

**T.R.  
SAKARYA UNIVERSITY  
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES**

**A NEW FRAMEWORK FOR DECENTRALIZED SOCIAL  
NETWORKS: HARNESSING BLOCKCHAIN, DEEP LEARNING,  
AND NATURAL LANGUAGE PROCESSING**

**MSc THESIS**

**AMIR AL KADAH**

**Software Engineering Department**

**Software Engineering Program**

**JUNE 2024**



**T.R.**  
**SAKARYA UNIVERSITY**  
**GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES**

**A NEW FRAMEWORK FOR DECENTRALIZED SOCIAL  
NETWORKS: HARNESSING BLOCKCHAIN, DEEP LEARNING,  
AND NATURAL LANGUAGE PROCESSING**

**MSc THESIS**

**AMIR AL KADAH**

**Software Engineering Department**

**Software Engineering Program**

**Thesis Advisor: Dr.Öğr.Üyesi Deniz BALTA**

**JUNE 2024**



The thesis work titled “A New Framework for Decentralized Social Networks: Harnessing Blockchain, Deep Learning, and Natural Language Processing” prepared by Amir Al Kadah was accepted by the following jury on 05/07/2024 by unanimously of votes as a MSc THESIS in Sakarya University Graduate School of Natural and Applied Sciences, Software Engineering department, Software Engineering programe.

### **Thesis Jury**

**Head of Jury :**

**Jury Member :**

**Jury Member :**



## **STATEMENT OF COMPLIANCE WITH THE ETHICAL PRINCIPLES AND RULES**

I declare that the thesis work titled " A New Framework for Decentralized Social Networks: Harnessing Blockchain, Deep Learning, and Natural Language Processing", which I have prepared in accordance with Sakarya University Graduate School of Natural and Applied Sciences regulations and Higher Education Institutions Scientific Research and Publication Ethics Directive, belongs to me, is an original work, I have acted in accordance with the regulations and directives mentioned above at all stages of my study, I did not get the innovations and results contained in the thesis from anywhere else, I duly cited the references for the works I used in my thesis, I did not submit this thesis to another scientific committee for academic purposes and to obtain a title, in accordance with the articles 9/2 and 22/2 of the Sakarya University Graduate Education and Training Regulation published in the Official Gazette dated 20.04.2016, a report was received in accordance with the criteria determined by the graduate school using the plagiarism software program to which Sakarya University is a subscriber, I have received an ethics committee approval document, I accept all kinds of legal responsibility that may arise in case of a situation contrary to this statement.

(05/07/2024)

( )

Amir Al Kadah





*To my family, whose belief in my potential lit the path of my scholarly pursuits.  
Your encouragement has been the wind beneath my wings, elevating me to heights  
unimagined.*



## **ABBREVIATIONS**

<b>BC</b>	: Blockchain
<b>DL</b>	: Deep Learning
<b>NLP</b>	: Natural language processing
<b>TF-IDF</b>	: Term Frequency-Inverse Document Frequency
<b>DApps</b>	: Decentralized applications
<b>API</b>	: Application Programming Interface
<b>HTML</b>	: HyperText Markup Language
<b>CSS</b>	: Cascading Style Sheets
<b>IPFS</b>	: InterPlanetary File System
<b>NPM</b>	: Node Package Manager
<b>JS</b>	: JavaScript
<b>IoT</b>	: Internet of Things
<b>EVM</b>	: Ethereum Virtual Machine
<b>URL</b>	: Uniform Resource Locators
<b>AI</b>	: Artificial Intelligence
<b>XAI</b>	: Explainable Artificial Intelligence
<b>NLG</b>	: Natural Language Generation
<b>NLU</b>	: Natural Language Understanding
<b>PoS</b>	: Proof of Stake
<b>PoW</b>	: Proof of Work
<b>PBFT</b>	: Practical Byzantine Fault Tolerance
<b>DPoS</b>	: Delegated Proof of Stake



## SYMBOLS

$t$	: Time [Unit]
$f_t$	: Forget gate activation at time $t$
$\sigma$	: Sigmoid function
$W_{fh}$	: Weight matrix for the hidden state in forget gate
$h_{t-1}$	: Hidden state at time $t-1$
$W_{fx}$	: Weight matrix for input in forget gate
$x_t$	: Input at time $t$
$b_f$	: Bias for forget gate
$i_t$	: Input gate activation at time $t$
$W_{ih}$	: Weight matrix for the hidden state in the input gate
$W_{ix}$	: Weight matrix for input in input gate
$b_i$	: Bias for input gate
$c_t$	: Candidate cell state at time $t$
$W_{ch}$	: Weight matrix for the hidden state in the candidate cell state
$W_{cx}$	: Weight matrix for input in candidate cell state
$b_c$	: Bias for candidate cell state
$c_t$	: Cell state at time $t$
$c_{t-1}$	: Cell state at time $t-1$
$o_t$	: Output gate activation at time $t$
$W_{oh}$	: Weight matrix for the hidden state in the output gate
$W_{ox}$	: Weight matrix for input in output gate
$b_o$	: Bias for output gate
$h_t$	: Hidden state at time $t$
$\tanh$	: Hyperbolic tangent function
$W_{ch}$	: Weight matrix for the hidden state in the candidate cell state
$LSTM$	: Long Short-Term Memory function
$h_t^{\rightarrow}$	: Forward hidden state at time $t$
$h_{t-1}^{\rightarrow}$	: Forward hidden state at time $t-1$

$\mathbf{h}_t^{\leftarrow}$  : Backward hidden state at time  $t$   
 $\mathbf{h}_{t+1}^{\leftarrow}$  : Backward hidden state at time  $t+1$   
 $[\mathbf{h}_t^{\rightarrow}; \mathbf{h}_t^{\leftarrow}]$  : Concatenation of forward and backward hidden states at time  $t$   
**Tp** : True Positives  
**Tn** : True Negatives  
**Fp** : False Positives  
**Fn** : False Negatives

## LIST OF TABLES

	<u>Page</u>
<b>Table 5.1.</b> Performance comparison between BiLSTM, GRU, and RNN.....	87
<b>Table 5.2.</b> Performance comparison between LSTM, GRU, and RNN .....	88
<b>Table 5.3.</b> The comparison of cosine similarity with TF-IDF and the other method.	90





## LIST OF FIGURES

	<u>Page</u>
<b>Figure 2.1.</b> Users interaction .....	42
<b>Figure 2.2.</b> Dataset categories .....	43
<b>Figure 2.3.</b> Class distribution. ....	44
<b>Figure 2.4.</b> Test length distribution.....	45
<b>Figure 2.5.</b> Word Cloud for suicide texts.....	46
<b>Figure 2.6.</b> Ganache interface.....	48
<b>Figure 2.7.</b> After inserting one of the address accounts in the metamask. ....	48
<b>Figure 3.1.</b> The Social Network Architecture.. ....	52
<b>Figure 3.2.</b> Sample Solidity Smart Contract Code.. ....	54
<b>Figure 3.3.</b> Sample of the code functions.....	55
<b>Figure 3.4.</b> Post prediction model architecture.....	58
<b>Figure 3.5.</b> Suicide detection model architecture. ....	61
<b>Figure 3.6.</b> Pseudo-Code for computing cosine similarity between text vectors. ....	62
<b>Figure 3.7.</b> Pseudo-Code for cosine similarity calculation in text analysis.....	63
<b>Figure 3.8.</b> Initializing Web3 provider, obtaining signer, and creating contract.....	65
<b>Figure 3.9.</b> API endpoint in Flask.....	65
<b>Figure 3.10.</b> JavaScript code for asynchronous interaction with a Flask API.....	66
<b>Figure 4.1.</b> Smart contract compilation outcomes. ....	67
<b>Figure 4.2.</b> Smart contract migration outcomes. ....	67
<b>Figure 4.3.</b> Flask Application Running Successfully.....	68
<b>Figure 4.4.</b> Metamask wallet interface.....	69
<b>Figure 4.5.</b> Ganache private key extraction.....	69
<b>Figure 4.6.</b> Metamask wallet address insertion. ....	70
<b>Figure 4.7.</b> Metamask address charged with 100 ethereum. ....	70
<b>Figure 4.8.</b> Sign up page. ....	71
<b>Figure 4.9.</b> Creating post page.....	73
<b>Figure 4.10.</b> Suicide detected. ....	75
<b>Figure 4.11.</b> Similarity result. ....	76
<b>Figure 4.12.</b> Category prediction result. ....	76
<b>Figure 4.13.</b> Gas fee transaction. ....	77
<b>Figure 4.14.</b> Profile page content.....	78
<b>Figure 4.15.</b> User verification page. ....	80
<b>Figure 4.16.</b> Main page.. ....	82
<b>Figure 4.17.</b> Searching page. ....	83



# **A NEW FRAMEWORK FOR DECENTRALIZED SOCIAL NETWORK: HARNESSING BLOCKCHAIN, DEEP LEARNING, AND NATURAL LANGUAGE PROCESSING**

## **SUMMARY**

This thesis explores the incorporation of blockchain technology, deep learning (DL), and natural language processing (NLP) to create a decentralized social network that aims to tackle privacy breaches and unethical data manipulation that are inherent in centralized social networks. The main goal is to provide a social media network that is secure, transparent, and focused on the needs of the users. The research is centered around three main areas: augmenting data security and privacy, boosting content authenticity and moderation, and empowering people through data control.

Blockchain technology offers a distributed and unchangeable record system that greatly improves the security and confidentiality of data. Blockchain reduces the dangers of centralized data storage, such as data breaches and illegal access, by distributing data across a network of nodes and using cryptographic mechanisms. The research provides evidence that blockchain is capable of securely protecting user data, guaranteeing both privacy and trust. Deep learning and natural language processing (NLP) algorithms are essential for improving the accuracy and control of content authenticity and moderation on social networks. Deep learning algorithms are extremely efficient at classifying content, forecasting user preferences, and identifying suicidal inclinations in posts.

This skill is crucial for delivering prompt interventions and guaranteeing a more secure online environment. In addition, the platform's capabilities are enhanced by employing NLP techniques, which enable the system to efficiently comprehend and analyze human language. This involves the identification of similarities between postings and the detection of plagiarism, which are crucial in upholding the validity of content and preventing the dissemination of disinformation. The blockchain-based social network's decentralized architecture grants users enhanced authority over their data, so empowering them. Conventional social networks frequently utilize user data for financial benefit without adequately compensating or empowering the users.

Conversely, the suggested solution guarantees that consumers maintain ownership of their data and possess enhanced authority over its utilization. Smart contracts enable transparent and secure interactions, enabling users to participate in transactions and data exchanges without the need for a central authority. Nevertheless, the establishment of a social network based on blockchain technology presents several obstacles. Scalability is a major concern. Public blockchain networks frequently encounter scalability issues as a result of the substantial computational resources needed for transaction validation and consensus methods. This can lead to decreased

transaction speed and increased operational expenses, thus impeding the mainstream acceptance of the system.

Another obstacle lies in the interpretability of deep learning models. DL algorithms are extremely efficient in data analysis, yet their intricate and obscure nature sometimes leads to them being referred to as "black boxes." The absence of openness can provide challenges, especially in situations when comprehending the decision-making process is essential.

Moreover, the incorporation of blockchain and distributed ledger (DL) technologies necessitates substantial processing capacity and infrastructure, which can pose a hindrance for smaller enterprises or individual users. Ensuring that decentralized social networks are accessible and affordable is crucial for their future development and adoption. Future research should prioritize the development and implementation of cutting-edge technologies, such as sharding and off-chain solutions, to improve scalability.

Furthermore, it is imperative to prioritize the development of DL models that are easier to understand and to integrate explainable AI techniques in order to improve transparency and foster user trust. By investigating the incorporation of additional cutting-edge technologies, such as edge computing and federated learning, it is possible to enhance the efficiency and safety of decentralized social networks. It is crucial to prioritize the accessibility and cost of these technologies for smaller companies and individual consumers.

Ultimately, the combination of blockchain, deep learning, and natural language processing presents a hopeful resolution for revolutionizing social networks. The suggested approach has the potential to revolutionize the social media ecosystem by addressing privacy concerns, improving content authenticity, and empowering users. This study presents an all-encompassing structure for creating a social networking platform that is more safe, transparent, and centered around the needs of the users. This framework sets the stage for future advancements in digital communication. The challenges identified emphasize the necessity of ongoing enhancement and originality to fully achieve the advantages of these technologies. In summary, this thesis adds to the current endeavors aimed at establishing a more secure, fair, and user-focused online atmosphere.

# **A NEW FRAMEWORK FOR DECENTRALIZED SOCIAL NETWORK: HARNESSING BLOCKCHAIN, DEEP LEARNING, AND NATURAL LANGUAGE PROCESSING**

## **ÖZET**

Bu tez, merkezi sosyal ağlarda doğal olan gizlilik ihlallerini ve etik dışı veri Günümüzde sosyal platformlar, sadece etkileşim alanlarından çok daha fazlası haline gelmiş ve modern hayatın ayrılmaz bir parçası olmuştur. Bu dönüşüm, veri gizliliği endişeleri, bilgi güvenilirliği ve ekonomik değerın adil dağıtımını gibi bir dizi sorunu da beraberinde getirmiştir. Bu tez, bu sorunlara çözüm olarak blockchain teknolojisi, derin öğrenme (DL) ve doğal dil işleme (NLP) entegrasyonunu önermektedir. Bu teknolojilerin bir araya getirilmesiyle, kullanıcıların bilgi doğrulama süreçlerine katılımının sağlanması, güvenlik düzeyinin artırılması ve kullanıcı etkileşiminin teşvik edilmesi hedeflenmektedir. Blockchain, verilerin merkezi bir otoriteye ihtiyaç duymadan dağıtık bir ağda tutulmasını sağlayan, değiştirilemez ve şeffaf bir dijital defter olarak tanımlanmaktadır.

Her bir blok, kriptografik hash'ler aracılığıyla önceki bloklarla bağlantılıdır, bu da veri manipülasyonunu neredeyse imkansız hale getirir. Blockchain'in merkezi olmayan yapısı, verilerin bütünlüğünü garanti eder ve kullanıcıların veri üzerindeki kontrolünü geri kazanmasını sağlar. Bu teknoloji, kripto paraların ötesinde finans, tedarik zinciri yönetimi ve belge doğrulama gibi alanlarda da uygulanmaktadır.

Derin öğrenme, büyük veri yığınlarını analiz ederek anlamlı desenler çıkarmak için kullanılan bir tekniktir. DL, özellikle sosyal ağlardaki kullanıcı tercihlerini tahmin etmek ve içerik sınıflandırmasında oldukça etkilidir. Bu bağlamda, farklı türde sinir ağları, özellikle de Recurrent Neural Networks (RNN) ve Long Short-Term Memory (LSTM) ağları, teorik arka planda incelenmektedir. LSTM'ler, ardışık verilerde uzun vadeli bağımlılıkları yakalamada etkilidir ve sosyal medya metinlerinde intihar tespiti gibi görevlerde kullanılabilir. Bidirectional LSTM (BiLSTM) ağları ise veriyi hem ileri hem de geri yönde işleyerek bağlamı daha etkili bir şekilde yakalar ve bu nedenle kategori tahmini gibi görevlerde yüksek doğruluk sağlar. NLP, insan dilini işleme ve anlama konusunda güçlü bir araçtır. Bu bölümde, TF-IDF vektörizasyonu ve kosinüs benzerliği algoritmaları gibi çeşitli NLP teknikleri anlatılmaktadır. TF-IDF (Term Frequency-Inverse Document Frequency) modeli, metin verilerini sayısal vektörlere dönüştürerek her kelimenin önemini hesaplar. Bu yöntem, özellikle belge içerisindeki anahtar kelimeleri belirlemede ve metinlerin birbirine olan benzerliğini ölçmede kullanılır. Sistemimizde TF-IDF, kullanıcı gönderilerinin orijinalliğini ve benzersizliğini değerlendirmek için kullanılmıştır. Bu sayede, içerik orijinalliğinin sağlanması ve intihalin önlenmesi amacıyla benzer veya tekrarlayan içerikler tespit edilebilir. NLP'nin blockchain ve DL ile entegrasyonu, sosyal ağların kullanıcılara kişiselleştirilmiş ve ilgili içerik sunma kapasitesini artırır. Blockchain, DL ve NLP'nin

bir araya getirilmesi, merkezi olmayan bir sosyal ağ oluşturma sürecinde önemli bir sinerji yaratır. Blockchain, veri güvenliği ve kullanıcı gizliliğini sağlarken, DL ileri düzey analizler sunar ve NLP etkin içerik moderasyonu ve etkileşimi sağlar. Önerilen sistem, kullanıcı kaydı, gönderi oluşturma ve veri alımı gibi temel işlevleri yönetmek için Ethereum blockchain üzerinde Solidity tabanlı akıllı sözleşmeler kullanır.

JavaScript'in Web3.js kütüphanesi aracılığıyla akıllı sözleşmelerle entegrasyonu, web arayüzü ile Ethereum blockchain arasındaki etkileşimleri kolaylaştırmak için kritik öneme sahiptir. Bu etkileşim, sayfa yüklendiğinde Web3'ün başlatılmasıyla başlar ve tarayıcıya uygun bir Ethereum uzantısının (örneğin MetaMask) olup olmadığını kontrol eder. Kullanıcı hesap erişimi sağlandığında, akıllı sözleşmenin ABI'si kullanılarak sözleşme örnekleri oluşturulur ve kullanıcı ekleme, gönderi oluşturma veya blockchain'den veri alma gibi işlemler gerçekleştirilir. JavaScript'teki olay işleme, kullanıcı eylemlerini yönetir ve işlem süreçlerini kesintisiz hale getirir, böylece kullanıcı deneyimini dinamik ve tepki veren bir yapıya kavuşturur.

Platformun yeteneklerini daha da artıran Flask sunucusu, JavaScript ön yüzü ile derin öğrenme (DL) ve doğal dil işleme (NLP) modelleri arasında bir köprü görevi görür. Sunucu, bu modelleri barındırır ve kategori tahmini ve gönderi benzerlik tespiti gibi işlevler için REST API uç noktalarını sağlar. Kullanıcı bir gönderiyi analiz için sunduğunda, içerik önceden eğitilmiş modeller tarafından işlenir ve kategorisi belirlenir veya mevcut gönderilerle benzerlikleri tespit edilir. Bu yapı, sistemin kullanıcı tarafından oluşturulan içerik üzerinde gerçek zamanlı, doğru geri bildirim sağlamasına imkan tanır ve ileri düzey DL ve NLP tekniklerinden yararlanarak içerik bütünlüğünü ve orijinalliğini korur.

İçerik orijinalliğini korumak ve tekrar eden içerikleri önlemek için sistem, ileri düzey metin benzerlik ölçümleri kullanır. TF-IDF vektörizasyonu ile birlikte Cosine Benzerliği entegrasyonu, farklı uzunluktaki metinleri açılımlarına göre değerlendirerek işler. TF-IDF'nin terimlerin önemini ağırlıklandırmasıyla bu yöntem, benzerlik tespitinde yüksek doğruluk sağlar ve nadir ancak anlamlı kelimeleri vurgular. Bu yaklaşım, gerçek zamanlı içerik moderasyonunu kolaylaştırmakla kalmaz, aynı zamanda karar alma sürecinde şeffaflık sağlar ve kullanıcı güveni ile içerik yönetiminde kritik bir rol oynar.

Bir Flask sunucusu, DL ve NLP modellerini barındırır ve veri alışverişi için REST API uç noktaları sağlar. Bu modeller, gönderi kategorisi tahmini ve intihar tespiti gibi görevleri yerine getirir. SocialNetwork adlı akıllı sözleşme, Solidity dilinde yazılmış ve Ethereum blockchain'ine dağıtılmıştır. Bu sözleşme, kullanıcı yönetimi ve gönderi yönetimi gibi işlevleri içerir. Kullanıcı ve gönderi verilerini saklamak için UserInfo ve PostContent gibi temel yapılar kullanılır. Kullanıcı kaydı, gönderi oluşturma ve gönderilere beğeni veya satın alma gibi etkileşimleri yöneten çeşitli işlevler mevcuttur.

Belirli etkinlikleri kaydeden olaylar, şeffaflığı sağlar. DL modeli, özellikle intihar tespiti gibi metin sınıflandırma görevlerine odaklanır. Süreç, veri hazırlama, metin tokenizasyonu, sıra doldurma ve etiket kodlama aşamalarını içerir. Model mimarisi, metindeki anlamsal ilişkileri ve bağlamsal bağımlılıkları yakalamak için Embedding ve Bidirectional LSTM katmanlarını içerir. Dropout katmanları aşırı uyumu önler ve son katman, ikili sınıflandırma için sigmoid aktivasyon fonksiyonuna sahip bir Dense katmandır. Eğitim için Adam optimize edici ve ikili çapraz entropi kayıp fonksiyonu kullanılır.

NLP teknikleri, metin benzerliklerini tespit etmek ve içerik orijinallliğini sağlamak için TF-IDF vektörizasyonu ve kosinüs benzerliği gibi yöntemler kullanır. Süreç, metin verilerini sayısal vektörlere dönüştürür ve benzerliği hesaplar. Bu teknolojilerin entegrasyonu, blockchain'in değişmezliği ile DL ve NLP'nin analitik gücünü birleştirir. Bu, veri bütünlüğünü sağlayarak kullanıcı etkileşimini ve güvenliğini artırır. Akıllı sözleşme, kullanıcı ve gönderi yönetim sistemi içerir. Kullanıcı kaydı (addUserInfo), gönderi oluşturma (createPost) ve etkileşim yönetimi (likePost, buyPost) gibi işlevler bulunmaktadır. Ayrıca gönderileri düzenleme ve silme desteği, blockchain ağı içinde veri bütünlüğünü korur.

Ön uç, akıllı sözleşme ile JavaScript ve Web3.js kütüphanesi aracılığıyla etkileşimde bulunur. Flask sunucusu, DL ve NLP model işlemlerini yönetir ve REST API uç noktaları aracılığıyla veri alışverişi yapar. JavaScript, eşzamanlı veri akışını yönetir, gönderi içeriğini analiz eder ve sonuçları kullanıcı arayüzüne entegre eder. Önerilen merkezi olmayan sosyal ağın uygulanmasının sonuçları, blockchain, DL ve NLP teknolojilerinin veri gizliliği, güvenliği, içerik güvenilirliği ve kullanıcı yetkilendirmesi üzerindeki etkilerini değerlendirmektedir. Performans, doğruluk, kesinlik, hatırlama ve F1 puanı gibi anahtar metrikler kullanılarak değerlendirilmiştir. Performans Metrikleri, doğruluk, doğru sınıflandırılan öğelerin toplam gözlem sayısına oranını ölçer. Kesinlik, yanlış pozitiflerin maliyetinin yüksek olduğu senaryolarda önemli olan, doğru pozitif tahminlerin toplam tahmin edilen pozitiflere oranını yansıtır. Hatırlama, modelin bir veri kümesindeki tüm ilgili örnekleri tanımlama kapasitesini ölçer. F1 Puanı, Kesinlik ve hatırlamanın ağırlıklı ortalamasını hesaplar, sınıf dağılımının dengesiz olduğu durumlarda kullanışlıdır.

Tartışma başlığında gizlilik ve veri güvenliği, blockchain teknolojisi, verilerin merkezi olmayan depolanması ve kriptografik teknikler kullanılarak güvenliğini ve gizliliğini önemli ölçüde artırır. İçerik güvenilirliği ve moderasyonu, DL ve NLP teknolojileri, içerik güvenilirliği ve moderasyonunu artırmada önemli bir rol oynar. Kullanıcı yetkilendirmesi, blockchain tabanlı sosyal ağın merkezi olmayan mimarisi, kullanıcıların verileri üzerinde kontrol sahibi olmalarını sağlar. Gelecek yönelimler ise ölçeklenebilirlik, blockchain ağlarının ölçeklenebilirliğini artırmak için yenilikçi stratejiler geliştirilmelidir. DL model yorumlanabilirliği, DL modellerinin karar verme süreçlerinin anlaşılabilirliğini artırmak için açıklanabilir AI teknikleri entegre edilmelidir.

Yeni teknolojilerle entegrasyon, kenar bilişim ve federatif öğrenme gibi yeni teknolojilerle entegrasyon araştırılmalıdır. Kullanıcı deneyiminin iyileştirilmesi, içerik önerme algoritmalarının geliştirilmesi ve AR/VR gibi yeni etkileşim paradigmalarının araştırılması kullanıcı deneyimini artırabilir. Erişilebilirlik ve uygunluk, küçük şirketler ve bireysel kullanıcılar için merkezi olmayan sosyal ağ teknolojilerinin erişilebilirliği ve uygunluğu sağlanmalıdır.

Etik ve düzenleyici hususlar, merkezi olmayan sosyal ağların etik kullanımı ve düzenleyici çerçeveler geliştirilmelidir. Uzunlamasına çalışmalar ise merkezi olmayan sosyal ağların kullanıcı davranışı, gizlilik ve veri güvenliği üzerindeki uzun vadeli etkilerini değerlendirmek için uzunlamasına çalışmalar yapılmalıdır. Bu genişletilmiş özet, blockchain teknolojisi, derin öğrenme ve doğal dil işlemenin entegrasyonu yoluyla merkezi olmayan bir sosyal ağ oluşturma sürecine kapsamlı bir bakış sağlamaktadır. Önerilen sistem, kullanıcıların veri güvenliğini, gizliliğini ve

kontrolünü artırmayı hedeflerken, aynı zamanda içerik güvenilirliği ve kullanıcı etkileşimini de teşvik etmektedir.



## **1. INTRODUCTION**

Social platforms have evolved from social interaction spaces to complex ecosystems that permeate all aspects of modern life. This transformation has not come without challenges, such as concerns over data privacy, information authenticity, and the fair and equitable distribution of the economic value tied to that information. A novel approach for addressing these issues, presented in this paper, explores the integration of blockchain technology, deep learning (DL), and natural language processing for enhancing security, creating a transparent and user-controlled information validation mechanism, and for promoting user engagement in the redefined social networking services.

### **1.1. Overview**

The integration of blockchain technology, deep learning (DL), and Natural Language Processing (NLP) has ushered in a new era in the design and operation of social networks, now increasingly founded on principles of data sovereignty, transparency, and user-centric authentication. Utilizing blockchain technology as a framework demonstrates how social networks can function without centralized control over data, effectively returning information ownership and oversight to users where it belongs [1]. Meanwhile, DL and NLP enable a nuanced analysis of the content that users create, meaning more than ever that we are what we post. These technologies analyze the content of our posts to uncover and highlight themes, interests, and preferences, thus enabling a rich and deeply personalized user experience [2]. NLP further processes extensive textual data within these networks, supporting functionalities such as chatbot communications, content recommendations, and trend analysis, and automating content moderation to ensure appropriate interactions, thereby enhancing user experiences by delivering more relevant and engaging content [3]. This dissertation examines the combined potential of blockchain, DL, and NLP technologies to transform the operation of social networks by providing unprecedented transparency of data flow, a robust and user-centric framework for secure account

authentication, and an unparalleled ability to tailor network content in real-time to the perceived interests and preferences of individual users.

Social networks enhanced by the powerful combination of blockchain technology, DL, and NLP can transform data privacy and security within the space. Blockchain's decentralized architecture utilizes an immutable ledger system ensuring that data integrity is never compromised [1]. These decentralization capabilities are crucial for creating a new social network where user privacy and data control are prioritized, with transactions and data exchanges within the network securely recorded without a central authority. DL, as a subset of machine learning, alongside NLP, drives complex algorithms to understand and analyze the vast sets of data created by users on social networks. For example, by examining patterns in posts, these algorithms can analyze user preferences, thus delivering content much more personalized and relevant to the user [2]. Additionally, NLP enhances platform interaction by enabling advanced processing and understanding of user-generated content, particularly in identifying similarities and differences between posts, which improves content relevance and authenticity [3].

Moreover, the integration of blockchain, DL, and NLP addresses several critical issues currently faced by traditional social networks, such as data control and account authentication. Blockchain provides a robust environment for identity verification and data exchange, ensuring that user accounts are authenticated and remain untampered [1]. Concurrently, DL and NLP use behavioral data to analyze user preferences, serving the dual purpose of defending the network from fraudulent activity and enhancing the user experience overall by ensuring that content delivery is relevant to individual interests [2].

The thesis will evaluate the transformative potential of blockchain, deep learning, and NLP in shaping a new paradigm for social networks that hinges upon user-controlled data transparency and personalized content delivery. The goal is to create a social network that blends the security and transparency of blockchain with the deep analytical capabilities of DL and the advanced processing power of NLP to achieve unprecedented levels of user privacy, security, and engagement.

## **1.2. Problem Statment**

Conventional social networks, despite being widely used and popular, encounter substantial and diverse obstacles that have extensive consequences for both consumers and authors of information. The difficulties can be classified into three primary domains: centralized data storage, limited transparency, and insufficient value distribution to content creators. Each of these concerns arises from the fundamental design and operating structures of these platforms, resulting in an intricate network of problems that impact users in diverse ways. An urgent concern with conventional social networks is the centralized storage of user data. Social media platforms such as Facebook, Twitter, and Instagram retain significant quantities of personal data, preferences, and content created by users on their servers. The concentration of data storage in a concentrated location presents considerable privacy hazards. The hazards associated with centralizing all user data have been brought to light by several data breaches and examples of misuse. An example of this is the Facebook-Cambridge Analytica incident, which revealed the unauthorized collection and utilization of millions of users' data for political advertising purposes, without their awareness or consent[4,5]. This incident not only emphasized the possibility of misuse but also brought attention to the wider risks associated with centralized data storage. Centralized systems facilitate the accessibility of substantial amounts of sensitive information to malicious individuals, resulting in substantial infringements of user privacy. Moreover, this concentration also empowers platforms to exert significant authority over the dissemination of information on their networks. This sort of control can result in censorship, when platforms possess the authority to selectively remove or obstruct content, often leading to allegations of prejudice and infringements upon freedom of expression. The absence of transparency and accountability in controlling and manipulating the dissemination of information intensifies the distrust that users have in these platforms [5]. The deficiency in transparency, or more precisely, the lack of it, is another significant issue afflicting conventional social networks. Users frequently lack transparency regarding the process of making content moderation choices. The criteria and techniques used to determine the display, concealment, or removal of content lack transparency and honesty. The lack of transparency causes confusion and dissatisfaction among users, who perceive the enforcement of

community norms as inconsistent and unjust. Studies on content moderation highlight the intricate nature of these judgments, underscoring the importance of more transparency and accountability [5,6]. Users' belief in the platforms' integrity and commitment to fair treatment diminishes when they lack a comprehensive comprehension of the moderation processes. The capriciousness of these judgments can also give rise to allegations of prejudice, as specific perspectives are seen as being unjustly singled out or repressed. The absence of transparency not only impacts user confidence but also weakens the overall credibility of the platform [5]. Aside from concerns regarding privacy and transparency, conventional social networks also fall short in effectively distributing the value derived from user-generated content to the creators of that content. These platforms mainly depend on user-generated content to attract and retain an audience, which subsequently boosts their advertising revenue. Nevertheless, the individuals responsible for producing this content frequently receive minimal or nonexistent remuneration for their efforts. This disparity is especially flagrant considering that the platforms' revenue is strongly linked to user involvement and the data derived from such engagement [5]. The matter of ownership and monetization is further convoluted by stringent copyright and licensing agreements, which confer onto the platforms substantial rights to utilize, alter, and disseminate user work without sufficient remuneration. Content creators, who are vital to the existence of these social networks, are increasingly being pushed to the sidelines and their contributions are not given the recognition they deserve. The digital economy has greatly altered the production and consumption of material, posing challenges to traditional concepts of ownership and value distribution in the creative industries [5]. Although these platforms produce substantial money, the creators, who play a crucial role in their success, frequently do not receive a fair share of the economic rewards. This scenario not only diminishes the motivation of current creators but also deters prospective artists from participating, ultimately impacting the variety and quality of content accessible on these platforms[5]. The issues linked to conventional social networks are fundamentally ingrained in their design and operating frameworks. Storing data in a centralized manner puts consumers at considerable risk of privacy violations since a huge amount of sensitive information becomes susceptible to breaches and unauthorized usage. The absence of transparency in the processes of content moderation results in user dissatisfaction, perplexity, and a

widespread skepticism regarding the platforms' impartiality and honesty. Moreover, the lack of sufficient compensation for content providers emphasizes the inequity in the value distribution systems of social networks. In light of the ongoing development of the digital economy, it is crucial for traditional social networks to thoroughly tackle these matters. Improving data privacy through decentralized storage options, promoting transparency in content moderation, and establishing fairer value-sharing models are crucial measures for creating platforms that are more egalitarian and trustworthy. In order to rebuild user confidence, preserve privacy, and create a more inclusive and rewarding environment for content creators, traditional social networks must solve these fundamental challenges[5].

### **1.3. Objectives**

The goal of the research focuses on integrating blockchain technology, deep learning (DL) and natural language processing (NLP) in social networks, which is formulated as follows.

- Research and apply blockchain technology to improve data authentication on social networks.
- Analyze deep learning capabilities to accurately predict user interests on social networks.
- Research the effectiveness of natural language processing in detecting similarities between messages.
- Consider the challenges and opportunities of implementing decentralized social networks based on blockchain and deep learning.
- Conduct a comprehensive review of integrating blockchain technology and deep learning into social networks, evaluating both advantages and disadvantages.

To achieve these goals, this study aims to highlight the general potentials and limitations of blockchain technology, deep learning, and natural language processing (NLP), and provide a roadmap for the development of Innovative, user-centric and more secure social platforms.

## **1.4. Research Questions**

To guide the research process, the following research questions will be addressed:

- How can blockchain technology be utilized to enhance data authentication processes in social networks?
- How can deep learning be applied to predict user preferences within social networks accurately?
- How can natural language processing be employed to detect similarities between posts in social networks?
- What are the challenges and opportunities associated with implementing a decentralized social network that leverages blockchain technology and deep learning?

In tackling these research inquiries, the objective of this study is to shed light on how blockchain technology and the Ethereum network can be effectively combined with DL and NLP. It seeks to evaluate the benefits and drawbacks of this method, and investigate possible.

## **1.5. Literature Review**

The literature review explores the integration of blockchain, natural language processing (NLP), and deep learning (DL) within social networks to address significant challenges like privacy breaches and unethical data manipulation. The review provides a comprehensive examination of existing research and developments in these technologies, highlighting their potential to enhance security, improve content authenticity, and empower users through greater control over their data. By analyzing various studies, this section underscores the innovative applications and theoretical underpinnings of blockchain, DL, and NLP technologies in transforming traditional centralized social networks into decentralized, user-centric platforms.

### **1.5.1. Blockchain in social networks**

In the realm of modern digital communication, blockchain technology is increasingly recognized for its potential to revolutionize social networking platforms by enhancing data security, privacy, and user governance.

“A decentralized social networking architecture enhanced by blockchain”

The paper introduces a novel decentralized social networking architecture leveraging blockchain technology to address privacy concerns in centralized online social networks (OSNs). It proposes a sharding framework for enhanced scalability, a blockchain system for data integrity and consistency, and a reputation-based authority control method to augment system security. The architecture comprises a peer-to-peer communication network, a blockchain network ensuring a trustless environment for safe interactions, and a social network that operates similarly to existing OSNs but with decentralized data services. The system categorizes users into different types based on their social reputation, assigning varying permissions and roles within the network. This model aims to give users more control over their privacy and data while improving the trustworthiness and efficiency of the social networking ecosystem. The methodology involves key components such as data storage distinguishing between social and hash data for integrity verification, a sharding-based two-layer blockchain for improved efficiency and security, a consensus mechanism employing verifiable random functions for miner selection and attack prevention, and a reputation-based permission assignment to enhance trust and system security. These strategies collectively aim to address major concerns associated with centralized OSNs by offering a decentralized solution that prioritizes privacy, data control, and trust [7].

“BCOSN: A Blockchain-Based Decentralized Online Social Network”

This paper presents an innovative framework for constructing a privacy-centric, decentralized online social network (OSN) leveraging blockchain technology. It introduces a novel architecture that distinctly separates control and storage services, integrating smart contracts for decentralized governance and data management to counteract the inefficiencies plaguing existing decentralized OSNs (DOSNs) by ensuring data privacy, integrity, and efficient access control. This architecture is validated using real-world datasets to demonstrate its efficacy in providing a secure,

privacy-protected social networking experience. The methodology underpinning BCOSN incorporates the development of smart contracts for user registration and management, the use of Ciphertext-Policy Attribute-Based Encryption (CP-ABE) for stringent data access control, and detailed experiments to assess the system's cost-efficiency and recommendation mechanisms. It also includes a comprehensive three-layer architecture comprising a blockchain layer for control, a storage layer for data, and an application layer for user interactions, aimed at enhancing scalability, security, and data management efficiency. Additionally, the paper explores decentralized storage solutions, attribute-based access control for enhanced privacy and security, social graph-based friend recommendation systems, and extended functionalities like message synchronization for offline users. Through rigorous experimental evaluation on the Ethereum platform, the paper delineates the cost implications of smart contract operations, the efficiency of its friend recommendation system, and the performance of its event notification mechanism, while also contemplating the limitations and potential future directions for research in creating secure, decentralized OSNs [8].

#### “DECENTRALISED SOCIAL MEDIA”

The paper presents a comprehensive approach to decentralized social media, leveraging blockchain technology to ensure data integrity, user privacy, and control. It proposes a web application-based social network, utilizing Ethereum blockchain and IPFS for distributed storage, focusing on user control over data and enabling seamless transitions between social networking services to enhance freedom and data privacy. The methodology includes implementing NFT smart contracts for user interactions, a unique OSN blockchain algorithm for secure data sharing, and a hybrid consensus mechanism combining Proof of Work and Proof of Stake to balance efficiency, scalability, and security. By advocating for a shift towards decentralized social networks, the paper aims to address key concerns associated with centralized social networks, such as data misuse, censorship, and lack of user control, proposing a decentralized model that fosters user-centric, secure, and transparent online social interactions [9].

#### “Social-Chain: Decentralized Trust Evaluation Based on Blockchain in Pervasive Social Networking”



The paper introduces Social-Chain, a blockchain-based decentralized system for evaluating trust within pervasive social networks (PSNs), confronting the challenges of trust evaluation where interactions often lack pre-existing relationships. It innovates with the Proof-of-Trust (PoT) consensus mechanism, integrating trust evaluation into blockchain consensus, designed to be lightweight and efficient, emphasizing security, efficiency, and decentralization. Social-Chain's methodology involves developing the PoT mechanism, distinct from traditional blockchain consensus mechanisms, to address PSN-specific challenges like limited computational resources and the need for swift, reliable trust evaluations. It includes four key algorithms—Block Generation, Timestamp Validation, Mining Winner Selection, and Consensus Policy Setting—working in tandem to generate new blocks from trust evidence, ensure block timestamp integrity, select mining winners to maintain blockchain integrity, and establish consensus policies for trust-based block acceptance. This system allows for transparent, verifiable trust assessments among users, employing a practical trust evaluation algorithm resistant to attacks, ensuring trustworthy trust evaluations directly within the blockchain consensus process, thus achieving trust consensus efficiently, preventing forking, and maintaining decentralization and security [10].

“Towards Trusted Social Networks with Blockchain Technology”

The paper show that the integration of blockchain technology to mitigate rumor spreading in social networks. It introduces a decentralized mechanism for information exchange, leveraging virtual credits within smart contracts to incentivize the sharing of trustworthy information and deter the spread of rumors. A blockchain-based SIR (Susceptible-Infected-Recovered) model is developed to simulate and analyze the dynamics of rumor propagation. Numerical simulations demonstrate the model's effectiveness in controlling rumor spread, highlighting the potential of blockchain in creating more trusted and secure social networking environments [11].

### **1.5.2. Deep learning in social networks**

Deep learning (DL) has emerged as a pivotal technology in social network analysis, providing sophisticated tools for processing and interpreting the vast amounts of unstructured data generated by users, thereby enhancing both the understanding and functionality of digital social interactions.

### “A Deep Learning Approach for Dengue Tweet Classification”

The paper presents a novel approach using Convolutional Neural Networks (CNNs) to classify tweets related to dengue into seven categories: Infected, Informative, Vaccination, News, Awareness, Concern, and Others. Highlighting the superiority of deep learning over traditional machine learning methods like SVM, Naive Bayes, and Decision Trees, the study showcases the effectiveness of CNNs in accurately categorizing tweets, emphasizing the potential of social media as a tool for real-time dengue surveillance and public health awareness. The proposed method addresses the limitations of traditional algorithms by leveraging CNNs for feature learning, automatically extracting features from the raw tweet data, thus improving classification accuracy and scalability. The comprehensive methodology includes data collection, preprocessing (removal of emoticons, punctuations, and URLs), and classification through a CNN model designed with layers for data processing, embedding, convolution, pooling, dropout, and output, demonstrating deep learning's capability in handling the complexities of natural language processing [12].

### “Deep Learning for Automated Sentiment Analysis of Social Media”

The paper introduces a deep learning-based framework aimed at processing social media content, particularly focusing on movie reviews. Highlighting the difficulties of managing informal language, including slang and emoticons, it addresses the inadequacies of traditional natural language processing tools in this context. The framework employs a Python web crawler to collect review data from platforms like YouTube and Facebook, and a preprocessing module to convert informal text elements into standardized forms using an Internet slang word dictionary. For sentiment analysis, it utilizes NLTK, Textblob, and the Google Cloud Natural Language API to assign sentiment scores to each sentence, further employing word embedding techniques like word2vec and GloVe. Deep learning models, including LSTM, BiLSTM, and GRUs, are trained on this preprocessed data to classify sentiments accurately. The effectiveness of this approach is tested through experimental evaluation, where trailer comments from YouTube are used to create a labeled dataset for model performance comparison in terms of precision, recall, F-measure, and accuracy. This comprehensive method aims to leverage the capability of deep learning

models to accurately interpret the sentiment of user-generated content on social media platforms, offering significant insights for marketing and opinion mining [13].

#### “Fake News Detection on Social Media using Geometric Deep Learning”

The paper introduces a method to detect fake news on Twitter utilizing geometric deep learning, focusing on understanding the propagation patterns of fake news through graph-structured data analysis, including user profiles, activities, and social network structures. It emphasizes training on verified news story datasets to achieve high accuracy in identifying fake news based on these patterns. The proposed model, specifically designed for graph-structured data, integrates heterogeneous data types into a unified framework, employing a supervised learning approach with a Graph CNN architecture. This architecture includes convolutional and fully connected layers, leveraging graph attention, mean-pooling for dimensionality reduction, and SELU for non-linearity, trained with hinge loss. Demonstrating nearly 93% ROC AUC accuracy and the capacity to detect fake news within hours of its spread, the model underscores the efficiency of geometric deep learning in processing graph-structured data and the importance of social network structure and propagation patterns in enhancing fake news detection strategies [14].

#### “Research on Social Media Feature Learning Algorithm Based on Deep Neural Network”

This paper presents a novel deep neural network model designed to optimize feature learning from social media text data. This model introduces a variable-speed learning algorithm, incorporating Mini-Batch Gradient Descent (MBGD), to enhance the training and parameter adjustment process efficiently, aiming to significantly improve accuracy and efficiency in social media data mining. Tailored to confront the challenges presented by the massive influx of information on social media platforms, this approach utilizes advanced deep learning techniques to refine text mining processes, ensuring better utilization of social media data. The proposed model is adept at navigating the complexities inherent in social media data, employing a strategic approach through the use of MBGD and deep neural network methodologies to not only improve the text mining processes but also to optimize the extraction and learning of features for diverse applications, thereby offering a robust solution for enhancing the utility of social media information [15].

### “Suicidal Text Detection on Social Media for Suicide Prevention Using Deep Learning Models”

The paper introduces two sophisticated models, LSTM and DistilBERT, tailored for identifying suicidal content on social media platforms. The LSTM model, known for its effectiveness in processing sequential data, includes components like Cell State, Input Gate, Forget Gate, Output Gate, and Hidden State, making it adept at distinguishing between suicidal and non-suicidal content. On the other hand, DistilBERT, a streamlined variant of the BERT model, utilizes Transformer Encoder, knowledge distillation, parameter sharing, and task-specific layers to efficiently process and classify text data. The paper demonstrates DistilBERT's superior performance over LSTM in accurately classifying texts, supported by an innovative Telegram bot application for real-time intervention. This research highlights the potential of deep learning in enhancing suicide prevention efforts, showcasing the effectiveness of DistilBERT in processing and recognizing patterns indicative of suicidal ideation within social media communications [16].

### **1.5.3. Natural language processing in social network**

Natural Language Processing (NLP) plays a crucial role in social network analysis by enabling the automated understanding and processing of vast amounts of unstructured textual data, thus facilitating enhanced insights into user behaviors, sentiments, and community dynamics.

#### “Natural Language Processing: Challenges and Future Directions”

The paper delves into the critical issue of fake news and its impact on democracy, journalism, and public trust, providing a thorough examination of the various methods employed to detect fake news using natural language processing (NLP). It emphasizes the significance of data preprocessing, vectorization, and the application of machine learning techniques in distinguishing between genuine and fraudulent information. The study adopts a stylistic-computational approach to NLP, focusing on the identification and analysis of specific linguistic and stylistic features that can verify the authenticity of news content. This methodology cleverly combines traditional NLP techniques with sophisticated machine learning algorithms to significantly enhance the accuracy and efficiency of fake news detection systems. By improving the detection

process, the paper aims to bolster information integrity and credibility on digital platforms, addressing the pervasive challenge of misinformation in modern media landscapes. This research contributes to the ongoing efforts to safeguard democratic processes and journalistic integrity by providing reliable tools for ensuring the veracity of digital information[17].

“Detection of Sociolinguistic Features in Digital Social Networks for the Detection of Communities”

This paper delves into the sociolinguistic dynamics of digital social networks by employing methodologies from Natural Language Processing (NLP), Computational Linguistics, and Artificial Intelligence. Specifically, it examines the Twitter accounts of Colombian universities to map community structures based on language usage, reflecting community identity and cohesion. Utilizing a Design Science Research approach, the study emphasizes how specific linguistic characteristics within these digital interactions indicate community affiliations. Techniques such as the Weirdness Index and various community detection algorithms are pivotal in identifying distinctive linguistic features that are markers of community boundaries. This approach not only highlights the role of language in digital community formation but also illustrates how language use within these networks can be systematically analyzed to reveal deeper sociocultural and institutional affiliations, providing insights into the dynamics that underlie digital social interactions [18].

“A Sensitive Stylistic Approach to Identify Fake News on Social Networking”

In this paper Nicollas R. de Oliveira and his colleagues present a nuanced method for detecting fake news utilizing a combination of computational stylistics and natural language processing (NLP). The paper outlines an innovative approach by analyzing approximately 33,000 tweets, categorizing them into genuine and deliberately false categories to refine the detection techniques. Emphasizing the application of machine learning algorithms, the research integrates stylistic analysis to evaluate the textual content of social media posts. The study showcases impressive detection metrics, achieving 86% accuracy and 94% precision by implementing a dimensional reduction strategy, which reduces the original feature set to a sixth of its size, thereby minimizing computational overhead while maintaining high reliability in distinguishing between authentic and fabricated information. This work significantly contributes to the

understanding of fake news dissemination mechanisms on social media, providing a scalable solution that enhances the trustworthiness of digital information channels [19].

#### **1.5.4. Blockchain and deep learning in social networks:**

“A Blockchain-based Autonomous Decentralized Online Social Network”

The paper introduces a blockchain-powered framework for a decentralized OSN, emphasizing user autonomy and democratic management. Incorporating the Interplanetary File System (IPFS) for data storage and a decentralized autonomous organization (DAO) for governance, the model targets the drawbacks of centralized OSNs, such as privacy and security. This approach aims to offer a secure, user-controlled OSN solution, addressing the significant challenges faced by conventional social networks and proposing a novel method for ensuring user privacy and network management through blockchain technology [20].

“A decentralized social network architecture”

This paper proposes a novel approach to social media that prioritizes user data privacy and ownership through decentralization. It identifies centralization issues like privacy breaches and unauthorized data monetization in traditional social networks and introduces a solution leveraging blockchain technology, Ethereum for smart contracts, IPFS for decentralized storage, and Web3.0 technologies. This architecture aims to provide standard social networking features while ensuring data privacy, security, and ownership by decentralizing data control, and utilizing a distributed ledger for secure, transparent transactions, and interactions. By integrating these technologies, the proposed solution seeks to address the limitations and privacy concerns of existing centralized platforms, offering a user-centric alternative that empowers individuals over their data [21].

“A decentralized social networking architecture enhanced by blockchain”

The paper introduces a revolutionary decentralized social networking architecture that leverages blockchain technology to enhance user privacy and data control. It proposes an innovative system incorporating a sharding framework for scalability, a blockchain network for maintaining data integrity and consistency, and a reputation-based

authority control for enhanced security measures. This system is built on a peer-to-peer communication network, ensuring a trustless environment where data transactions are immutable and securely recorded on distributed nodes. By integrating these components, the architecture aims to overcome the privacy and security limitations of existing centralized and blockchain-enhanced social networks, offering users a more secure, efficient, and control-oriented social networking experience [22].

“A Privacy-Preserving-Framework-Based Blockchain and Deep Learning for Protecting Smart Power Networks”

The paper introduces a comprehensive privacy-preserving framework that leverages the synergy of blockchain and deep learning technologies to safeguard smart power networks from privacy threats and attacks. It establishes a two-tier privacy protection module alongside an anomaly detection mechanism. The initial tier utilizes an advanced blockchain infrastructure to ensure data integrity and thwart data poisoning attacks by authenticating meter nodes and preserving the authenticity of data chains. Concurrently, the second tier deploys a Variational Autoencoder (VAE) to transform data in a manner that significantly mitigates the risk of inference attacks, making it challenging for adversaries to glean sensitive information. Anomaly detection is orchestrated via a Long Short-Term Memory (LSTM) model, which proficiently discriminates between normal operations and potential threats, thereby enhancing the system's resilience to attacks. This methodology is substantiated through empirical validation on public datasets, demonstrating its superiority in data protection and anomaly detection over prevailing approaches. By amalgamating blockchain's robustness in data integrity, VAE's efficacy in data transformation, and LSTM's precision in anomaly detection, the framework presents a holistic and robust defense mechanism against a spectrum of privacy risks and security threats targeting smart power networks [23].

“An Enhanced Decentralized Social Network based on Web3 and IPFS using Blockchain”

The document outlines the development of a decentralized social network utilizing Web3 and IPFS technologies with blockchain to bolster privacy and security, addressing centralization problems of current social media platforms by proposing a system granting users greater data control. It employs Ethereum for blockchain

processes and IPFS for file storage, striving for a secure, scalable network. The described method integrates Ethereum for smart contracts, a genesis file to set the network's initial parameters, and web app development with Angular.js and web3.js for user interface and blockchain interaction. Additionally, it employs IPFS for decentralized storage, ensuring data availability. This multifaceted approach combines blockchain, smart contracts, decentralized storage, and web development to create a privacy-focused, user-centric social networking platform, aiming to overcome traditional platforms' limitations in privacy, security, and data ownership, while acknowledging challenges in scalability and user adoption, suggesting avenues for future research and enhancement [24].

“Fake Media Detection Based on Natural Language Processing and Blockchain Approaches”

The document discusses a sophisticated method for identifying fake media through the synergistic use of Natural Language Processing (NLP) and blockchain technologies. It introduces an integrated framework aimed at enhancing the detection of counterfeit news and user accounts on social media platforms by deploying advanced machine learning strategies, particularly reinforcement learning, alongside a decentralized blockchain system to safeguard digital content's security and authenticity. The proposal highlights the creation of a fake news prevention mechanism, the implementation of a proof of authority protocol for authentication purposes, and the application of reinforcement learning for predictive analytics. This research underscores the efficacy of melding NLP, blockchain, and machine learning to confront the complexities associated with fake media on social networks. The model amalgamates NLP for the initial processing and analysis of textual data, Deep Reinforcement Learning (DRL) for decision-making based on unstructured data to refine fake news detection capabilities, and blockchain technology to ensure the detection process's security and to verify news sources and outcomes transparently. By integrating these technologies, the model aims to exploit each one's strengths to effectively tackle the multifaceted problem of fake news dissemination within social media landscapes [25].

“Performance Evaluation of Decentralized Social Media on Near Protocol Blockchain”



The paper presents a comprehensive evaluation of decentralized social media performance on the Near Protocol Blockchain, with a focus on throughput and scalability. It introduces a prototype application featuring a fully decentralized architecture that combines the Near Protocol for storing states and the InterPlanetary File System (IPFS) for storing post content. However, it was found that directly storing post content on the blockchain hindered scalability. The solution proposed involves utilizing IndexedDB for off-chain storage of post content, which significantly improved response times and scalability. This hybrid approach, leveraging both on-chain and off-chain storage strategies, was suggested to enhance the performance of decentralized social media applications. The prototype, which includes functionalities for posting content and managing friend connections, also features a Web 3.0 graphical user interface for ease of user interaction. Performance was assessed by measuring transaction response times via the NEAR command line interface, calculating throughput as the number of successful transactions per second, and evaluating scalability with a specific formula, focusing on text string content for this study [26].

“SentiTrust: A New Trust Model for Decentralized Online Social Media”

The paper presents SentiTrust, a groundbreaking trust model for decentralized online social media, which employs AI-powered sentiment analysis to meticulously evaluate interpersonal trust among users through direct social interactions. This innovative model is distinguished by its use of social features, such as the number of common friends and interaction evaluations, for estimating trust levels, and it incorporates sentiment analysis to provide a richer, more nuanced trust assessment. Designed to be adaptable across various decentralization technologies, SentiTrust is part of the HELIOS project, aiming to forge a contextually aware, privacy-conscious social network. The model's trust computation framework, informed by properties indicating that trust is continuous, dynamic, non-transitive, asymmetrical, subjective, and context-dependent, uses the Neuro-Behavioral Module (NBM) for analyzing textual and graphical content. This approach allows for the integration of optional features like profile similarity and common friends to refine trust evaluations. Tested with real participants, SentiTrust demonstrates promising results in evaluating positive interactions and multimedia exchanges, showcasing its potential to significantly

influence trust computations in decentralized social networks by mirroring real-world human relationships [27].

## **1.6. Organization**

This thesis is meticulously structured into several chapters, each meticulously crafted to elucidate distinct facets of designing and implementing a blockchain social network, augmented by deep learning and natural language processing. The organization of the thesis is as follows:

1. **Introduction:** This initial chapter sets the context for the research by delineating the topic, articulating the problem statement, outlining the research objectives, and formulating the research questions. It provides an exposition of the prevailing challenges inherent in traditional social networks and introduces the innovative solutions posited through the integration of blockchain, deep learning, and natural language processing technologies.
2. **Literature Review:** The subsequent chapter systematically reviews the extant literature, examining the roles and impacts of blockchain technology, deep learning, and natural language processing within the domain of social networks. It critically analyzes previous studies, identifies gaps in the current technological landscape, and establishes the theoretical framework underpinning the thesis.
3. **Theoretical Background:** This chapter delves into the foundational theories and technologies of blockchain, deep learning, and natural language processing. It elucidates the principles, operational mechanisms, and recent advancements in each area, providing a robust background necessary for understanding their applicational synergy in social networks.
4. **Implementation:** This pivotal chapter describes the practical implementation of the proposed blockchain-based social network. It details the system architecture, data structures, functional capabilities, and the integration of deep learning and natural language processing to enhance privacy measures and user engagement within the network.

5. **Application:** In this chapter, the application of the developed system is showcased, including comprehensive details on the deployment environment, user interface design, and interactions within the blockchain network.
6. **Results and Discussion:** The penultimate chapter presents a critical evaluation of the system's performance. It assesses the efficacy of the blockchain network, the deep learning models, and the NLP functionalities deployed. The discussion extends to the implications of these technologies on privacy, data security, and user empowerment within social networks.
7. **Conclusion and Future Work:** The concluding chapter synthesizes the findings, discusses the limitations of the current study, and proposes avenues for future research. It contemplates the potential broader impacts of the thesis work on reshaping social networking practices and advancing digital communication.

Each chapter is logically sequenced to build upon the insights gained from the preceding sections, ensuring a coherent flow of information throughout the thesis. This structured approach facilitates a comprehensive understanding of the integration and challenges of blockchain, deep learning, and natural language processing within social networks, thereby enhancing the academic rigor of the study.



## **2. THEORETICAL BACKGROUND**

### **2.1. Blockchain**

#### **2.1.1. Whats blockchain**

Blockchain appears a revolutionary technology that functions as a decentralized digital ledger. It continually grows by adding a sequence of records, known as 'blocks', across a widespread network of computers. Every block is securely connected to the one before it through cryptographic hashes. This connection creates a traceable path, guaranteeing the transparency and unchangeability of the data [28]. Beyond mere transaction recording, blockchain's attributes like decentralization, dependability, and irrefutability raise its status to an innovative level in managing data securely and privately [29]. The advantages of blockchain technology go far beyond its primary use in cryptocurrencies such as Bitcoin and Ethereum Its versatility and application reach far and wide, transcending its initial association with digital currencies. [28]. Its applications permeate multiple industries, encompassing finance, supply chain oversight, and the authentication of documents [28]. The distinguishing feature of blockchain lies in its capacity to engender trust amongst users without relying on a central entity or middlemen, thereby guaranteeing that the data preserved is uniform and invulnerable to alteration throughout the network of participants [30].

#### **2.1.2. Blockchain working principle:**

The operational essence of blockchain technology unfolds through a harmonious synergy of decentralization, transparency, and cryptographic security. At its heart lies a distributed ledger system, a platform crafted not just for recording transactions but also for embedding trust into every digital interaction [31].

When we delve into the specifics of its functioning, we witness a transaction initiation sparking a cascade of events. This transaction, once broadcast, is not left to fend for itself. Instead, it is embraced by a network of nodes, each a custodian of the blockchain's integrity. These nodes, operating on a consensus mechanism-be it proof of work or proof of stake-collectively validate the transaction. Upon achieving consensus, the transaction is sealed within a block [31].

Each block is a chapter in a larger saga, woven together through cryptographic hashes. Cryptographic hashes serve as the lifeblood of continuity and integrity within the blockchain, with each block carrying not just its own distinct hash, but also a nod to its forebear's hash. This creates an inseparable linkage, a perpetual chain of blocks tracing all the way back to the blockchain's very origin [31].

The inherent strength of blockchain lies in its distributed structure, operating without a centralized governing body. Its robustness stems from its network of nodes, each serving as a vault for the complete blockchain, perpetually synchronizing and safeguarding the ledger. This sophisticated symphony of technology ensures that the information within the blockchain is not only protected and transparent but also permanent and reliable. Blockchain stands as a beacon of innovation, poised to transform current systems and pave new digital pathways [31].

### **2.1.3. Blockchain architecture**

The structure of blockchain is like a finely crafted mosaic, composed of three essential elements: blocks, chains, and the network [30]. Picture each block as a digital record keeper, storing transaction details, time stamps, and its own distinctive cryptographic code, referred to as a hash [29] [30]. This hash not only seals the contents within but also serves as an indelible identifier [29]. These blocks are then chronologically linked in a chain through their hashes, each block securely connected to its predecessor, forming an unbreakable sequence that traces its lineage back through time [29]. Completing this architectural triad is the network, a global coalition of nodes [29]. Each node acts as a vigilant sentinel, validating new blocks and transactions [29]. Their collective consensus is the bulwark against any unauthorized changes, a testament to the decentralized nature of blockchain [29] [30]. This triad-blocks, chains, and the

network-unites to create the resilient and transparent architecture that is the hallmark of blockchain technology [29].

#### **2.1.4. Consensus mechanism**

Consensus mechanisms are fundamental protocols within blockchain technology, crucial for establishing uniform agreement on the validity and sequence of transactions across all participants. These mechanisms not only ensure transaction integrity but also influence the overall network security and resource consumption[32].

- **Proof of Work (PoW):** Predominantly utilized by cryptocurrencies like Bitcoin and Ethereum 1.0, PoW requires nodes to solve cryptographic puzzles, termed mining, to form new blocks. The mechanism dictates that transaction fees fluctuate based on transaction supply and demand dynamics, with higher fees enticing quicker mining prioritization. Despite its widespread adoption, PoW is resource-intensive, requiring substantial computational power which can lead to inefficiencies and high energy costs. The cost-effectiveness of mining under PoW is directly influenced by the cryptocurrency's market value relative to operational expenses, often rendering the mining process non-viable if potential rewards do not justify the energy costs[32].
- **Proof of Stake (PoS):** This mechanism stands out for its low energy requirements compared to PoW. In PoS, the likelihood of a node being chosen to validate and create new blocks is proportional to its stake in the network, typically represented by the amount of cryptocurrency held and "staked" by the node. This method significantly reduces operational costs and, consequently, transaction fees. Cryptocurrencies such as Ethereum 2.0, Cardano, Solana, and Polkadot employ PoS, highlighting its scalability and cost-efficiency benefits[32].

Further advancements in blockchain technology have led to the development of other consensus models like Delegated Proof of Stake (DPoS) and Practical Byzantine Fault Tolerance (PBFT), among others. These include Proof of Elapsed Time, Proof of Weight, Proof of Burn, Proof of Capacity, and Proof of Space. Although innovative, these models have not reached the same level of adoption as PoW and PoS. Each consensus protocol comes with unique advantages and trade-offs in terms of

decentralization, security, and scalability, necessitating careful consideration based on the specific requirements and objectives of the blockchain application in question[32].

### **2.1.5. Blockchain characteristics**

Blockchain technology is very valuable for many reasons. It lets people do transactions and store data in a safe and simple way. It does not have a central authority that controls it, which makes it harder for hackers and fraudsters to attack it. This is very important especially in fields like finance where people need to trust and see what is happening. Also, blockchain makes things faster and cheaper by cutting out middlemen, which saves money and time. In things like building and funding projects, blockchain brings a better way of managing everything that is more effective, secure, and clear.

Key advantages include:

- **Decentralization:** The absence of a central authority in blockchain technology ensures enhanced security and reduced susceptibility to attacks or system failures [29].
- **Transparency:** Blockchain offers a transparent and verifiable record of transactions, fostering increased trust and accountability across various sectors [29].
- **Immutability:** Once recorded on the blockchain, transactions are permanent and unmodifiable, guaranteeing data integrity [29].
- **Security:** Cryptographic algorithms employed by blockchain safeguard transactions and prevent unauthorized access or alterations [29].
- **Efficiency:** By bypassing intermediaries, blockchain enables quicker, more cost-effective transactions with lower fees [29].
- **Anonymity:** Blockchain provides anonymous transaction capabilities, negating the need for third-party involvement in centralized systems [29].
- **Innovation:** As a rapidly developing field, blockchain technology holds the potential to transform various industries and spawn new business models [29].

### **2.1.6. Blockchain areas**

- **Cryptocurrencies:** Blockchain is the backbone of digital money like Bitcoin, Ethereum, and Litecoin. It gives a safe and distributed way to do transactions [29].



- **Financial Services:** Blockchain is used for many services in the money sector, such as sending money across borders, funding trade, and settling deals. This helps to cut costs, improve work efficiency, and increase security [31].
- **Supply Chain Management:** Blockchain technology is very important for making supply chains more transparent and accountable. It lowers fraud and boosts efficiency by carefully tracking products and checking their quality.
- **Healthcare:** Blockchain offers a fortified platform for the exchange of patient information among medical professionals, with the potential to better patient care outcomes and decrease operational costs [31].
- **Real Estate:** Blockchain technology makes real estate deals easier, giving a safe and clear way to record and transfer property rights, which reduces fraud and improves efficiency [31].
- **Identity Verification:** Blockchain's use in identity verification makes security better and lowers fraud in different industries. It lets people store and share their personal identity information safely, without needing a central authority [31].
- **Voting Systems:** Blockchain can make voting systems more secure and clear, making sure that elections are fair [29].
- **Internet of Things (IoT):** In the world of IoT, blockchain enables small transactions between devices, creating a safe and distributed platform for IoT payments and markets [29].
- **Identity Management:** Blockchain gives a strong and distributed way to manage identity, greatly lowering the chance of identity theft and other frauds [29].

### **2.1.7. Blockchain advantages and disadvantages**

#### **2.1.7.1. Advantages of blockchain technology:**

1. **Elimination of Trusted Intermediaries:** Blockchain can potentially remove the need for trusted third parties in transactions involving value transfer, ensuring parties act as agreed upon without an intermediary [31].
2. **Versatility in Applications:** It can help mitigate counterfeit goods in retail, facilitate data sharing in healthcare, streamline credential verification in academia, and more [31].

3. **Secure and Transparent Data Management:** Blockchain provides a secure, transparent way to store and transfer data [31].
4. **Decentralization:** Distributes authority among all nodes, ensuring system redundancy and reducing failure risk [29].
5. **Immutability:** Transactions are permanent and distributed among nodes, making them unalterable and preserving data integrity [29].
6. **Enhanced Security:** The decentralized nature and cryptographic measures safeguard against unauthorized alterations and enhance overall security [29].

#### **2.1.7.2. Disadvantages of blockchain technology:**

1. **Nascent Stage Challenges:** Being in the early stages, blockchain faces numerous technical and regulatory hurdles [31].
2. **Performance and Cost Issues:** It can be slow and costly, particularly for large-scale applications [31].
3. **Vulnerability to Threats:** Even with stringent security protocols in place, blockchain isn't impervious to cyber threats and potential attacks [31].
4. **Issues with Scalability:** Present blockchain solutions face constraints in scalability, causing transactions to lag and fees to escalate [29].
5. **High Energy Demand:** The computational power needed for transaction validation results in significant energy consumption [29].
6. **Regulatory Ambiguity:** A lack of clear regulation can lead to security concerns, fraud, and legal issues such as data privacy and intellectual property rights [29] [30].
7. **Scalability Issues:** Like the increasing nodes in the network can lead to slower validation times and related scalability challenges [30].
8. **Legal and Regulatory Hurdles:** The implementation of blockchain may intensify the challenges related with compliance, anti-money laundering, and adhering to know your customer regulations [30].

## **2.1.8. Blockchain types**

### **2.1.8.1. Public blockchains:**

Public blockchains are completely open networks where anyone can participate, such as Bitcoin and Ethereum. These blockchains are unique because they are distributed, meaning no single organization controls the network. They are clear, as all the transactions are visible to everyone, and they are safe, using math and rules like Proof of Work (PoW) or Proof of Stake (PoS) to secure the network. Also, these blockchains are inclusive, letting anyone take part in the decision-making process. Public blockchains are commonly used for digital money and apps that work on the network (DApps) [33].

### **2.1.8.2. Private blockchains:**

A private blockchain network is controlled by a single organization or company, often used within an organization or enterprise. These have restricted access, centralized control over reading, writing, or auditing the blockchain, and are generally more efficient and faster than public blockchains due to fewer participants in the consensus process. Use cases include supply chain management, internal auditing, and enterprise resource planning [33].

### **2.1.8.3. Consortium blockchains:**

Consortium blockchains are not fully decentralized and are run by a bunch of organizations instead of one. The blockchain is controlled by everyone, more flexible, and quicker than public blockchains, giving a mix of the strong safety of public blockchains and the power of private blockchains. They are helpful in money and business, and sharing data between different organizations [33].

### **2.1.8.4. Hybrid blockchains:**

Hybrid blockchains combine between of both private and public blockchains, offering a balanced approach with controlled access and privacy while maintaining the security and transparency of public blockchains. They feature selective transparency, controlled participation, and flexibility for organizations to set up the blockchain to

suit specific requirements. Use cases include real estate transactions, voting systems, and supply chain management [33].

### **2.1.9. Smart contract**

A key include of blockchain innovation, shrewd contracts mechanize the execution of contracts and guarantee compliance without the require for outside applications. The terms of these self-executing contracts are composed straightforwardly in code and run on stages such as Ethereum and Hyperledger Texture, each of which offers distinctive capabilities, such as: Ethereum's Turing-complete programming environment and Fabric's permissioned arrange capabilities. Keen contracts are imperative in a assortment of businesses, counting back, healthcare, and supply chain, since they streamline operations by expelling brokers, lessening costs, and expanding straightforwardness in exchanges. Within the budgetary segment, they oversee complex monetary understandings extending from subsidiaries to protections claims, minimizing the chance of extortion. They are useful to the healthcare industry, as they empower straightforward trades between substances whereas guaranteeing the keenness and security of delicate understanding information. In genuine domain, these contracts robotize resource deals, and lease installments, essentially shortening customarily time-consuming verification forms. Within the supply chain industry, keen contracts are broadly utilized to progress traceability and responsibility by consequently upgrading records after conveyance, subsequently decreasing debate In contracts and misfortunes. Be that as it may, challenges such as security vulnerabilities, execution bottlenecks, and uncertain administrative suggestions stay major impediments. High-profile occurrences such as the DAO assault have highlighted these vulnerabilities and started continuous investigate pointed at making strides the security and effectiveness of keen contracts through strategies such as key approval modes and execution optimizations. The independent nature of shrewd contracts not as it were diminishes authoritative burden but too guarantees that exchanges are executed agreeing to predefined rules, bringing modern levels of unwavering quality and proficiency to trade exercises. As innovation propels, keen contracts are anticipated to become indeed more effective, giving dependable and adaptable arrangements for any industry, whereas empowering modern developments

that offer assistance make forms cheaper, quicker, and compliant with administrative prerequisites[34].

## **2.2. Deep Learning**

### **2.2.1. Deep learning overview**

DL remains a powerful force in various fields, utilizing intricate artificial neural networks to analyze large amounts of data. DL excels in identifying patterns and features in unstructured data, leading to breakthroughs in computer vision, NLP, and autonomous systems [35]. Its hierarchical structures mimic how the human brain processes information, enabling automatic learning of complex features across multiple abstraction levels for tasks like image recognition and predictive analytics. DL's progress relies on increased computational power and vast datasets for training DNNs. Challenges include the need for substantial computational resources, model interpretability concerns, and potential overfitting risks. Recent innovations like transformer models for NLP and advancements in GANs illustrate the rapid development in the field [36].

### **2.2.2. Deep learning areas**

#### **2.2.2.1. Healthcare**

Deep Learning has transformed healthcare by enhancing diagnostics, personalizing medicine, and improving treatment efficiency.

- **Disease Detection:** DL models like Convolutional Neural Networks (CNNs) have shown proficiency in identifying diseases from medical images. For example, CNNs have been successfully utilized for skin cancer classification using dermatoscopic images [37].
- **Drug Discovery:** DL methods have been used to forecast molecular bioactivity, enhance drug characteristics, and develop new drug options. Apps like Generative Adversarial Networks (GANs) are also being used to speed up drug discovery for COVID-19 [36].

#### **2.2.2.2. Autonomous vehicles**

Deep learning plays a vital role in assisting autonomous vehicles by enabling them to perceive their environment and make decisions on the road if something happens.

- **Image and Sensor Data Interpretation:** DL models are used to process and interpret vast amounts of data from cameras and sensors, enhancing vehicle perception [38].
- **Path Planning:** DL algorithms help in predicting pedestrian movement and optimizing routes [39].

#### **2.2.2.3. Natural language processing (NLP)**

DL has led to breakthroughs in understanding and generating human language.

- **Machine Translation:** Neural Machine Translation (NMT) models have significantly improved the quality of automated translation [35].
- **Sentiment Analysis:** DL models analyze text data to determine sentiment, aiding in market analysis and social media monitoring [40].

#### **2.2.2.4. Computer vision**

DL models, especially CNNs, have become the backbone of computer vision tasks.

- **Facial Recognition:** DL has improved the accuracy and reliability of facial recognition systems, with applications in security and authentication [41].
- **Object Detection:** DL models identify and locate objects within images with high precision [42].

#### **2.2.2.5. Finance**

DL models are employed in predicting market trends and detecting fraudulent transactions.

- **Fraud Detection:** Deep learning algorithms help in identifying patterns indicative of fraudulent activity [43].
- **Algorithmic Trading:** DL models are used for predicting stock prices and optimizing trading strategies [44].

### **2.2.3. The advantages and disadvantages of the deep learning**

#### **2.2.3.1. Advantages of deep learning**

- **Superior Data Handling and Feature Extraction:** DL models excel in managing large datasets, automatically identifying complex patterns and features without the need for manual extraction. This capability has been pivotal in enhancing image and speech recognition technologies [45].
- **Improved Accuracy with Large Datasets:** As the amount of training data increases, DL models tend to improve in accuracy and efficiency, outperforming traditional machine learning models in tasks such as predictive analytics and pattern recognition [46].
- **Versatility Across Domains:** DL has found applications in a wide array of fields, demonstrating its versatility. From healthcare diagnostics to autonomous vehicle navigation, DL models have been instrumental in driving innovation and solving complex problems [47].

#### **2.2.3.2. Disadvantages of deep learning**

- **High Resource Consumption:** Training DL models requires substantial computational resources, including powerful GPUs and large amounts of memory, making it inaccessible for some organizations and researchers [48].
- **Dependency on Large Datasets:** The performance of DL models is heavily reliant on the availability of large, annotated datasets. The scarcity of such datasets in certain domains can limit the applicability and effectiveness of DL [46].
- **Lack of Interpretability:** DL models, especially those with numerous layers, are often considered "black boxes" due to their complex nature, making it challenging to understand how decisions are made. This lack of interpretability can be a significant drawback in fields requiring transparency and accountability, such as healthcare and criminal justice [49].

#### **2.2.4. LSTM**

Long Short-Term Memory (LSTM) networks are a specialized form of Recurrent Neural Networks (RNNs) that are designed to address the vanishing gradient problem associated with standard RNNs. This problem typically arises during the training of

traditional RNNs, where gradient values diminish as they are propagated back through each time step of the input sequence, making it challenging to capture long-range dependencies within the sequence data[50]

The core innovation in LSTM networks is the introduction of a memory cell that effectively maintains information over extended sequences. Each LSTM unit comprises three distinct gates: the input gate, the forget gate, and the output gate, which collectively regulate the flow of information into and out of the cell, as well as the retention of information over time[50].

The operations within an LSTM cell can be mathematically represented as follows:

1. Forget Gate:

$$f_t = \sigma(W_{f_h}[h_{t-1}], W_{f_x}[x_t], b_f) \quad (2.1)$$

This gate decides which information is discarded from the cell state.

2. Input Gate:

$$i_t = \sigma(W_{i_h}[h_{t-1}], W_{i_x}[x_t], b_i) \quad (2.2)$$

$$c_t^{\sim} = \tanh(W_{c_h}[h_{t-1}], W_{c_x}[x_t], b_c) \quad (2.3)$$

It determines what new information is added to the cell state.

3. Update to Cell State:

$$c_t = f_t * c_{t-1} + i_t * c_t^{\sim} \quad (2.4)$$

This equation updates the cell state by blending old and new information.

4. Output Gate:

$$o_t = \sigma(W_{o_h}[h_{t-1}], W_{o_x}[x_t], b_o) \quad (2.5)$$

$$h_t = o_t * \tanh(c_t) \quad (2.6)$$

The output gate determines what part of the cell state is passed to the output.

### 2.2.5. BiLSTM

Bidirectional LSTM (BiLSTM) networks extend the traditional LSTM architecture by processing the data in both forward and reverse directions with two separate hidden layers, which are then fed forwards to the same output layer. This bidirectional



structure allows the networks to have both backward and forward information about the sequence at every point in time[50].

And there are mathematical representations For the BiLSTM:

- Forward LSTM:

$$h_t \rightarrow = LSTM(x_t, h_{t-1} \rightarrow) \quad (2.7)$$

- Backward LSTM:

$$h_t \leftarrow = LSTM(x_t, h_{t+1} \leftarrow) \quad (2.8)$$

- Output Concatenation:

$$h_t = [h_t \rightarrow; h_t \leftarrow] \quad (2.9)$$

This represents the concatenation of the forward and backward LSTM outputs.

The utilization of both past and future input information allows BiLSTMs to capture context more effectively than traditional unidirectional LSTMs. This makes BiLSTMs particularly useful for applications where context from both directions is crucial, such as time series prediction[50].

### 2.2.6. Comparative analysis of LSTM and BiLSTM

Empirical studies, as highlighted in the document, demonstrate that BiLSTM networks often outperform their LSTM counterparts in tasks involving complex dependencies, such as financial time series forecasting. The bidirectional approach of BiLSTMs, which leverages information from both past and future contexts, contributes to a more robust model with improved prediction accuracy[50].

This chapter of the thesis provides a foundational understanding of LSTM and BiLSTM architectures, illustrating their relevance and superiority in handling long-range dependencies within sequential data, particularly in the context of time series forecasting. The subsequent sections will explore practical implementations and specific case studies where these models have been effectively applied[50].

### 2.2.7. Blockchain and DL in social networks:

Blockchain technology and deep learning are playing increasingly pivotal roles in enhancing the functionality and security of social networks. Blockchain, with its

decentralized and immutable ledger system, ensures the integrity and transparency of transactions and interactions on social platforms. It mitigates risks such as data tampering and unauthorized access, thereby fostering trust among users [7]. On the other hand, deep learning, a subset of machine learning, leverages complex neural networks to analyze vast amounts of data generated by social network users. This analysis helps in personalizing user experiences, optimizing content delivery, and improving targeted advertising, all while maintaining user identity [8].

The integration of blockchain and deep learning in social networks addresses critical challenges, including data privacy, and fake news dissemination [8]. Blockchain's transparent and secure environment, combined with deep learning's predictive capabilities, enhances content authenticity and user privacy [7]. However, this integration is not without its hurdles. Deep learning models, often seen as "black boxes," pose challenges in interpretability and transparency, which are crucial for user trust and regulatory compliance [49]. Additionally, the computational intensity of deep learning models demands significant processing power, potentially limiting their deployment on less powerful devices.

## **2.3. Natural Language Processing (NLP)**

### **2.3.1. Whats the NLP**

Natural Language Processing (NLP) is a branch of Artificial Intelligence and Linguistics focused on enabling computers to understand statements or words written in human languages. It was developed to simplify user interactions with computers, allowing communication in natural language without the need for users to learn specialized computer languages. NLP is divided into two main areas: Natural Language Understanding (NLU), which involves the comprehension and interpretation of human language, and Natural Language Generation (NLG), which concerns the production of text that is meaningful to humans. NLP covers various aspects of language, including phonology (sound), morphology (word formation), syntax (sentence structure), semantics (meaning), and pragmatics (contextual understanding) [51].

### **2.3.2. How NLP works**

NLP involves several steps and techniques to process and understand natural language:

- **Preprocessing:** This includes cleaning and normalizing text data through tokenization, stemming, lemmatization, and removing stop words [52].
- **Feature Extraction:** Techniques like Bag of Words, TF-IDF, and word embeddings are used to convert text into a form that can be processed by machine learning models[52].
- **Model Training:** Machine learning and deep learning models are trained on preprocessed text data for various tasks such as classification, sentiment analysis, named entity recognition, and more[52].
- **Evaluation and Application:** The performance of NLP models is evaluated using metrics like accuracy, precision, recall, and F1 score. Once trained and evaluated, these models are deployed to perform the intended NLP tasks in real-world applications[52].

### **2.3.3. NLP benefites**

NLP is used to bridge the gap between human communication and computer understanding, allowing humans to connect with computers through natural language. This interaction enables multiple applications in a variety of domains, including machine translation, email spam detection, information extraction, summarization, medical applications, and question answering, among others. NLP technologies are intended to improve the efficiency of linguistic data processing, facilitate decision-making processes, and make information more accessible. NLP automates operations requiring human language understanding by transforming natural language into a form that computers can understand, saving time and minimizing the need for specialized expertise. The usage of NLP greatly contributes to the improvement of human-computer interaction, allowing for more natural and intuitive ways of communication with digital systems and applications [51].

### **2.3.4. NLP advantages and disadvantages:**

Natural Language Processing (NLP) stands as a pivotal branch of artificial intelligence that focuses on the interaction between computers and humans through the natural

language. The aim is to read, decipher, understand, and make sense of the human languages in a manner that is valuable. Here are the advantages and disadvantages of NLP:

#### **2.3.4.1. Advantages:**

- **Enhanced Communication:** NLP facilitates more intuitive interactions between computers and humans, improving user experience through more natural human-computer interaction interfaces[53].
- **Automation and Scalability:** It allows for the automation of numerous tasks such as data entry, translation, and customer service, offering significant improvements in efficiency and scalability of operations[53].
- **Sentiment Analysis:** NLP excels in analyzing the sentiments behind vast amounts of text data through social media, reviews, and feedback, enabling businesses to gauge public opinion and customer satisfaction more effectively[52].
- **Error Reduction:** By automating translation and data interpretation tasks, NLP reduces the potential for human error, contributing to more accurate data handling and decision-making processes[53].
- **Accessibility:** It enhances accessibility, particularly through applications like speech recognition systems, which assist individuals with disabilities in interacting with technology seamlessly[53].

#### **2.3.4.2. Disadvantages:**

- **Complexity of Language:** NLP systems often struggle with the complexities and nuances of human language, including slang, irony, and context-dependent meanings, leading to misunderstandings and inaccurate interpretations[52].
- **High Resource Requirements:** Developing proficient NLP systems requires extensive computational resources and large datasets, which can be costly and resource-intensive[53].
- **Maintenance and Adaptation:** Language evolves continuously, and NLP systems require ongoing updates and maintenance to adapt to new languages, dialects, or changes in speech patterns[53].
- **Bias and Ethical Concerns:** NLP can inadvertently propagate bias if the training data contains bias. This is particularly concerning in scenarios involving sentiment

analysis or predictive typing, where biased outputs might affect decision-making processes or perpetuate stereotypes[53].

- **Data Privacy Issues:** NLP often processes sensitive personal information, leading to potential risks and concerns related to data privacy and security[53].

## 2.4. Traditional Social Networks

Online social networks (OSNs) have revolutionized the way individuals connect, communicate, and share information. These platforms, such as Facebook, Twitter, LinkedIn, and YouTube, provide users with the ability to express their thoughts, voice their opinions, and engage with others globally. OSNs have facilitated the creation of virtual communities where members can share common interests, discuss various topics, and form connections that often transcend geographical boundaries[54].

One of the defining characteristics of OSNs is their heterogeneous nature. They encompass multiple types of entities, including users, content, tags, and interactions, which are interwoven into a complex network. For instance, in content-sharing platforms like Flickr and YouTube, users can upload media, tag content, comment on posts, and form friendships or follow other users. This multifaceted interaction fosters a rich environment for human interaction and collective behavior on a large scale[54]

The evolution of OSNs has seen them grow from niche platforms to significant components of the internet landscape. With millions of active users, these networks are responsible for a substantial portion of web traffic and user engagement. They not only provide social connectivity but also serve as platforms for information dissemination, entertainment, and even business opportunities. Major social networking sites have integrated social capabilities to enhance user experience and leverage the power of massive user bases[54].

The academic interest in OSNs has kept pace with their growth, as researchers seek to understand the structure, dynamics, and implications of these networks. Studies have explored various aspects such as the formation and evolution of communities, the role of key influencers, and the impact of social interactions on behavior. These investigations often utilize large datasets from platforms like Flickr and Yahoo! 360, analyzing the temporal aspects of network growth and user engagement[54].

Social networks, especially in the context of online platforms like Facebook, Twitter, Instagram, and LinkedIn, consist of several key components that allow them to function as vast, interconnected systems where users can share information, interact, and create content[54].

#### **2.4.1. Fundamental elements for social network**

##### **1. Data Centers**

Social networks rely on data centers equipped with high-performance servers that store vast quantities of data, including user profiles, multimedia content (photos, videos, posts), and activity logs. These data centers are strategically situated across the globe to ensure redundancy, reliability, and swift data access. The scalability and availability of these networks are paramount as they must accommodate millions, sometimes billions, of users simultaneously. Techniques such as load balancing, distributed databases, and cloud computing are pivotal in managing the load and maintaining the platform's availability around the clock without disruptions[55].

##### **2. Networking Infrastructure**

The global accessibility of social networks necessitates a robust networking infrastructure capable of efficiently managing large volumes of data traffic. This infrastructure includes routers, switches, and other networking devices essential for data transmission. Additionally, Content Delivery Networks (CDNs) play a critical role in distributing the load of delivering content such as images and videos to users rapidly, irrespective of their geographic location, by caching content in multiple locations worldwide[55].

##### **3. Databases**

Social networks employ sophisticated database systems for storing a wide array of data, from user information to user-generated content and interaction data. These databases are optimized for fast read and write operations and are often distributed to improve performance and fault tolerance. Big Data technologies such as Hadoop, Spark, and NoSQL databases are frequently utilized to manage and analyze large datasets, which aids in personalizing content recommendations, targeting advertising, and enhancing user experience[55].

##### **4. Application Servers**

The backbone of a social network's functionality is its application servers. These servers execute the backend logic, processing user requests, serving dynamic content, and managing critical operations like user authentication, news feed generation, and notification dispatching. Application Programming Interfaces (APIs) further extend the network's functionality by allowing integration with third-party applications and services, thereby enriching the user's experience beyond the social network's core offerings[55].

#### 5. User Interface

Social networks offer user interfaces through web browsers and mobile apps, enabling users to interact with the network. These interfaces are designed to be user-friendly, responsive, and provide a seamless experience across various devices and platforms. This approach ensures that all users, regardless of their device type or operating system, have a consistent and engaging interaction with the platform[55].

#### 6. Security Systems

Data security is a cornerstone of social networks due to the sensitivity of the personal data they handle. Robust security measures, including encryption, secure authentication methods, and regular security audits, are crucial to protect user data and prevent unauthorized access. Privacy controls are also vital, allowing users to manage the visibility of their information and posts, thus safeguarding their privacy within the digital space[55].

#### 7. Analytics and Monitoring Systems

Analytical systems within social networks scrutinize user behavior to tailor content, suggest connections, and serve targeted advertisements effectively. System performance monitoring tools are critical in ensuring the network's health and performance by swiftly identifying and resolving any potential issues. This continuous monitoring helps maintain the quality of service and ensures the platform remains reliable and efficient for its users[55].

### **2.4.2. Online social network disadvantages**

#### 1. Loss of Information Control

Users may lose control over their personal information, which can be accessed and shared by third parties, such as advertisers. This includes profile information and

user-generated content that can be exploited through well-developed application programming interfaces (APIs)[5].

## 2. Exploitation of Social Connections

There is a risk of exposing information of close ties. Firms might intercept and exploit exchanges between connected contacts, potentially leading to breaches of communication privacy[5].

## 3. Location Tracking

Social networks often use geospatial technology to collect location data. This can lead to confidentiality issues, as both travel history and real-time positions of individuals may be disclosed. Organizations can pinpoint exact user locations and reach them, which raises concerns about surveillance and intrusion[5].

## 4. Biometric Data Risks

The collection and sharing of biometric data (physiological and behavioral data) pose severe risks, including lack of control over highly sensitive information, potential identity theft, stalking, and emotional manipulation[5].

## 5. Web Tracking

Technologies like cookies and web beacons can create extensive profiles of users by tracking their activities across multiple websites, often without their knowledge or consent. This can lead to significant privacy invasions and the risk of sophisticated manipulation through predictive analytics[5].

## 6. Data Aggregation and Profiling

The aggregation of data from multiple sources can reveal highly sensitive attributes such as sexual orientation, age, and political views. This comprehensive profiling increases individual vulnerability and susceptibility to targeted marketing and other forms of exploitation[5].

## 7. IoT and Smart Devices

The integration of Internet of Things (IoT) devices increases the risk of unauthorized data access and cyberattacks. These devices can capture and transmit



sensitive information, including communications between users, leading to significant privacy concerns[5].

These privacy tensions are compounded by the inherent difficulty in controlling the flow of information once it is shared on social networks, leading to significant concerns about individual, information, and communication privacy[5].

## **2.5. Datasets**

### **2.5.1. Posts dataset**

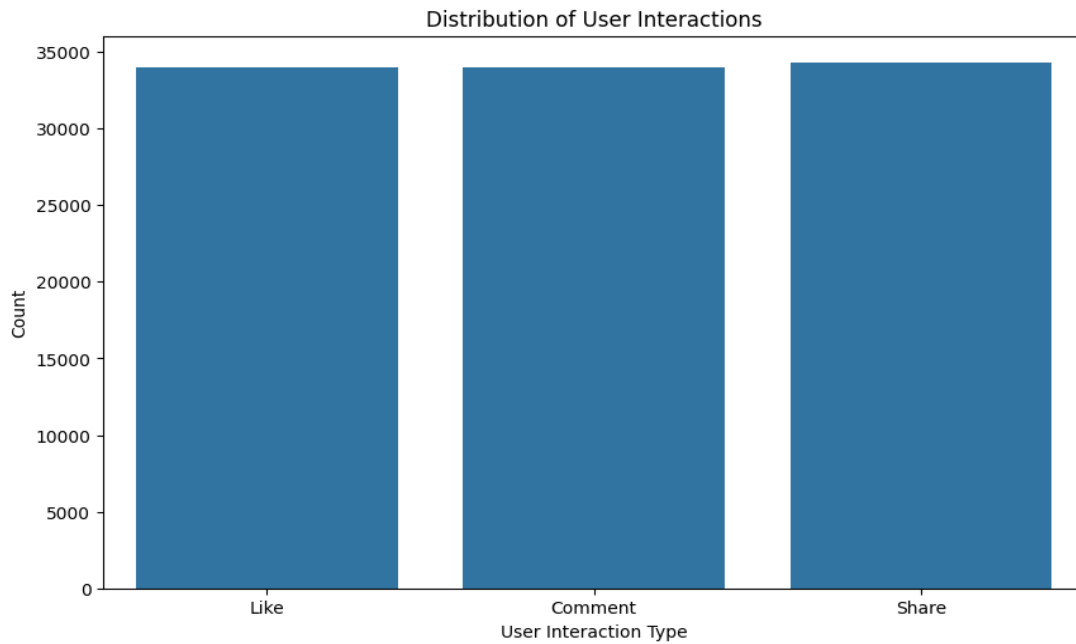
For the development of our study aimed at unraveling the nuances of user engagement across social media platforms, we meticulously curated a dataset consisting of 102,282 entries. Each entry serves as a vital piece of the puzzle in deciphering the complex interplay between content type and user interaction on these digital platforms. Our dataset is structured into three principal columns, each offering unique insights into the social media landscape:

#### **2.5.1.1. Post content**

This segment of the dataset captures a rich tapestry of social media posts, totaling 74,461 unique entries. These entries span a wide range of subjects and activities, reflecting the diversity of user interests and expressions on social media. From discussions on the latest video games to insights into news and current events, each post is a window into the myriad ways individuals share and consume content online.

#### **2.5.1.2. User interaction type**

As shown in Figure 2.1, the nature of user engagement with the posts is categorized into six distinct types, including 'Like', 'Comment', 'Share', among others. This extended classification allows us to delve deep into the anatomy of user interaction, revealing the predominance of 'Shares' as the most frequent form of engagement. Understanding these interaction patterns is pivotal in gauging the immediate appeal of various types of content and in modeling the dynamics of social media engagement.

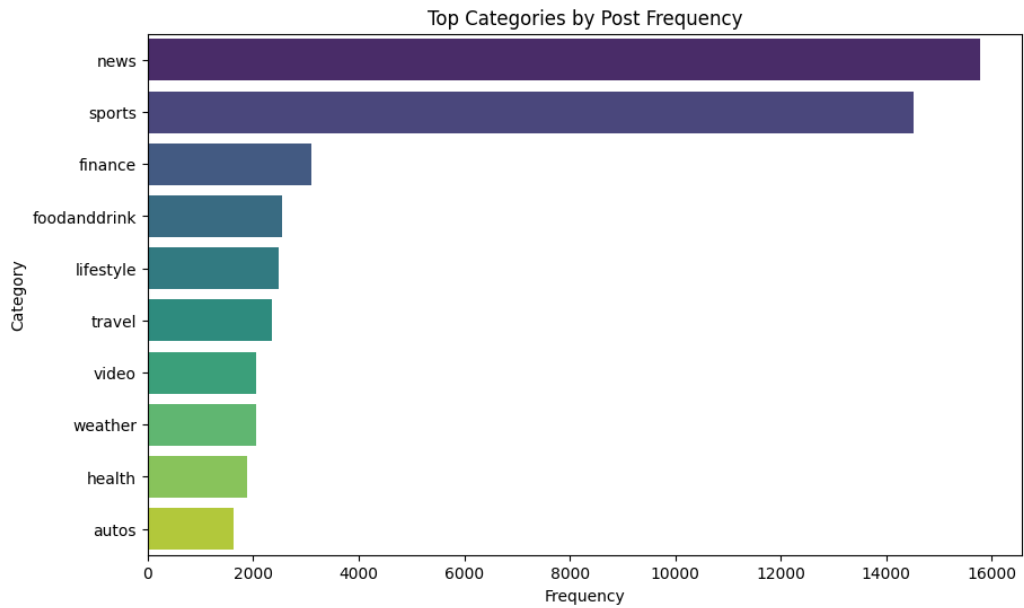


**Figure 2.1.** Users interaction.

### 2.5.1.3. Categories

Our dataset intricately classifies the post content into 130 diverse categories, ranging from 'Art' and 'Real Estate' to 'News' and 'Sustainable Living'. This broad categorization not only underscores the extensive range of topics that captivate social media users but also serves as a cornerstone for analyzing engagement trends. The prevalence of the 'News' category, which emerges as the most represented, highlights the universal appeal of certain themes on social media platforms.

The distribution of dataset categories are shown in the Figure 2.2.



**Figure 2.2.** Dataset categories.

#### **2.5.1.4. Application in deep learning model training**

The dataset was meticulously assembled with the specific aim of training a deep learning model to predict user engagement patterns on social media. The diversity encapsulated in the post content and categories, along with the detailed breakdown of user interaction types, provides a rich and nuanced foundation for the model's training. This comprehensive dataset enables the development of advanced algorithms capable of discerning user preferences and forecasting engagement levels with remarkable accuracy, based on the thematic content of social media posts.

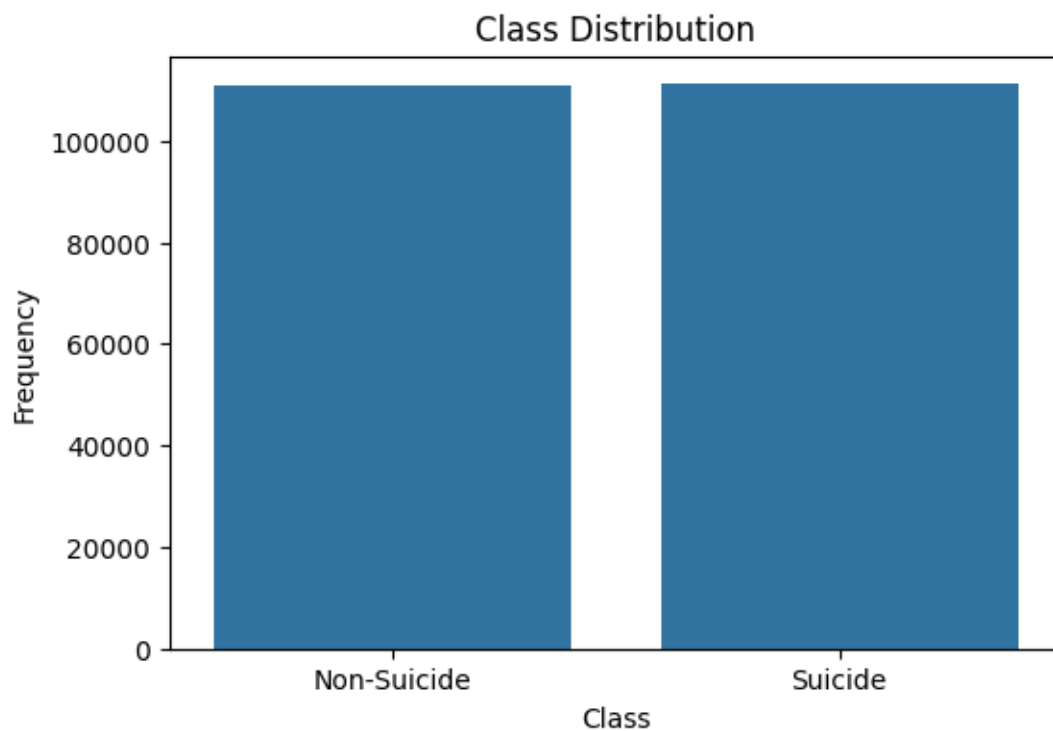
#### **2.5.1.5. Ethical considerations and data preparation**

In compiling the dataset, utmost care was taken to adhere to stringent ethical and privacy guidelines. All data were either anonymized or derived from simulations to circumvent any potential privacy violations. Prior to its deployment in training the deep learning model, the dataset underwent an exhaustive preprocessing phase. This phase included steps such as normalization, categorization, and the removal of duplicates or irrelevant entries, thereby ensuring that the dataset was optimized for the intricate requirements of deep learning algorithms. Through this careful and considered approach, the dataset stands as a testament to the ethical and methodological rigor that underpins our study.

## 2.5.2. The suicide dataset

### 2.5.2.1. Overview

This dataset comprises textual data collected from a variety of sources, aimed at facilitating the study of mental health, particularly focusing on identifying expressions related to suicidal ideation versus non-suicidal content. The dataset is structured into two primary columns: 'text' and 'class'. As illustrated in **Figure 2.3** (insert the actual figure number), the class distribution within the dataset is balanced between 'Non-Suicide' and 'Suicide' categories, each containing approximately equal numbers of entries. This balanced distribution is critical for ensuring that the predictive models trained on this dataset can accurately learn to distinguish between the two classes without bias towards one class due to uneven sample sizes.



**Figure 2.3.**Class distribution.

### 2.5.2.2. Description

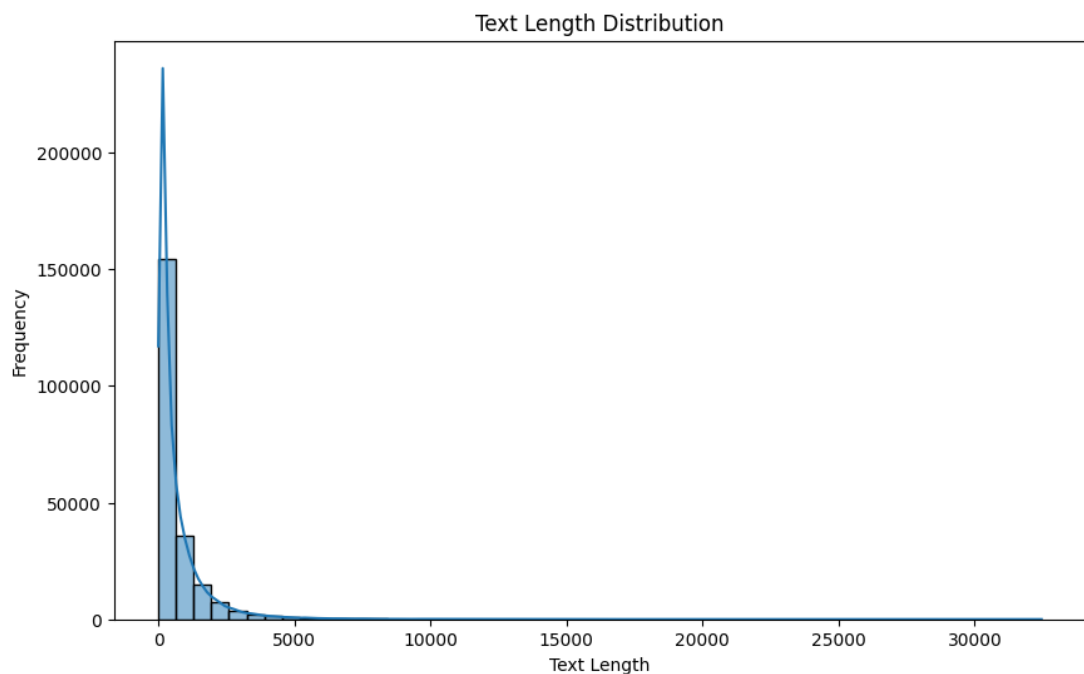
- Text: This column contains user-generated content, ranging from personal anecdotes and expressions of feelings to general statements. The textual data provides insight into the mental state and emotional expressions of the individuals,

offering a rich source for natural language processing (NLP) and sentiment analysis.

- **Class:** The classification column categorizes each entry into either 'suicide' or 'non-suicide', indicating whether the text reflects suicidal ideation or not. This binary classification makes the dataset particularly valuable for training machine learning models aimed at detecting signs of suicidal tendencies in text-based communication.

### 2.5.2.3. Data characteristics

The dataset showcases a wide range of linguistic expressions, emotional states, and personal experiences, reflecting the complexity and sensitivity of the subject matter. As demonstrated in the Figure 2.4, which depicts the distribution of text lengths, the textual content varies significantly from concise statements to elaborate narratives. The Figure 2.5, a word cloud, visually emphasizes the frequency of key terms that appear in the dataset, highlighting predominant themes of emotional and psychological distress. These graphical representations serve not only to summarize the data's extensive variability in length and detail but also underscore the context and depth of the personal expressions contained within.



**Figure 2.4.** Test length distribution.



for Ethereum blockchain development, Solidity enables the creation of complex contracts for decentralized applications (DApps), including voting, crowdfunding, and decentralized finance (DeFi) applications. Its syntax is similar to that of JavaScript and C++, making it accessible to a wide range of developers [58].

### **2.6.3. Truffle**

Truffle is a popular development framework for Ethereum, aimed at making life easier for blockchain developers. It provides a suite of tools for writing, testing, and deploying smart contracts. Truffle's built-in smart contract compilation, linking, deployment, and binary management simplify the development process, while its testing framework and network management for deploying to any number of public and private networks enhance productivity [59].

### **2.6.4. Ganache**

Ganache is a part of the Truffle Suite that provides a personal blockchain for Ethereum development you can use to deploy contracts, develop applications, and run tests. It's available as both a desktop application and a command-line tool. Ganache is designed to streamline the development process by making it easy to explore different states and scenarios on a blockchain without the costs and delays of the real Ethereum network [60].

#### **2.6.4.1. Ganache interface**

When users establish a workspace, they receive server details and a 10 wallet address each one of them has 100 eth and we can see that at the **figure 2.6** his allocation of ether in every account enables developers to concentrate on building applications without worrying about initial financial resources. It simplifies the workflow, facilitating smooth testing and deployment of smart contracts in the workspace setting.

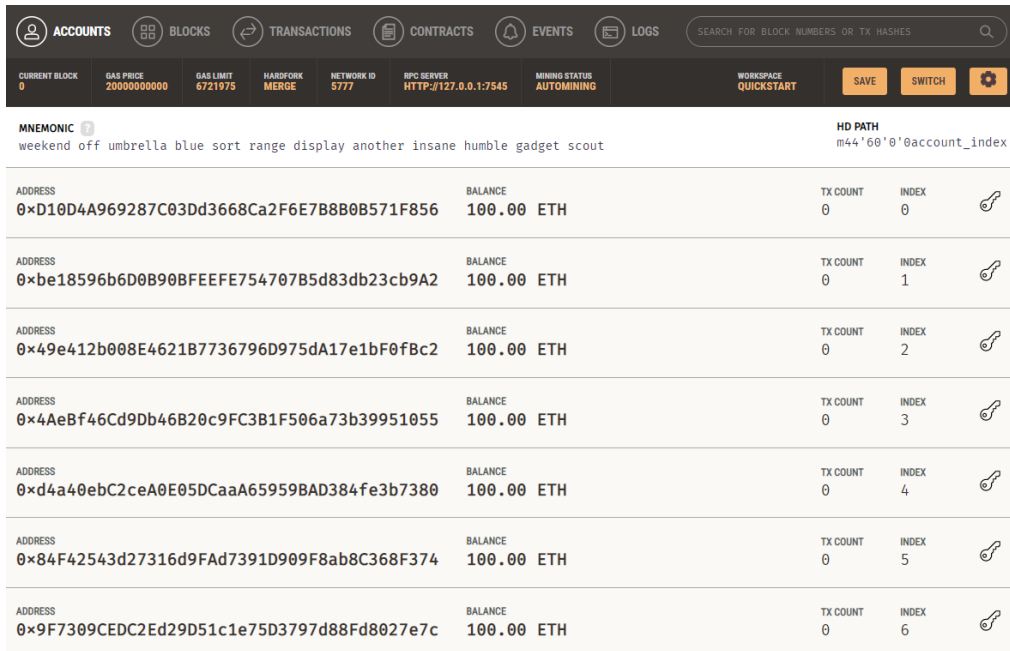


Figure 2.6. Ganache interface.

### 2.6.5. Metamask

MetaMask is a crypto wallet and gateway to blockchain apps, available as a browser extension and a mobile app and **figure 4.2** shows how it looks like after sign in to it. It allows users to interact with the Ethereum blockchain easily, managing accounts and assets, including ETH and ERC-20 tokens. MetaMask also provides a user-friendly platform to connect with decentralized applications (DApps) without running a full Ethereum node [61].

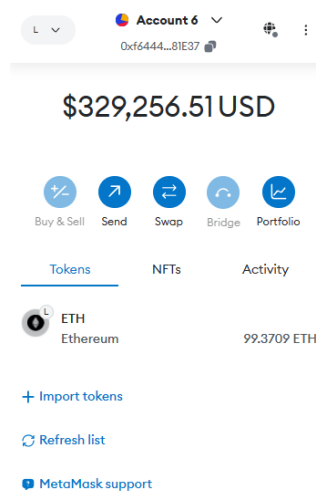


Figure 2.7. After inserting one of the address accounts in the metamask.



### **2.6.6. Ethers.js**

Ethers.js is a lightweight JavaScript library that aims to make it easier to interact with the Ethereum blockchain and its ecosystem. It provides developers with tools to write JavaScript applications that utilize the Ethereum blockchain, including functions for creating wallets, writing smart contracts, and connecting to Ethereum nodes. Ethers.js is designed for ease of use, minimalistic, and complete with features needed for many Ethereum development tasks [62].

### **2.6.7. Python & Flask**

Python is a versatile, high-level programming language known for its readability and broad applicability in areas such as web development, data analysis, artificial intelligence, and scientific computing. It's favored for its simplicity and the vast ecosystem of libraries and frameworks it supports [63].

Flask is a lightweight WSGI (Web Server Gateway Interface) web application framework for Python. It is designed to make getting started with web application development quick and easy, with the ability to scale up to complex applications. Flask offers suggestions but does not enforce any dependencies or project layout [64].

### **2.6.8. HTML, Css, and JS**

HTML (HyperText Markup Language) is the standard markup language used to create web pages. It forms the structural foundation of all websites on the internet [65].

CSS (Cascading Style Sheets) is used to control the layout and appearance of HTML elements on a webpage. It allows for the separation of presentation from content, enabling more flexibility and control in the specification of presentation characteristics [66].

JavaScript is a programming language that enables interactive web pages. It is a part of most web browsers and allows for client-side script to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed [67].

### **2.6.9. WEB3**

Web3 is envisioned as the next evolutionary stage of the internet, distinct from the preceding eras of Web 1.0 and 2.0. Web3's infrastructure is composed of several key elements, including clients (browsers or DApps), digital wallets for managing cryptocurrencies and user data, and blockchain servers that record online activities. This decentralized setup contrasts sharply with traditional, centralized web services [68].

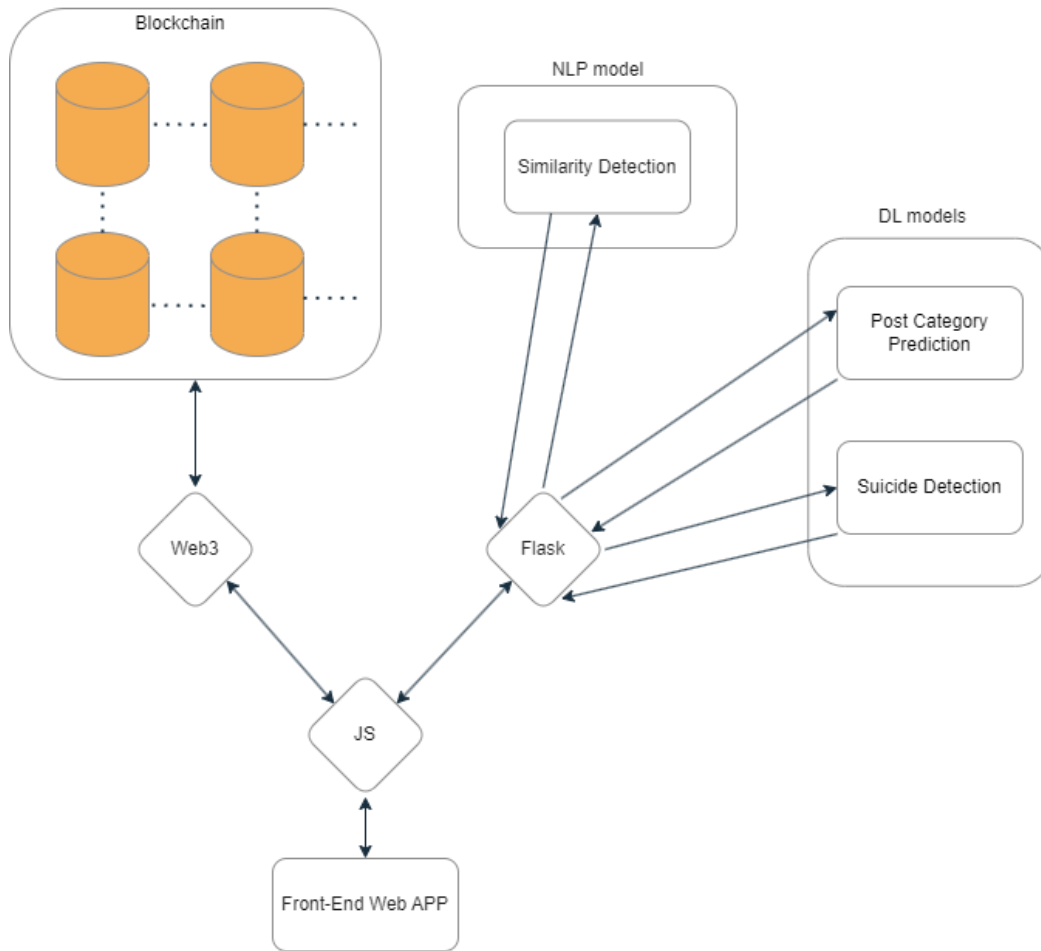
It represents a paradigm shift towards a decentralized and user-empowered digital ecosystem. At its heart, Web3 utilizes blockchain technologies and cryptocurrency to establish a serverless internet where content creation directly benefits the creators, promoting ownership and empowerment [68].

#### **2.6.9.1. Key components of Web3:**

- **Decentralized Applications (DApps):** These are applications that function without centralized control, running on a peer-to-peer network facilitated by blockchain technology [68].
- **Smart Contracts:** Self-executing contracts with the terms of the agreement directly written into code, enabling trustless and automated transactions and agreements [68].
- **Digital Currencies and Tokens:** Including cryptocurrencies and assets like ERC-20 tokens, these digital forms of value are integral to Web3 transactions [68].
- **User Sovereignty:** Users maintain control over their data, benefiting from enhanced privacy and data ownership [68].

### 3. IMPLEMENTATION

Refers to **Figure 3.1** The diagram illustrates a sophisticated architecture that integrates blockchain technology, deep learning (DL), and natural language processing (NLP) within a decentralized social networking platform. At its core, the system utilizes Solidity-based smart contracts deployed on the Ethereum blockchain to manage key functionalities such as user registration, post creation, and data retrieval, thus ensuring robust data integrity and security. The front-end web application facilitates user interaction, leveraging JavaScript coupled with the Web3.js library to bridge communication between the user interface and blockchain operations, allowing for real-time interaction with smart contracts directly from users' browsers. Additionally, a Flask server acts as a conduit between the front-end and the backend, hosting DL and NLP models while providing REST API endpoints for data exchange, enabling tasks like post category prediction and suicide detection by processing and analyzing social media text. The NLP model employs TF-IDF vectorization and cosine similarity to detect text similarities, ensuring content originality and reducing redundancy. This integrated approach showcases a dynamic system that combines the immutability of blockchain with the analytical power of machine learning and NLP to enhance user engagement and security on the platform.



**Figure 3.1.** The Social Network Architecture.

### 3.1. Blockchain Implementation

Decentralized applications (dApps) have gained prominence due to their ability to offer decentralized solutions that mitigate issues associated with centralized platforms, such as data privacy concerns and single points of failure. and we will present here a smart contract, SocialNetwork, developed in Solidity for the Ethereum blockchain, facilitating a decentralized social networking platform where interactions are immutable and transparently recorded on the blockchain.

#### 3.1.1. System architecture

The smart contract, named SocialNetwork, is initiated with a pragma directive to specify the compiler version, ensuring compatibility and security. It encapsulates functionalities related to user and post management within the Ethereum blockchain, including new features for following and unfollowing users, buying posts, and editing

post prices. The contract operates on the Ethereum blockchain, which has transitioned to a Proof-of-Stake (PoS) consensus mechanism, enhancing security, reducing energy consumption, and ensuring efficient transaction validation.

### **3.1.2. Data structures**

The contract meticulously defines two primary structures: `UserInfo` and `PostContent`. The `UserInfo` struct is essential for storing user-related data, encompassing a unique ID, username, email, and wallet address. Conversely, the `PostContent` struct captures the essence of a social media post, detailing attributes such as post ID, username, user address, content, and the number of likes, along with user preferences, the sellability of the post, and its price. These structs are instrumental in facilitating the storage of user and post data within the blockchain framework, thereby enabling a complex array of social network interactions. Additionally, the contract incorporates state variables such as `postCount` and `userID` to monitor the number of posts and manage incremental user IDs, respectively. It also employs mappings to organize user information (`userInformation`, `userInformationBna`), posts (`posts`), likes (`likes`), and a catalog of posts by individual users (`userPosts`). Moreover, the contract declares several events—`PostLiked`, `PostUnliked`, `UserInfoEvent`, `PostContentUser`, `PostPurchased`, `PostDeleted`, `UserAdded`—each designed to log specific activities on the blockchain. These events ensure transparency and are pivotal for tracking interactions like post creation, liking, purchasing, and the dynamics of user follow and unfollow actions within the network.

And we can see in Figure 3.2 a sample from the code:

```
contract SocialNetwork {
    uint public postCount = 0;
    uint public userID = 1;

    struct UserInfo {
        uint userID;
        string username;
        string email;
        address userAddress;
    }

    struct PostContent {
        uint postId;
        string username;
        address userAddress;
        string postContent;
        uint likes;
        string userPreference;
        bool isPayable;
        uint price;
    }

    mapping(address => UserInfo) private userInformation;
    mapping(uint => PostContent) public posts;
    mapping(uint => mapping(address => bool)) public likes;
    mapping(address => uint[]) private userPosts;

    event PostLiked(uint indexed postId, address indexed user);
    event PostUnliked(uint indexed postId, address indexed user);
    event UserInfoEvent(string method, UserInfo userInfo, address caller);
    event PostContentUser(string action, PostContent content, address sender);
    event PostPurchased(uint postId, address indexed oldOwner, address indexed newOwner, uint price);
    event PostDeleted(uint postId);
}
```

**Figure 3.2.** Sample Solidity Smart Contract Code.

### 3.1.3. Functionalities

The contract encapsulates a robust User Management system and Post Management functionalities alongside specific Interaction Functions, effectively orchestrating the dynamics of a blockchain-based social network. User management is facilitated through functions such as `addUserInfo`, which registers users by archiving their data, and retrieval functions like `getUserInfo` and `getUserInfoBname`, which extract user information using wallet addresses or usernames. Additionally, functions like `verifiedUser` and `doesUserExist` are integral for verifying the completeness of a user's information and confirming the existence of users within the system. In terms of Post Management, the `createPost` function allows users to initiate new posts. This is complemented by a suite of functions—`getPostContent`, `getUserPosts`, `getAllPostIds`, and other getters—that provide detailed access to post contents, user-specific posts, and aggregate post counts. The function `buyPost` is particularly notable for enabling the purchase and ownership transfer of posts designated as sellable, which also involves updating the post's price. Further, `editPostPrice` permits post owners to modify the prices of sellable posts. Interaction functions such as `likePost` and `unlikePost` not only allow users to express their preferences by liking or unliking posts but also update the likes mapping and trigger corresponding events,

while `hasUserLikedPost` checks user interactions with specific posts. The system also supports post editing and deletion through the `deletePost` function, which removes posts from the blockchain and updates relevant mappings and lists, thus maintaining the integrity and relevance of the data stored within the network.

And we can see in Figure 3.3 a sample from the code that shows number of functions that we used in our application

```
function addUserInfo(string memory _username, string memory _email, address _userAddress) public {
    userInfo[_userAddress] = UserInfo(userID++, _username, _email, _userAddress);
    emit UserInfoEvent("AddUserInfo", userInfo[_userAddress], msg.sender);
    emit UserAdded(_username, _userAddress); // Emit the event
}

function createPost( string memory _username, string memory _postContent, string memory _userPreference, bool _isPayable, uint _price) public {
    posts[postCount] = PostContent(postCount, _username, msg.sender, _postContent, 0, _userPreference, _isPayable, _price);
    userPosts[msg.sender].push(postCount);
    emit PostContentUser("AddPost", posts[postCount], msg.sender);
    postCount++;
}

function verifiedUser(address _userAddress) public view returns (bool) {
    return bytes(userInfo[_userAddress].username).length > 0 && bytes(userInfo[_userAddress].email).length > 0;
}

function doesUserExist(address _userAddress) public view returns (bool) {
    return userInfo[_userAddress].userAddress != address(0);
}

function likePost(uint _postId) public {
    require(_postId < postCount, "Post does not exist.");
    require(!likes[_postId][msg.sender], "Post already liked.");
    posts[_postId].likes++;
    likes[_postId][msg.sender] = true;
    emit PostLiked(_postId, msg.sender);
}

function unlikePost(uint _postId) public {
    require(_postId < postCount, "Post does not exist.");
    require(likes[_postId][msg.sender], "Post not liked before.");
    posts[_postId].likes--;
    likes[_postId][msg.sender] = false;
    emit PostUnliked(_postId, msg.sender);
}

function hasUserLikedPost(uint _postId, address _user) public view returns (bool) {
    return likes[_postId][_user];
}

function buyPost(uint _postId) public payable {
    require(_postId < postCount && posts[_postId].isPayable, "Post not available for purchase.");
    PostContent storage post = posts[_postId];
    require(msg.value == post.price, "Incorrect Ether value.");
    require(post.userAddress != msg.sender, "Cannot buy your own post.");
    address payable oldOwner = payable(post.userAddress);
    oldOwner.transfer(msg.value);
    removePostFromUser(post.userAddress, _postId);
    userPosts[msg.sender].push(_postId);
    post.userAddress = msg.sender;
    UserInfo storage buyer = userInfo[msg.sender];
    post.username = buyer.username;
    emit PostPurchased(_postId, oldOwner, msg.sender, msg.value);
}
```

**Figure 3.3.** Sample of the code functions.

### 3.1.4. Implementation details

The smart contract uses Ethereum's blockchain to store and manage data related to users and posts in a decentralized manner. It leverages mappings for efficient data retrieval and updates, ensuring that interactions are immutable and verifiable. Solidity's event mechanism provides a transparent audit trail of actions performed on the network. Additional features like post purchasing, and price editing enrich the social network functionality, making it more interactive and dynamic.

## 3.2. Deep Learning Implementation

### 3.2.1. Deep learning approach to post category prediction in social media posts

The proliferation of social media platforms has led to an exponential increase in the volume of textual data generated daily. This data, rich in information, presents both opportunities and challenges in information retrieval, sentiment analysis, and content categorization. Automating the classification of social media posts can aid in efficiently managing this data, enabling applications such as targeted advertising, content recommendation, and monitoring public sentiment.

#### 3.2.1.1. Data preparation and preprocessing

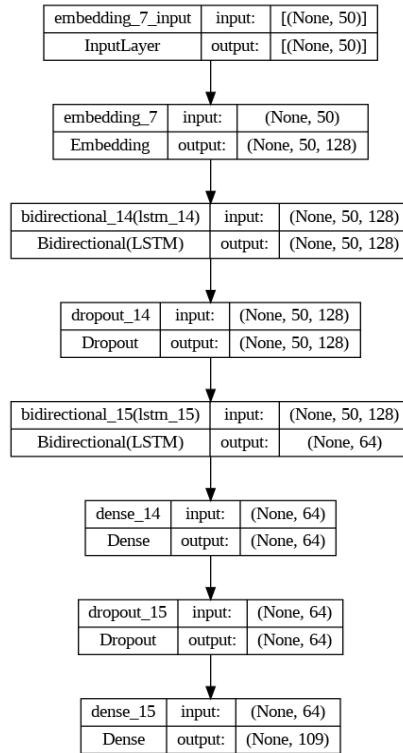
- **Dataset Loading:** The process begins with loading the dataset from a CSV file into a pandas DataFrame. The dataset contains social media posts alongside their corresponding categories or labels, facilitating supervised learning. The columns 'Post' and 'Category' represent the text data and labels, respectively.
- **Text Tokenization:** Tokenization is performed using Keras's Tokenizer, which converts the textual data into sequences of integers where each integer represents a unique word in the dataset. This step is crucial for preparing the text data for numerical processing by the deep learning model. A vocabulary size is predefined to limit the number of unique words, enhancing model efficiency and reducing computational complexity.
- **Sequence Padding:** Given the variable lengths of text posts, sequence padding ensures that all input sequences to the model have a uniform length. This uniformity is achieved by either truncating longer sequences or padding shorter sequences with zeros. This step is vital for batch processing of data during model training.
- **Label Encoding and Conversion to Categorical:** The categories or labels are transformed from textual to numerical form using LabelEncoder, facilitating their processing by the neural network. Subsequently, these numerical labels are converted into a one-hot encoded format, a necessary step for categorical classification.



### **3.2.1.2. Model architecture and training**

The deep learning model is conceptualized as a Sequential model, which is fundamentally an ordered aggregation of layers specifically tailored for text classification tasks. The architectural framework of the model begins with an Embedding layer, which serves the critical function of transforming integer-encoded vocabulary into dense vectors of a fixed size. This transformation is pivotal as it captures the semantic relationships between words, thereby enriching the model's understanding of language nuances. Further enhancing the model's capability to comprehend and process textual data, Bidirectional LSTM layers are incorporated. These layers are adept at learning dependencies and contextual nuances in both the forward and backward directions of the sequence, substantially improving the model's ability to grasp the overall context of the sequence. To mitigate the risk of overfitting, Dropout layers are strategically placed within the model to randomly nullify a fraction of the input units during the training phase. The model also includes Dense layers that are crucial for the classification process, culminating in a final layer equipped with a softmax activation function. This function is essential for outputting a probability distribution across the predefined categories, thus enabling precise classification. The optimization and loss calculation of the model are managed by the Adam optimizer and the categorical crossentropy loss function, respectively, both of which are well-suited for multi-class classification scenarios. This meticulously designed model structure ensures robust performance in text classification by effectively handling various linguistic elements and their contextual relationships.

And in Figure 3.4 we can see the model architecture



**Figure 3.4.** Post prediction model architecture.

The training of the model incorporates an Early Stopping mechanism, where it is initially trained on a designated subset of the data known as the training set and validated against a separate subset referred to as the validation set. This technique is crucial as it halts the training process when there is no further decrease in validation loss, effectively preventing overfitting and ensuring the model's ability to generalize well to unseen data. Following the training phase, the model undergoes a rigorous evaluation on a test dataset to ascertain its accuracy and generalization capabilities. The evaluation process includes a detailed classification report that provides metrics such as precision, recall, and f1-score for each category, thereby offering a comprehensive view of the model's predictive performance across different classes. Additionally, the practical application of the model is demonstrated through a function designed to predict the category of a new post. This function involves tokenizing the post, padding the sequence to a fixed length, and subsequently utilizing the model to predict the category, thereby showcasing the model's capability to process and classify raw text from inception to conclusion. This holistic approach from training to prediction underscores the model's utility in real-world applications, ensuring its efficacy in delivering accurate and reliable text classification.

### **3.2.2. Deep learning approach to suicide detection in social media texts**

The advent of social media has led to an immense increase in the generation of digital textual content, offering significant insights into users' mental states and inclinations. Among these, the detection of suicidal tendencies in textual data has become a paramount concern, aiming to provide timely interventions and support. Leveraging deep learning for suicide detection involves nuanced text analysis to discern patterns indicative of suicidal thoughts or tendencies, distinguishing them from non-suicidal content.

#### **3.2.2.1. Data preparation and preprocessing**

The process of leveraging a dataset for training a deep learning model begins with the initial step of loading the dataset into a pandas DataFrame. This dataset, consisting of social media texts categorized as 'suicide' or 'non-suicide', is structured to facilitate ease of manipulation and preprocessing. The subsequent text preprocessing phase involves multiple steps tailored to prepare the text data for modeling. Firstly, the 'text' column is converted to a string type to maintain uniformity across data types, simplifying further manipulations. Additionally, the 'class' column, which delineates the labels 'suicide' and 'non-suicide', is encoded into numeric forms (0 and 1) to facilitate computational processing. Following preprocessing, the dataset is strategically divided into training and testing sets with an 80-20 split, designed to assess the model's effectiveness on previously unseen data. The final preparatory steps involve tokenization and sequence padding. Utilizing Keras's Tokenizer, the texts are tokenized to transform them into sequences of integers that represent words, while setting a vocabulary limit to concentrate on the most frequently occurring words. To ensure that all input sequences maintain a consistent length suitable for the neural network, these sequences are either padded or truncated to a fixed length, thereby standardizing the input data for optimal model training.

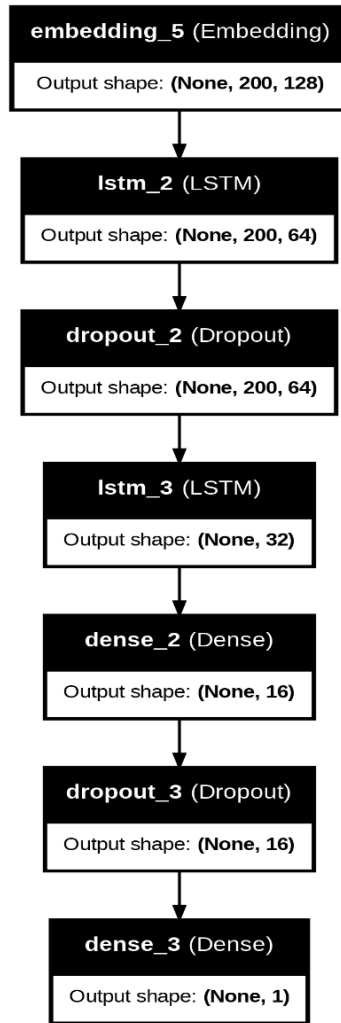
#### **3.2.2.2. Model architecture and training**

The construction of the deep learning model follows a strategic design aimed at tackling binary classification tasks, such as suicide detection in text data. The model architecture begins with an Embedding layer, which is pivotal for transforming integer-encoded vocabulary into dense vectors that effectively capture semantic

relationships within the text. This layer is followed by a series of LSTM layers, which are essential for learning dependencies and contexts across the text. To counteract overfitting—a common challenge in deep learning models—dropout layers are interspersed among the LSTM layers. The final layer of the model is a Dense layer equipped with a sigmoid activation function, which is particularly suitable for binary classification tasks, providing a probabilistic output that categorizes the input texts.

For compilation and training, the model employs the Adam optimizer, known for its efficiency in handling sparse gradients on noisy problems, along with the binary cross-entropy loss function, which is standard for binary classification models. Additionally, an early stopping callback is integrated into the training process. This feature is designed to halt training if there is no improvement in validation loss over a set number of epochs, effectively preventing overfitting and ensuring the model does not learn idiosyncrasies of the training data at the expense of its ability to generalize to new, unseen data. This careful orchestration of model architecture, compilation settings, and training interventions is engineered to optimize performance for its specific application in detecting suicidal intent within text data.

And in Figure 3.5 we can see the model architecture



**Figure 3.5.** Suicide detection model architecture.

### 3.2.2.3. Evaluation and application

Post-training, the model undergoes a rigorous evaluation on the test set, a crucial step in determining its accuracy and ability to generalize across new, unseen data. This assessment is essential for gauging the model's effectiveness in its designated task of detecting suicidal content within texts. By analyzing how well the model performs on this independent data set, researchers can infer the robustness of the learned patterns and the likelihood of the model to maintain high performance in practical applications. This evaluation not only highlights the predictive power of the model but also provides insights into any potential biases or weaknesses, ensuring that the model is reliable and effective in real-world scenarios where the stakes of accurate detection are high.

### 3.3. Natural language processing implementation

In the rapidly evolving domain of digital communication, our social network exemplifies the influence of shared ideas and expressions. Given the ceaseless influx of user-generated content, it is crucial to ensure the originality and uniqueness of posts. To tackle this challenge, we have incorporated advanced text similarity detection techniques, utilizing TF-IDF vectorization and cosine similarity algorithms to discern and manage duplicate or highly similar content within our platform. The process of ensuring content originality begins by transforming textual data into a structured, machine-readable format through vectorization. As depicted in the pseudo-code snippet in Figure 3.6, we employ the TF-IDF method to convert text into vectors, which emphasizes the importance of unique terms across documents.

```
DEFINE FUNCTION vectorize(text1, text2)
  # Create a TF-IDF vectorizer instance
  vectorizer = CREATE TF-IDF Vectorizer

  # Convert text1 and text2 into numerical vectors
  using the vectorizer
  vectors = vectorizer.fit_transform([text1, text2])

  # Convert vectors to an array format
  vector_array = CONVERT vectors to array

  RETURN vector_array
```

**Figure 3.6.** Pseudo-Code for computing cosine similarity between text vectors.

The text data is first converted into numerical vectors by creating a TF-IDF vectorizer instance, which are then arrayed for subsequent analysis. This methodical approach not only prepares our data but also focuses our analysis on the most significant words, setting the groundwork for precise evaluation of text similarity. Further, as shown in Figure 3.7, the similarity among posts is quantified using a function called `calculate_similarity`, which applies cosine similarity to measure the angle between two vectors.

```

DEFINE FUNCTION calculate_similarity(text1, text2)
    # Convert text1 and text2 into numerical vectors
    vectors = CALL vectorize(text1, text2)
    # Extract the numerical vector for each text
    vector1 = vectors[0]
    vector2 = vectors[1]
    # Calculate cosine similarity between vector1 and vector2
    similarity_score = CALCULATE cosine similarity(vector1,
vector2)
    RETURN similarity_score

```

**Figure 3.7.** Pseudo-Code for cosine similarity calculation in text analysis.

This calculation follows the conversion of text into numerical vectors and extraction of individual vectors for each text. Cosine similarity adeptly evaluates the orientation of these TF-IDF vectors, providing a nuanced comparison of textual content ranging from 0 (no similarity) to 1 (identical texts), thus enabling us to accurately determine the relatedness of texts based on their vector representations. In practical application within our social network, these text similarity detection techniques are crucial for maintaining content integrity and originality. By identifying posts that are duplicates or highly similar, we can initiate appropriate measures to prevent content redundancy and promote the generation of fresh, unique content. Our system automatically flags potential plagiarism by comparing new posts against existing content for similarity, ensuring that all shared material is original and enhancing content diversity. By actively monitoring and managing similar posts, we contribute to a more diverse and engaging content landscape, enriching the user experience on our platform.

### **3.4. Integration JavaScript with Smart Contract, DL, and NLP**

the blockchain component is a set of Solidity smart contracts deployed on the Ethereum blockchain. And those contracts manage social network functionalities, including user registration, post creation, and data retrieval, while ensuring data integrity and security.

User and Post Management: The Social network contract includes structures for user information (userinfo) and posts (postcon), alongside mappings to store these entities.

Functions to add and retrieve user and post information are implemented, facilitating interactions with the blockchain data.

### **3.4.1. Integration of javascript and smart contract via Web3**

JavaScript, with the power of the Web3.js library, acts as a translator between a web interface and the Ethereum blockchain (or similar blockchains). Web3.js is a popular JavaScript library that provides the tools to interact with the blockchain, allowing users to directly interact with smart contracts from their browser.

Here's how it works:

**Web3 Initialization:** When the web page loads, the script checks for a compatible Ethereum browser extension installed, like MetaMask. It then requests permission from the user to connect their Ethereum account. This is necessary because user accounts hold the keys to interact with the blockchain.

**Smart Contract Interaction:** Once connected, the script uses the user's account information to interact with specific smart contracts. To do this, it relies on the contract's ABI (Application Binary Interface). The ABI acts like an instruction manual for the contract, defining how to call its functions from outside. It specifies details like function names, the type of data they accept (parameters), and what kind of data they return (if any). With the ABI and the contract address, the script can create instances of the smart contract and call its functions directly from the web interface. This allows users to perform actions like adding new users, creating posts, or fetching stored data on the blockchain.

**Event Handling:** JavaScript listens for user actions, such as form submissions. These actions trigger interactions with the blockchain. Event handling manages both initiating transactions (writing data to the blockchain) and retrieving data (reading from the blockchain).

Below in figure 3.8 we can see lines of code establish a connection to the Ethereum blockchain, obtain the user's Ethereum account signer, and create an instance of a smart contract that can be used to interact with the contract's functions and data on the blockchain.



```
const provider = new ethers.providers.Web3Provider(window.ethereum);
const signer = provider.getSigner();
const contract = new ethers.Contract(address, abi, signer);
```

**Figure 3.8.** Initializing Web3 provider, obtaining signer, and creating contract.

### 3.4.2. Integration of javascript with deep learning and natural language processing models via flask

A Flask server provides a critical bridge between the JavaScript front end and the deep learning (DL) and natural language processing (NLP) models, serving as the interface that facilitates the operation of these advanced analytical tools. Hosting both DL and NLP models, the server exposes a REST API endpoint, such as `/predict`, which is designed to accept JSON payloads containing social network post content. When such a request is received, the Flask application processes the content through a pre-trained deep learning model to categorize the post. The functionality and workflow of this API endpoint are exemplified in the code illustrated in Figure 3.9, highlighting how the server handles incoming data and returns analysis results, thus demonstrating the practical application of interfacing DL and NLP models within a web environment.

```
@app.route('/predict', methods=['POST'])
def predict():
    data = request.get_json()
    post_content = data['postContent']

    # Preprocess the post content
    test_sequence = tokenizer.texts_to_sequences([post_content])
    test_padded = pad_sequences(test_sequence, maxlen=50, padding='post')

    # Make prediction
    prediction = model.predict(test_padded)
    predicted_category_index = np.argmax(prediction, axis=1)
    predicted_category = label_encoder.inverse_transform(predicted_category_index)

    return jsonify({"predictedCategory": predicted_category[0]})
```

**Figure 3.9.** API endpoint in Flask.

- JavaScript Communication with Flask

JavaScript plays a pivotal role in interfacing with the Flask server to submit post content for deep learning (DL) or natural language processing (NLP) analysis, managing the asynchronous data flow between the client-side application and the server. This asynchronous data submission is exemplified

when a user initiates a request for post content analysis. In this scenario, JavaScript captures the content and utilizes the fetch API to send it to the Flask server's /predict endpoint. This process is designed to be asynchronous to ensure that the web page remains responsive during the data transmission and analysis phases. Upon receiving the prediction from the Flask server, JavaScript then handles the server's response by processing and integrating the analysis results into the web interface. This integration may involve displaying the predicted category of the post or other pertinent results. Figure 3.10 illustrates the implementation of this asynchronous communication with the Flask server for predictive analysis. It details the JavaScript code that manages the flow of data from capturing and sending post content to processing the server's response, thereby seamlessly integrating it into the client-side application.

```
fetch('http://127.0.0.1:5000/predict', {
  method: 'POST',
  headers: {
    'Content-Type': 'application/json',
  },
  body: JSON.stringify({ postContent: content }),
})

.then(response => response.json())
.then(data => {
  return contract.createApost(username, useraddress, content, like, data.predictedCategory);
})
.catch((err) => console.error("Error sending post content to Flask:", err));
```

**Figure 3.10.** JavaScript code for asynchronous interaction with a Flask API.

## 4. APPLICATION

### 4.1. Compiling the Smart Contract

1. Ensure that Ganache is actively running.
2. Store the smart contract in the directory "~/contracts" using the ".sol" extension.
3. Access the "~/contracts" directory by opening the terminal and navigating to this location.
4. Compile the Solidity contract through Truffle by executing the command "truffle compile". After successful compilation like in Figure 4.1, migrate the contract to the Ganache network with the command "truffle migrate".
5. Once migration is complete, retrieve the contract address to interact with the contract, as illustrated in Figure 4.2.

```
C:\Users\amerk\Desktop\thesis codes\ThesisCode>truffle compile

Compiling your contracts...
=====
> Compiling .\contracts\Socialnetwork.sol
> Artifacts written to C:\Users\amerk\Desktop\thesis codes\ThesisCode\build\contracts
> Compiled successfully using:
  - solc: 0.8.10+commit.fc410830.Emscripten.clang
```

Figure 4.1. Smart contract compilation outcomes.

```
C:\Users\amerk\Desktop\thesis codes\ThesisCode>truffle migrate

Compiling your contracts...
=====
> Compiling .\contracts\Socialnetwork.sol
> Artifacts written to C:\Users\amerk\Desktop\thesis codes\ThesisCode\build\contracts
> Compiled successfully using:
  - solc: 0.8.10+commit.fc410830.Emscripten.clang

Starting migrations...
=====
> Network name: 'development'
> Network id: 5777
> Block gas limit: 6721975 (0x6691b7)

1_migration.js
=====

Replacing 'SocialNetwork'
-----
> transaction hash: 0x60b9660f1c754d6a8411a2a18a18d0e5396c2fcef11b41e91fb312525192de8
> Blocks: 0
> contract address: 0xb378d1a2B8C46b096980016BA995E521207a262A
> block number: 27
> block timestamp: 1715541592
> account: 0xf6444EE67b07B1946Ea57DF621DEd3E0ff581E37
> balance: 99.360924716191610955
> gas used: 3920255 (0x3bd17f)
> gas price: 2.542720756 gwei
> value sent: 0 ETH
> total cost: 0.00996811375731278 ETH

> Saving artifacts
-----
> Total cost: 0.00996811375731278 ETH

Summary
=====
> Total deployments: 1
> Final cost: 0.00996811375731278 ETH
```

Figure 4.2. Smart contract migration outcomes.

## 4.2. Run the Flask Server

To successfully run a Flask application on your device, follow these corrected steps:

1. Ensure Flask is installed on your device. If not, install it using the command ‘pip install flask’.
2. Navigate to the directory containing your Flask application by changing the directory in your terminal.
3. Execute the Flask application using the command ‘python -m flask --app pythonAppName run --port=5001’. Replace pythonAppName with the name of your Flask application file without the .py extension.
4. Once the command is executed, the application will be accessible via ‘http://127.0.0.1:5001’, as shown in figure 4.3.

A terminal window with a black background and white text. The text shows the output of running a Flask application. It starts with '\* Serving Flask app '\CDUM.py'', followed by '\* Debug mode: off'. A red warning message reads 'WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.'. Below that, it says '\* Running on http://127.0.0.1:5001' and 'Press CTRL+C to quit'.

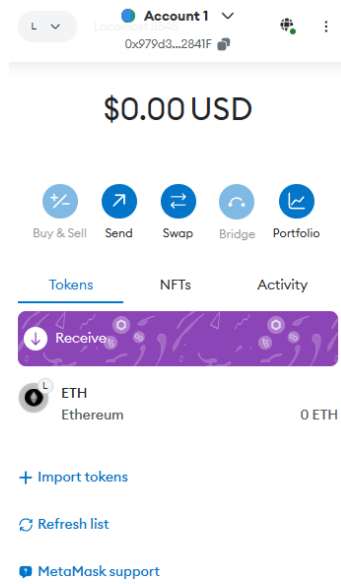
```
* Serving Flask app '\CDUM.py'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5001
Press CTRL+C to quit
```

**Figure 4.3.** Flask Application Running Successfully.

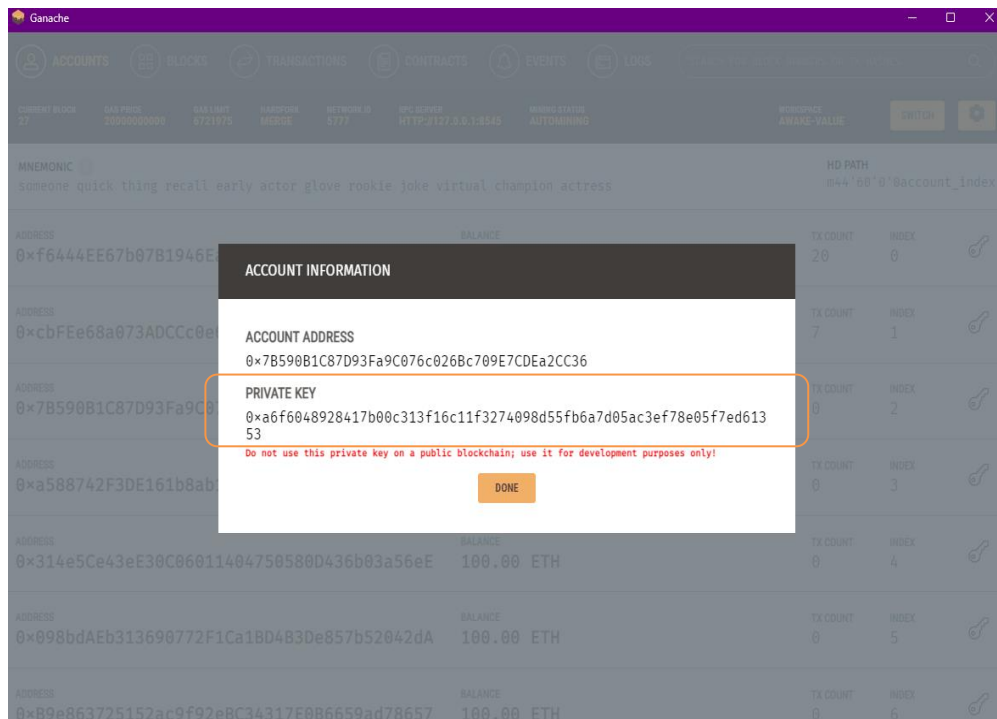
## 4.3. Metamask Registration

After launching the Flask server, proceed with the following steps to set up your wallet for use with your application:

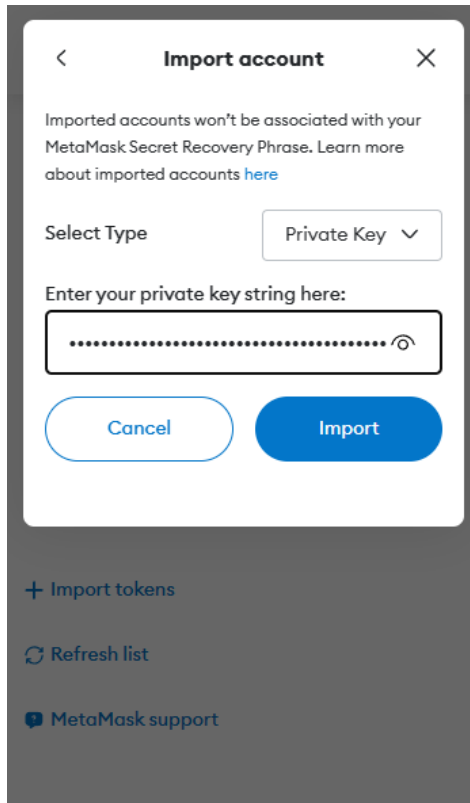
1. Create a new wallet address in MetaMask. Upon logging in, the wallet interface will appear as shown in Figure 4.4.
2. Utilize a dummy address from Ganache by copying the private key from Ganache as illustrated in Figure 4.5.
3. Import the copied private key into your MetaMask wallet as depicted in Figure 4.6. Each dummy address provided by Ganache contains 100 test Ethereum we can see that in the figure 4.7, which can be used to interact with your social network application.
4. Now you can use this setup to run and test our social network application.



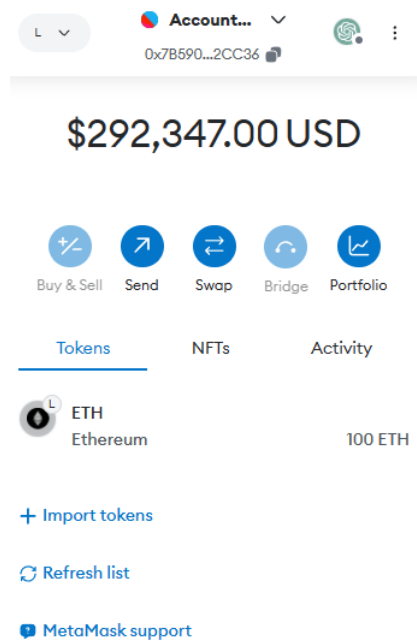
**Figure 4.4.** Metamask wallet interface.



**Figure 4.5.** Ganache private key extraction.



**Figure 4.6.** Metamask wallet address insertion.

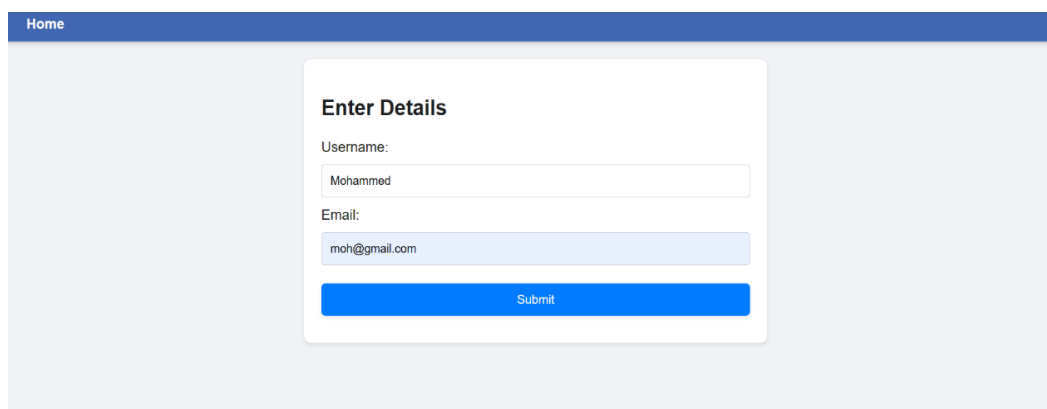


**Figure 4.7.** Metamask address charged with 100 ethereum.

## 4.4. Application Interface

### 4.4.1. Sign up page

After saving the necessary HTML, CSS, JavaScript, and smart contract ABI files into the public directory, you can start the server by running the command `node server.js`. This action enables the website to operate at `http://127.0.0.1:5500/`. Upon entering the website, users are initially prompted to sign up. Clicking the 'Sign Up' button will display the registration page, as illustrated in Figure 4.8. Once setup is complete, the web interface is accessible via this link, providing a functional and interactive online platform for users.

The image shows a web browser window with a dark blue header bar containing the word "Home" in white. Below the header is a light gray background. In the center, there is a white rectangular form titled "Enter Details" in bold black text. The form contains two input fields: "Username:" with the value "Mohammed" and "Email:" with the value "moh@gmail.com". Below these fields is a blue button with the text "Submit" in white.

**Figure 4.8.** Sign up page.

The Signup interface, as depicted in Figure 4.8, is structured to facilitate user interaction for entering personal details. It consists of the following key elements:

1. **Navigation Bar:** Positioned at the top, it includes a link to the "Home" page, enhancing the user's navigation experience across different sections of the website.
2. **Form Container:** Centrally located, this container houses a form titled "Enter Details." It provides a straightforward mechanism for users to input their username and email address. Each field is clearly labeled and has a placeholder text to guide the user on what information to enter.
3. **Input Fields:** There are two main input fields within the form
  - **Username Field:** Allows the user to enter their desired username.

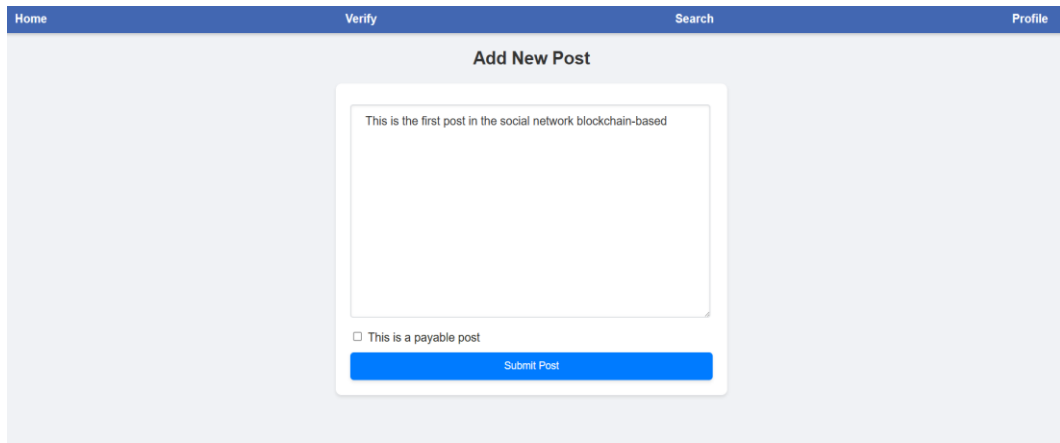
- Email Field: For entering the user's email address. Both fields are designed to capture essential user information required for registration or login processes.
4. Submit Button: A prominent blue button labeled "Submit" is provided to finalize the data entry process. This button is intended to submit the form data, triggering backend processes or further interactions like form validation or data processing.

In addition to the visual elements of the web interface, there's an underlying interactive component handled by JavaScript, crucial for integrating blockchain functionalities:

5. Blockchain Integration: The JavaScript code includes an interaction with a blockchain via a smart contract. This is facilitated by the `addUserinfo` (username, email, useraddress) function. Here's how it operates within the system:
  - Event Listener: JavaScript listens for the "click" event on the "Submit" button. Upon activation, it captures the username and email from the input fields.
  - Smart Contract Interaction: The captured details, along with the user's Ethereum address (`currentAccount`), are submitted to the blockchain through the `addUserinfo` function of the deployed smart contract. This function call is designed to securely register or update the user's information on the blockchain.
  - Transaction Processing: Once the data is submitted, the transaction goes through the blockchain network where it is verified and added to the blockchain. Upon successful transaction completion, feedback is provided to the user in the form of a success message, enhancing the user experience by confirming the registration process.



## 4.4.2. Creating post page



**Figure 4.9.** Creating post page.

The web interface that show in the Figure 4.9 for the "Add New Post" section of a social media platform is designed to not only facilitate the creation and submission of posts but also to integrate sophisticated blockchain interactions that enhance security and functionality. Here's a detailed overview of the entire process, including blockchain functions:

- **Navigation Bar:** Located at the top, this bar facilitates easy navigation with links to Home, Verify, Search, and Profile sections, ensuring users can easily move between different functionalities of the site.
- **Post Submission Form:** The main feature of this interface is the form where users can input and submit new posts:
- **Text Area:** Users can type their post content here, intended for social sharing or discussions.
- **Payable Post Option:** A checkbox allows users to mark the post as payable, which, when activated, displays an input for setting the price in Ethereum, demonstrating the integration of blockchain for financial transactions.
- **Submit Button:** Pressing this button triggers multiple blockchain interactions through embedded JavaScript functions that enhance the platform's interaction with the Ethereum blockchain.
- **Account and ABI Initialization:** Upon loading, the JavaScript checks for Ethereum wallet access (`ethereum.request({method: "eth_requestAccounts"}`

))). If available, it fetches the ABI from the server and initializes the application to interact with the smart contract at a specified address.

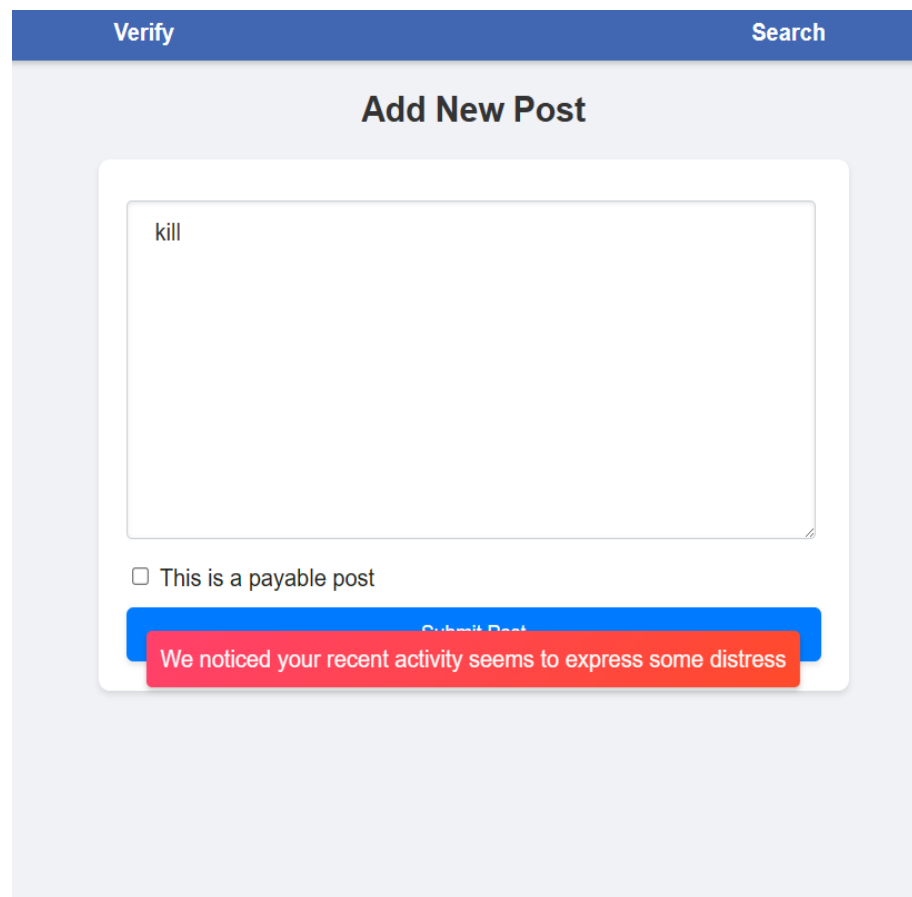
- Smart Contract Functions

1. User Verification Check: Immediately after the environment loads and the user's Ethereum account is connected, the `contract.verifiedUser(currentAccount)` function is invoked. This smart contract function checks whether the current user's account is registered and verified on the blockchain. This step is essential for ensuring that only registered and authenticated users can interact with the platform's features, particularly those involving blockchain transactions like posting content.

#### Handling Verification Results:

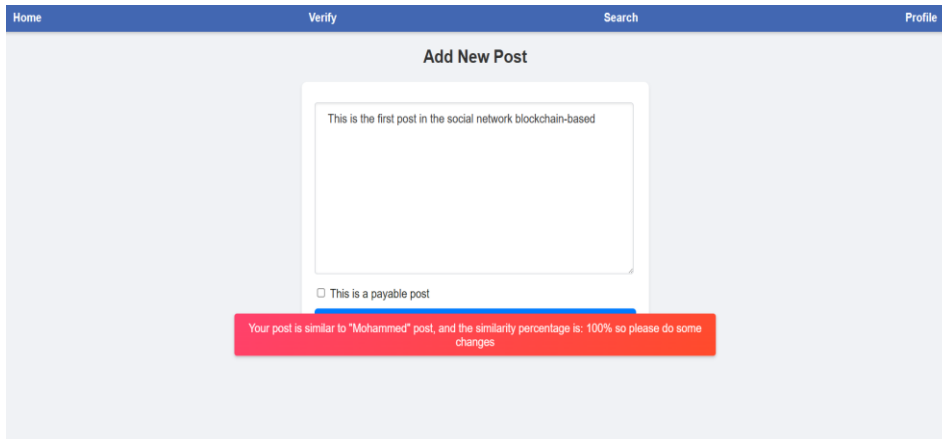
- If Verified: If the user is successfully verified, the system allows them to proceed with the content submission process, which includes content sensitivity checks, similarity analysis, and category prediction as previously outlined.
  - If Not Verified: If the verification process fails (i.e., the user is not registered or recognized by the smart contract), the user interface updates to display a notification or redirection option. Typically, this might include a message such as "You are not registered on this website. Please register and come back," and a prompt or button leading to the registration page. This ensures that only verified users can post content, maintaining the platform's integrity and compliance with user management policies.
2. Get All Post IDs (`getAllPostIds()`): This function fetches all existing post IDs to ensure unique identifiers for new posts.
  3. Get User Info (`getUserInfo(currentAccount)`): Retrieves user-specific information from the blockchain, to use them in posts creation.
  4. CreatePost(`contract.createPost(username, postContent, data. Predicted Category, isPayable, postPrice )`) This function is central to the post submission process on the social media platform, called upon form submission after thorough validations. The detailed steps of the process are as follows:

- Initial Content Assessment for Sensitive Themes: Prior to submission, the post content is first sent to a server-side API (`fetch('http://127.0.0.1:5001/predictSui')`) which checks for any indications of harmful content, such as mentions of suicide. If the API predicts the content to be sensitive, a visible alert (`suimessage`) is shown, advising the user to modify the content, thereby ensuring the community's safety and we can see the Figure 4.10.



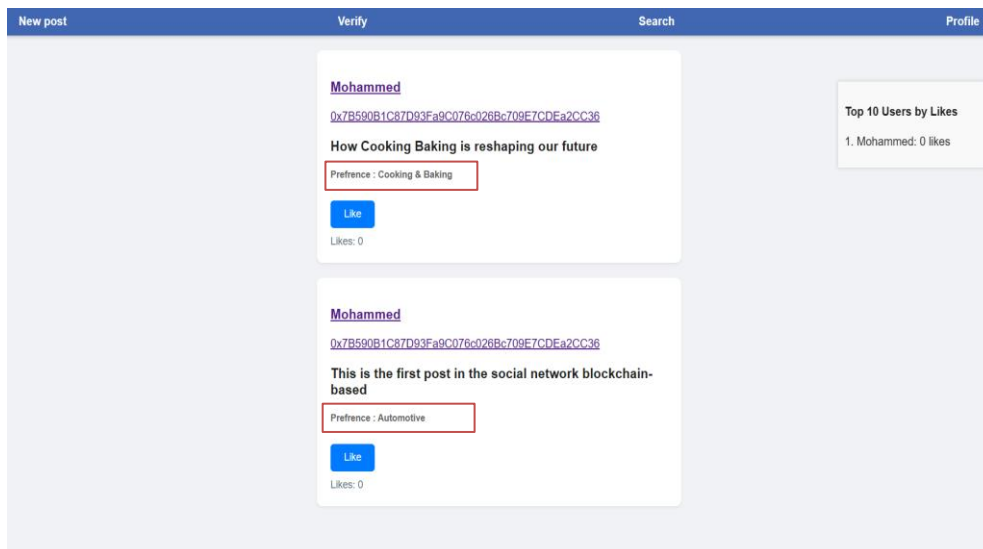
**Figure 4.10.** Suicide detected.

- Content Similarity Analysis: If the content is deemed safe, it undergoes a similarity analysis via another server call (`fetch('http://127.0.0.1:5001/compare_texts')`). This function compares the new post against existing posts to avoid content redundancy and promote original contributions on the platform, as we shown in the figure 4.11.



**Figure 4.11.** Similarity result.

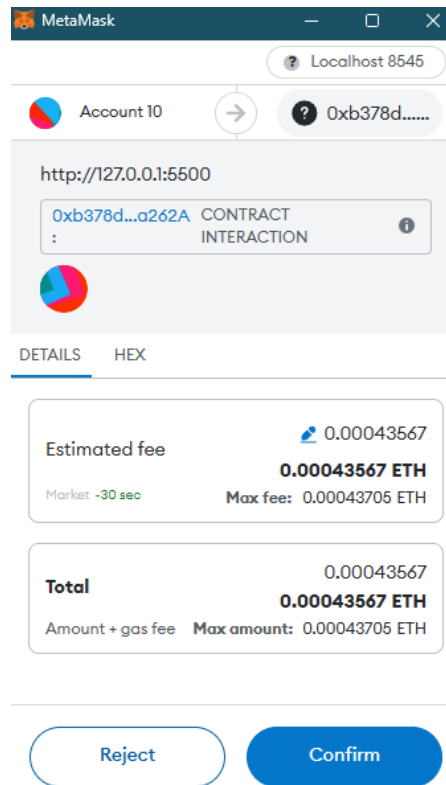
- **Category Prediction and Post Processing:** Concurrently, another predictive model is accessed via (`fetch('http://127.0.0.1:5001/predict')`) which classifies the post content into predefined categories and that shown in figure 4.12. This categorization helps in organizing the posts effectively on the platform, enhancing the browsing and search experience for users.



**Figure 4.12.** Category prediction result.

- **Blockchain Transaction for Post Creation:** If all the above validations are passed — no sensitive content, acceptable uniqueness, and a defined category — the `createPost` function is executed. This function submits the post along with critical metadata such as the username, content category as determined by the predictive analysis, whether it's payable or not, and the price if applicable. This payment step is crucial as it ensures that the transaction is processed by

the network, and the details are immutable and reliably stored on the blockchain, as depicted in Figure 4.13, then the information is securely recorded on the blockchain, ensuring immutability and reliability.

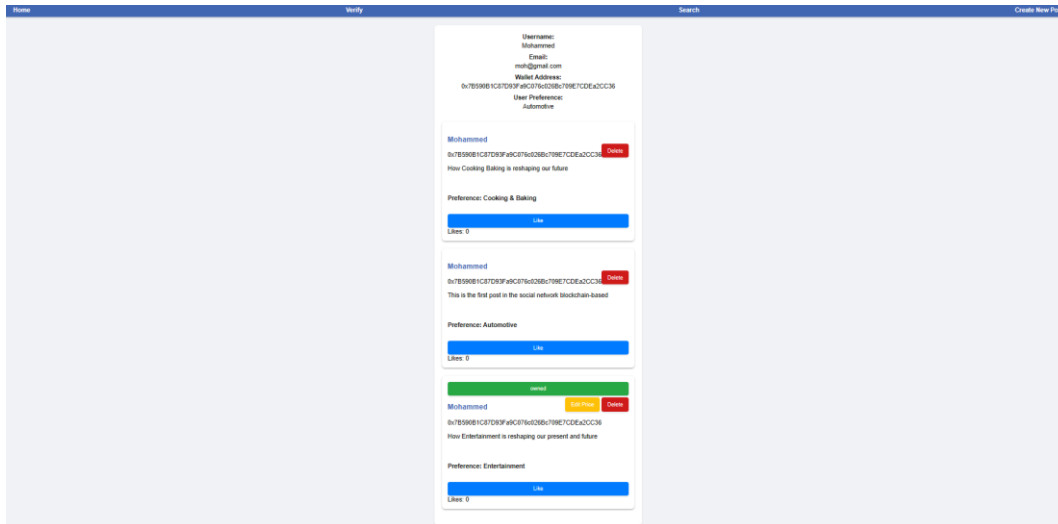


**Figure 4.13.** Gas fee transaction.

5. Feedback Mechanisms: Success and error messages are dynamically displayed based on the results of blockchain transactions
  - Success Messages: Displayed when a post is successfully added to the blockchain, providing positive feedback to the user.
  - Error Handling: If there are issues during the submission process, such as blockchain transaction failures or data validation errors, appropriate messages guide the user to rectify the situation.

#### 4.4.3. Profile page

The figure 4.14 demonstrates a web interface designed for a blockchain-based social network, showcasing user profiles and interactive posts. The layout provides a clear view of individual posts along with the user's details and interaction options such as liking or deleting posts.



**Figure 4.14.** Profile page content.

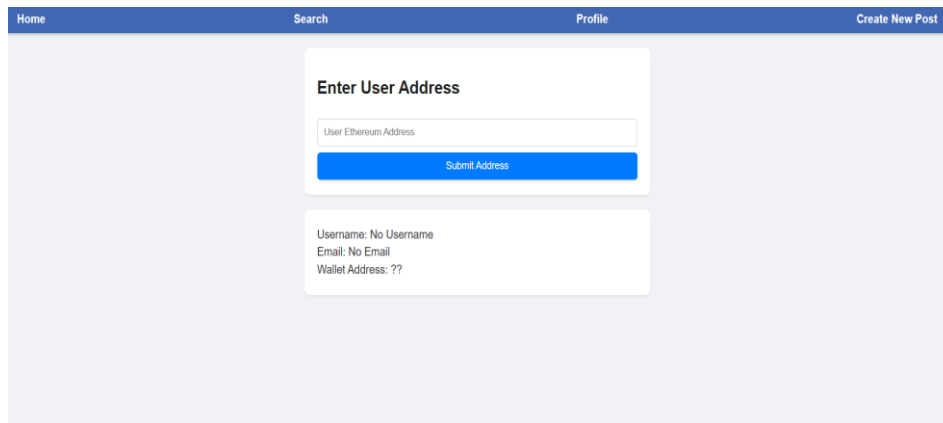
Here's a detailed explanation of how this interface integrates with blockchain technology and handles user interactions:

1. **User Profile Display:** At the top, the user's information is displayed, including the username, email, wallet address, and user preferences. This information is retrieved from the blockchain, ensuring that all displayed data is up-to-date and securely managed.
2. **Dynamic Content Loading**
  - **Post Retrieval:** The `getUserPosts(currentAccount)` function fetches all post IDs linked to the current user's account. Each post ID is utilized to retrieve complete post content using the `getPostContent(postId)` function.
  - **Post Display:** Individual cards are dynamically generated to display each post's content, user preference, and interaction options. These cards include the username, content, and interaction buttons like 'Like', with additional 'Delete' and 'Edit Price' options for posts owned by the user, reflecting the post's status (e.g., owned, payable).
3. **Interactions**
  - **Liking Posts:** Users can like or unlike posts by interacting with like buttons. The state of each post (liked or not) is managed by checking `hasUserLikedPost(postId, currentAccount)`. Users can toggle their like status, which updates the blockchain record via `likePost(postId)` or `contract.unlikePost(postId)`.

- **Deleting Posts:** Owners of posts can delete their content using `deletePost(postId)`, which removes the post from the blockchain and updates the UI to reflect these changes.
  - **Buying and Selling Posts:** For posts marked as payable and not owned by the current user, a 'Buy' option is available, allowing users to acquire the post by transferring the set amount of ETH specified in the post. This transaction is processed through `buyPost(postId, {value: post.Content.price})`.
4. **Price Management**
    - **Editing Post Price:** Owners of payable posts can adjust the post price. An 'Edit Price' button triggers a prompt for the user to input a new price, which is then updated on the blockchain through `editPostPrice(postId, newPriceInWei)`.
  5. **User Preferences and Content Filtering**
    - **Preference Aggregation:** The system aggregates user preferences based on the content of the posts they interact with. This aggregation helps in tailoring the content feed to match user interests more closely.
    - **Most Common Preference Display:** The most frequently occurring preference across all posts is displayed in the user profile area, giving a snapshot of the user's main interest or activity on the network.
  6. **Blockchain Interaction Initialization**
    - **Smart Contract Connection:** Upon loading the webpage, the script checks for an Ethereum provider (e.g., MetaMask) and requests access to the user's account if he was not connected with any account. It then loads the ABI from a local source and initializes the contract interaction.
    - **Account Management:** The system listens for changes to the Ethereum account (e.g., switching accounts in MetaMask), which triggers a reload of the user data and posts to reflect the information of the newly active account.

#### **4.4.4. Verification users**

In the verification user page as we shown in the figure 4.15 we are use it to verify the users and what the posts that they post on the social network.



**Figure 4.15.** User verification page.

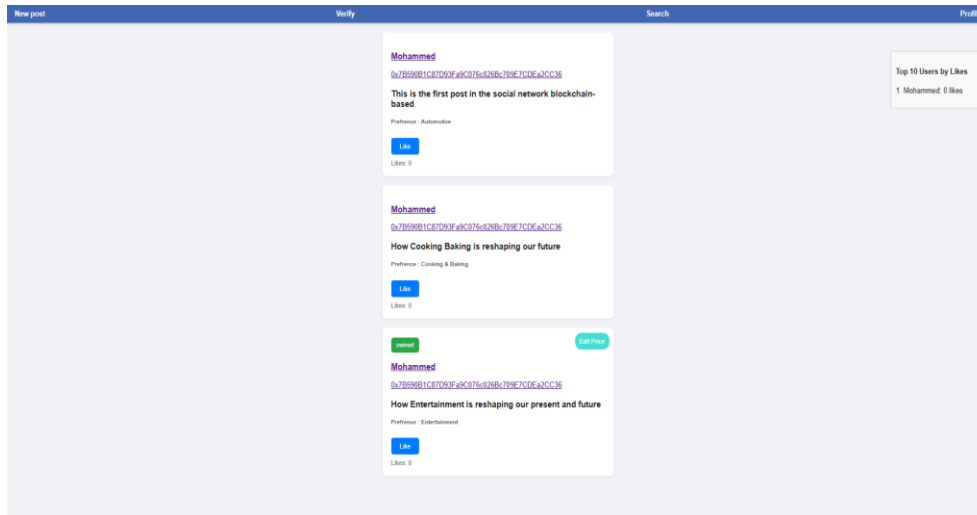
1. User Address Input: The interface includes a section for users to enter an Ethereum address. This is facilitated through an input field where users can submit the address they wish to query or interact with on the blockchain.
2. User Information Display
  - Initial Display: Before any interaction, the interface shows placeholders for username, email, and wallet address indicating no user data is currently loaded.
  - Post-Interaction Display: Upon submitting an address, if the address is verified and user data is available on the blockchain, their information (username, email, wallet address) will be updated and displayed. If not verified or data is unavailable, the interface shows default messages indicating absence of data.
3. Blockchain Interaction
  - Contract Connection: When the webpage loads, it checks for an Ethereum provider (e.g., MetaMask) and requests access to the user's account. If found, it loads the ABI from a predefined location and initializes a contract instance.
  - User Verification: Upon entering an Ethereum address and clicking 'Submit Address', the `verifiedUser(userAddress)` function checks if the user is verified on the blockchain. If verified, further details are fetched; if not, a 'Verification Failed' message is displayed.



- **Fetching User Data:** For verified users, the `getUserInfo(userAddress)` function retrieves detailed user information such as username and email, which are then displayed on the interface.
4. **Dynamic Interaction Handling**
    - **Handling Account Changes:** The interface listens for changes in the Ethereum account (e.g., account switches in MetaMask), prompting a reload of user data to reflect the currently active account.
    - **User Feedback:** The interface provides feedback via the `verificationStatus` div, showing whether the verification was successful, failed, or if an error occurred. This section uses different background colors and icons to visually communicate the status (green for success, red for failure, and a distinct style for errors).
  5. **Styling and Usability Enhancements**
    - **Responsive Design:** The design uses a responsive navigation bar and input groups styled with CSS for better user experience on different devices.
    - **Interactive Elements:** Interactive buttons for submitting addresses and links to other pages (Home, Search, Profile, Create New Post) enhance the usability of the interface.

#### **4.4.5. Main page**

The user interface seamlessly integrates blockchain technology to provide a dynamic and interactive experience for users on a social network platform. It features a navigation bar for easy access to various functionalities such as creating new posts, verifying user, searching for content, and managing personal profiles. Below this, posts are displayed in structured cards that include essential information like the username, the content of the post, user preferences, and interactive options such as liking, editing, or purchasing posts and we can see that in the figure 4.16.



**Figure 4.16.** Main page.

## 1. Interactive Features

- **Liking Posts:** Users can express their preferences by liking or unliking posts, with the interface updating in real-time to reflect changes in the like count without needing to reload the page.
- **Editing Post Prices:** Users who own posts have the ability to adjust prices for payable content directly through the interface, which then updates these details on the blockchain.
- **Purchasing Posts:** Payable posts feature a 'Buy' button enabling users to acquire posts by transferring Ethereum, which also updates ownership details on the blockchain.

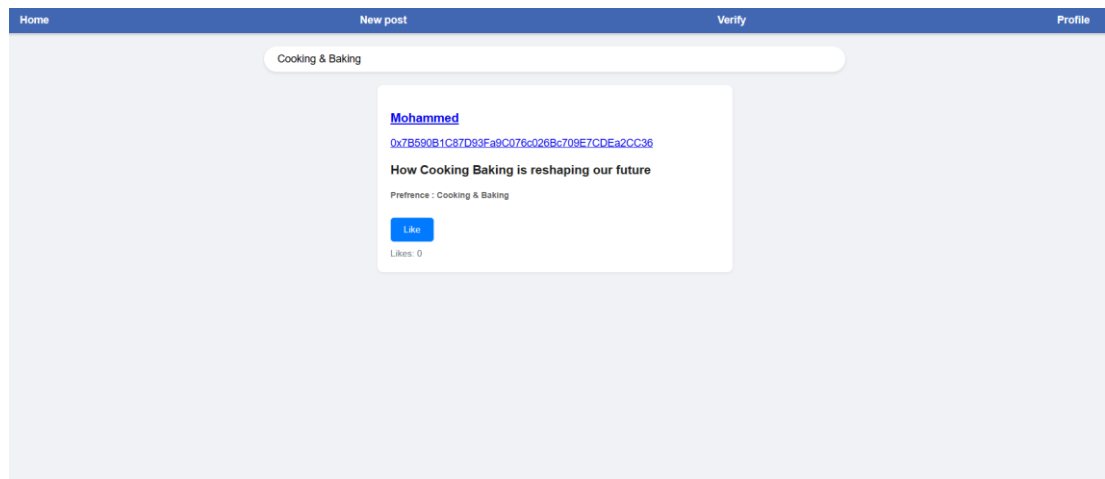
## 2. Blockchain Integration

- **Initial Setup:** Upon accessing the website, the interface interacts with MetaMask to secure user authentication and load the necessary blockchain application binary interface (ABI) from local storage.
- **User Verification:** User verification is performed through blockchain checks, with the interface updating to show personalized user data or prompt registration depending on the verification outcome.
- **Dynamic Content Loading:** Posts are dynamically fetched and displayed based on user interactions, with JavaScript managing backend calls and front-end updates efficiently.

3. The `displayTopUsers()` function significantly enhances user engagement on the social network platform by dynamically showcasing the most popular users based on their posts' like counts. Upon invocation, it initially calls `getTopUsers()` to retrieve all post IDs, aggregate likes per user, and compile a list of top contributors. This leaderboard is then populated with usernames and their respective like counts, formatted in a clear list, providing immediate visual insight into the most active users. Stylistically, the leaderboard is designed to be visually distinct and accessible, fixed on the right side of the screen to remain visible as users scroll. This not only acts as a gamification element, encouraging more interaction and content creation by adding a competitive layer, but also leverages real-time data to reflect community engagement, thus enhancing the interactive appeal and dynamic nature of the platform.

#### 4.4.6. Post searching

In the Figure 4.17, we can see the page that contains a searching bar to search for the posts based on the preferences.



**Figure 4.17.** Searching page.

The JavaScript and HTML snippet provided outlines an Ethereum blockchain-integrated application for a social network, managing posts interactively. Upon page load, it confirms Ethereum connectivity via MetaMask, requesting user account access and listening for account changes to refresh the application setup using the

loadABIAndInitApp function. This function fetches the ABI from a JSON file, essential for smart contract interaction.

Within the initApp function, after verifying the user with verifiedUser, the application introduces a search bar allowing users to input preferences for filtering posts. The loadPosts function fetches all post IDs with getAllPostIds, retrieves each post's content via getPostContent, and dynamically displays posts that match the entered preference in the search bar. Each post is shown on a "post card" displaying the username, wallet address (linked to their profile), content, preference, and number of likes.

Interaction features include liking or unliking posts, managed by hasUserLikedPost, likePost, and unlikePost functions, updating like counts and states dynamically without reloading the page. For posts marked as payable, the "Buy" button triggers the buyPost method, enabling users to purchase posts with Ethereum, thus transferring ownership. Post owners can modify prices through editPostPrice. The script ensures robust user interaction by updating the UI responsively to reflect changes, providing real-time feedback, and facilitating user engagement through a search-driven filtering system that tailors content display based on user preference.

## 5. RESULTS AND DISCUSSION

### 5.1. Results

This section presents the key findings from our research on creating a decentralized social network based on the blockchain with DL and NLP models for classifying social network posts using them.

#### 5.1.1. Performance metrics

First digging into the precise technologies and methodologies employed, it is crucial to establish the performance indicators that were employed to assess the efficacy of our models. These measurements are crucial for understanding the capabilities and efficiency of the suggested solutions in real-world scenarios.

- Accuracy: Often the primary metric selected for assessing the performance of algorithms in classification tasks, accuracy is defined as the proportion of correctly classified items to the overall number of observations we can see the formula (5.1). Although accuracy is commonly employed, it is not always the best indicator of performance, particularly in situations where the target[69].

$$\text{Accuracy} = \frac{T_p + T_n}{T_p + T_n + F_p + F_n} \quad (5.1)$$

- Precision: Also known as the positive predictive value, this metric reflects the ratio of true positive predictions to the total predicted positives we can see that in the formula (5.2). It is crucial for scenarios where the cost of false positives is high[69].

$$\text{Precision} = \frac{T_p}{T_p + F_p} \quad (5.2)$$

- Recall: This metric quantifies a model's capacity to identify all the pertinent instances in a given dataset and formula (5.3) show us the Recall equation. High recall is crucial in situations where failing to identify a positive event has severe consequences[69].

$$\text{Recall} = \frac{T_p}{T_p + F_n} \quad (5.3)$$

- F1-Score: The F1-Score is a metric that calculates the weighted average of Precision and Recall and formula (5.4) represent the F1-Score equation. Thus, this score considers both incorrect positive and incorrect negative results. It is especially useful when the class distribution is uneven[69].

$$\text{F1 - Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (5.4)$$

So now let us know why we are using these technologies in our social network instead of others.

### 5.1.2. Category prediction and suicide detection

The decision to use a Bidirectional LSTM (BiLSTM) for post-category prediction and a regular LSTM for suicide detection can be influenced by the specific characteristics and requirements of each task.

Here are some reasons for these choices:

- BiLSTM for Post-Category Prediction

In the domain of post-category prediction, the ability to comprehend the context encompassing both the antecedent and subsequent text is pivotal. Often, the interpretation and classification of posts are contingent upon nuanced language that may derive significant meaning from sentences or phrases positioned both prior to and following a particular segment. Bidirectional Long Short-Term Memory networks (BiLSTMs) excel in such environments by processing text from both directions, thus capturing an extensive range of contextual information, which enhances the accuracy of categorizing posts. This dual-directional approach is particularly advantageous when the categories are varied and necessitate a profound understanding of the linguistic elements within the posts, as it provides a holistic view of language patterns crucial for precise classification. Additionally, BiLSTMs are

adept at leveraging features such as specific keywords or phrases, regardless of their placement within the post be it at the beginning or the end ensuring that these critical indicators are effectively utilized in prediction processes. This capability of BiLSTMs to integrate and synthesize contextual cues from both directions of the text substantiates their utility in complex categorization tasks, where both contextual richness and feature utilization are essential for achieving high accuracy in predictions[70][71]. Empirical evidence supporting this claim can be seen in the comparative results presented in Table 5.1, where BiLSTMs significantly outperform GRU and RNN models across various metrics such as Accuracy, F1-Score, Precision, and Recall, underscoring the effectiveness of BiLSTMs in handling complex linguistic tasks with higher precision and reliability.

**Table 5.1.** Performance comparison between BiLSTM, GRU, and RNN.

Model	Accuracy	F1-Score	Precision	Recall
BiLSTM	80.58%	79.62%	80.10%	80.58%
GRU	78.40%	76.08%	75.47%	78.40%
RNN	30.71%	21.38%	18.26%	30.71%

- LSTM for Suicide Detection

In tasks such as suicide detection, it is crucial to identify specific expressions or distress signals that typically manifest in certain sections of the text. The ability to understand text sequentially from the beginning to the identified point of interest is essential, as these indicators are usually confined to specific statements or follow a distinct narrative. Considering the sensitive and urgent nature of suicide detection, employing a computationally less intensive model like the Long Short-Term Memory (LSTM) network may offer significant advantages. LSTMs can be trained and deployed more rapidly than their bidirectional counterparts (BiLSTMs), making them more suitable for real-time or large-scale applications where processing speed is paramount. Furthermore, if empirical testing indicates that an LSTM achieves comparable accuracy to a BiLSTM in detecting suicidal tendencies, the simpler LSTM architecture would be favored due to its reduced computational requirements and ease

of management. This approach not only aligns with the need for prompt and efficient processing but also ensures that the system remains robust and effective in high-stakes environments. Performance comparisons such as those detailed in Table 5.2 between LSTM, GRU, and RNN models illustrate the relative effectiveness of these technologies in such critical applications[72].

**Table 5.2.** Performance comparison between LSTM, GRU, and RNN.

Model	Accuracy	F1-Score	Precision	Recall
LSTM	94.30%	94%	94%	95%
GRU	93.78%	94%	95%	93%
RNN	64.04%	47%	89%	32%

### 5.1.3. Similarity detection

In the development of our social network, addressing the challenge of effectively detecting similarities between posts, which vary significantly in length from brief updates to extensive articles, necessitated a judicious choice in the method of similarity detection. Among the various methods available for assessing text similarity, we selected Cosine Similarity coupled with TF-IDF over alternative approaches for several compelling reasons. A key advantage of employing Cosine Similarity with TF-IDF lies in its adaptability to texts of varying lengths. Unlike other methods that may struggle with or require preprocessing to effectively handle texts of disparate lengths, Cosine Similarity inherently accommodates this variability by focusing on the angle between vectors in a multi-dimensional space, rather than their magnitude, making it particularly well-suited to the diverse content on our social network[73]. Furthermore, TF-IDF augments Cosine Similarity by weighting terms based on their importance within the text and across the corpus. This feature proves advantageous for a social network platform, where the significance of specific terms can vary greatly between posts. Other methods might neglect the nuanced importance of less frequent but more meaningful words, resulting in less accurate similarity assessments. By integrating TF-IDF, our similarity detection emphasizes semantic relevance, which aligns with the varied nature of posts on our platform [74]. Additionally, our decision was influenced



by the computational efficiency of the TF-IDF and Cosine Similarity combination. In a dynamic social network environment, where posts are continually created and compared, computational resources are at a premium. Some alternative methods, especially those involving deep learning or extensive lexical databases, can impose significant computational demands. In contrast, Cosine Similarity with TF-IDF offers an optimal balance between accuracy and computational cost, facilitating real-time similarity detection without compromising the platform's responsiveness [73]. Moreover, for content creators and moderators on our social network, understanding why certain posts are flagged as similar is as important as the detection itself. Many advanced similarity detection methods, particularly those based on neural networks, act as "black boxes," offering minimal insight into their decision-making processes. However, Cosine Similarity with TF-IDF provides greater transparency and interpretability, ensuring that decisions based on similarity detection are justifiable and understandable, which fosters trust in the platform's content management processes [74].

Here in, we present a detailed comparison utilizing texts selected to test the methods that exhibit an approximate 80% similarity between Post 1 and Post 2, derived from our post's dataset. For this comparison, we specifically chose two categories of posts: two long, two normal-length posts and two short posts, to conduct the analysis.

In the realm of text analysis, different methods of similarity measurement provide unique insights depending on the context and nature of the data being examined. Cosine Similarity, both with and without TF-IDF, serves as a robust tool for measuring the angle between two vectors, with the integration of TF-IDF offering a more sophisticated analysis by weighting terms according to their relevance across documents. This advanced similarity measure is particularly advantageous in document retrieval systems or plagiarism detection where the relevance of terms is critical. In contrast, Levenshtein Distance, which calculates the minimum number of single-character edits required to transform one string into another, is ideally suited for applications that deal with minor text variations such as auto-correction or search queries. It focuses on character-level changes rather than semantic similarity. On the other hand, the Jaccard Index, measuring similarity based on the intersection over the union of sets, is appropriate for comparing sets of words or tags, useful in simpler,

unweighted comparisons such as keyword matching in content recommendation systems. These distinctions underscore the varying applicability and effectiveness of each method in different scenarios. For a comprehensive evaluation, refer to Table 5.3, which quantitatively compares the performance of Cosine Similarity with TF-IDF against Jaccard Index, Levenshtein Distance, and Cosine Similarity alone across different post lengths. The results elucidate the relative strengths and weaknesses of each method, providing empirical support for choosing the appropriate similarity measurement technique based on specific requirements of text analysis.

**Table 5.3.** The comparison of cosine similarity with TF-IDF and the other methods.

Model	Long post	Normal post	Short post
<b>Cosine Similarity with TF-IDF</b>	63.4%	32%	60.2%
<b>Jaccard Index</b>	6.94%	8.16%	40%
<b>Levenshtein Distance</b>	33.7%	29.4%	74.5%
<b>Cosine Similarity alone</b>	63.4%	32%	60.2%

## 5.2. Discussion

The integration of blockchain technology, deep learning (DL), and natural language processing (NLP) into the design and implementation of a decentralized social network represents a significant advancement in addressing key challenges inherent in traditional, centralized social networks. This section delves into the implications, benefits, and potential limitations of the implemented system, based on the results obtained from the experimental analysis.

### **5.2.1. Implications for privacy and data security**

One of the foremost benefits of employing blockchain technology within social networks is the enhanced security and privacy it offers. Traditional social networks often store user data in centralized servers, making them vulnerable to data breaches and unauthorized access. The decentralized nature of blockchain ensures that data is distributed across a network of nodes, making it significantly harder for malicious entities to compromise the system. Additionally, blockchain's immutable ledger ensures that once data is recorded, it cannot be altered or deleted, thus maintaining data integrity.

The implementation demonstrated that blockchain could effectively safeguard user data, ensuring privacy and trust. By decentralizing data storage and employing cryptographic techniques, the system mitigates risks associated with centralized data management, such as data tampering and unauthorized data access.

### **5.2.2. Enhanced content authenticity and moderation**

Deep learning and NLP play a crucial role in enhancing content authenticity and moderation within the social network. DL algorithms are adept at identifying patterns and extracting features from large datasets, which is essential for categorizing and filtering content. The results showed that DL models could accurately predict user preferences and detect suicidal tendencies in posts, which is critical for providing timely interventions and ensuring a safer online environment.

NLP techniques further enhance the platform's capabilities by enabling the system to understand and process human language effectively. This includes detecting similarities between posts and identifying plagiarism, which are pivotal in maintaining content authenticity and preventing the spread of misinformation. The integration of NLP allowed for more sophisticated content analysis, enabling automated content moderation and enhancing user experience by delivering relevant and engaging content.

### **5.2.3. User empowerment and control**

A significant advantage of the proposed system is the empowerment of users through enhanced control over their data. Traditional social networks often exploit user data for commercial gain without providing adequate compensation or control to the users. In contrast, the decentralized architecture of the blockchain-based social network ensures that users retain ownership of their data and have greater control over how it is used.

The system's use of smart contracts facilitates transparent and secure interactions, allowing users to engage in transactions and data exchanges without relying on a central authority. This not only enhances user trust but also promotes a more equitable distribution of value generated from user-generated content.

### **5.2.4. Challenges and limitations**

Despite the numerous advantages, the implementation of a blockchain-based social network is not without challenges. One of the primary issues is scalability. Blockchain networks, especially public blockchains, often face scalability problems due to the extensive computational resources required for transaction validation and consensus mechanisms. This can result in slower transaction times and higher operational costs, which may hinder the widespread adoption of the system.

Another challenge is the interpretability of deep learning models. While DL algorithms are highly effective at analyzing data, they are often considered "black boxes" due to their complex and opaque nature. This lack of transparency can be problematic, particularly in applications where understanding the decision-making process is crucial.

Furthermore, the integration of blockchain and DL technologies requires significant computational power and infrastructure, which can be a barrier for smaller organizations or individual users. Ensuring accessibility and affordability of the technology remains a critical consideration for the future development and adoption of decentralized social networks.

### **5.2.5. Future directions**

The findings from this study suggest several avenues for future research and development. Enhancing the scalability of blockchain networks through innovations such as sharding and off-chain solutions could address current limitations and improve the system's efficiency. Additionally, developing more interpretable DL models and incorporating explainable AI techniques could enhance transparency and user trust in the system.

Exploring the integration of other emerging technologies, such as edge computing and federated learning, could further optimize the performance and security of decentralized social networks. By continuing to refine and innovate upon the existing framework, it is possible to create a more secure, transparent, and user-centric social networking platform that addresses the challenges of modern digital communication.



## **6. CONCLUSION AND FUTURE WORK**

### **6.1. Conclusion**

This thesis explores the fusion of blockchain technology, deep learning (DL), and natural language processing (NLP) to create a decentralized social network with the goal of tackling privacy infringements and unethical data manipulation that are inherent in centralized social networks. The main goal was to develop a social media platform that is secure, transparent, and focused on the needs of the users. The research findings suggest that the decentralized and immutable ledger system of blockchain greatly improves data security and privacy by distributing data across a network of nodes and using cryptographic techniques. This technique successfully reduces the dangers linked to centralized data storage, such as data breaches and illegal access. Deep learning (DL) and natural language processing (NLP) algorithms are essential for improving the accuracy and control of content authenticity and moderation on social networks. The deep learning models exhibited a notable level of effectiveness in classifying material, forecasting user preferences, and identifying suicidal inclinations in posts. These capabilities are crucial for delivering prompt responses and guaranteeing a more secure online setting. In addition, the application of NLP techniques has enhanced the system's capacity to comprehend and process human language with efficacy. This includes the ability to discover similarities between postings and detect instances of plagiarism, which are crucial in upholding the validity of content and curbing the dissemination of disinformation. The blockchain-based social network's decentralized architecture empowers users by granting them enhanced authority over their data. Conventional social networks frequently utilize user data for financial benefit without offering sufficient compensation or control to the users. On the other hand, the suggested solution guarantees that users maintain ownership of their data and exert more authority over its usage. Smart contracts enable transparent

and secure interactions, enabling users to participate in transactions and data exchanges without the need for a central authority. Nevertheless, the establishment of a social network based on blockchain technology presents several obstacles. Scalability is a significant concern. Blockchain networks, particularly public blockchains, frequently encounter scalability issues as a result of the substantial computational resources needed for transaction validation and consensus methods. This can lead to decreased transaction speed and increased operational expenses, thus impeding the mainstream acceptance of the system. Deep learning models pose another obstacle in terms of their interpretability. DL algorithms are extremely efficient in data analysis, but their intricate and obscure nature often leads to them being referred to as "black boxes". The absence of transparency can pose challenges, especially in applications where comprehending the decision-making process is vital. Furthermore, the integration of blockchain and DL technologies requires significant computational power and infrastructure, which can be a barrier for smaller organizations or individual users. Ensuring that decentralized social networks are accessible and affordable is crucial for their future development and adoption. Future research should prioritize the development and implementation of cutting-edge technologies, such as sharding and off-chain solutions, to improve scalability. Furthermore, it is imperative to prioritize the development of DL models that are easier to understand and to integrate explainable AI techniques in order to improve transparency and foster user trust. By examining the incorporation of additional cutting-edge technologies like edge computing and federated learning, it is possible to enhance the efficiency and safety of decentralized social networks. It is crucial to prioritize the accessibility and cost of these technologies for smaller companies and individual consumers. Ultimately, the combination of blockchain, deep learning, and natural language processing presents a hopeful resolution for revolutionizing social networks. The suggested approach has the potential to revolutionize the social media ecosystem by addressing privacy concerns, improving content authenticity, and empowering users. This study offers a thorough structure for creating a social networking platform that is more safe, transparent, and user-oriented. It sets the stage for future advancements in digital communication. The challenges identified emphasize the necessity of ongoing enhancement and originality in order to fully achieve the advantages of these technologies. In general, this thesis adds to the



continuous endeavors to establish a more secure, fair, and user-focused online atmosphere.

## **6.2. Future work**

The incorporation of blockchain, DL, and NLP in social networks signifies a notable progress. However, there are numerous aspects that require additional investigation and enhancement in order to fully exploit the capabilities of these technologies. An essential obstacle found in this study is the capacity for blockchain networks to scale. Future research should prioritize the creation and execution of inventive strategies to improve scalability, such as sharding, which involves dividing the blockchain network into smaller, more easily manageable sections, and off-chain solutions, which involve processing transactions outside of the primary blockchain to alleviate congestion. In addition, investigating layer-two alternatives, including as state channels and sidechains, could additionally mitigate scalability concerns and enhance transaction performance. Although DL models have shown considerable efficacy in many tasks, their opaque nature presents difficulties in comprehending and analyzing their decision-making mechanisms. Subsequent investigations should focus on creating DL models that are easier to understand, by integrating techniques from explainable AI (XAI) to improve transparency and user confidence. It will be essential to provide tools and frameworks that can effectively display and analyze the outputs of deep learning models. The integration of blockchain, DL, and NLP with other new technologies offers a promising opportunity for investigation. Edge computing allows decentralized social networks to handle data in close proximity to its origin, resulting in reduced latency and enhanced real-time capabilities. Similarly, federated learning, a technique that enables training models on several decentralized devices while keeping data localized, can improve privacy and minimize the requirement for centralized data aggregation. Subsequent research should explore the integration of these technologies to establish decentralized social networks that are more efficient, safe, and scalable. Furthermore, it is imperative to prioritize research efforts towards improving user experience and involvement in decentralized social networks. This can be achieved by creating advanced algorithms for content recommendation and investigating novel interaction paradigms like augmented reality (AR) and virtual reality (VR). It is crucial

to prioritize the accessibility and affordability of decentralized social network technologies for smaller companies and individual users. This requires investigating cost-effective deployment options and creating user-friendly interfaces. It is of utmost importance to solve ethical and regulatory problems as decentralized social networks become more popular. Subsequent studies should focus on constructing frameworks and norms to guarantee the ethical use of these technologies, specifically in domains such as data privacy, content moderation, and user permission. Additionally, it is crucial to collaborate with legislators in order to influence the creation of favorable regulations. Ultimately, performing longitudinal studies to evaluate the enduring effects of decentralized social networks on user conduct, privacy, and data security, in addition to implementing them in real-world scenarios and conducting pilot projects, will yield valuable knowledge and pinpoint practical obstacles. This will establish a feedback loop for ongoing improvement and enhancement.

## REFERENCES

- [1] Tama, D., & Arya Wicaksana. (2023). Performance Evaluation of Decentralized Social Media on Near Protocol Blockchain. 2023 17th International Conference on Ubiquitous Information Management and Communication (IMCOM). <https://doi.org/10.1109/imcom56909.2023.10035654>
- [2] Guidi, B., Michienzi, A., Ricci, L., Baiardi, F., Lucía Gómez-Zaragozá, Carrasco-Ribelles, L. A., & Marín-Morales, J. (2023). SentiTrust: A New Trust Model for Decentralized Online Social Media. *IEEE Access*, 11, 53401–53417. <https://doi.org/10.1109/access.2023.3281194>
- [3] de Oliveira, N. R., Pisa, P. S., Lopez, M. A., de Medeiros, D. S. V., & Mattos, D. M. F. (2021). Identifying Fake News on Social Networks Based on Natural Language Processing: Trends and Challenges. *Information*, 12(1), 38. <https://doi.org/10.3390/info12010038>
- [4] Cadwalladr, C., & Graham-Harrison, E. (2018, March 17). Revealed: 50 million facebook profiles harvested for cambridge analytica in major data breach. *The Guardian*. <https://www.theguardian.com/news/2018/mar/17/cambridge-analytica-facebook-influence-us-election>
- [5] Quach, S., Thaichon, P., Martin, K. D., Weaven, S., & Palmatier, R. W. (2022). Digital technologies: Tensions in Privacy and Data. *Journal of the Academy of Marketing Science*, 50(1), 1299–1323. Springer. <https://link.springer.com/article/10.1007/s11747-022-00845-y>
- [6] Gorwa, R., Binns, R., & Katzenbach, C. (2020). Algorithmic Content moderation: Technical and Political Challenges in the Automation of Platform Governance. *Big Data & Society*, 7(1), 205395171989794. <https://doi.org/10.1177/2053951719897945>
- [7] Zeng, S., Yuan, Y., & Wang, F.-Y. (2019, November 1). A decentralized social networking architecture enhanced by blockchain. *IEEE Xplore*. <https://doi.org/10.1109/SOLI48380.2019.8955104>
- [8] Jiang, L., & Zhang, X. (2019). BCOSN: A Blockchain-Based Decentralized Online Social Network. *IEEE Transactions on Computational Social Systems*, 6(6), 1454–1466. <https://doi.org/10.1109/tcss.2019.2941650>
- [9] Bhusare, P. K., Mannem, B. M. R., & K, A. P. (2023, May 1). Decentralised Social Media. *IEEE Xplore*. <https://doi.org/10.1109/ViTECoN58111.2023.10157136>
- [10] Yan, Z., Li Ning Peng, Feng, W., & Yang, L. T. (2021). Social-Chain. 21(1), 1–28. <https://doi.org/10.1145/3419102>
- [11] Chen, Yize & Li, Quanlai & Wang, Hao. (2018). Towards Trusted Social Networks with Blockchain Technology.

- [12] Bharambe, A., Akshaya Arun Chandorkar, & Dhanajay Kalbande. (2021). A Deep Learning Approach for Dengue Tweet Classification. 2021 Third International Conference on Inventive Research in Computing Applications(ICIRCA). <https://doi.org/10.1109/icirca51532.2021.9544862>
- [13] Cheng, L.-C., & Tsai, S.-L. (2019). Deep learning for automated sentiment analysis of social media. Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. <https://doi.org/10.1145/3341161.3344821>
- [14] Monti, F., Frasca, F., Davide Eynard, Mannion, D., & Bronstein, M. M. (2019). Fake News Detection on Social Media using Geometric Deep Learning. ArXiv (Cornell University). <https://doi.org/10.48550/arxiv.1902.06673>
- [15] Zhang, R. (2022). Research on Social Media Feature Learning Algorithm Based on Deep Neural Network. 2022 IEEE 2nd International Conference on Power, Electronics and Computer Applications (ICPECA). <https://doi.org/10.1109/icpeca53709.2022.9719080>
- [16] Sharma, N., & Prashant Karwasra. (2023). Suicidal Text Detection on Social Media for Suicide Prevention Using Deep Learning Models. <https://doi.org/10.1109/tencon58879.2023.10322499>
- [17] Kaddari, Z., Mellah, Y., Berrich, J., Belkasmi, M. G., & Bouchentouf, T. (2020). Natural Language Processing: Challenges and Future Directions. Artificial Intelligence and Industrial Applications, 236–246 .[https://doi.org/10.1007/978-3-030-53970-2\\_22](https://doi.org/10.1007/978-3-030-53970-2_22)
- [18] Puertas, E., Moreno-Sandoval, L. G., Redondo, J., Alvarado-Valencia, J. A., & Pomares-Quimbaya, A. (2021). Detection of Sociolinguistic Features in Digital Social Networks for the Detection of Communities. Cognitive Computation, 13(2),518–537. <https://doi.org/10.1007/s12559-021-09818-9>
- [19] Puertas, E., Moreno-Sandoval, L. G., Redondo, J., Alvarado-Valencia, J. A., & Pomares-Quimbaya, A. (2021). Detection of Sociolinguistic Features in Digital Social Networks for the Detection of Communities. Cognitive Computation, 13(2), 518–537. <https://doi.org/10.1007/s12559-021-09818-9>
- [20] Chen, N., & Cho, D. S.-Y. (2021, January 1). A Blockchain based Autonomous Decentralized Online Social Network. IEEE Xplore. <https://doi.org/10.1109/ICCECE51280.2021.9342564>
- [21] Tharuka Sarathchandra, & Damith Jayawikrama. (2021). A decentralized social network architecture. <https://doi.org/10.1109/scse53661.2021.9568334>
- [22] Zeng, S., Yuan, Y., & Wang, F.-Y. (2019, November 1). A decentralized social networking architecture enhanced by blockchain. IEEE Xplore. <https://doi.org/10.1109/SOLI48380.2019.8955104>
- [23] Keshk, M., Turnbull, B., Moustafa, N., Vatsalan, D., & Choo, K.-K. R. (2020). A Privacy-Preserving-Framework-Based Blockchain and Deep Learning for Protecting Smart Power Networks. IEEE Transactions on Industrial Informatics, 16(8), 5110–5118. <https://doi.org/10.1109/tii.2019.2957140>

- [24] D. Palanikkumar, Arun, G., R Arunadevi, Gayathri, S. R., & P Dharun. (2023). An Enhanced Decentralized Social Network based on Web3 and IPFS using Blockchain. <https://doi.org/10.1109/icoei56765.2023.10125612>
- [25] Shahbazi, Z., & Byun, Y.-C. (2021). Fake Media Detection Based on Natural Language Processing and Blockchain Approaches. *IEEE Access*, 9,128442–128453. <https://doi.org/10.1109/access.2021.3112607>
- [26] Tama, D., & Arya Wicaksana. (2023). Performance Evaluation of Decentralized Social Media on Near Protocol Blockchain. 2023 17th International Conference on Ubiquitous Information Management and Communication (IMCOM). <https://doi.org/10.1109/imcom56909.2023.10035654>
- [27] Guidi, B., Michienzi, A., Ricci, L., Baiardi, F., Lucía Gómez-Zaragozá, Carrasco-Ribelles, L. A., & Marín-Morales, J. (2023). SentiTrust: A New Trust Model for Decentralized Online Social Media. *IEEE Access*, 11,53401–53417. <https://doi.org/10.1109/access.2023.3281194>
- [28] Wang, H., Zheng, Z., Xie, S., Dai, H. N., & Chen, X. (2018). Blockchain challenges and opportunities: a survey. *International Journal of Web and Grid Services*, 14(4),352–375. <https://doi.org/10.1504/ijwgs.2018.10016848>
- [29] Saputhanthri, A., De Alwis, C., & Liyanage, M. (2022). Survey on Blockchain-Based IoT Payment and Marketplaces. *IEEE Access*, 10, 103411–103437. <https://doi.org/10.1109/access.2022.3208688>
- [30] Ali, O., Jaradat, A., Kulakli, A., & Abuhlimeh, A. (2021). A Comparative Study: Blockchain Technology Utilization Benefits, Challenges and Functionalities. *IEEE Access*, 9(1), 12730–12749. <https://doi.org/10.1109/access.2021.3050241>
- [31] Dozier, P. D., & Montgomery, T. A. (2020). Banking on Blockchain: An Evaluation of Innovation Decision Making. *IEEE Transactions on Engineering Management*, 67(4), 1129–1141. <https://doi.org/10.1109/tem.2019.2948142>
- [32] Wang, Q., Huang, J., Wang, S., Chen, Y., Zhang, P., & He, L. (2020). A Comparative Study of Blockchain Consensus Algorithms. *Journal of Physics: Conference Series*, 1437, 012007. <https://doi.org/10.1088/1742-6596/1437/1/012007>
- [33] Parizo, C. (2021, May 28). What are the 4 different types of blockchain technology? SearchCIO. <https://www.techtarget.com/searchcio/feature/What-are-the-4-different-types-of-blockchain-technology>
- [34] Khan, S. N., Loukil, F., Ghedira-Guegan, C., Benkhelifa, E., & Bani-Hani, A. (2021). Blockchain smart contracts: Applications, challenges, and future trends. *Peer-To-Peer Networking and Applications*, 14(1), 2901–2925. <https://doi.org/10.1007/s12083-021-01127-0>
- [35] Subakan, C., Ravanelli, M., Cornell, S., Bronzi, M., & Zhong, J. (2021, June 1). Attention Is All You Need In Speech Separation. *IEEE Xplore*. <https://doi.org/10.1109/ICASSP39728.2021.9413901>

- [36] Zhavoronkov, A., Ivanenkov, Y. A., Aliper, A., Veselov, M. S., Aladinskiy, V. A., Aladinskaya, A. V., Terentiev, V. A., Polykovskiy, D. A., Kuznetsov, M. D., Asadulaev, A., Volkov, Y., Zholus, A., Shayakhmetov, R. R., Zhebrak, A., Minaeva, L. I., Zagribelnyy, B. A., Lee, L. H., Soll, R., Madge, D., & Xing, L. (2019). Deep learning enables rapid identification of potent DDR1 kinase inhibitors. *Nature Biotechnology*, 37(9), 1038–1040. <https://doi.org/10.1038/s41587-019-0224-x>
- [37] Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639), 115–118. <https://doi.org/10.1038/nature21056>
- [38] Grigorescu, S., Trasnea, B., Cocias, T., & Macesanu, G. (2019). A survey of deep learning techniques for autonomous driving. *Journal of Field Robotics*, 37(3). <https://doi.org/10.1002/rob.21918>
- [39] You, C., Lu, J., Filev, D., & Tsiotras, P. (2019). Advanced planning for autonomous vehicles using reinforcement learning and deep inverse reinforcement learning. *Robotics and Autonomous Systems*, 114, 1–18. <https://doi.org/10.1016/j.robot.2019.01.003>
- [40] Zhang, L., Wang, S., & Liu, B. (2018). Deep Learning for Sentiment Analysis : A Survey. *ArXiv (Cornell University)*. <https://doi.org/10.48550/arxiv.1801.07883>
- [41] Masi, I., Wu, Y., Hassner, T., & Natarajan, P. (2018). Deep Face Recognition: A Survey. 2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI). <https://doi.org/10.1109/sibgrapi.2018.00067>
- [42] Zhao, Z.-Q., Zheng, P., Xu, S.-T., & Wu, X. (2019). Object Detection With Deep Learning: A Review. *IEEE Transactions on Neural Networks and Learning Systems*, 30(11), 3212–3232. <https://doi.org/10.1109/tnnls.2018.2876865>
- [43] El Naby, A. A., El-Din Hemdan, E., & El-Sayed, A. (2021). Deep Learning Approach for Credit Card Fraud Detection. 2021 International Conference on Electronic Engineering (ICEEM). <https://doi.org/10.1109/iceem52022.2021.9480639>
- [44] Théate, T., & Ernst, D. (2021). An application of deep reinforcement learning to algorithmic trading. *Expert Systems with Applications*, 173, 114632. <https://doi.org/10.1016/j.eswa.2021.114632>
- [45] Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data*, 8(1). <https://doi.org/10.1186/s40537-021-00444-8>
- [46] Martin, C. H., Peng, T. (Serena), & Mahoney, M. W. (2021). Predicting trends in the quality of state-of-the-art neural networks without access to training or testing data. *Nature Communications*, 12(1), 4122. <https://doi.org/10.1038/s41467-021-24025-8>

- [47] Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., Bridgland, A., Meyer, C., Kohl, S. A. A., Ballard, A. J., Cowie, A., Romera-Paredes, B., Nikolov, S., Jain, R., Adler, J., & Back, T. (2021). Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873),583–589. <https://doi.org/10.1038/s41586-021-03819-2>
- [48] Strubell, E., Ganesh, A., & McCallum, A. (2019). Energy and Policy Considerations for Deep Learning in NLP. ArXiv (Cornell University). <https://doi.org/10.48550/arxiv.1906.02243>
- [49] Rudin, C. (2018). Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead. <https://doi.org/10.48550/arxiv.1811.10154>
- [50] Siami-Namini, S., Tavakoli, N., & Namin, A. S. (2019). The Performance of LSTM and BiLSTM in Forecasting Time Series. 2019 IEEE International Conference on Big Data (Big Data), 3285–3292. <https://doi.org/10.1109/bigdata47090.2019.9005997>
- [51] Khurana, D., Koli, A., Khatter, K., & Singh, S. (2022). Natural Language processing: State of the art, Current Trends and Challenges. *Multimedia Tools and Applications*, 82(3), 3713–3744. <https://doi.org/10.1007/s11042-022-13428-4>
- [52] Artificial Intelligence and Industrial Applications. (2021). In *Advances in intelligent systems and computing*. Springer Nature. <https://doi.org/10.1007/978-3-030-51186-9>
- [53] Li, H. (2017). Deep learning for natural language processing: advantages and challenges. *National Science Review*, 5(1),24–26. <https://doi.org/10.1093/nsr/nwx110>
- [54] Yu, P. S., Han, J., & Christos Faloutsos. (2010). Link Mining: Models, Algorithms, and Applications. In *Springer eBooks*. Springer Nature. <https://doi.org/10.1007/978-1-4419-6515-8>
- [55] Datta, A., Dikaiakos, M. D., Haridi, S., & Iftode, L. (2012). Infrastructures for Online Social Networking Services [Guest editorial]. *IEEE Internet Computing*, 16(3), 10–12. <https://doi.org/10.1109/mic.2012.53>
- [56] Index | Node.js v20.2.0 Documentation. (n.d.). Nodejs.org. <https://nodejs.org/docs/latest/api/>
- [57] npm Documentation. (n.d.). Docs.npmjs.com. <https://docs.npmjs.com/>
- [58] Solidity — Solidity 0.8.25 documentation. (n.d.). Docs.soliditylang.org. <https://docs.soliditylang.org/en/v0.8.25/>
- [59] Truffle Documentation - Truffle Suite. (n.d.). Archive.trufflesuite.com. Retrieved May 20, 2024, from <https://archive.trufflesuite.com/docs/>
- [60] Ganache | Overview - Truffle Suite. (n.d.). Archive.trufflesuite.com. <https://archive.trufflesuite.com/docs/ganache/>
- [61] Home | MetaMask developer documentation. (n.d.). Docs.metamask.io. <https://docs.metamask.io/>

- [62] Documentation. (n.d.). Docs.ethers.org. <https://docs.ethers.org/v5/>
- [63] Python Software Foundation. (2019). 3.7.3 Documentation. Python.org. <https://docs.python.org/3/>
- [64] Flask. (2010). Welcome to Flask — Flask Documentation (3.0.x). Flask.palletsprojects.com. <https://flask.palletsprojects.com/en/3.0.x/>
- [65] HTML Standard. (2019). Whatwg.org. <https://html.spec.whatwg.org/multipage/>
- [66] Cascading Style Sheets. (n.d.). Wwv.w3.org. <https://www.w3.org/Style/CSS/>
- [67] MDN Contributors. (2023, September 25). JavaScript. MDN Web Docs. <https://developer.mozilla.org/en-US/docs/Web/javascript>
- [68] Lai, Y., Yang, J., Liu, M., Li, Y., & Li, S. (2023). Web3: Exploring Decentralized Technologies and Applications for the Future of Empowerment and Ownership. *Blockchains*, 1(2), 111–131. <https://doi.org/10.3390/blockchains1020008>
- [69] Vakili, M., Ghamsari, M., & M. Katayoon Rezaei. (2020). Performance Analysis and Comparison of Machine and Deep Learning Algorithms for IoT Data Classification. *ArXiv* (Cornell University). <https://doi.org/10.48550/arxiv.2001.09636>
- [70] Xu, G., Meng, Y., Qiu, X., Yu, Z., & Wu, X. (2019). Sentiment Analysis of Comment Texts Based on BiLSTM. *IEEE Access*, 7, 51522–51532. <https://doi.org/10.1109/access.2019.2909919>
- [71] Ezen-Can, A. (2020, September 11). A Comparison of LSTM and BERT for Small Corpus. *ArXiv*. <https://doi.org/10.48550/arXiv.2009.05451>
- [72] Bhattacharya, D., N, S. H. K., & A, S. (2021, November 1). Early Detection of Suicidal Tendencies from Text Data using LSTM. *IEEE Xplore*. <https://doi.org/10.1109/i-PACT52855.2021.9696630>
- [73] Sitikhu, P., Pahi, K., Thapa, P., & Shakya, S. (2019). A Comparison of Semantic Similarity Methods for Maximum Human Interpretability. *2019 Artificial Intelligence for Transforming Business and Society (AITB)*. <https://doi.org/10.1109/aitb48515.2019.8947433>
- [74] Resume Shortlisting and Grading using TF-IDF, Cosine Similarity and KNN. (n.d.). *Www.jetir.org*. Retrieved May 20, 2024, from <https://www.jetir.org/view?paper=JETIR2305459>



## **CURRICULUM VITAE**

Name Surname : Amir Al Kadah

### **EDUCATION:**

- **Undergraduate** : 2022, Yarmouk University, Hijjawi Faculty for Engineering Technology, Computer Engineering
- **Graduate** : 2024, Sakarya University, Software Engineering Department, Software Engineering

### **PROFESSIONAL EXPERIENCE AND AWARDS:**

- 2023-2024 Desktop and web applications programmer, Freelancer
- 2022 worked as android developer at orange yarmouk innovation lab “OYIL”, Irbid-Jordan