FAKE NEWS DETECTION IN SOCIAL MEDIA USING BLOCKCHAIN

Kaustubh Katkar, Nikhilesh Reddy Tummala and Santosh Kannan

(3147-0922 8350-1593 9095-5971)

CISE University of Florida, 2020

***Abstract* – Fake news has become a long-standing problem on social media platforms. Such false information is used by malicious attackers to defame people, manipulate elections and much more. Over the course, many different methods have been proposed to tackle this problem. In this paper, we present a blockchain based model, which can help detect fake news in social media. We leverage the peer to peer architecture of the public blockchain network in deeming a news authentic or fake. This is achieved by a rating scheme we propose for each post, wherein, end users of the system are encouraged to rate a post as valid or invalid. All such functionalities are achieved using the smart contracts system provided in the Ethereum blockchain model. This paper also presents a prototype model for the ideology and backs the proposal with some experimental results. Furthermore, this paper identifies the challenges our implementation might face as well as acknowledges the technologies whose collaboration can improve the efficiency of this research.**

I. INTRODUCTION

Blockchain in social media is novel concept which has numerous possible applications. Our model leverages the security advantages of a blockchain network to authenticate shared news on social media. We provide an Ethereum blockchain based prototype to effectively distinguish true and false information shared on the social media platform.

Social media is an easy and quick source for all kinds of global and local news. However, it backfires when malicious individuals spread false information. These individuals / organisations seek to take advantage of people's tendency to share appealing information without verifying the authenticity of the news. Such acts could potentially influence elections negatively, defame well-known people and much more.

To deeply understand the issue regarding fake news, a recent paper titled "*The science of fake news*" [1] has attempted to study the science behind fake news. This answers the questions, what is fake news? how is fake news generated? what are the essential elements in spreading false information? Researchers over the last decade have proposed multiple ways to nullify the effects of fake news. For instance, authors Youngkyung Seo, Deokjin Seo, Chang-Sung Jeong provided a fake news detection model based on text classifier system [2]. Few creators recently explored the Blockchain route to develop a smart contract based fake news detection model [3]. Another paper demonstrates a model based on leveraging user inputs to verify the authenticity of the news [4]. Although recent researches have proposed few ideas using the Blockchain system, none of the solutions have been implemented in the real-world applications.

We build our model using the user input to authenticate news as valid or invalid. News posts are deemed as valid or invalid based on a rating scheme defined in the later sections. We discuss the design of our model and the functioning of our prototype in the following sections.

II. SYSTEM ARCHITECTURE

A. Overview

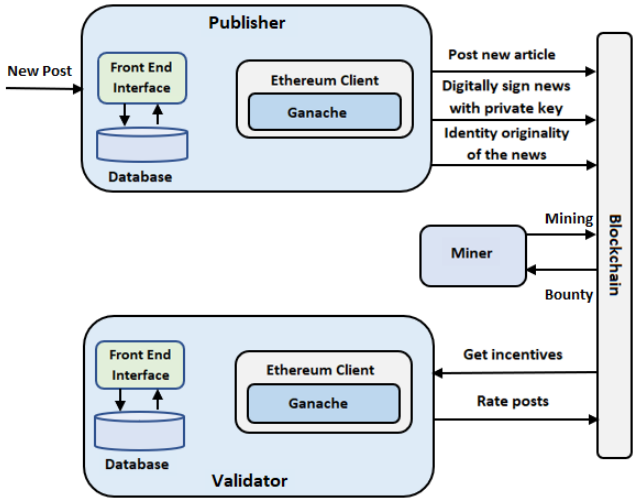
The model proposed below presents the use of blockchain for publishing the news and computing ratings provided by end users on a peer to peer network. We leverage the use of "smart contracts" in blockchain technology to track the published news and change in ratings (such as origin of a news article, users who repeatedly publish false information). We use the Ethereum blockchain [10] model to log time and id of the publisher, Vote count and the corresponding ratings on external databases. On the block of the blockchain, we store the hashed value for the published text to maintain data integrity and reduce the overhead for data storage. Publishers can publish new news articles and validators will provide a corresponding rating for each of the posts. The ratings from the validators are calculated with a weighted model, which ensures the integrity of the system.

Our application also provides a smart contract for publishing a news article. We assign new credentials for each publisher block chain address and confirm the identity of each publisher through public keys. In the following sections, we present the design and working of our model

B. System Description

The primary focus of our design is to build a prototype which leverages blockchain to nullify the effect of false information on the social media platforms. The design for our model broadly assumes two flows in the system: *publishing a news and validating a news*. The system is deployed on a peer to peer network with Ethereum Blockchain client. At a high level, this design contains three components: Ethereum Client, Database and frontend. These are typically hosted on servers in a social media platform. In this report, we provide a prototype implementation with focus on the Ethereum Client which acts as an interface for the blockchain and a social media application. This model can further be utilised

in any social media system by integrating the blockchain model into the existing ecosystem.



Front End Interface: Typically, a front-end system consists of a user interface which allows the user to access the data. In our application, the front-end system would assist the user with features such as login, publishing and rating the news feeds. These actions are performed on the blockchain through the Ethereum client interface with the public / private keys.

Database: The database is a data repository for the entire distributed network. All the posts and images would be physically stored in the databases, designed with data marts. The databases along with the front-end system are responsible for the client and server-side communication of the network.

Ethereum Client: This component is the primary module for implementing the functionality of the blockchain. Briefly, this component is tasked with connecting to the peer to peer network, sending encoded ratings to the network and maintaining a local copy of the blockchain. In our prototype, we use Truffle and Ganache client. The smart contracts are deployed through this client and the addresses of participants are resolved. The smart contracts for publishers and validators would be continuously running within the client. In any event, these contracts would be triggered, and corresponding transactions would be executed on the blockchain.

Blockchain: Blockchain is a continuous chain of blocks. Each block stores a transaction which occurs in the network. Smart contracts are designed and deployed on the blockchain. These contracts are continuously running within the system and are triggered every time a publisher publishes news, or a validator provides rating.

Blockchain plays a key role in two perspectives in the network. For a publisher, the blockchain will store the address of the publisher, timestamp of the post and hashed value of the post. Once a news feed has been posted, validators would be able to access the post through their public keys and rate the feed accordingly. These ratings are stored as a count against the publisher and the post in the blockchain. This information is utilized by the smart contracts for calculating the rating of the publisher as well as the post.

C. Smart Contracts:

Our prototype incorporates smart contracts for publishing posts, voting mechanism for validators and audiences, and providing incentives to the validators.

Publisher: Every news published by a publisher is subject to the terms provided in the smart contract for publishing. The smart contract determines whether the news published was an original or a duplicate post, since it is unnecessary to spend time validating redundant posts. The smart contract would identify the original post and assign the same ratings to the duplicate post. The contract would also calculate the rating for a publisher based on the votes received for each post published by the publisher. This is essential in flagging malicious users who spread false information.

*Voting Validators***:** The validators are verified and authentic accounts who receive incentive for their input. To determine whether an account is authentic will be the responsibility of the social media platform. They can achieve this by associating accounts with their businesses or brands and provide our module with that information. The validators are tasked with rating all the news posts. These ratings are processed through smart contracts to compute the overall rating of the post, rating of the publisher and the individual rating of the validator.

Voting Audiences: The general public or unverified accounts will also be allowed to vote albeit their votes will be displayed in a different format. These ratings will be displayed immediately on the post and will not be an indicator of whether the post is legitimate or not unless it satisfies a geographic constraint. General audiences are not rewarded for their input in our model. Our primary focus will be to determine what format of news attracts the general public to assist analysts of the social media platform.

The geography of an account is an important factor to determine the accuracy of a news story as people belonging to the region can attest to it. In this regard there is a research towards a protocol named FOAM [5] which addresses how they obtain geospatial data and can be incorporated to determine the location of an address. This protocol has matured over 5 years, at the time of writing this report, which is why we are comfortable recommending it. Thus, based on the location we can add the unverified votes to the legitimacy score as well however with a low multiplier. These votes will also be added to the audience votes.

III. FUNCTIONING OF THE PROTOTYPE

As mentioned earlier, the model relies on the communication across the application system (front and back end) and the blockchain ecosystem. Our prototype is deployed using Ganache Ethereum client and Truffle development environment. A detailed working of the model is presented below:

A. Publishing a News

Each publisher is identified as a verified / non verified publisher by the social media organization. This information is used by the smart contract to assign them corresponding user ids. Once a publisher has registered with the system, they can use their credentials to login into the system. This would

navigate them to the user interface of the social media application. The publisher can then upload the post and publish it on the peer to peer network.

B. Publisher Smart Contract

Once the publisher uploads a post, the Ethereum client is responsible for interfacing the social media application and the blockchain. The post will include a digital signature of the publisher, signed with its private key. The post goes through the *'publisher'* smart contract which identifies whether the post is original. This is achieved using the language *now* which returns the current Unix Epoch time of the transaction that increments each second. So, the smallest value is always the earliest. We maintain the timestamp as a constant variable because if all the blocks are not up to date and if within a vast network a post appears where the blocks updated are not yet synchronized, when the entire network gets updated, the timestamp of the post will be checked against same post if such a post exists and the smaller timestamp value wins. This will require another mapping of post to its Epoch time which will be in uint256 format. A private function is used to update this mapping with access privileges only to the owner of the contract.

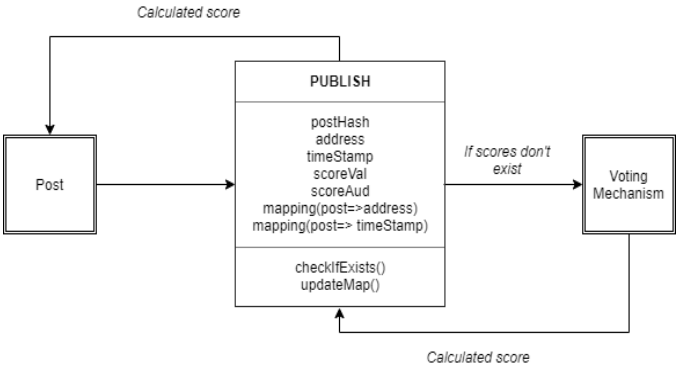


Fig 1. Flow of operations through Publish smart contract

Since we are not concerned with how many times the post has been republished, we will identify the first instance when the post is published and maintain a hash value of the post. Solidity provides *keccak256()* function to achieve this. We map the hash value with the address of the source. At any instance when a post is published the hash value of the newly published post will be calculated and checked if it already exists in the hash. If it does, then the already acquired audience and validator ratings are displayed over the post. Otherwise, the new hash is added to the map along with the address of the sender. The voting mechanism is initiated to compute the scores. We will discuss this in more detail in the Voting smart contract section.

Thus, this smart contract will store the cryptographic hash value of the post, the timestamp and the address of the publisher on the blockchain.

C. Voting by Validators

After a news has been successfully published on the blockchain, the validators can access these posts through their public key and send their ratings. There are three types of ratings associated with this model: rating for a post, rating for the publisher (*rpub*) and rating for the validator (*rval*).

1. Rating for the post

Each validator can rate the authenticity of a post on a scale of 10. These ratings are then calculated with a weighted scheme according to the individual rating of the validator. The validators rating is taken into consideration to factor in the legitimacy of their vote. A validator with a higher *rval* will have a greater influence on the rating of a post compared to a validator with a lower *rval*.

Post Rating Scheme: Consider a news article is published and two validators A and B with respective *rvals* 5 and 10 decide to vote on the article. A voted 5 whereas B voted 6. Our scheme calculates the average rating as weighted scheme as follows:

$$\textit{Average Rating} = \sum \frac{(\textit{rval} \text{ for user} * \text{rating by user})}{\sum \textit{rval} \text{ for user}}$$

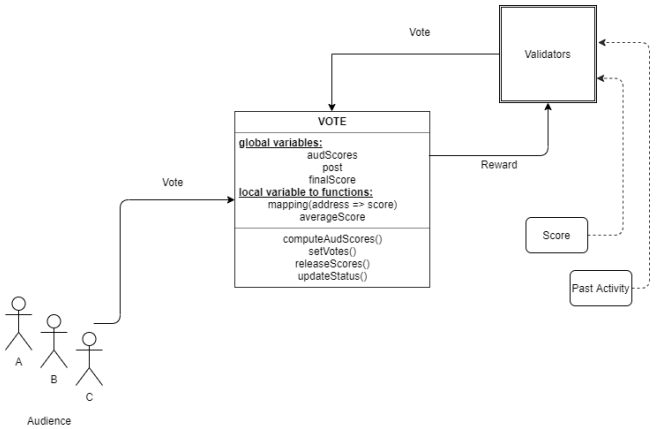
Although, the ratings 5 and 6 are not far apart, the weighted value (*rval* for user * rating by user) i.e. 25 and 60 are significantly different. Since B had a *rval* of 10, it has a higher influence on the overall rating of the post.

Each vote is stored in local variables of functions, which are not visible to the public. This is essential in calculating the *rval* of the validator.

2. Rating for the validator (*rval*)

As presented in the above section, the validator's rating plays an important role in computing the rating of the post. Thus, it is important to implement a method to efficiently calculate the validator rating. Along with this, it is equally important to provide the validators an incentivization scheme for rating as many posts sincerely.

Our smart contract efficiently computes the *rvals* for the validators based on the rating of the posts. For each post, the smart contract will consider a score bracket of + and – 1 of the average rating for the post. All validators who rated within this range will a positive effect on their rating. The ratings of all validators who rated outside the bracket will be negatively affected by a factor of how far their rating was off.



Rval Rating Scheme: Suppose a news article receives an overall rating of 7 after the 24hr voting window. A validator

who voted 8 would have an increase in *rval*, whereas, validator who voted 4 would have a decrease in *rval*.

Incentivization: Each rating by the validator will qualify for an incentive. Since the unit of currency will depend on the organization, we propose the incentivization in terms of factors. Suppose the maximum reward for providing a rating is 1 unit, a validator with a *rval* of 8.5 will qualify for a maximum incentive for 0.85 units. For fairness, we will award the scores of 9 and above full amount i.e. 1 × max reward, as maximum reward as it will be very unlikely for any account have a perfect score. Initially, all validators will begin with a rval of 0.7. This should motivate the validators to provide sincere and accurate ratings to qualify for higher incentives. Do note that the time specific multiplier, which we discuss ahead, with still affect the maximum reward value for each validator.

Since we are tackling spread of fake News, we need to limit the duration within which the voting takes place. Having people vote when the news becomes irrelevant is not a practical scenario. So, we maintain a time limit within which voting will take place. Through our simulations, we decided a decay pattern as below:

All votes received in the first hour after the news is published are eligible for full rewards. Votes received in the next 2 hours have 10% rewards deducted. Every subsequent hour thereafter has an additional penalty of 4%. It follows the pattern:

0.1, 0.1, 0.14, 0.18, 0.22, 0.26, 0.30, …, 0.84

The below chart shows reward eligibility per hour for a validator with *rval* 8.

Hour	Reward
1st	0.8 × 1
2nd	0.8 – (0.8 × 0.1) = 0.72
3rd	0.72 – (0.72 × 0.1) ≈ 0.65
4th	0.65 – (0.65 × 0.14) ≈ 0.56
5th	0.56 – (0.56 × 0.18) ≈ 0.46
6th	0.46 – (0.46 × 0.22) ≈ 0.36
…	…

** This decrement scheme does not affect the *rval* score

After the end of the voting period the rval scores for the validator are updated. The scores and voting average is stored in a private function with access only to the owner of the smart contract and the scores are updated and released after the status of voting period changes to *false*.

3. Rating for the Publisher

Like the rating for each validator, each publisher is rated based on rating history of their news. As seen above, every post is rated by validators and these votes are stored on the blockchain. These scores help determine the authenticity of the uploader as well. This is done by storing the validator score of the post against the address of the uploader. Recurring behaviour of malicious or misleading uploads will gradually determine the status of the account for which we propose a three-strike policy. This is a scheme where a yellow, orange and red tag is incorporated for strike 1, strike 2 and strike 3 respectively. Initially, we keep a buffer of five posts where the publisher will not be assigned any tags.

IV. EXPERIMENTS AND RESEARCH

We implemented our model as a prototype on Ganache (Ethereum based client). In our tests we attained an approximate legitimacy with score 6 and issued a bracket of 3 which deemed the scores 7 and 5 valid as well. In our implementation we rounded off the average score to its nearest integer for the bracket. The validators with scores within this bracket i.e. 5,6 and 7 will thus be receiving full credit for their votes increasing their individual score.

We determined that the scale of 10 works better to identify instances of prejudice and bias as a vote further apart from the consensus will have reasonable doubt. We are currently determining the magnitude of impact it should have on the scores with different distribution models and will provide a more detailed analysis in the next iteration of our report as to visualize how the different distributions of score will impact the validator's reputation.

Weighing the scores: In this setting we had to make sure that validators with a good record of honest votes should be allowed to have a bigger contribution to the voting process than their counterparts with a lower score. Thus, we added weights to their score when averaging the total score.

The test example along with the description of its result is mentioned with its formulation in the FUNCTIONING OF THE PROTOTYPE section (section III).

Rewarding System: Our knowledge of rewards for the input of validators is based on the real-world instances where Google, Facebook and Amazon are rewarding users for their data either through questionnaires or other programs.

As in our case since the validators are professionals in their fields, the reward should be more lucrative. However, we ourselves refrain from taking any liberties and suggesting the reward as we are aware that each industry will have varying economic standing and different allotment of funds therein. Hence, we believe this decision is best left with the social media platform itself.

IV. LIMITATIONS

Since our proposed solution of leverages user feedback to detect fraudulent news on social media through blockchain, the model is susceptible to any security limitations the blockchain inherently possess, such a spartitioning attacks [7], de-anonymization attacks [6], DDos[8] etc. Although the paper does not address these attacks, future researchers can provide enhancements on this system.

Another potential risk in the system is due to the validators having access to the existing average rating of a newsfeed. They could exploit a potential loophole in our system by giving a biased rating to the post to gain more incentives. One way to handle this problem is through hiding the average rating of the posts during the voting window and releasing it thereafter. However, this would be troublesome for the general public to know whether a post is false or genuine, if they view it in the voting window.

The model design presented only identifies replicas of news posted on the social media. It falls short in detecting posts of similar news, written in different tones. Also, providing incentives through cryptocurrencies does not guarantee that the voters would participate in all voting processes.

V. CONCLUSION AND FUTURE SCOPE

Although the limitations presented in the previous section, the proposed model will help tackle the fake news detection problem in any social media ecosystem. Leveraging anonymous user input for determining the authenticity of the news and providing incentives for accurate and honest feedback allows the model to maintain its integrity. Through our prototype we were able to successfully verify the results of our approaches.

Future advancement to this current proposed solution would be to incorporate Machine Learning Algorithms to identify duplicate posts which are written in a different tone or format and categorize it in a fashion that would prevent redundant voting for the same information which was already posted on our system. This could also save the organization from over-spending the incentives to the evaluators for validating the same newsfeed repeatedly

Another enhancement on the proposed system is to eradicate the problem of validators having access to the existing post rating. We aim to design a robust scheme to deal with this potential problem in the next iteration of the paper.

VI. REFERENCES

[1] David MJ Lazer, Matthew A Baum, Yochai Benkler, Adam J Berinsky, Kelly M Greenhill, Filippo Menczer, Miriam J Metzger, Brendan Nyhan, Gordon Pennycook, David Rothschild, et al. The science of fake news. Science, 359(6380):1094–1096, 2018.

[2] FaNDeR: Fake News Detection Model Using Media Reliability -IEEE Conference Publication. [Accessed 22 Mar. 2019].

[3] Adnan Qayyum, Junaid Qadir, Muhammad Umar Janjua, and Falak Sher. Using Blockchain to Rein in The New Post-Truth World and Check the Spread of Fake News. Information Technology University (ITU), Lahore, Pakistan, 2019

[4] Shovon Paul, Jubair Islam Joy, Shaila Sarker, Amit Kumar Das, Sharif Ahmed, Abdullah - Al - Haris Shakib. Fake News Detection in Social Media using Blockchain. 7th International Conference on Smart Computing & Communications (ICSCC), 2019.

[5] E. Leka, L. Lamani, B. Selimi and E. Deçolli, "Design and Implementation of Smart Contract: A use case for geo-spatial data sharing," 2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, Croatia, 2019, pp. 1565-1570

[6] G. Wondracek, T. Holz, E. Kirda and C. Kruegel, "A Practical Attack to De-anonymize Social Network Users," 2010 IEEE Symposium on Security and Privacy, Berkeley/Oakland, CA, 2010, pp. 223-238.

[7] M. Saad, V. Cook, L. Nguyen, M. T. Thai and A. Mohaisen, "Partitioning Attacks on Bitcoin: Colliding Space, Time, and Logic," 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS), Dallas, TX, USA, 2019, pp. 1175-1187.