

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belagavi -590014, Karnataka State, India



A PROJECT REPORT ON
“**DETERMINATION OF FAKE NEWS USING BLOCKCHAIN AND
IBM WATSON**”

Submitted on partial fulfillment of academic requirement for the award the degree

**BACHELOR OF ENGINEERING
IN
INFORMATION SCIENCE AND ENGINEERING**

Submitted by:-

ABISHEK GOWDA B K	1SJ14IS001
ACHYUTH N S	1SJ16IS002
DHANANJAY S	1SJ16IS027
MRUDULA P B	1SJ16IS051

Under the guidance of:-

Prof. NAGARAJ G

Associate Professor,
Dept. Of ISE, SJCIT



S J C INSTITUTE OF TECHNOLOGY
Department of Information Science and Engineering
Chickaballapur – 562101
2019-2020

||Jai Sri Gurudev||

Sri Adichunchanagiri Shikshana Trust®

S J C INSTITUTE OF TECHNOLOGY

Information Science and Engineering Department

Chickballapur-562101



CERTIFICATE

This is to certify that the report entitled **“DETERMINATION OF FAKE NEWS USING BLOCKCHAIN AND IBM WATSON”** is a bonafide work carried out at **SJC Institute Of Technology** by **ABISHEK GOWDA B K (1SJ14IS001)**, **ACHYUTH N S (1SJ16IS002)** , **DHANANJAY S (1SJ16IS027)**, **MRUDULA P B (1SJ16IS051)** in partial fulfilment for the award of **Bachelor of Engineering in Information Science and Engineering of the Visvesvaraya Technological University**, Belagavi during the year **2019-2020**. It is certified that all corrections/suggestions indicated for internal assessment have been incorporated in the report deposited in the department library. The project report has been approved as it satisfies the academic requirements with respect to project work prescribed for the Bachelor of Engineering.

Signature of the Guide

Prof. Nagaraj G
Associate Professor
Dept. of ISE, SJCIT

Signature of the HOD

Prof. Satheesh Chandra Reddy
Associate Prof & HOD
Dept. of ISE, SJCIT

Signature of the Guide

Dr.Ravi Kumar K M
Principal
SJCIT

External Examiners:
Name of the Examiners

signature with Date:

1.

2

DECLARATION

We **ABHISHEK GOWDA B K, ACHYUTH N S, DHANANJAY S, MRUDULA P B** members of 8th Semester B.E Information Science and Engineering, S J C Institute of Technology, Chikaballapur, hereby declare that the Project Work entitled **“DETERMINATION OF FAKE NEWS USING BLOCKCHAIN AND IBM WATSON”** has been independently carried out by us under the supervision of our guide, **Prof. Nagaraja G ,Associate Professor ,S J C Institute of Technology, Chikaballapur** and submitted in fulfilment for the award of degree in Bachelor of Engineering in Information Science and Engineering of Visvesvaraya Technological University, Belagavi, during the academic year 2019-2020. We further declare that the report has not been submitted to any other university for award of any other degree.

ABHISHEK GOWDA BK	(1SJ14IS001)
ACHYUTH N S	(1SJ16IS001)
DHANANJAY S	(1SJ16IS027)
MRUDULA PB	(1SJ16IS051)

ABSTRACT

Due to the exponential growth of information online, it is becoming impossible to decipher the true from the false. Thus, this leads to the problem of fake news. This research considers previous and current methods for fake news detection in textual formats while detailing how and why fake news exists. The extensive spread of fake news has the potential for extremely negative impacts on individuals and society. Therefore, fake news detection has recently become an emerging research that is attracting tremendous attention. Fake news detection presents unique characteristics and challenges that make existing detection algorithms from traditional news media ineffective or not applicable.

The survey identifies and specifies fundamental theories across various disciplines, e.g., psychology and social science, to facilitate and enhance the interdisciplinary research of fake news. Current fake news research is reviewed, summarized and evaluated. These studies focus on fake news from four perspective: (1) the false knowledge it carries, (2) its writing style, (3) its propagation patterns, and (4) the credibility of its creators and spreaders. We characterize each perspective with various analyzable and utilizable information provided by news and its spreaders, various strategies and frameworks that are adaptable, and techniques that are applicable.

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of the people who made it possible, whose constant guidance and encouragement crowned the efforts with success.

We would like to profoundly thank **Management** of **S J C Institute of Technology** for providing such a healthy environment for the successful completion of Seminar work.

We would like to express my thanks to the Principal **Dr. K M RAVIKUMAR** for their encouragement that motivated us for the successful completion of Seminar work.

It gives us immense pleasure to thank **Prof. SATHEESH CHANDRA REDDY** Professor and Head of the Department for their constant support and encouragement.

Also, we would like to thank the Project Coordinator **Prof. NAGARAJA G** ,**Assistant Professor**, Department of Information Science and Engineering and all other teaching and non-teaching staff of Information Science Department who has directly or indirectly helped us in the completion of the project work.

Last, but not the least, we would hereby acknowledge and thank my parents who have been a source of inspiration and also instrumental in the successful completion of the Project work.

ABISHEK GOWDA B K 1SJ14IS001

ACHYUTH N S 1SJ16IS002

DHANANJAY S 1SJ16IS027

MRUDULA P B 1SJ16IS051

TABLE OF CONTENTS

Abstract	i
Table of contents	ii
List of figures and tables	iii
Chapter 1: Introduction	01
Chapter 2: Problem identification and definition	03
Chapter 3: Literature survey	05
Chapter 4: Software Requirement Specifications	08
Chapter 5: Project objectives	10
Chapter 6: Design And Analysis	11
Chapter 7: Implementation	22
Chapter 8: Testing	67
Conclusion	75
Bibliography	76

LIST OF FIGURES

Figure/Code no.	Name	Page no.
Fig. 6.1	Architecture Diagram	11
Fig. 6.2	Flow chart	13
Fig. 6.3	Use case diagram	15
Fig 6.4	Sequence Diagram	16
Fig 6.5	Class Diagram	17
Fig 6.6.1	Data Flow Diagram level: 0	18
Fig 6.6.2	Data Flow Diagram level:1	19
Fig 6.6.3	Data Flow Diagram level:2	21

CHAPTER 1

INTRODUCTION

Fake news is now viewed as one of the greatest threats to democracy, journalism and freedom of expression. In recent years, the topic of fake news has experienced a resurgence of interest in society. The increased attention stems largely from growing concerns around the widespread impact of fake news on public opinion and events. Trending topics make the search works easier and also help people to look for the events happening. However, this freedom to the user to post anything leads to spread of fake information. People are also less likely to check the news and hence it leads to spreading of false misleading information at a faster rate.

It is important to think and evaluate the news critically and it can be done by encountering a claim in the news by checking it against reliable sources and everyone should do their part in not spreading it further. Many incidents have been previously reported of Fake News causing chaos and troubles in certain parts of India and even at an International level. It has weakened public trusting government and its potential impact on the contentious “Brexit” referend amend the equally divisive 2016 U.S. presidential election—which it might have affected—is yet to be realized.

While fake news is not a new phenomenon, questions such as why has it emerged as a world topic and why is it attracting increasingly more public attention are particularly relevant at this time. The leading cause is that fake news can be created and published online faster and cheaper when compared to traditional news media such as newspapers and television. The topic of fake news has increasing. In this regard, there have been attempts to attention from the academic community survey and summarize the literature on fake news detection

Furthermore, as an ideal platform to accelerate fake news disseminations social media breaks the physical distance barrier among individuals, provides rich platforms to share, forward, vote, and review, and encourages users to participate and discuss online news. This surge of activity around online news can lead to grave repercussions, but also substantial potential political and economic benefits. Such generous benefits encourage malicious entities to create, publish and spread fake news. But, how can fake news gain public trust?

Fake news gaining public trust and further facilitate the spread of fake news. For instance, humans have been proven to be irrational and vulnerable when differentiating between truth

and falsehood while over loaded with deceptive information. Studies in social psychology and communications have demonstrated that human ability to detect deception is only slightly better than chance. In addition, individuals tend to trust fake news after repeated exposures validity effect, or if it confirms their pre-existing knowledge. Peer pressure can also at times “control” our perception and behaviour. The theories are provided along with short descriptions. These theories can be used to study fake news from three different perspectives: (I) style: how fake news is written, (II) propagation: how fake news spreads, and (III) users: how users engage with fake news and the role users play in fake news creation, propagation, and intervention. In the following, we detail how each perspective and its corresponding theories facilitate fake news analysis.

I. Style-based Fake News Analysis, these fundamental theories address how fake news content and writing style can be different from true news. For instance, reality monitoring indicates that actual events can be expressed by higher levels of sensory-perceptual information.

II. Propagation-based Fake News Analysis. Epidemic models, which can mathematically model the progression of an infectious disease, can be used or extended to model fake news propagation. However, selecting or developing proper epidemic models relies on making reasonable assumptions. Some real-world phenomena can help simplify these assumptions and in turn, simplify such epidemic models. Examples include backfire effect, conservatism bias and Semmelweis reflex, which indicate that “fake news is incorrect but hard to correct”, i.e., it propagates with minimum resistance.

III. User-based Fake News Analysis. These theories investigate fake news from a user’s perspective, considering how users engage with fake news and what roles users play in fake news creation, propagation and intervention. In sum, users that participate in fake news activities can be grouped into

(i) malicious users, who intentionally create and/or propagate fake news motivated by some benefits and

(ii) normal users, some of whom spread fake news along with malicious users. These normal users are often called naïve users as their engagement is unintentional and driven by self-influence or social influence, e.g., naïve users can participate in fake news spreading due to their preexisting knowledge.

CHAPTER 2

PROBLEM IDENTIFICATION AND DEFINITION

2.1 PROBLEM STATEMENT

The false information is often caused by reporters paying sources for stories, an unethical practice called checkbook journalism. Digital news has brought back and increased the usage of fake news, or yellow journalism. The news is then often reverberated as misinformation in social media but occasionally finds its way to the mainstream media as well.

We have theoretically differentiated between fake news and fake news-related terms such as rumors, but empirical comparative studies are limited leaving many questions unanswered, e.g., how similar (or specific) are writing style or propagation patterns of fake news compared to that of related concepts (e.g. disinformation and rumors)? Does having different characteristics lead to different detection strategies? Can we automatically distinguish these concepts from fake news? We have also provided two definitions for fake news, with the narrow definition being the most accurate; however, ground-truth datasets for fake news supporting the narrow definition are rarely seen. Systematically analyzing, identifying, and blocking fake news still has many unexplored arenas

2.2 EXISTING SYSTEM

The deception detection in online reviews & fake news has an important role in business, law enforcement, national security, political due to the potential impact fake reviews can have on consumer behavior and purchasing decisions. Researchers used deep learning with the large dataset to increase in learning and thus get the best results by using word embedding for extract features or cues that distinguish relations between words in syntactic and semantic. The implementation of RNN technique models (Vanilla, GRU) and LSTMs that have been proposed for the detection of online fake news after we prepare our LAIR dataset applying to prepare data to word embedding to get vectors of words then entering this vectors to deep learning technique, the results of experiments are close but GRU(Gated Recurrent Unit) is the best because it's solving the problems of Vanilla that popular of gradient vanishing problem and LSTMs (long short-term memories) which GRU is easy to modify and doesn't need memory units .

2.3 PROPOSED SYSTEM

With the increasing popularity of peer to peer network, more and more people consume news from peer to peer network instead of traditional news media. However, peer to peer network has also been used to spread fake news, which has strong negative impacts on individual users and broader society. Here we solve the problem of fake news by splitting the project in two phases: blockchain & NLP. In the blockchain phase, we enable any users to upload the news. In the NLP phase, we reviewed existing fake news detection approaches from a data mining perspective, including feature extraction and model construction

CHAPTER 3

Literature Survey

3.1“Learning to detect phishing emails”

It is possible to detect phishing emails with high accuracy by using a specialized filter, using features that are more directly applicable to phishing emails than those employed by general purpose spam filters. Although phishing is a subset of spam .it is characterized by certain unique properties that they have identified. One might be inclined to think that phishing emails should be harder to detect than general spam emails. After all, phishing emails are designed to sound like an email from a legitimate company, often a company with which the attacker hopes the user has a pre-existing relationship. Models based on “native” assumptions, such as certain words like “Viagra” being indicative of a class of un-desirable emails, no longer hold when the attackers are using the same words and the same overall “feel” to lure the user into a false sense of security. At the same time, phishing emails present unique opportunities for detection that are not present in general spam emails. Fette et al. used machine learning to classify an email as phishing or not by using features such as age of URL, number of dots in URL and HTML content of email while obtaining a high accuracy of 99.5%. [1]

3.2“Characterizing and identifying fake images on twitter during Hurricane Sandy”

Online social media has the capability of playing the role of, either a life saver or that of a daemon during the times of crisis. In this research work, they highlighted one of the malicious intended usage of Twitter during a real-world event. they analyzed the activity on the online social networking website Twitter, during Hurricane Sandy (2012) that spread of fake images. they identified 10,350 unique tweets containing fake images that were circulated on Twitter, during Hurricane Sandy. they performed a characterization analysis, to understand the temporal, social reputation and influence patterns of the spread of these fake images. they found that 86% tweets spreading the fake images were retweets, hence very few were original tweets by users. Also, their results showed that top 30 users (0.3% of the users) resulted in 90% of retweets of the fake image. Hence, they concluded that only a handful of users contributed to majority of the damage, via the retweeting activity on the Twitter.

They analyzed the role of Twitter social graph in propagating the fake images. they crawled the network links, that is, the follower relationships of the users and applied their algorithm to compute the overlap. They found only a 11% overlap between the retweet and follower graphs for the users who tweeted fake images of Sandy. This result highlights the fact that, at the time of crisis, users retweet information from other users irrespective of the fact whether they follow them or not. Next, they used classification models, to identify fake images from real images of Hurricane Sandy. Best results were obtained from Decision Tree classifier, they got 97% accuracy in predicting fake images from real. Tweet based features are very effective in distinguishing fake images tweets from real, while the performance of user-based features was very poor. Their research work provided insights into the behavioral pattern of the spread of fake image tweets. Also, their results provided a proof of concept that, automated techniques can be used in identifying real images from fake images posted on Twitter [2].

3.3 “Fake news detection on social media: A data mining perspective.”

Shu et al. said that malicious accounts can be easily and quickly created to boost the spread of fake news, such as social bots, cyborg users, or trolls. With the increasing popularity of social media, more and more people consume news from social media instead of traditional news media. However, social media has also been used to spread fake news, which has strong negative impacts on individual users and broader society. In article, they explored the fake news problem by reviewing existing literature in two phases: characterization and detection. In the characterization phase, they introduced the basic concepts and principles of fake news in both traditional media and social media. In the detection phase, they reviewed existing fake news detection approaches from a data mining perspective, including feature extraction and model construction. They also further discussed the datasets, evaluation metrics, and promising future directions in fake news detection research and expand the field to other applications. The existing algorithms for detection of fake news are either i). News Content Based or ii). Social Context Based. News content-based approaches focus on extracting various features in fake news content, including knowledge-based and style based. Social context-based approaches aim to utilize user social engagements as auxiliary information to help detect fake news.[3]

3.4“Automatic Detection of Fake News on Social Media Platforms.”

Janze et al. trained a variety of machine learning classifiers suitable for binary classification problem especially logistic regression, Support vector machines, Random forest and extreme gradient boosting. Because of the societal transformation induced by fake news and the difficulties people have when asked to identify them, their explorative study investigates how to fully automatically identify fake news using information immediately apparent on social media platforms. Specifically, building on the ELM and existing works in the realm of UGC and social psychology, they design an exploratory Automatic Detection of Fake News research model to study how cognitive, visual, affective and behavioral cues of a Facebook news posting as well as the associated comments allow for the prediction of fake news using machine learning classifiers .They evaluated the classification models (LOG, SVM, DTR, RFO and XGB) via different metrics which are based on a stratified 10-fold cross validation approach. They divided the data set of n=460 posts into 10 equally sized folds containing the same amount of fake and non-fake observations randomly selected from the total sample. Then, they took out onefold and trained the models with the nine remaining folds. They obtained an accuracy of nearly 80%[4]

3.5 “Combating fake news: An agenda for research and action”

Laser et al. identified the source of fake news and offered feedback to users that a particular news may be fake. They detected information being promoted by bots and ‘cyborg’ accounts Also, they found that older and more extreme individuals on the political spectrum appear to share fake news more than others The concrete steps for making the science of fake news more inclusive for researchers across the political spectrum, detailed strategies for making the truth “louder,” and introduced an interdisciplinary initiative for advancing the study of misinformation online. Finally, they recognized areas where additional research is needed to provide a better understanding of the fake news phenomenon and ways to mitigate it.[5]

CHAPTER 4

SYSTEM REQUIREMENT SPECIFICATION

4.1 FUNCTIONAL REQUIREMENTS

This section describes the functional requirements of the system for those requirements which are expressed in the natural language style

- 1.Installing the Framework
- 2.Create application which contains scheduler.
- 3.User will input the dataset to our system.
4. System automatically analysis dataset and schedule the process.
5. Apply efficient and optimal scheduling algorithm.

Application should provide efficient task scheduling process

4.2 NON-FUNCTIONAL REQUIREMENTS

These are requirements that are not functional in nature, that is, these are constraints within which the system must work.

1.The program must be self-contained so that it can easily be moved from one Computer to another. It is assumed that network connection will be available on the computer on which the program resides.

2. Capacity, scalability and availability.

The system shall achieve 100 per cent availability at all times.

The system shall be scalable to support additional clients and volunteers.

3. Maintainability.

The system should be optimized for supportability, or ease of maintenance as far as possible. This may be achieved through the use documentation of coding standards,

naming conventions, class libraries and abstraction.

4 Randomness, verifiability and load balancing.

The system should be optimized for supportability, or ease of maintenance as far as possible. This may be achieved through the use documentation of coding standards, naming conventions, class libraries and abstraction. It should have randomness to check the nodes and should be load balanced.

4.3 SOFTWARE REQUIREMENTS

- Operating system : Windows 8 & above , UNIX like OS
- Programming Language : Python/R

4.4 HARDWARE REQUIREMENTS

- System : Intel Core i5 & above
- Hard Disk : 512GB
- Monitor : 15''LED and above
- Input Device : Keyboard , Mouse
- Ram : 8GB

CHAPTER 5

PROJECT OBJECTIVES

- To Distinguish between fake news and correct news in the real time.
- To Eliminate the Fake News Which is being Spread.
- To Reduce the Misinformation Risks in the Society.
- To Determine the reliability of sources of news.

Chapter 6

DESIGN AND ANALYSIS

6.1 ARCHITECTURE DIAGRAM

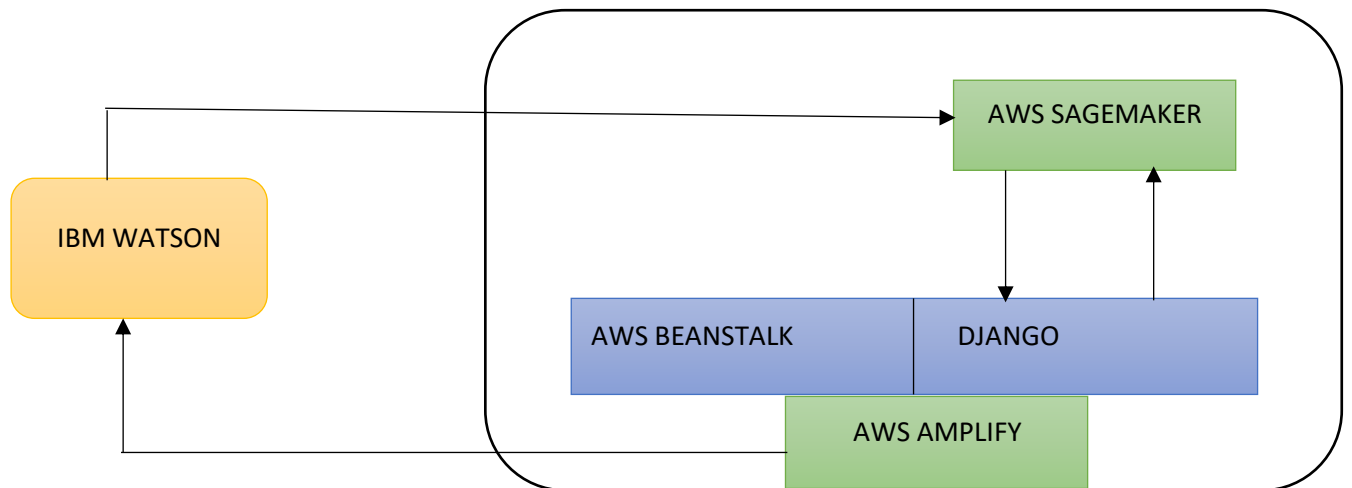


Fig 6.1 ARCHITECTURE DIAGRAM

The project works on the decentralised architecture and maintains the distributed ledger. The ibm watson is used in order to maintain the natural language processing module. Ibm watson provides the rest api to the application server that runs on the amazon servers. The entire project runs on django framework inturn on the aws benstalk. Since the clients can access the data from many devices in order to synchronize amplify is used.

The proposed project contains differnet modules that involves:

- Blockchain maintained by the aws
- Natural language processing module running on ibm watson
- Machine learning module on aws sagemaker
- Synchronization of the user data flowing in the system by aws amplify.

IBM WATSON- Watson is a question-answering computer system capable of answering questions posed in natural language and able to understand from the natural languages. The entire NLP process is on the IBM Watson.

AWS SAGEMAKER

Amazon Sage Maker is a cloud machine-learning platform. Sage Maker enables to operate at a number of levels of abstraction when training and deploying machine learning models. At its highest level of abstraction. The entire machine learning methodologies is performed by the sage maker.

AWS AMPLIFY

AWS Amplify makes it straightforward to make configure, and implement ascendable mobile and internet apps power-driven by AWS. Amplify seamlessly provisions and manages the mobile backend and provides an easy framework to simply integrate your backend along with their iOS, Android, Web, and React Native frontends. Amplify conjointly automates the application release method of your frontend and backend permitting you to deliver options quicker.

NLP

Natural Language Processing, or *NLP*, is the sub-field of AI that is focused on enabling computers to understand and process human languages. The learning procedures used during machine learning automatically focus on the most common cases, whereas when writing rules by hand it is often not at all obvious where the effort should be directed.

Django

Django is a high level python web framework that encourages rapid development and clean, pragmatic design. It takes care of much of the hassle of Web development, so you can focus on writing your app without needing to reinvent the wheel .

AWS MANAGED BLOCK CHAIN

Manage scalable blockchain networks using popular open source frameworks Hyperledger Fabric and Ethereum. Amazon Managed Blockchain eliminates the overhead required to create the network, and automatically scales to meet the demands of thousands of Amazon Managed Blockchain is a fully managed service that makes it easy to create and applications running millions of transactions. Once your network is up and running, Managed Blockchain makes it easy to manage and maintain your blockchain network. It manages your certificates, lets you easily invite new members to join the network, and tracks operational metrics such as usage of compute, memory, and storage resources.

6.2 FLOW CHART

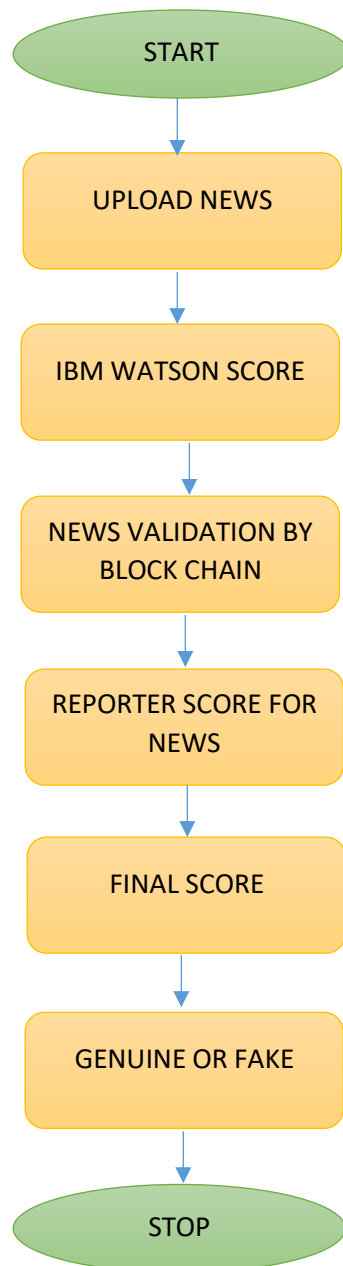


Fig 6.2 FLOW CHART

The above chart explains how the process of fake news detection takes place. Firstly the user uploads the news and then the process starts. IBM WATSON starts its work once the news is uploaded. Watson is a question-answering computer system capable of answering questions posed in natural language and able to understand from the natural languages. The entire NLP process is on the IBM Watson.

Watson has a small number of potential solutions, it is able to check against its database to ascertain whether the solution makes sense or not. Once Watson completes its job Block chain comes into picture.

Block chain manages the network and it is concerned about who enters and exits the network. It manages your certificates, lets you easily invite new members to join the network, and tracks operational metrics such as usage of compute, memory, and storage resources. It validates the news and decides whether it is acceptable or unacceptable.

Then the reporter has an opportunity to give a score to news which has been uploaded. The news initially gets the neutral score and then viewers who read the news may give score to it.

We get a final score of news that has been uploaded. Based on this we decide whether the news is genuine or fake which is one of the main aims of the process.

To resolve these challenges aforementioned, in this paper, we will introduce a new fake news detection framework, namely FAKE NEWS DETECTOR. In FAKE NEWS DETECTOR, the fake news detection problem is formulated as a credibility score inference problem, and FAKE NEWS DETECTOR aims at learning a prediction model to infer the credibility labels of news articles, creators and subjects simultaneously. FAKE NEWS DETECTOR deploys a new hybrid feature learning unit for learning the explicit and latent feature representations of news articles, creators and subjects respectively, and introduce a novel deep diffusive network model with the gated diffusive unit for the heterogeneous information fusion within the social networks. The attribution span is divided into two searchable sub-spans called the forward and trail attribution spans. The classifier tool was built to search inside the forward and trail attribution space and to classify the quote as either attributed or not. The resulting binary classification label is based on the presence of learned source and cue information inside the attribution spans. To identify a source, the custom classifier searched for named-entities or persons or organizations that could be attributed as having made a quote using named-entity recognition methods. Cue identification is based on learning associated cueing verbs or cue information contained inside the training set. Most informative cue words or phrases will be added to a living attribution “bag of words” model. Attribution feature extraction comes from applying algorithms to the forward and trailing attribution spans.

6.3 USE CASE DIAGRAM

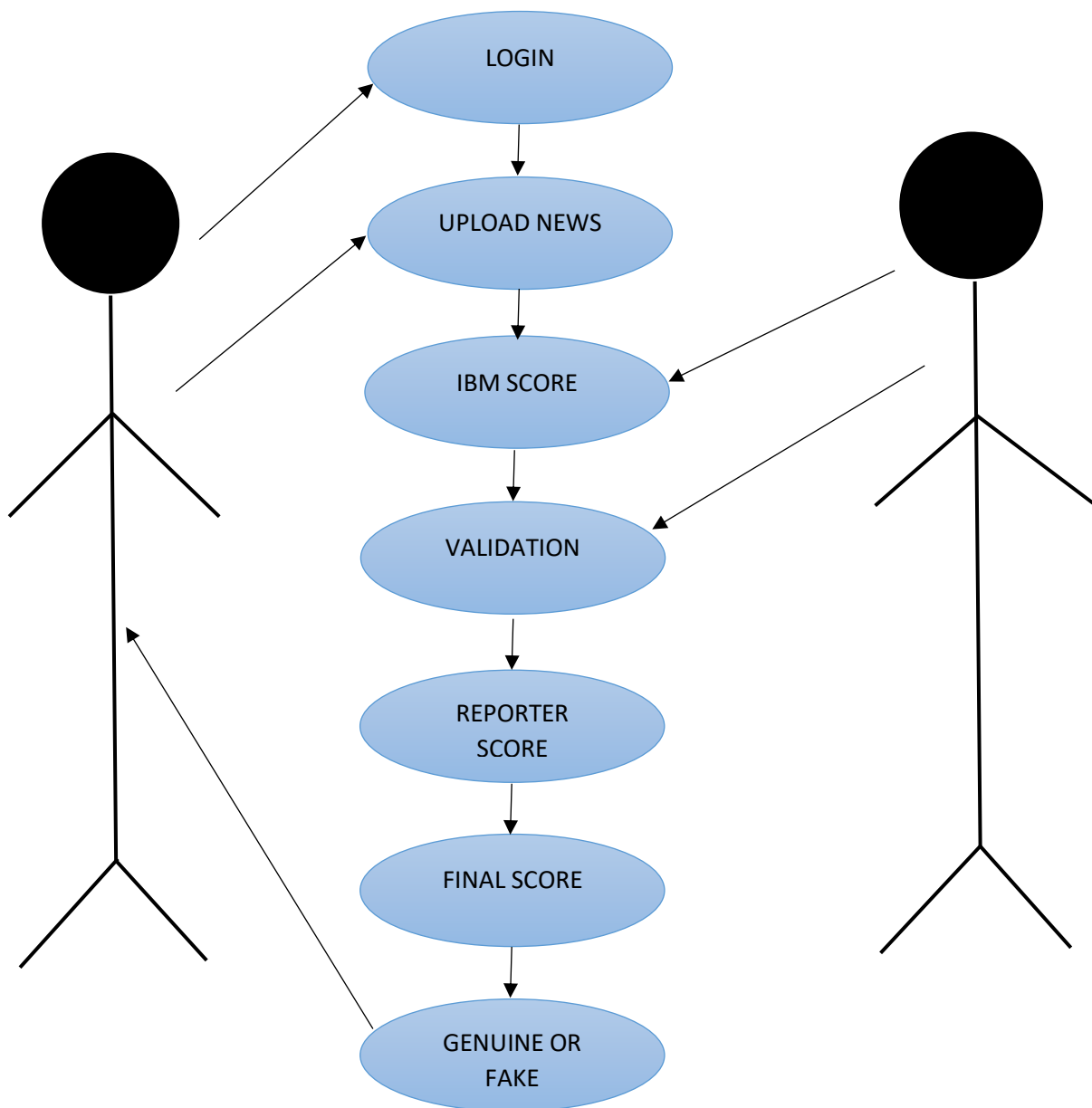


Fig 6.3 USE CASE DIAGRAM

The fake news detection tool uses the results of the outputs from the attribution classifications to assign a final label for the entire document. A simple scoring system, described in the next sub-section, was used to construct a final attribution score (called the attribution score or A-score) and assign a fake versus real classification label for every document containing quotes.

6.4 SEQUENCE DIAGRAM

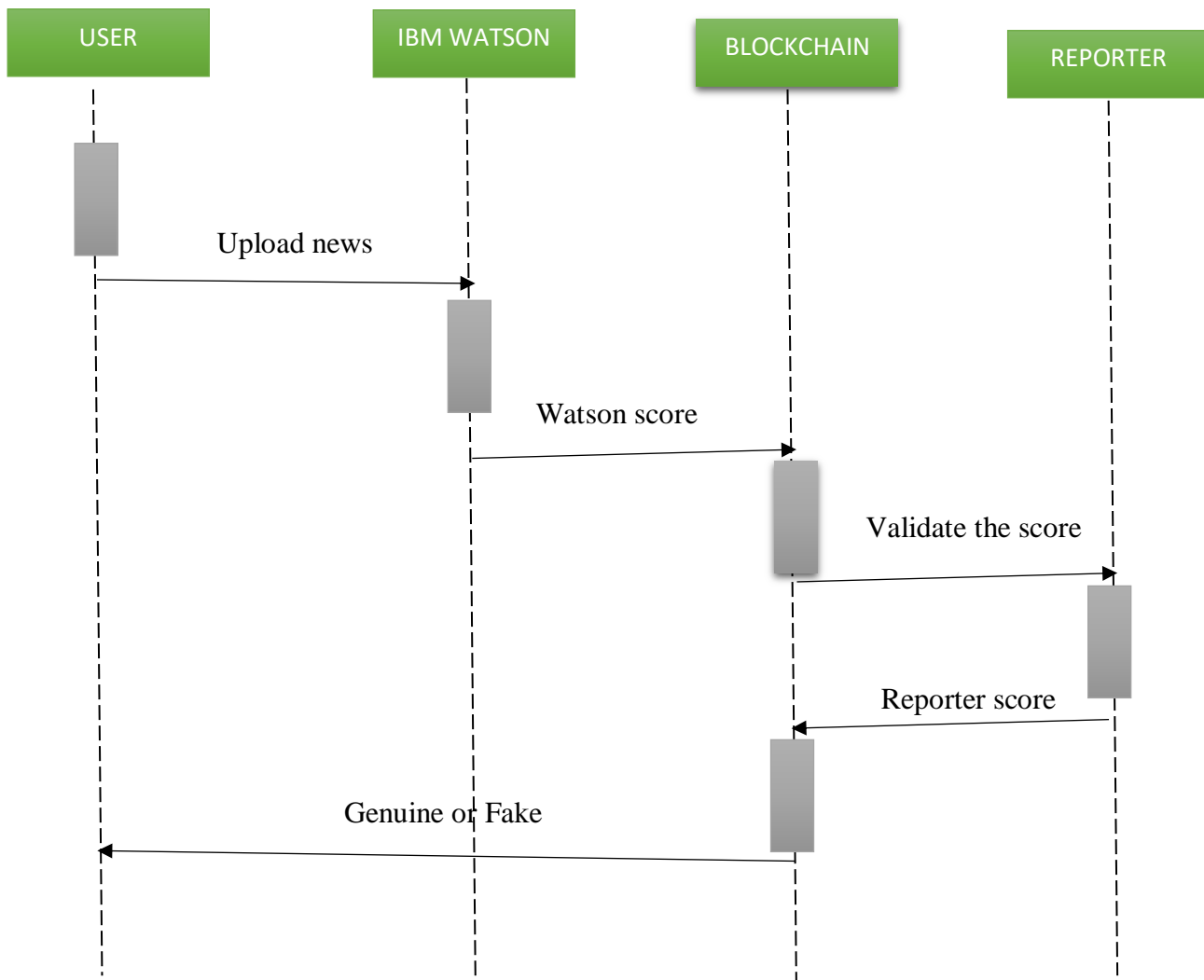


Fig 6.4 SEQUENCE DIAGRAM

The fake news detection algorithm is as follows. For each document in the document collection, the document's paragraphs are counted and tokenized. Each paragraph is also checked for quotes. If a paragraph has quotes, then these are processed using the custom attribution classifier (which uses the A-score algorithm). Positive attributions receive a +1 score and negative attribution classifications receive a -1 score. If the overall A-score (the sum of positives and negatives) is greater than or equal to 0, then the document is assigned a label of real. If the A-score is less than 0 then the document is assigned a label of fake. Note that the A-score threshold is, thus, a key area of potential configuration for this algorithm.

The A-score algorithm is used to label quotes as either real or fake based on the results of the machine learning classification

6.5 Class diagram

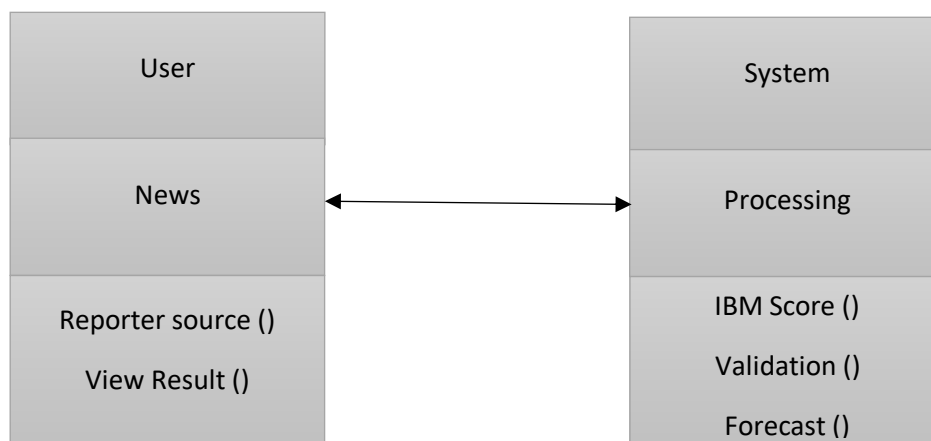


Fig 6.5 Class diagram

Pre-training data preparation focuses on removing extraneous common words to prevent these words from influencing association scores. Because common word data is not necessary for presentation to the classifier, no additional data preparation is done to the corpus for testing. As can be seen in the pseudo code and algorithm description, it was possible to tune the attribution span during testing, but it was decided to perform a simple run with the attribution space at $d=45$ for simplicity. Three types of experimental validations were conducted during experimentation. The accuracy of the quote attribution identifier, the custom quote attribution classifier, and finally the overall performance of the one feature fake news detection tool were tested. The quote attribution classifier, which is the core of the system, functioned well, but did not perform at an acceptable level. Several runs were needed to properly calibrate the classifier and account for linguistic processing issues such as quotes inside quotes and single quotes inside double quotes. The classifier also had issues handling multiple quotes within short proximity to each other inside a text. For example, if one quote came right behind another quote and the source and cue data were within the joint attribution space (in front of or behind a quote) for both quotes, the system encountered challenges processing both quotes. The final overall Classifier Accuracy is 0.69 and the overall Classifier Error is 0.31. Additional classifier performance metrics. While these numbers are subpar for attribution classification work, research is on going to improve the performance of

the classifier. Tuning the attribution distance and potentially developing a fake news attribution dictionary are methods being used to improve the classifier performance.

6.6 Data Flow diagram

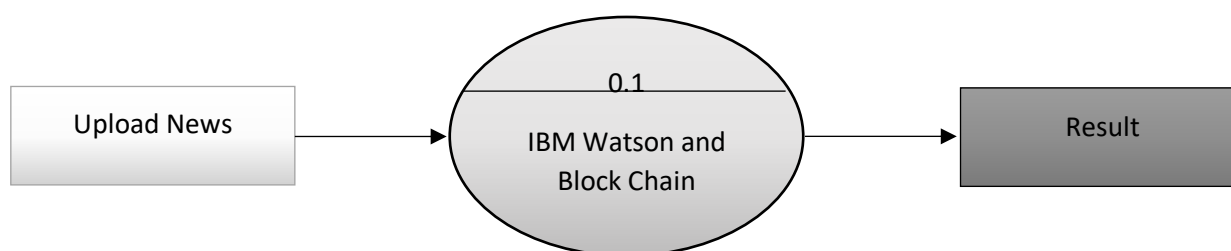


Fig 6.6.1 Data Flow diagram Level:0

Level 0 describes the overall process of this project. we are uploading the news as input and the system will determine the fake news

The true and false positive and negative rates are the number correctly (true) or incorrectly (false) identified over the total number identified with the relevant classification (positive or negative). For example, the true positive rate is the number of true positives divided by the number of true positives and false negatives. The precision is the number of correctly labelled items divided by the total number of elements belonging to the positive class. The total number of elements in the positive class includes both the true positives or correctly labelled items and the false positives or incorrectly labelled items. The score is a metric used in binary classification problems that measures the accuracy of a test. The score combines the precision and recall (or true positive rate) for a binary classification problem and is the harmonic mean of the precision and recall. The attribution-based fake news detection tool that uses the quote attribution classifier, performed suitably for a detection tool using only one feature extraction to classify a document; however, like the attribution classifier, it did not perform well enough for production use. After training and configuration, the tool correctly identified 69.4% of the fake and real news documents in the test set. Upon review, some of the missed labels were attributable to fake news documents with no quotations, fake news documents with attributed quotes of inaccurate statements, and fake news documents that quoted or cited other fake news documents. While, the overall performance results for this system are not as strong as

desired, the initial performance is generally encouraging, because fake news is designed to deceive human targets, so an initial classification tool with only one extraction feature seems to perform well, given the complexity of the topic and the aims of the project.

Level : 1

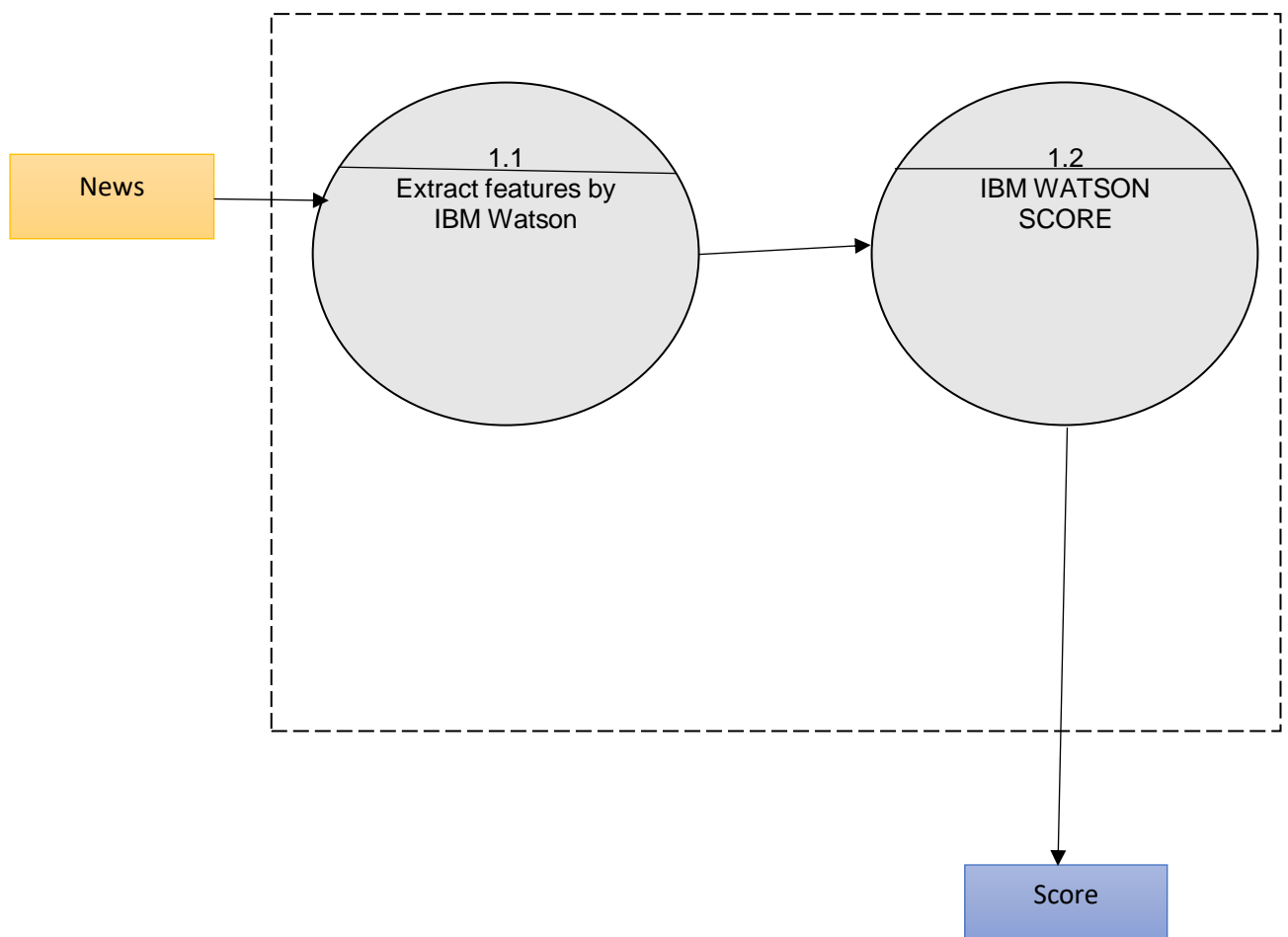


Fig 6.6.2 Data Flow diagram Level:1

Level 1 Describes the first stage process of this project. We are passing the news as an input and the IBM Watson will give the score.

The attribution classifier work and definitions originally put forth by multiple researchers were extended to build the attribution classifier and resultant one-feature fake news identification system defined attribution as a linguistic convention where a verb or attribution

cue links a source to a quoted piece of text called content. Specifically, an attribution for a quote has a source span, cue span, and content span. The trends and patterns that emerge from the analysis are grouped into codes and themes. The true and false positive and negative rates are the number correctly (true) or incorrectly (false) identified over the total number identified with the relevant classification (positive or negative). For example, the true positive rate is the number of true positives divided by the number of true positives and false negatives. The precision is the number of correctly labelled items divided by the total number of elements belonging to the positive class. Additional classifier performance metrics. While these numbers are subpar for attribution classification work, research is ongoing to improve the performance of the classifier. Tuning the attribution distance and potentially developing a fake news attribution dictionary are methods being used to improve the classifier performance.

After training and configuration, the tool correctly identified 69.4% of the fake and real news documents in the test set. Upon review, some of the missed labels were attributable to fake news documents with no quotations, fake news documents with attributed quotes of inaccurate statements, and fake news documents that quoted or cited other fake news documents. While, the overall performance results for this system are not as strong as desired, the total number of elements in the positive class includes both the true positives or correctly labelled items and the false positives or incorrectly labelled items. The score is a metric used in binary classification problems that measures the accuracy of a test. Over time, the codes and themes become categories and form the basis for a new theory. Then, prior fake news detection efforts are reviewed. Finally, fake news as a communication phenomenon (including attribution considerations) is discussed.

Level :2

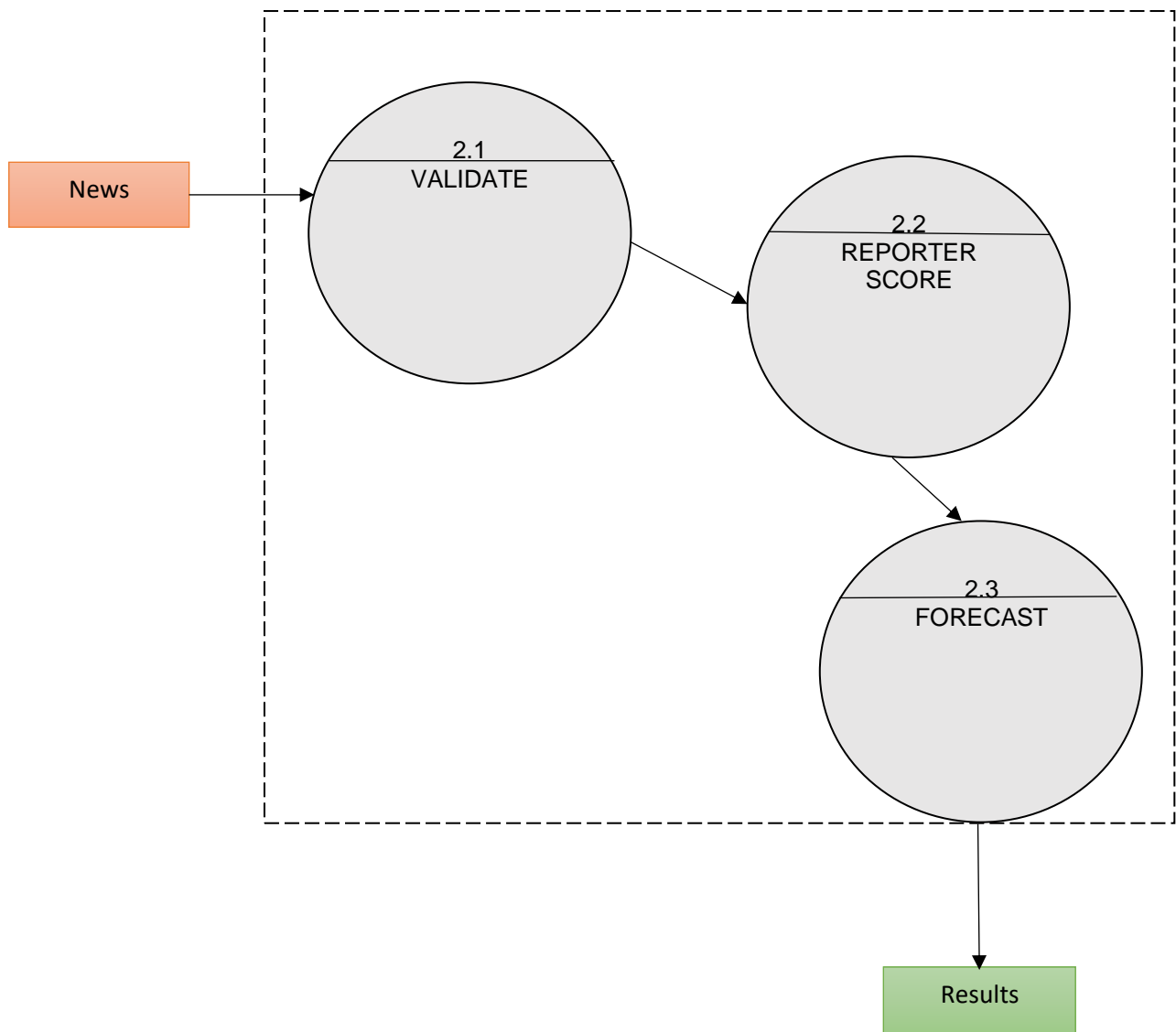


Fig 6.6.3 Data Flow diagram Level:2

Level 2 Describes the final stage process of this project. This will determine the fake news.

To produce tools that not only identify potential false content, but influence based content designed to compel a reader or target audience to make inaccurate or altered decisions. Tuning the attribution distance and potentially developing a fake news attribution dictionary are methods being used to improve the classifier performance. The total number of elements in the positive class includes both the true positives or correctly labelled items and the false positives or incorrectly labelled items.

CHAPTER 7

IMPLEMENTATION

In this section we implement the Fake News detection using the following three modules:

7.1 User Module

7.2 IBM Watson Module

7.3 Blockchain Module

7.1 User Module

The user module allows users to register, log in, and log out. Users benefit from being able to sign on because this associates content they create with their account and allows various permissions to be set for their roles.

The user module supports user roles, which can be set up with fine-grained permissions allowing each role to do only what the administrator permits. Each user is assigned one or more roles. By default there are three roles: anonymous (a user who has not logged in) and authenticated (a user who is registered), and administrator (a signed in user who will be assigned site administrator permissions).

Users can use their own name or handle and can fine tune some personal configuration settings through their individual my account page. Registered users need to authenticate by supplying their username and password, or alternately an login.

A visitor accessing your website is assigned a unique ID, the so-called session ID, which is stored in a cookie. For security's sake, the cookie does not contain personal information but acts as a key to retrieving the information stored on your server.

User Module Pseudo code:

Begin

Step 1: login

Step 2:if user wants to know whether news is real or fake then he has to register himself in blockchain

Step 3:gets approved by the block

Step 4: Upload the news

Step 5 :get the result

Stop.

7.2 IBM Watson Module

The relevance of natural language query results can be improved in IBM Watson™ Discovery for IBM Cloud Pak for Data with training.

Relevancy training is optional; if the results of your queries meet your needs, no further training is necessary. For information about use cases for relevancy training, see [Improve your natural language query results from Watson Discovery](#).

In order to train Watson, you'll need to:

- Identify natural language queries that are representative of the queries your users would request.
- Rate the results of each query as relevant OR not relevant.

Once Watson has enough training input, the information you have provided about which results are relevant or not relevant for each query will be used to learn about your collection. Watson does not memorize, it learns from the specific information about individual queries and applies the patterns it has detected to all new queries. It does this with machine learning Watson techniques that find signals in your content and questions. After training is applied, Discovery for Cloud Pak for Data then reorders the query results to display the most relevant results at the top. As you add more and more training data, Discovery for Cloud Pak for Data should become more accurate in the ordering of query results. Discovery for Cloud Pak for Data returns a score for natural language queries of trained collections. This score is not interchangeable with scores returned by untrained collections. The score can range from 1.0 to 100.0. The higher the number, the more relevant the result. The score can be found in the query results, under the `result_metadata` for each document. This number is calculated based on how relevant the result is estimated to be, compared to the trained model. Curations (beta) can be used to specify the exact document returned in response to a specific query.

Curations can guarantee that frequent or important questions always return the most valuable document. The score for a curated query will always be 100.000.

This beta feature is only available when using the API and can be used to specify up to 1,000 curations

Training consists of three parts:

- A natural language query
- The results of that query
- The rating you apply to each result

To train a collection Watson will learn which the best results for queries are after rated enough section will indicate your status by striking out the requirements as meet them:

- Add more queries
- Rate more results
- Add more variety to ratings

A minimum of 50 unique queries must be trained, though more may be required to meet the training threshold.

7.3 Blockchain Module

Blockchain a "phenomenal technology" that will change industries, but its adoption will be slow because it can't happen in a vacuum; it will require a shared ecosystem among enterprises; one company can't simply adopt it and expect returns on its investment without others climbing on board. Blockchain will more quickly take root in financial services for security and management of identities. The online credentials would be akin to identify information a person might have in his or her wallet: a driver's license, a bank debit card or a company ID. Instead of a physical card, however, the IDs in digital wallets would be encrypted on a blockchain ledger and link back to the institutions that created them, such as a bank, a government agency or even an employer. Through blockchain and a smart contract, the digital information could automatically verify information to a requestor. a smart contract can add functionality to a DLT as it is a computer program that is stored in the distributed

database Smart contracts allow for the addition of validations, constraints and business logic to transactions in a form of an agreement between parties. For example, a smart contract can be used to store relevant information such as publisher identity, status, reputation score, public key, timestamp, and then broadcast the content to the P2P network. Moreover, smart contracts can be used to register, update and revoke the identities of different organizations (e.g., publishers), as well as to determine their status and reputation score automatic management of non-tampered content and multi-node content verification can help to overcome the problem of verifying big data news streams. DLTs inherently guarantee data integrity once transactions are stored. This feature makes DLTs an essential infrastructure for notarization services. Nevertheless, a central problem show to ensure that data are not forged before they are inserted into a block. Service providers can take a fundamental role to provide a tamper-proof way to notarize content (e.g., by generating a digital signature) using a Public Key Infrastructure (PKI). When creating a P2P platform for tackling digital deception. These analysed some applications currently under development and proposed a number of additional mechanisms to control content.

7.4 Source Code:

```
(function() {  
    $('answers').hide();  
    $('loading').hide();  
    $('#form').submit(onFormSubmit);  
    $('dropdown-menu li > a').click(onExamplesClick);  
    $('classify-text').val("");  
  
    function onFormSubmit() {  
        var text = $('classify-text').val();  
        $('loading').show();  
        $('answers').hide();  
        $('classify-btn').prop('disabled', true);  
    }  
})
```

```
$.post("/classify", {text: text}, function(data) {
    renderAnswer(data)
}).fail(function(err) {
    renderAnswer(err);
});
return false;
}

function onExamplesClick() {
    var text = this.innerHTML;
    $('#classify-text').val(text);
    if (text && text.length > 1) {
        $('#form').submit();
    }
}

function renderAnswer(data) {
    if (!data.classes || !data.classes.length > 0) {
        $('#answer').html('Something went wrong :-(');
    } else {
        var top = data.classes[0]
        $('#answer').html(top.class_name.toUpperCase());
        $('#confidence').html('Confidence: '+Math.floor(top.confidence*100
).toFixed(0)+'%');
    }

    $('#classify-btn').prop('disabled', false);
}
```

```
    $('answers').show();  
    $('loading').hide();  
  }  
});  
  
/**  
 * Copyright 2017 IBM Corp. All Rights Reserved.  
 *  
 * Licensed under the Apache License, Version 2.0 (the 'License'); you may not  
 * use this file except in compliance with the License. You may obtain a copy of  
 * the License at  
 *  
 * http://www.apache.org/licenses/LICENSE-2.0  
 *  
 * Unless required by applicable law or agreed to in writing, software  
 * distributed under the License is distributed on an 'AS IS' BASIS, WITHOUT  
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the  
 * License for the specific language governing permissions and limitations under  
 * the License.  
 */  
  
'use strict';  
  
require('dotenv').config({  
  silent: true,
```

```
});

const express = require('express');
const router = express.Router();
const request = require('request');
const NLC = require('watson-developer-cloud/natural-language-classifier/v1');

var classifierId = process.env.CLASSIFIER_ID
var uname = process.env.NATURAL_LANGUAGE_CLASSIFIER_USERNAME
var pword = process.env.NATURAL_LANGUAGE_CLASSIFIER_PASSWORD

console.log("CLASSIFIER_ID:")
console.log(classifierId)

router.post('/', function(req, res, next) {
  classify(req, res);
});

function classify(req, res) {
  console.log("Classifying:")
  console.log(req.body);

  var nlc = new NLC({username: uname, password: pword});

  nlc.classify({
    text: req.body.text,
```

```
    classifier_id: classifierId
  }, function(err,response){
    if (err) {
      console.log(err)
    } else {
      res.json(response)
    }
  })
}
```

```
module.exports = router;
```

```
/**
```

```
* Copyright 2017 IBM Corp. All Rights Reserved.
```

```
*
```

```
* Licensed under the Apache License, Version 2.0 (the "License");
```

```
* you may not use this file except in compliance with the License.
```

```
* You may obtain a copy of the License at
```

```
*
```

```
* http://www.apache.org/licenses/LICENSE-2.0
```

```
*
```

```
* Unless required by applicable law or agreed to in writing, software
```

```
* distributed under the License is distributed on an "AS IS" BASIS,
```

```
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or  
implied.
```

* See the License for the specific language governing permissions and

* limitations under the License.

*/

```
/jslint node: true/
```

```
/*jslint es6 */
```

```
"use strict";
```

```
var express = require('express');
```

```
var path = require('path');
```

```
var favicon = require('serve-favicon');
```

```
var logger = require('morgan');
```

```
var cookieParser = require('cookie-parser');
```

```
var bodyParser = require('body-parser');
```

```
var classify = require('./routes/classify');
```

```
var app = express();
```

```
app.use(favicon(__dirname + '/public/favicon.ico'));
```

```
app.use(logger('dev'));
```

```
app.use(bodyParser.json());
```

```
app.use(bodyParser.urlencoded({ extended: false }));
```

```
app.use(cookieParser());
```

```
app.use(express.static(path.join(__dirname, 'public')));
```

```
app.use('/classify', classify);

// catch 404 and forward to error handler
app.use(function(req, res, next) {
  var err = new Error('Not Found');
  err.status = 404;
  next(err);
});

// development error handler
// will print stacktrace
if (app.get('env') === 'development') {
  app.use(function(err, req, res, next) {
    res.status(err.status || 500);
    res.json({
      message: err.message,
      error: err
    });
  });
}

// production error handler
// no stacktraces leaked to user
app.use(function(err, req, res, next) {
  res.status(err.status || 500);
  res.json({
```

```
    message: err.message,
    error: {}
  });
});

const port = process.env.PORT || process.env.VCAP_APP_PORT || 3000;

app.listen(port, function () {
  console.log("Server running on port: %d", port);
});

//module.exports = app;

{
  "name": "Fake_news_detection",
  "description": "A server using Watson Natural Language Classifier to classify News as either real or fake ",
  "version": "0.0.1",
  "main": "app.js",
  "scripts": {
    "start": "node app.js"
  },
  "dependencies": {
    "body-parser": "^1.18.3",
    "cookie-parser": "~1.3.4",
    "debug": "^3.1.0",
    "dotenv": "^2.0.0",
```

```
"express": "^4.16.3",
"morgan": "^1.9.1",
"request": "^2.61.0",
"serve-favicon": "^2.5.0",
"vcap_services": "^0.2.0",
"watson-developer-cloud": "^3.5.1"
}
}
{
  "name": "FAke_news_detection",
  "version": "0.0.1",
  "lockfileVersion": 1,
  "requires": true,
  "dependencies": {
    "@types/caseless": {
      "version": "0.12.1",
      "resolved": "https://registry.npmjs.org/@types/caseless/-/caseless-0.12.1.tgz",
      "integrity": "sha512-FhlMa34NHp9K5MY1Uz8yb+ZvuX0pnav3jScRSNAb75KHGB8d3rEU6hqMs3Z2vjuytcMfRg6c5CHMc3wtYyD2/A=="
    },
    "@types/csv-stringify": {
      "version": "1.4.2",
      "resolved": "https://registry.npmjs.org/@types/csv-stringify/-/csv-stringify-1.4.2.tgz",
      "integrity": "sha512-OxBpngVW09fD5S90Fi2ihWJa9gWk/cYSeZ05T5PiElOJw4OjKq9bXyVEkEPpHS/1Vr/gKzAgv7okuvHLpb5CSA=="
    },
    "requires": {

```

```

    "@types/node": "*"
  }
},
"@types/extend": {
  "version": "3.0.0",
  "resolved": "https://registry.npmjs.org/@types/extend/-/extend-3.0.0.tgz",
  "integrity": "sha512-
Eo8NQCbglMPQarIFAE3vpyCvFda4dg1Ob5ZJb6BJI9x4NAZVWowyMNB8GJJDgDI4lr2
oqiQvXIPB0Fn1NoXnQ=="
},
"@types/file-type": {
  "version": "5.2.1",
  "resolved": "https://registry.npmjs.org/@types/file-type/-/file-type-5.2.1.tgz",
  "integrity": "sha512-
Im0cJaIPJbbpuW91OrjXnqWPZCJK/tcFy2cFX+1qjG1gubgVZPPO9OVsTVAjotN4I1E6FA
V0eIqt+rR8Y1c3iA=="
  "requires": {
    "@types/node": "*"
  }
},
"@types/form-data": {
  "version": "2.2.1",
  "resolved": "https://registry.npmjs.org/@types/form-data/-/form-data-2.2.1.tgz",
  "integrity": "sha512-
JAMFhOaHIciYVh8fb5/83nmuO/AHwmtO+Hq7a9y8FzLDcC1KCU344XDOMEmahnrTFI
Hjgh4L0WJFczNIX2GxnQ=="
  "requires": {
    "@types/node": "*"
  }
}

```

```
},
"@types/is-stream": {
  "version": "1.1.0",
  "resolved": "https://registry.npmjs.org/@types/is-stream/-/is-stream-1.1.0.tgz",
  "integrity": "sha512-jkZatu4QVbR60mpIzjINmtS1ZF4a/FqdTUTBeQDVOQ2PYyidtwFKr0B5G6ERukKwliq+7mIXvxypwzG5EgRYg==",
  "requires": {
    "@types/node": "*"
  }
},
"@types/node": {
  "version": "10.3.6",
  "resolved": "https://registry.npmjs.org/@types/node/-/node-10.3.6.tgz",
  "integrity": "sha512-h7VDRFL8IhdPw1JjiNVvhr+WynfKW09q2BOfIIOA0yeuXNeXBP1bIRuBrysSryH4keaJ5bYUNp63aIyQL9YpDQ==",
  "requires": {
    "@types/request": {
      "version": "2.47.1",
      "resolved": "https://registry.npmjs.org/@types/request/-/request-2.47.1.tgz",
      "integrity": "sha512-TV3XLvDjQbleVxJ1Z3oCTDk/KuYwwcNKVwz2YaT0F5u86Prgc4syDAp6P96rkTQQ4bIdh+VswQIC9zS6NjY7/g==",
      "requires": {
        "@types/caseless": "*",
        "@types/form-data": "*",
        "@types/node": "*",
        "@types/tough-cookie": "*"
      }
    }
  }
}
```

```
}  
},  
"@types/tough-cookie": {  
  "version": "2.3.3",  
  "resolved": "https://registry.npmjs.org/@types/tough-cookie/-/tough-cookie-2.3.3.tgz",  
  "integrity": "sha512-MDQLxNFRLasqS4UlkWMSACMKeSm1x4Q3TxzUC7KQUsh6RK1ZrQ0VEyE3yzXcBu+K8ejVj4wuX32eUG02yNp+YQ=="  
},  
"accepts": {  
  "version": "1.3.5",  
  "resolved": "https://registry.npmjs.org/accepts/-/accepts-1.3.5.tgz",  
  "integrity": "sha1-63d99gEXI6OxTopywIBcjoZ0a9I=",  
  "requires": {  
    "mime-types": "~2.1.18",  
    "negotiator": "0.6.1"  
  }  
},  
"ajv": {  
  "version": "5.5.2",  
  "resolved": "https://registry.npmjs.org/ajv/-/ajv-5.5.2.tgz",  
  "integrity": "sha1-c7Xuyj+rZT49P5Qis0GtQiBdyWU=",  
  "requires": {  
    "co": "^4.6.0",  
    "fast-deep-equal": "^1.0.0",  
    "fast-json-stable-stringify": "^2.0.0",  
    "json-schema-traverse": "^0.3.0"
```

```
}  
},  
"array-flatten": {  
  "version": "1.1.1",  
  "resolved": "https://registry.npmjs.org/array-flatten/-/array-flatten-1.1.1.tgz",  
  "integrity": "sha1-m19pkFGx5wczKPKgCJaLZOopVdI=",  
},  
"asn1": {  
  "version": "0.2.3",  
  "resolved": "https://registry.npmjs.org/asn1/-/asn1-0.2.3.tgz",  
  "integrity": "sha1-2sh4dxPJlmhJ/IGAd36+nB3fO4Y=",  
},  
"assert-plus": {  
  "version": "1.0.0",  
  "resolved": "https://registry.npmjs.org/assert-plus/-/assert-plus-1.0.0.tgz",  
  "integrity": "sha1-8S4PPF13sLHN2RRpQuTpbB5N1SU=",  
},  
"async": {  
  "version": "2.6.1",  
  "resolved": "https://registry.npmjs.org/async/-/async-2.6.1.tgz",  
  "integrity": "sha512-fNEiL2+AZt6AlAw/29Cr0UDe4sRAHCpEHh54WMz+Bb7QfNcFw4h3loofyJpLeQs4Yx7y  
uqu/2dLgM5hKOs6HIQ==",  
  "requires": {  
    "lodash": "^4.17.10"  
  }  
},
```

```
"async-kit": {
  "version": "0.4.0",
  "resolved": "https://registry.npmjs.org/async-kit/-/async-kit-0.4.0.tgz",
  "integrity": "sha1-x57Zf380y48robyXkLzDZkdLS3k="
},
"aws-sign2": {
  "version": "0.7.0",
  "resolved": "https://registry.npmjs.org/aws-sign2/-/aws-sign2-0.7.0.tgz",
  "integrity": "sha1-tG6JCTSpWR8tL2+G1+ap8bP+dqg="
},
"aws4": {
  "version": "1.7.0",
  "resolved": "https://registry.npmjs.org/aws4/-/aws4-1.7.0.tgz",
  "integrity": "sha512-32NDda82rhwD9/JBCCkB+MRYDp0oSvlo2IL6rQWA10PQi7tDUM3eqMSltXmY+Oyl/7N3P3qNtAlv7X0d9bI28w=="
},
"basic-auth": {
  "version": "2.0.1",
  "resolved": "https://registry.npmjs.org/basic-auth/-/basic-auth-2.0.1.tgz",
  "integrity": "sha512-NF+epuEdnUYVIGuhaxbbq+dvJttwLnGY+YixIXIME5KpQ5W3CnXA5cVTneY3SPbPDRkcjMbifrwfFYcClgOZeg==",
  "requires": {
    "safe-buffer": "5.1.2"
  },
  "dependencies": {
    "safe-buffer": {

```

```
"cookie-signature": "1.0.6"
}
},
"cookie-signature": {
  "version": "1.0.6",
  "resolved": "https://registry.npmjs.org/cookie-signature/-/cookie-signature-1.0.6.tgz",
  "integrity": "sha1-4wOogrNCzD7oylE6eZmXNNqzriw="
},
"core-util-is": {
  "version": "1.0.2",
  "resolved": "https://registry.npmjs.org/core-util-is/-/core-util-is-1.0.2.tgz",
  "integrity": "sha1-tf1UIgqivFq1eQtxQMlAdUUDwac="
},
"cryptiles": {
  "version": "3.1.2",
  "resolved": "https://registry.npmjs.org/cryptiles/-/cryptiles-3.1.2.tgz",
  "integrity": "sha1-qJ+7Ig9c4l7FboxKqKT9e1sNKf4=",
  "requires": {
    "boom": "5.x.x"
  },
  "dependencies": {
    "boom": {
      "version": "5.2.0",
      "resolved": "https://registry.npmjs.org/boom/-/boom-5.2.0.tgz",
      "integrity": "sha512-4tJ9P6n/5Kd8NiKLqV/5WwF0I5W1MnD5nz1tW1J1GZuGR3IuNq6Z1FbC03HozxNF0U038kF311Wz3G72A=="
    }
  }
}
```

```
"requires": {
  "hoek": "4.x.x"
}
},
"csv-stringify": {
  "version": "1.0.4",
  "resolved": "https://registry.npmjs.org/csv-stringify/-/csv-stringify-1.0.4.tgz",
  "integrity": "sha1-vBi6ua1M7zGV/SV5gLWLR5xC0+U=",
  "requires": {
    "lodash.get": "^4.0.0"
  }
},
"dashdash": {
  "version": "1.14.1",
  "resolved": "https://registry.npmjs.org/dashdash/-/dashdash-1.14.1.tgz",
  "integrity": "sha1-hTz6D3y+L+1d4gMmuN1YEDX24vA=",
  "requires": {
    "assert-plus": "^1.0.0"
  }
},
"debug": {
  "version": "3.1.0",
  "resolved": "https://registry.npmjs.org/debug/-/debug-3.1.0.tgz",
```

```
"integrity": "sha512-
OX8XqP7/1a9cqkxYw2yXss15f26NKWBpDXQd0/uK/KPqdQhxbPa994hznzcE2VqQpDslf
55723cKPUOGSmMY3g==",
```

```
"requires": {
```

```
  "ms": "2.0.0"
```

```
}
```

```
},
```

```
"delayed-stream": {
```

```
  "version": "1.0.0",
```

```
  "resolved": "https://registry.npmjs.org/delayed-stream/-/delayed-stream-1.0.0.tgz",
```

```
  "integrity": "sha1-3zrhmayt+31ECqrgsp4icrJOxhk="
```

```
},
```

```
"depd": {
```

```
  "version": "1.1.2",
```

```
  "resolved": "https://registry.npmjs.org/depd/-/depd-1.1.2.tgz",
```

```
  "integrity": "sha1-m81S4UwJd2PnSbJ0xDRu0uVgtak="
```

```
},
```

```
"destroy": {
```

```
  "version": "1.0.4",
```

```
  "resolved": "https://registry.npmjs.org/destroy/-/destroy-1.0.4.tgz",
```

```
  "integrity": "sha1-l4hXRCxEJ5CBmE+N5RiBYJqvYA="
```

```
},
```

```
"dotenv": {
```

```
  "version": "2.0.0",
```

```
  "resolved": "https://registry.npmjs.org/dotenv/-/dotenv-2.0.0.tgz",
```

```
  "integrity": "sha1-vXWcNXqqcDZeAclrewvsCKbg2Uk="
```

```
},
```

```
"ecc-jsbn": {
  "version": "0.1.1",
  "resolved": "https://registry.npmjs.org/ecc-jsbn/-/ecc-jsbn-0.1.1.tgz",
  "integrity": "sha1-D8c6ntXw1Tw4GTOYUj735UN3dQU=",
  "optional": true,
  "requires": {
    "jsbn": "~0.1.0"
  }
},
"ee-first": {
  "version": "1.1.1",
  "resolved": "https://registry.npmjs.org/ee-first/-/ee-first-1.1.1.tgz",
  "integrity": "sha1-WQxhFWsK4vTwJVcyoViyZrxWsh0="
},
"encodeurl": {
  "version": "1.0.2",
  "resolved": "https://registry.npmjs.org/encodeurl/-/encodeurl-1.0.2.tgz",
  "integrity": "sha1-rT/0yG7C0CkyL1oCw6mmBs1bP1k="
},
"escape-html": {
  "version": "1.0.3",
  "resolved": "https://registry.npmjs.org/escape-html/-/escape-html-1.0.3.tgz",
  "integrity": "sha1-Aljq5NPQwJdN4cFpGI7wBR0dGYg="
},
"etag": {
  "version": "1.8.1",
```

```
"resolved": "https://registry.npmjs.org/etag/-/etag-1.8.1.tgz",
"integrity": "sha1-Qa4u62XvpiJorr/qg6x9eSmbCIc="
},
"express": {
  "version": "4.16.3",
  "resolved": "https://registry.npmjs.org/express/-/express-4.16.3.tgz",
  "integrity": "sha1-avilAjUNsyRuzEvs9rWjTSL37VM=",
  "requires": {
    "accepts": "~1.3.5",
    "array-flatten": "1.1.1",
    "body-parser": "1.18.2",
    "content-disposition": "0.5.2",
    "content-type": "~1.0.4",
    "cookie": "0.3.1",
    "cookie-signature": "1.0.6",
    "debug": "2.6.9",
    "depd": "~1.1.2",
    "encodeurl": "~1.0.2",
    "escape-html": "~1.0.3",
    "etag": "~1.8.1",
    "finalhandler": "1.1.1",
    "fresh": "0.5.2",
    "merge-descriptors": "1.0.1",
    "methods": "~1.1.2",
    "on-finished": "~2.3.0",
    "parseurl": "~1.3.2",
```

```
"path-to-regexp": "0.1.7",
"proxy-addr": "~2.0.3",
"qs": "6.5.1",
"range-parser": "~1.2.0",
"safe-buffer": "5.1.1",
"send": "0.16.2",
"serve-static": "1.13.2",
"setprototypeof": "1.1.0",
"statuses": "~1.4.0",
"type-is": "~1.6.16",
"utils-merge": "1.0.1",
"vary": "~1.1.2"
},
"dependencies": {
  "body-parser": {
    "version": "1.18.2",
    "resolved": "https://registry.npmjs.org/body-parser/-/body-parser-1.18.2.tgz",
    "integrity": "sha1-h2eKGdhLR9hZuDGZvVm84iKxBFQ=",
    "requires": {
      "bytes": "3.0.0",
      "content-type": "~1.0.4",
      "debug": "2.6.9",
      "depd": "~1.1.1",
      "http-errors": "~1.6.2",
      "iconv-lite": "0.4.19",
      "on-finished": "~2.3.0",
```

```
"qs": "6.5.1",
"raw-body": "2.3.2",
"type-is": "~1.6.15"
}
},
"cookie": {
  "version": "0.3.1",
  "resolved": "https://registry.npmjs.org/cookie/-/cookie-0.3.1.tgz",
  "integrity": "sha1-5+Ch+e9DtMi6klxcWpboBtFoc7s="
},
"debug": {
  "version": "2.6.9",
  "resolved": "https://registry.npmjs.org/debug/-/debug-2.6.9.tgz",
  "integrity": "sha512-bC7EldJaJnPbAP+1EotYvqZsb3ecl5wi6Bfi6BJTUcNowp6cvspg0jXznRTKDJm/E7AdgFBVeAPVMNcKGsHMA==",
  "requires": {
    "ms": "2.0.0"
  }
},
"iconv-lite": {
  "version": "0.4.19",
  "resolved": "https://registry.npmjs.org/iconv-lite/-/iconv-lite-0.4.19.tgz",
  "integrity": "sha512-oTZqweIP51xaGPI4uPa56/Pri/480R+mo7SeU+YETByQNhDG55ycFyNLIgta9vXhILrxXDmF7ZGhqZlCuN0gJQ=="
},
"qs": {
```

```
"version": "6.5.1",
"resolved": "https://registry.npmjs.org/qs/-/qs-6.5.1.tgz",
"integrity": "sha512-eRzhrN1WSINYCDCbrz796z37LOe3m5tmW7RQf6oBntukAG1nmovJvhnwHHRMAfeoItc
1m2Hk02WER2aQ/iqs+A=="
},
"raw-body": {
"version": "2.3.2",
"resolved": "https://registry.npmjs.org/raw-body/-/raw-body-2.3.2.tgz",
"integrity": "sha1-vNYMd9Prk83gBQKVw/N5OJvIj4k=",
"requires": {
"bytes": "3.0.0",
"http-errors": "1.6.2",
"iconv-lite": "0.4.19",
"unpipe": "1.0.0"
},
"dependencies": {
"depd": {
"version": "1.1.1",
"resolved": "https://registry.npmjs.org/depd/-/depd-1.1.1.tgz",
"integrity": "sha1-V4O04cRZ8G+lyif5kfPQbnoxA1k="
},
"http-errors": {
"version": "1.6.2",
"resolved": "https://registry.npmjs.org/http-errors/-/http-errors-1.6.2.tgz",
"integrity": "sha1-CgAsyFcHGSsp+eUbO7cERVfYOxzY=",
"requires": {
```

```

    "depd": "1.1.1",
    "inherits": "2.0.3",
    "setprototypeof": "1.0.3",
    "statuses": ">= 1.3.1 < 2"
  }
},
"setprototypeof": {
  "version": "1.0.3",
  "resolved": "https://registry.npmjs.org/setprototypeof/-/setprototypeof-1.0.3.tgz",
  "integrity": "sha1-ZIZ+NwQ+608E2RvWWMDL77VbjgQ="
}
}
},
"statuses": {
  "version": "1.4.0",
  "resolved": "https://registry.npmjs.org/statuses/-/statuses-1.4.0.tgz",
  "integrity": "sha512-zhSctt8v2NDRRIPQpCNTw/heZLtfUDqxBM1udqikb/Hbk52LK4nQSwr10u77iopCW5LsyHpuXS0GnEc48mLeew=="
}
}
},
"extend": {
  "version": "3.0.2",
  "resolved": "https://registry.npmjs.org/extend/-/extend-3.0.2.tgz",
  "integrity": "sha512-1220k0hhz6ccBAPY010Wp2e6o3V89jS4D94+kIdiRPVr9vjS6D869311DQ3gKxhXjY74L3Lw6q3Y2+0oUw=="
}
}
}

```

```
  },
  "extsprintf": {
    "version": "1.3.0",
    "resolved": "https://registry.npmjs.org/extsprintf/-/extsprintf-1.3.0.tgz",
    "integrity": "sha1-lpGEQOMEGnpBT4xS48V06zw+HgU="
  },
  "fast-deep-equal": {
    "version": "1.1.0",
    "resolved": "https://registry.npmjs.org/fast-deep-equal/-/fast-deep-equal-1.1.0.tgz",
    "integrity": "sha1-wFNHeBfla1HaqFPIHgWbcz0CNhQ="
  },
  "fast-json-stable-stringify": {
    "version": "2.0.0",
    "resolved": "https://registry.npmjs.org/fast-json-stable-stringify/-/fast-json-stable-stringify-2.0.0.tgz",
    "integrity": "sha1-1RQsDK7msRifh9OnYREGT4bIu/I="
  },
  "file-type": {
    "version": "7.7.1",
    "resolved": "https://registry.npmjs.org/file-type/-/file-type-7.7.1.tgz",
    "integrity": "sha512-bTrKkzzZI6wH+NXhyD3SOXtb2zXTw2SbwI2RxUIRcXVsnN7jNL5hJzVQLYv7FOQhxFkK4XWdAflEaWFpaLLWpQ=="
  },
  "finalhandler": {
    "version": "1.1.1",
    "resolved": "https://registry.npmjs.org/finalhandler/-/finalhandler-1.1.1.tgz",
```

```
"integrity": "sha512-
Y1GUDo39ez4aHAw7MysnUD5JzYX+WaIj8I57kO3aEPT1fFRL4sr7mjei97FgnwhAyyzR
YmQZaTHb2+9uZ1dPtg==",

"requires": {

  "debug": "2.6.9",

  "encodeurl": "~1.0.2",

  "escape-html": "~1.0.3",

  "on-finished": "~2.3.0",

  "parseurl": "~1.3.2",

  "statuses": "~1.4.0",

  "unpipe": "~1.0.0"

},

"dependencies": {

  "debug": {

    "version": "2.6.9",

    "resolved": "https://registry.npmjs.org/debug/-/debug-2.6.9.tgz",

    "integrity": "sha512-
bc7EldJrJaJnPbAP+1EotYvqZsb3ecl5wi6Bfi6BJTUcNowp6cvspg0jXznRTKDjm/E7AdgFB
VeAPVMNcKGsHMA==",

    "requires": {

      "ms": "2.0.0"

    }

  },

  "statuses": {

    "version": "1.4.0",

    "resolved": "https://registry.npmjs.org/statuses/-/statuses-1.4.0.tgz",

    "integrity": "sha512-
zhSctt8v2NDRRIPQpCNTw/heZLtfUDqxBM1udqikb/Hbk52LK4nQSwr10u77iopCW5LsyH
puXS0GnEc48mLeew=="

  }

}
```

```
    }  
  }  
},  
"forever-agent": {  
  "version": "0.6.1",  
  "resolved": "https://registry.npmjs.org/forever-agent/-/forever-agent-0.6.1.tgz",  
  "integrity": "sha1-+8cfDEGt6zf5bFd60e1C2P2sypE=",  
},  
"form-data": {  
  "version": "2.3.2",  
  "resolved": "https://registry.npmjs.org/form-data/-/form-data-2.3.2.tgz",  
  "integrity": "sha1-SXBJi+YEgwAXU9cI67NIda0kJk=",  
  "requires": {  
    "asynckit": "^0.4.0",  
    "combined-stream": "1.0.6",  
    "mime-types": "^2.1.12"  
  }  
},  
"forwarded": {  
  "version": "0.1.2",  
  "resolved": "https://registry.npmjs.org/forwarded/-/forwarded-0.1.2.tgz",  
  "integrity": "sha1-mMI9qx F1ZXuMBXPozszZGw/xjIQ=",  
},  
"fresh": {  
  "version": "0.5.2",  
  "resolved": "https://registry.npmjs.org/fresh/-/fresh-0.5.2.tgz",
```

```
"integrity": "sha1-PYyt2Q2XZWn6g1qx+OSyOhBWBac="
},
"getpass": {
  "version": "0.1.7",
  "resolved": "https://registry.npmjs.org/getpass/-/getpass-0.1.7.tgz",
  "integrity": "sha1-Xv+OPmhNVprkyysSgmBOi6YhSfo=",
  "requires": {
    "assert-plus": "^1.0.0"
  }
},
"har-schema": {
  "version": "2.0.0",
  "resolved": "https://registry.npmjs.org/har-schema/-/har-schema-2.0.0.tgz",
  "integrity": "sha1-qUwiJOvKwEeCoNkDVSHyRzW37JI="
},
"har-validator": {
  "version": "5.0.3",
  "resolved": "https://registry.npmjs.org/har-validator/-/har-validator-5.0.3.tgz",
  "integrity": "sha1-ukAsJmGU8VIW7xXg/PJcMT9qff0=",
  "requires": {
    "ajv": "^5.1.0",
    "har-schema": "^2.0.0"
  }
},
"hawk": {
  "version": "6.0.2",
```

```
"resolved": "https://registry.npmjs.org/hawk/-/hawk-6.0.2.tgz",
  "integrity": "sha512-miowhl2+U7Qle4vdLqDdPt9m09K6yZhkLDTWGoUiUzrQCn+mHHSmfJgAyGaLRZbPmTqfFFjRV1QWCW0VWUJBbQ==",
  "requires": {
    "boom": "4.x.x",
    "cryptiles": "3.x.x",
    "hoek": "4.x.x",
    "sntp": "2.x.x"
  },
  "hoek": {
    "version": "4.2.1",
    "resolved": "https://registry.npmjs.org/hoek/-/hoek-4.2.1.tgz",
    "integrity": "sha512-QLg82fGkfnJ/4iy1xZ81/9SIJiq1NGFUMGs6ParyjBZr6jW2Ufj/snDqTHixNIHdPNwN2RLVD0Pi3igeK9+JfA=="
  },
  "http-errors": {
    "version": "1.6.3",
    "resolved": "https://registry.npmjs.org/http-errors/-/http-errors-1.6.3.tgz",
    "integrity": "sha1-i1VoC7S+KDoLW/TqLjhYC+HZMg0=",
    "requires": {
      "depd": "~1.1.2",
      "inherits": "2.0.3",
      "setprototypeof": "1.1.0",
      "statuses": ">= 1.4.0 < 2"
    }
  }
}
```

```
},
"http-signature": {
  "version": "1.2.0",
  "resolved": "https://registry.npmjs.org/http-signature/-/http-signature-1.2.0.tgz",
  "integrity": "sha1-muzZJRFHcvPZW2WmCruPfbj7rOE=",
  "requires": {
    "assert-plus": "^1.0.0",
    "jsprim": "^1.2.2",
    "sshpk": "^1.7.0"
  }
},
"iconv-lite": {
  "version": "0.4.23",
  "resolved": "https://registry.npmjs.org/iconv-lite/-/iconv-lite-0.4.23.tgz",
  "integrity": "sha512-neyTUVFtahjf0mB3dZT77u+8O0QB89jFdnBkd5P1JgYPbPaia3gXXOVL2fq8VyU2gMMD7SaN7QukTB/pmXYvDA==",
  "requires": {
    "safer-buffer": ">= 2.1.2 < 3"
  }
},
"inherits": {
  "version": "2.0.3",
  "resolved": "https://registry.npmjs.org/inherits/-/inherits-2.0.3.tgz",
  "integrity": "sha1-Yzwsg+PaQqUC9SRmAiSA9CCCYd4="
},
"ipaddr.js": {
```

```
"version": "1.8.0",
"resolved": "https://registry.npmjs.org/ipaddr.js/-/ipaddr.js-1.8.0.tgz",
"integrity": "sha1-6qM9bd16zo9/b+DJygRA5wZzix4=",
},
"is-extendable": {
  "version": "1.0.1",
  "resolved": "https://registry.npmjs.org/is-extendable/-/is-extendable-1.0.1.tgz",
  "integrity": "sha512-arnXMxT1hhoKo9k1LZdmlNyJdDDfy2v0fXjFlmOK4+i8ul/6WlbVge9bhM74OpNPQPMG
  UToDtZ+KXAlPneJxOA==",
  "requires": {
    "is-plain-object": "^2.0.4"
  }
},
"is-plain-object": {
  "version": "2.0.4",
  "resolved": "https://registry.npmjs.org/is-plain-object/-/is-plain-object-2.0.4.tgz",
  "integrity": "sha512-h5PpgXkWitc38BBMYawTYMWJHFZJVnBquFE57xFpjB8pJFiF6gZ+bU+Wyl/yqXiFR5m
  dLsgYNaPe8uao6Uv9Og==",
  "requires": {
    "isobject": "^3.0.1"
  }
},
"is-typedarray": {
  "version": "1.0.0",
  "resolved": "https://registry.npmjs.org/is-typedarray/-/is-typedarray-1.0.0.tgz",
  "integrity": "sha1-5HnICFjfDBsR3dppQPlgEfzaSpo="
```

```
},
"isobject": {
  "version": "3.0.1",
  "resolved": "https://registry.npmjs.org/isobject/-/isobject-3.0.1.tgz",
  "integrity": "sha1-TkMekrEalzFjaqH5yNHMvP2reN8="
},
"isstream": {
  "version": "0.1.2",
  "resolved": "https://registry.npmjs.org/isstream/-/isstream-0.1.2.tgz",
  "integrity": "sha1-R+Y/evVa+m+S4VAOaQ64uFKcCZo="
},
"jsbn": {
  "version": "0.1.1",
  "resolved": "https://registry.npmjs.org/jsbn/-/jsbn-0.1.1.tgz",
  "integrity": "sha1-peZUwuWi3rXyAdls77yoDA7y9RM=",
  "optional": true
},
"json-schema": {
  "version": "0.2.3",
  "resolved": "https://registry.npmjs.org/json-schema/-/json-schema-0.2.3.tgz",
  "integrity": "sha1-tlDikuWALwWVTOcvt8qTogvnhM="
},
"json-schema-traverse": {
  "version": "0.3.1",
  "resolved": "https://registry.npmjs.org/json-schema-traverse/-/json-schema-traverse-0.3.1.tgz",
  "integrity": "sha1-NJptRMU6Ud6JtAgFxdXlm0F9M0A="
}
```

```
  },
  "json-stringify-safe": {
    "version": "5.0.1",
    "resolved": "https://registry.npmjs.org/json-stringify-safe/-/json-stringify-safe-5.0.1.tgz",
    "integrity": "sha1-Epai1Y/UXxmg9s4B1lcB4sc1tus="
  },
  "jsprim": {
    "version": "1.4.1",
    "resolved": "https://registry.npmjs.org/jsprim/-/jsprim-1.4.1.tgz",
    "integrity": "sha1-MT5mvB5cwG5Di8G3SZwuXFastqI=",
    "requires": {
      "assert-plus": "1.0.0",
      "extsprintf": "1.3.0",
      "json-schema": "0.2.3",
      "verror": "1.10.0"
    }
  },
  "lodash": {
    "version": "4.17.14",
    "resolved": "https://registry.npmjs.org/lodash/-/lodash-4.17.14.tgz",
    "integrity": "sha512-mmKYbW3GLuJeX+iGP+Y7Gp1AiGHGbXHCOh/jZmrawMmsE7MS4znI3RL2FsjsbqOyM  
ayHIInjOeykW7PEajUk1/xw=="
  },
  "lodash.get": {
    "version": "4.4.2",
    "resolved": "https://registry.npmjs.org/lodash.get/-/lodash.get-4.4.2.tgz",
```

```
"integrity": "sha1-LRd/ZS+jHpObRDjVNBSZ36OCXpk="
},
"media-typer": {
  "version": "0.3.0",
  "resolved": "https://registry.npmjs.org/media-typer/-/media-typer-0.3.0.tgz",
  "integrity": "sha1-hxDXrwqmJvj/+hzgAWhUUmMIV0g="
},
"methods": {
  "version": "1.1.2",
  "resolved": "https://registry.npmjs.org/methods/-/methods-1.1.2.tgz",
  "integrity": "sha1-VSmk1nZUE07cxSZmVoNbD4Ua/O4="
},
"mime": {
  "version": "1.4.1",
  "resolved": "https://registry.npmjs.org/mime/-/mime-1.4.1.tgz",
  "integrity": "sha512-KI1+qOZu5DcW6wayYHSzR/tXKCDc5Om4s1z2QJdDULzLcmf3DvzS7oluY4HCTrc+9Fi
KmWUgeNLg7W3uIQvxtQ=="
},
"mime-db": {
  "version": "1.33.0",
  "resolved": "https://registry.npmjs.org/mime-db/-/mime-db-1.33.0.tgz",
  "integrity": "sha512-BHJ/EKruNIqJf/QahvxwQZXKygoQ256myeN/Ew+THcAa5q+PjyTTMMeNQC4DZw5Aw
fvlsUrA6B67NKMqXDbzQ=="
},
"mime-types": {
  "version": "2.1.18",
```

```
"resolved": "https://registry.npmjs.org/mime-types/-/mime-types-2.1.18.tgz",

"integrity": "sha512-
lc/aahn+t4/SWV/qcmumYjymLsWfN3ELhpmVuUFjgsORruuZPVSwaQryq+HHGvO/SI2K
VX26bx+En+zhM8g8hQ==",

"requires": {

  "mime-db": "~1.33.0"

},

},

"morgan": {

"version": "1.9.1",

"resolved": "https://registry.npmjs.org/morgan/-/morgan-1.9.1.tgz",

"integrity": "sha512-
HQStPIV4y3afTiCYVxirakhlCfGkI161c76kKFca7Fk1JusM//Qeo1ej2XaMniiNeaZklMVrh3
vTtIzpzwbpmA==",

"requires": {

  "basic-auth": "~2.0.0",

  "debug": "2.6.9",

  "depd": "~1.1.2",

  "on-finished": "~2.3.0",

  "on-headers": "~1.0.1"

},

"dependencies": {

"debug": {

"version": "2.6.9",

"resolved": "https://registry.npmjs.org/debug/-/debug-2.6.9.tgz",

"integrity": "sha512-
bC7EldrdJaJnPbAP+1EotYvqZsb3ecl5wi6Bfi6BJTUcNowp6cvspg0jXznRTKDjm/E7AdgFB
VeAPVMNcKGsHMA==",

"requires": {
```

```
    "ms": "2.0.0"
  }
}
},
"ms": {
  "version": "2.0.0",
  "resolved": "https://registry.npmjs.org/ms/-/ms-2.0.0.tgz",
  "integrity": "sha1-VgiurfwAvmwpAd9fmGF4jeDVl8g="
},
"nan": {
  "version": "2.10.0",
  "resolved": "https://registry.npmjs.org/nan/-/nan-2.10.0.tgz",
  "integrity": "sha512-bAdJv7fBLhWC+/BlS0Oza+mvTaNQtP+1RyhhhvD95pgUJz6XM5IzgmXOkItJ9tkoCiplvAnXI1tNmmUD/eScyA=="
},
"negotiator": {
  "version": "0.6.1",
  "resolved": "https://registry.npmjs.org/negotiator/-/negotiator-0.6.1.tgz",
  "integrity": "sha1-KzJxhOiZIQEXeyhWP7XnECrNDKk="
},
"oauth-sign": {
  "version": "0.8.2",
  "resolved": "https://registry.npmjs.org/oauth-sign/-/oauth-sign-0.8.2.tgz",
  "integrity": "sha1-Rqarfwrq2N6unsBWV4C31O/rnUM="
},
```

```
"requires": {
  "forwarded": "~0.1.2",
  "ipaddr.js": "1.8.0"
},
"punycode": {
  "version": "1.4.1",
  "resolved": "https://registry.npmjs.org/punycode/-/punycode-1.4.1.tgz",
  "integrity": "sha1-wNWmOycYgArY4esPpSachN1BhF4="
},
"qs": {
  "version": "6.5.2",
  "resolved": "https://registry.npmjs.org/qs/-/qs-6.5.2.tgz",
  "integrity": "sha512-N5ZAX4/LxJmF+7wN74pUD6qAh9/wnvdQcjq9TZjev vXzSUo7bfmw91saqMjzGS2xq91/o
dN2dW/WOI7qQHNDGA=="
},
"range-parser": {
  "version": "1.2.0",
  "resolved": "https://registry.npmjs.org/range-parser/-/range-parser-1.2.0.tgz",
  "integrity": "sha1-9JvmtIeJTdxA3MIKMi9hEJLgDV4="
},
"raw-body": {
  "version": "2.3.3",
  "resolved": "https://registry.npmjs.org/raw-body/-/raw-body-2.3.3.tgz",
  "integrity": "sha1-FXq7F8A0wRnTcn7R6bJW0oIqY+Q="
},
"vcap_services": {
  "version": "0.2.0",
  "resolved": "https://registry.npmjs.org/vcap_services/-/vcap_services-0.2.0.tgz",
  "integrity": "sha1-zlNqhWniczyznmzFOUVvxXZ1nyg="
}
```

```
},
"verror": {
  "version": "1.10.0",
  "resolved": "https://registry.npmjs.org/verror/-/verror-1.10.0.tgz",
  "integrity": "sha1-OhBcoXBTr1XW4nDB+CiGguGNpAA=",
  "requires": {
    "assert-plus": "^1.0.0",
    "core-util-is": "1.0.2",
    "extsprintf": "^1.2.0"
  }
},
"watson-developer-cloud": {
  "version": "3.5.1",
  "resolved": "https://registry.npmjs.org/watson-developer-cloud/-/watson-developer-cloud-3.5.1.tgz",
  "integrity": "sha512-9Gy9L//CscG3ACJAKo6ETLPuK0rhzgONLi5K7Ttr3GA4HGPNfEAQpXvyb38rC8DMQW9kCvD7EeIVsk3jCk1E5Q==",
  "requires": {
    "@types/csv-stringify": "~1.4.2",
    "@types/extend": "~3.0.0",
    "@types/file-type": "~5.2.1",
    "@types/is-stream": "~1.1.0",
    "@types/node": "~10.3.5",
    "@types/request": "~2.47.1",
    "async": "~2.6.1",
    "buffer-from": "~1.1.0",
    "csv-stringify": "~1.0.2",
```

```
"extend": "~3.0.1",
"file-type": "^7.7.1",
"isstream": "~0.1.2",
"mime-types": "~2.1.18",
"object.omit": "~3.0.0",
"object.pick": "~1.3.0",
"request": "~2.87.0",
"vcap_services": "~0.3.4",
"websocket": "~1.0.26"
},
"dependencies": {
  "qs": {
    "version": "6.5.2",
    "resolved": "https://registry.npmjs.org/qs/-/qs-6.5.2.tgz",
    "integrity": "sha512-N5ZAX4/LxJmF+7wN74pUD6qAh9/wnvdQcj9TZjvXzSUo7bfmw91saqMjzGS2xq91/odN2dW/WOI7qQHNDGA=="
  },
  "request": {
    "version": "2.87.0",
    "resolved": "https://registry.npmjs.org/request/-/request-2.87.0.tgz",
    "integrity": "sha512-fcogkm7Az5bsS6SI0sibkbhcKsnyon/jV1kF3ajGmF0c8HrttdKTPRT9hieOaQHA5HEq6r8OyWOO/o781C1tNw==",
    "requires": {
      "aws-sign2": "~0.7.0",
      "aws4": "^1.6.0",
      "caseless": "~0.12.0",

```

```
"combined-stream": "~1.0.5",
"extend": "~3.0.1",
"forever-agent": "~0.6.1",
"form-data": "~2.3.1",
"har-validator": "~5.0.3",
"http-signature": "~1.2.0",
"is-typedarray": "~1.0.0",
"isstream": "~0.1.2",
"json-stringify-safe": "~5.0.1",
"mime-types": "~2.1.17",
"oauth-sign": "~0.8.2",
"performance-now": "^2.1.0",
"qs": "~6.5.1",
"safe-buffer": "^5.1.1",
"tough-cookie": "~2.3.3",
"tunnel-agent": "^0.6.0",
"uuid": "^3.1.0"
}
},

"resolved": "https://registry.npmjs.org/request/-/request-2.87.0.tgz",
"integrity": "sha512-
fcogkm7Az5bsS6Sl0sibkbhcKsnyon/jV1kF3ajGmF0c8HrttdKTPRT9hieOaQHA5HEq6r8Oy
W0o/o781C1tNw==",
"requires": {
  "aws-sign2": "~0.7.0",
```

```
"aws4": "^1.6.0",
"caseless": "~0.12.0",
"combined-stream": "~1.0.5",
"extend": "~3.0.1",
"forever-agent": "~0.6.1",
"form-data": "~2.3.1",
"har-validator": "~5.0.3",
"http-signature": "~1.2.0",
"is-typedarray": "~1.0.0",
"isstream": "~0.1.2",
"json-stringify-safe": "~5.0.1",
"mime-types": "~2.1.17",
"oauth-sign": "~0.8.2",
"performance-now": "^2.1.0",
"qs": "~6.5.1",
"safe-buffer": "^5.1.1",
"tough-cookie": "~2.3.3",
"tunnel-agent": "^0.6.0",
"uuid": "^3.1.0"
}
},
"vcap_services": {
  "version": "0.3.4",
  "resolved": "https://registry.npmjs.org/vcap_services/-/vcap_services-0.3.4.tgz",
  "integrity": "sha1-FUv5QEAlEqzKI98iY/xg72aEWto="
}
```

```
}  
},  
"websocket": {  
  "version": "1.0.26",  
  "resolved": "https://registry.npmjs.org/websocket/-/websocket-1.0.26.tgz",  
  "integrity": "sha512-fjcrYDPIQxpTnqFQ9JjxUQcdvR89MFAOjPBIF+vjOt49w/XW4fJknUoMz/mDIn2eK1Adsl  
VojcaOxOqyZZV8rw==",  
  "requires": {  
    "debug": "^2.2.0",  
    "nan": "^2.3.3",  
    "typedarray-to-buffer": "^3.1.2",  
    "yaeti": "^0.0.6"  
  },  
  "dependencies": {  
    "debug": {  
      "version": "2.6.9",  
      "resolved": "https://registry.npmjs.org/debug/-/debug-2.6.9.tgz",  
      "integrity": "sha512-bC7ElrdJaJnPbAP+1EotYvqZsb3ecl5wi6Bfi6BJTUcNowp6cvspg0jXznRTKDjm/E7AdgFB  
VeAPVMNcKGSHMA==",  
      "requires": {  
        "ms": "2.0.0"  
      }  
    },  
    "ms": {  
      "version": "2.0.0",  
      "resolved": "https://registry.npmjs.org/ms/-/ms-2.0.0.tgz",
```

```
"integrity": "sha1-VgiurfwAvmwpAd9fmGF4jeDVl8g="
}
}
},
"yaeti": {
  "version": "0.0.6",
  "resolved": "https://registry.npmjs.org/yaeti/-/yaeti-0.0.6.tgz",
  "integrity": "sha1-8m9ITXJoTPQr7ft2lwqhYI+/lXc="
}
}
}
```

applications:

- path: ./server

memory: 512M

instances: 1

domain: mybluemix.net

name: Fake_news_detection

disk_quota: 1024M

env:

CLASSIFIER_ID: placeholder

NATURAL_LANGUAGE_CLASSIFIER_USERNAME: placeholder

NATURAL_LANGUAGE_CLASSIFIER_PASSWORD: placeholder

CHAPTER 8

TESTING

8.1 SYSTEM TESTING

System testing is a level of testing that validates the complete and fully integrated software product. The purpose of a system test is to evaluate the end-to-end system specifications. The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement. System Testing is actually a series of different tests whose sole purpose is to exercise the full computer-based system.

- Testing the fully integrated applications including external peripherals in order to check how components interact with one another and with the system as a whole. This is also called End to End testing scenario.
- Verify thorough testing of every input in the application to check for desired outputs.
- Testing of the user's experience with the application.

In simple terms, Software Testing means Verification of Application Under Test (AUT). It involves execution of a software component or system component to evaluate one or more properties of interest Testing is important because software bugs could be expensive or even dangerous. Software bugs can potentially cause monetary and human loss, System testing is vital to the success of the system. System testing makes a logical assumption. Its utility like as user -oriented vehicle before implementation. It is required for an effective performance of system.

- The testing is done in the same environment as of the Production environment which helps to understand the user perspective and prevents the issues which can occur when the system goes live.
- If this testing is done in a systematic and proper manner, then it would help in mitigating the post-production issues.
- This testing tests both the application architecture and business requirements

TYPES OF TESTS

Unit testing

Strategy: Black Box and White Box Testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results. Unit testing is a level of software testing where individual units/ components of a software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output. In procedural programming, a unit may be an individual program, function, procedure, etc. In object-oriented programming, the smallest unit is a method, which may belong to a base/ super class, abstract class or derived/ child class. (Some treat a module of an application as a unit. This is to be discouraged as there will probably be many individual units within that module.) Unit testing frameworks, drivers, stubs, and mock/ fake objects are used to assist in unit testing.

Integration testing

Strategy: Black Box and White Box Testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components. Integration testing is a level of software testing

where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units.

Functional test

Strategy: Black Box Testing

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals. Functional testing is centred on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

System Test

Strategy: Black Box Testing

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software

Regression Testing

Strategy: Black Box and White Box Testing

Regression testing will be executed to check whether components of the entire application still working well after some modification had been made. This testing method is important because the improvement that had been made after receive the feedbacks from previous testing may cause the system become unstable. Hence, regression testing will re-test the process after a significant modification made on the application

Beta Testing

For the final stage for product testing, Beta testing will be implemented. After the application is consider stable, a beta version is published. This phase involve sending the product to beta sites outside the company for real-world exposure for the product. A Beta testing is regarding a list of a partial or full version application by Beta users (usually experienced application user) to explore unexpected error or bug exist within the application

TESTING STRATEGY:

A strategy for system testing integrates system test cases and design techniques into a well-planned series of steps that results in the successful construction of software. The testing strategy must co-operate test planning, test case design, test execution, and the resultant data collection and evaluation .A strategy for software testing must accommodate low-level tests

that are necessary to verify that a small source code segment has been correctly implemented as well as high level tests that validate major system functions against user requirements. Software testing is a critical element of software quality assurance and represents the ultimate review of specification design and coding. Testing represents an interesting anomaly for the software. Thus, a series of testing are performed for the proposed system before the system is ready for user acceptance testing.

SYSTEM TESTING:

Software once validated must be combined with other system elements (e.g. Hardware, people, and database). System testing verifies that all the elements are proper and that overall system function performance is achieved. It also tests to find discrepancies between the system and its original objective, current specifications and system documentation.

UNIT TESTING:

In unit testing different are modules are tested against the specifications produced during the design for the modules. Unit testing is essential for verification of the code produced during the coding phase, and hence the goals to test the internal logic of the modules. This testing is carried out during the programming stage itself. In this type of testing step, each module was found to be working satisfactorily as regards to the expected output from the module.

The test cases of unit testing are as follows:

Name of the Test	Testing for IBM Watson Score
Feature being Tested	Watson Score
Description	When we upload the news to classify the IBM Watson will classify the news and gives the score.
Sample input	News
Expected output	Score for the news .
Actual Output	Score
Remarks	Test case Pass

TEST CASE-1

Name of the Test	Testing for Blockchain
Feature being Tested	Blockchain
Description	Building the p-2-p nodes using ethereum.
Sample input	Token to enter into network
Expected output	Validation from super user or minor.
Actual Output	Validates or not validates
Remarks	Test case Pass

TEST CASE-2

The results of integration testing as follows:

Name of the Test	Testing for Fake news
Feature being Tested	Type of News
Sample input	News
Expected output	News type.
Actual Output	Fake News
Remarks	Test case Pass

TEST CASE-3



Name of the Test		Testing for Real news
Feature being Tested	Type of News	
Sample input	News	
Expected output	News type.	
Actual Output	Real News	
Remarks	Test case Pass	

TEST CASE-4



CONCLUSION

With the increasing popularity of peer to peer network, more and more people consume news from peer to peer network instead of traditional news media. However, peer to peer network has also been used to spread fake news, which has strong negative impacts on individual users and broader society. Here we solve the problem of fake news by splitting the project in two phases: blockchain & NLP. In the blockchain phase, we enable any users to upload the news. In the NLP phase, we reviewed existing fake news detection approaches from a data mining perspective, including feature extraction and model construction.

BIBLIOGRAPHY

- [1] Fette, N. Sadeh, and A. Tomasic, "Learning to detect phishing emails," in Proceedings of the 16th international conference on World Wide Web ACM.
- [2] Ponnurangam Kumaraguru, Aditi Gupta, Hemank Lamba. Faking sandy : "Characterizing and identifying fake images on twitter during Hurricane Sandy ". ACM,2013.
- [3] Shu, Kai, et al. "Fake news detection on social media: A data mining perspective." ACM SIGKDD Explorations Newsletter19.1 (2017): 22-36.
- [4] Janze, Christian, and Marten Risius. "Automatic Detection of Fake News on Social Media Platforms." (2017)
- [5] David Laser, Matthew Baum, Nir Grinberg, Lisa Friedland. "Combating fake news: An agenda for research and action". 2017.
- [6] Conroy N., Rubin V and Chen Y(2015) "Automatic deception detection: Methods for finding fake news".
- [7] Fan C(2017).Classifying fake news. Retrieved February 18,2018 from <http://www.conniefan.com/2017/03/classifying-fake-news>.
- [8] Xinyi zhou, Reza zafarani, kai Shu and Huan Liu 2019. "Fakenews: Fundamental theories, Detection Strategies and Challenges"
<https://doi.org/10.1145/3289600.3291382>.