

Project Name: Fake News

**Big Data Science Track (3 months)**

Presented By:

|  |  |
| --- | --- |
| Abdel-Rahman Helmy | Ahmed Nagy |
| Ahmed Matboly | Tarek Samir |

Supervised By:

*Eng. Sayed A. Omar*

*Dr. Doaa Hassan*

# TABLE OF CONTENTS

[**TABLE OF CONTENTS**](#_duvj7n8k6h13) **2**

[**Acknowledgement**](#_6trluz9ygru0)  **3**

**Abstract 3**

[**1 INTRODUCTION**](#_prp1z8losif5) **4**

[**2 BACKGROUND**](#_2y862ke34tlk) **5**

[2.1 OVERVIEW](#_algq7i234mw4) 5

[2.1.1 ARTIFICIAL INTELLIGENCE](#_tmzmxjivegm1) 6

[Machine Learning](#_x5i2a58fhdve) 6

[Data Science Workflow](#_rxmhkqk7jech) 7

[2.1.2 Data Gathering](#_plbkrfos8500) 7

[2.1.3 Data Preprocessing](#_z9dynxloa7k1) 8

[2.1.4 Model Creation and Training](#_wa5s65mbkffr) 8

[TF-IDF](#_2swa74i7qarf) 8

[Random Forest Classifier](#_rgr1gxrmwff6) 9

[Logistic Regression](#_vhj3rbspfnyk) 9

[Testing](#_73lnbexgj6q6) 10

**3** [**Implementation**](#_bcw0emi2qn15) **11**

**4 Conclusion** 12

# 

# Acknowledgements:

I would like to express my deepest appreciation to all those who provided me the possibility to complete this project. A special thanks to our instructor Eng. Sayed Omar in teaching us everything that we have learning during this course .And another special thanks to our supervisor Dr. Doaa Hassan, whose contribution in simulating suggestions and encouragement, helped us to coordinate our project especially in choosing this project and writing this documentation.

# 

# 

# 

# 

# ABSTRACT

In this research summary, we explore the application of NLP techniques to the detection of ‘fake news’, which is, misleading news stories that come from non-reputable sources. Using a dataset obtained from Kaggle (fake\_or\_real\_news.csv), (Train.csv) and (test.csv). We are using classification models and applying Term Frequency-Inverse Document Frequency (TF-IDF) detection to a corpus of about 166086 rows. We are skeptical about the generalizability of these findings, however, and include ample discussion on next steps for exploration in this space.

Chapter One

# 1 INTRODUCTION

Any company can grow fast rapidly if it is only can make their decision in the right way. Competitive companies always using fake news to distract others companies from making their own decision correctly.

We as a Data Scientists and Big Data Specialists can predict and classify if this news is fake or true.

The model aims to provide us with the result of an article's correctness by preprocessing the data and cleaning it and modeling it with the use of Logistic Regression Algorithm.

For this project to be properly evaluated, the following is delivered in the body of this documentation and attached to it:

1. *Background*: For every aspect of the project, a background of the tools and methods referenced is provided in full to provide context for the implementation of this project.
2. *Design*: The complete design of the project, including the requirements, specifications, and any diagrams used to describe the project in all stages of its execution.
3. *Implementation:* Complete source code (attached), along with specific technical documentation for every function, class, or method written, and build instructions.
4. *Prototype:* Completely functioning.
5. *Future Steps:* What can be improved, and how we will improve it.

Chapter Two

# 2 BACKGROUND

## 

## 2.1 OVERVIEW

The purpose of this section is to provide a comprehensive understanding or context for the implementation of this project; the theories referenced later, in the implementation, along with any terminology, or technology, will be introduced in full here.

In our project we are working on *sklearn* as a Data Science, and *spark* as a Big Data. We managed to do the preprocessing and modeling for the same dataset in the two approaches as a practice and implementing the work we have.

We used some Machine Learning Algorithms to model the dataset such as Random Forest and Logistic Regression Algorithms. To do the modeling correctly we have to preprocess the dataset in a write manner so the resulting dataset will be functional.

## 2.1.1 ARTIFICIAL INTELLIGENCE

## 

## Machine Learning

Machine Learning is the subfield of Computer Science that, according to Arthur Samuel in 1959, gives “computers the ability to learn without being explicitly programmed.” It explores the study and construction of algorithms that can learn from and make predictions on data.

Evolved from the study of pattern recognition and computational learning theory in artificial intelligence, machine learning explores the study and construction of algorithms that can learn from and make predictions on data – such algorithms overcome following strictly static program instructions by making data-driven predictions or decisions, through building a model from sample inputs. Machine learning is employed in a range of computing tasks where designing and programming explicit algorithms with good performance is difficult or infeasible; example applications include email filtering, detection of network intruders or malicious insiders working towards a data breach, optical character recognition (OCR), learning to rank, and computer vision.

Machine learning is closely related to (and often overlaps with) computational statistics, which also focuses on prediction-making through the use of computers. It has strong ties to mathematical optimization, which delivers methods, theory and application domains to the field. Machine learning is sometimes conflated with data mining, where the latter subfield focuses more on exploratory data analysis and is known as **unsupervised** learning. Machine learning can also be **supervised** and be used to learn and establish baseline behavioral profiles for various entities and then used to find meaningful anomalies[[1]](#footnote-1).

#### 

#### Data Science Workflow

The core steps of any machine learning workflow regardless of the choice of algorithms, are as follows:

1. *Data Gathering:* Get the data.
2. *Data Preprocessing:* Clean, prepare, and manipulate the data.
3. *Model Creation and Training*: Create the machine learning model, and train it using a subset of the data.
4. *Testing:* Test the model against the remaining data, and measure the success of the model in its initial purpose.
5. *Improvement*: Based on the testing results, improve the model, test again, and repeat until the desired result has been achieved.

In this section, we will explain the Machine Learning concepts/background relative to this workflow.

### 

### 2.1.2 Data Gathering



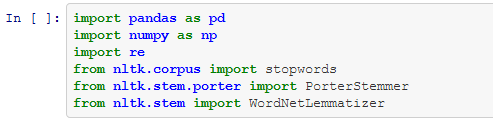
Data Gathering usually involves a process that ensures the integrity, and relevance of the gathered data, and that is thoroughly designed and carefully implemented. However, in the context of this project, we will rely on the *Kaggle* dataset as a base, which means we will make the assumption that the data gathered and presented by Kaggle is reliable. This provides us with:

1. *title,* which is*:*
   * *The title of an article.*
2. *text*, which is:
   * The body of the article
3. *Label*, which only have the name of the class if it is true or fake.

Therefore, we still need to gather more data. So, we made this dataset is the training dataset and we had another dataset that contains the same features as a test dataset.

### 

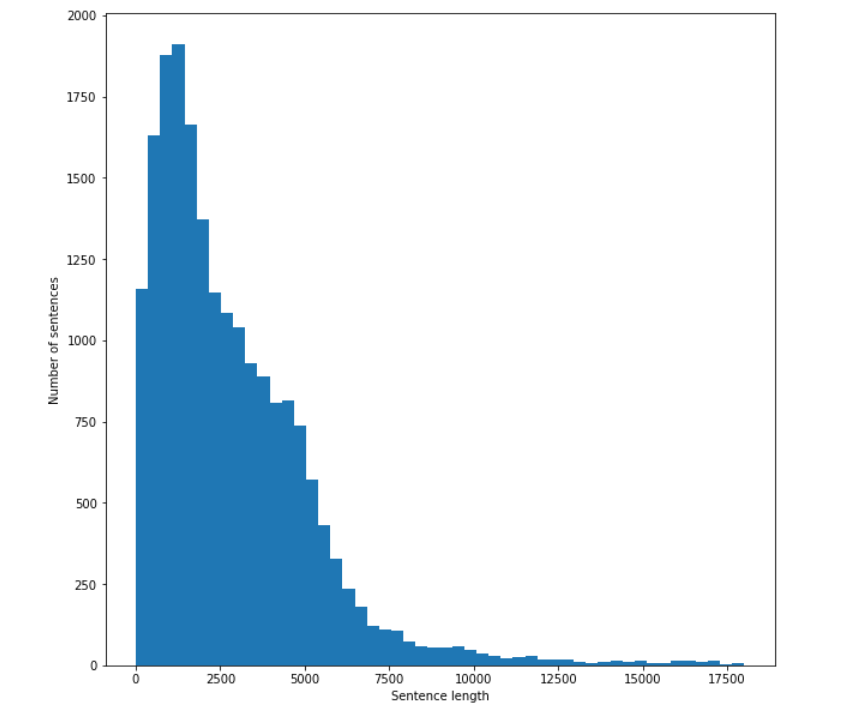
### 2.1.3 Data Preprocessing

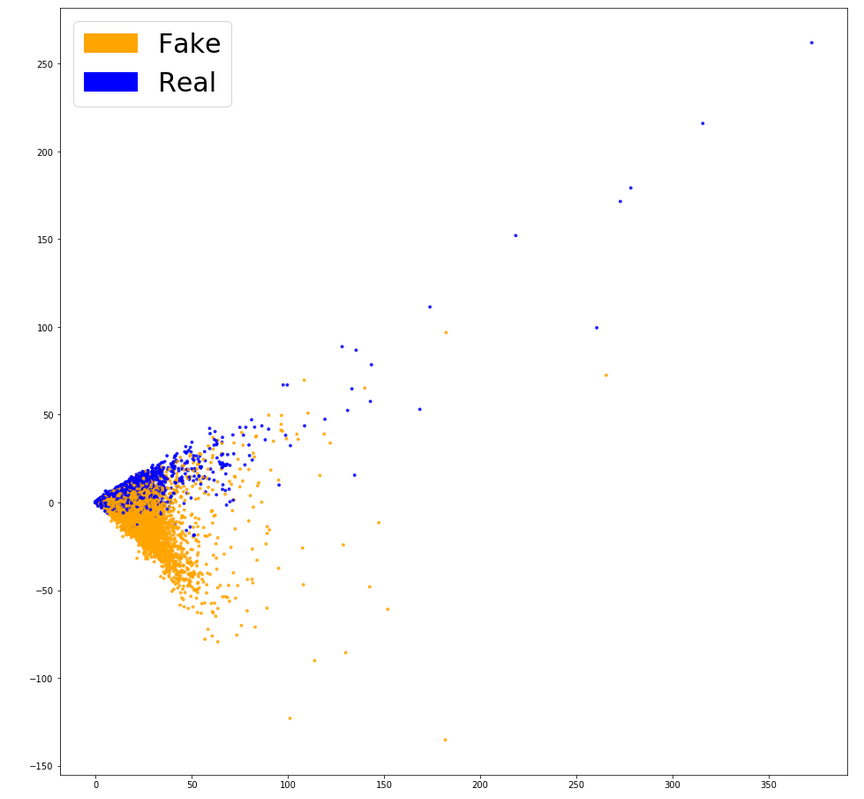


The aim of this step is to prepare the data to be used by our model. This means that we will have to represent it in a way that is consumable by our model and that makes sense in its context, regardless of the way it is stored (in a database, or extracted files, etc.).

Here are the checklist to use to clean our data:

* Remove all irrelevant characters such as any non-alphanumeric characters.
* Tokenize your text by separating it into individual words.
* Remove words that are not relevant, such as "@".
* Convert all characters to lowercase, in order to treat words such as "hello", "Hello" and "HELLO".
* Consider combining misspelled or alternately spelled words to a single representation such as (e.g. "cool"/"coool")
* Consider lemmatization (reduce stopwords such as "am", "the", "are", and "is" to a common from such as "be")



Plotting Sentence length vs Number of sentences in a histogram

Labels of the

datasets

### 2.1.4 Model Creation, Training and Testing

#### Term Frequency-Inverse Document Frequency (TF-IDF)

Now that we have the necessary *cleaned dataset*, we can use them to score measure widely used in information retrieval (IR) or summarization. TF-IDF is intended to reflect how relevant a term is in a given document.

The intuition behind it is that if a word occurs *multiple times in a document*, we should boost its relevance as it should be more meaningful than other words that appear fewer times (TF). At the same time, if a word occurs many times in a document but also *along many other documents*, maybe it is because this word is just a frequent word; not because it was relevant or meaningful (IDF).

Defining what a “relevant word” means:

We can come up with a more or less subjective definition driven by our intuition: a word’s relevance is proportional to the amount of information that it gives about its context (a sentence, a document or a full dataset). That is, the most relevant words are those that would help us, as humans, to better understand a whole document without reading it all.

#### Random Forest Classifier

Random Forest is a flexible, easy to use machine learning algorithm that produces, even without hyper-parameter tuning, a great result most of the time. It is also one of the most used algorithms, because its simplicity and the fact that it can be used for both classification and regression tasks. In this post, you are going to learn, how the random forest algorithm works and several other important things about it.

To classify a new object from an input vector, put the input vector down each of the trees in the forest. Each tree gives a classification, and we say the tree "votes" for that class. The forest chooses the classification having the most votes (over all the trees in the forest).

Each tree is grown as follows:

1. If the number of cases in the training set is N, sample N cases at random - but with replacement, from the original data. This sample will be the training set for growing the tree.
2. If there are M input variables, a number m<<M is specified such that at each node, m variables are selected at random out of the M and the best split on these m is used to split the node. The value of m is held constant during the forest growing.
3. Each tree is grown to the largest extent possible.

After we had used the TF-IDF Vectorizer and split the dataset to (x) and (y). We used Random Forest Classifier, as a first test model, and we calculated the accuracy and make prediction with it.

We found the score of this Algorithm is 84.8%.

#### 

#### 

#### Logistic Regression

Logistic regressionis a statistical method for analyzing a dataset in which there are one or more independent variables that determine an outcome. The outcome is measured with a dichotomous variable (in which there are only two possible outcomes).

The goal of logistic regression is to find the best fitting (yet biologically reasonable) model to describe the relationship between the dichotomous characteristic of interest (dependent variable = response or outcome variable) and a set of independent (predictor or explanatory) variables. Logistic regression generates the coefficients (and its standard errors and significance levels) of a formula to predict a logit transformation of the probability of presence of the characteristic of interest:



Where p is the probability of presence of the characteristic of interest. The logit transformation is defined as the logged odds:

  
And  
   
  
Rather than choosing parameters that minimize the sum of squared errors (like in ordinary regression), estimation in logistic regression chooses parameters that maximize the likelihood of observing the sample values.

We found the score of this Algorithm is 92.9%.

### 

### 

### Testing

This is the data typically used to provide an unbiased evaluation of the final model that are completed and then fit on the training dataset. Actually, such data is used for testing the model whether it is responsive or working appropriately or not.

We used our dataset *test.csv to* know exactlythe accuracy of our models.

Chapter Three

# 3 Implementation

# To-do…!!

# API sescription --------------------------

Flask is a lightweight [WSGI](https://wsgi.readthedocs.io/) web application framework. It is designed to make getting started quick and easy, with the ability to scale up to complex applications. It began as a simple wrapper around [Werkzeug](https://www.palletsprojects.com/p/werkzeug/) and [Jinja](https://www.palletsprojects.com/p/jinja/)and has become one of the most popular Python web application frameworks.

Flask offers suggestions, but doesn’t enforce any dependencies or project layout. It is up to the developer to choose the tools and libraries they want to use. There are many extensions provided by the community that make adding new functionality easy.

# API images -------------------------------

# Website images -------------------------------

Chapter Four

**4 Conclusion**

# To-do… !!

1. “Machine Learning.” *Wikipedia*, Wikimedia Foundation, 11 Nov. 2017, https://en.wikipedia.org/wiki/Machine\_learning [↑](#footnote-ref-1)