WWW.IJRPR.COM

**International Journal of Research Publication and Reviews**

Journal homepage: www.ijrpr.com ISSN 2582-7421

# Fake Reviews Detection for Online Product Using Machine Learning

***Nikita Kardile[1], Anuja Gosavi[2], Dr. Monika Rokade[3], Dr. Sunil Khatal[4]***

[1]BE Computer engineering, Shree Gajanan Maharaj Prasarak Mandal Sharadchandra Pawar College Of Engineering Dumbarwadi kardilenikita772@gmail.com

[2]BE Computer engineering Shree Gajanan Maharaj Prasarak Mandal Sharadchandra Pawar College Of Engineering, Dumbarwadi anujagosavi1048@gmail.com

[3]Professor Department of CS, Shree Gajanan Maharaj Prasarak Mandal, Sharadchandra Pawar College Of Engineering Dumbarwadi Monikarokade4@gmail.com

[4]Professor Department of CS, Shree Gajanan Maharaj, Prasarak Mandal Sharadchandra Pawar College Of Engineering Dumbarwadi drsunilkhatal@gmail.com

**ABSTRACT—**

The emergence of e-commerce platforms has revolutionized information exchange and consumer behavior, providing a convenient alternative to traditional shopping. As online shopping grows, reviews play a crucial role in influencing customer decisions. However, some businesses attempt to manipulate this process by using fake reviews, which can mislead consumers and impact business reputations. The proposed system detects and removes fake reviews, ensuring that customers access genuine feedback. It uses keyword-based identification, sentiment analysis, and tracks IP addresses to block suspicious users. Naive Bayes Classifier (NBC) is employed for feature selection and sentiment evaluation, with performance metrics like accuracy, precision, and processing time compared against classifiers such as Random Forest, Naive Bayes, and Support Vector Machine (SVM).

***Keywords—e-commerce, online reviews, fake review detection, Naive Bayes Classifier, machine learning, sentiment analysis, review credibility.***

## I. INTRODUCTION

### *1.1 Motivation*

The development of social network platforms has given people a new way to generate and consume a great deal of information on the web. In the past, people used to get information from portal websites. However, since the arrival of online social network platforms, people tend to get information from these platforms because of their fast and efficient features. These platforms are available for users to choose the information source they are interested in. And also a large number of social network platforms such Amazon, Google+, and Facebook provide information for users.

### *1.2 Background*

Technologies are changing rapidly. Old technologies are continuously being replaced by new and sophisticated ones. These new technologies are enabling people to have their work done efficiently. Such an evolution of technology is online marketplace. We can shop and make reservation using online websites. Almost, every one of us checks out reviews before purchas- ing some products or services. Hence, online reviews have become a great source of reputation for the companies. Also, they have large impact on advertisement and promotion of products and services. With the spread of online marketplace, fake online reviews are becoming great matter of concern. People can make false reviews for promotion of their own products that harms the actual users. Also, competitive companies can try to damage each other's reputation by providing fake negative reviews. Researchers

have been studying about many approaches for detection of these fake online reviews. Some approaches are review content based and some are based on behavior of the user who is posting reviews.

Amazon is the most popular microblogging platform in the world. It is also the fastest growing social network platform and has a dominant position in the area of mircroblogging. More than 500 million registered users post 340 million Amazon messages every day, sharing their opinions and daily activities. Compared with regular microblogging platforms, Amazon messages are much shorter. You are only allowed to post 140 characters or less in one Amazon message. With all of the advantages mentioned above, Amazon thus has become a powerful platform with many kinds of information from worldwide breaking news to purchasing prod- ucts at home.

### *1.3 Need*

In our daily life Fake reviews is an important issue on social media. Using fake reviews the more criminal activity are happening in the world it causes defect on human life to avoid and stop criminal activities by using these techniques Our work considers crowd signals for detect- ing fake reviews and is motivated by tools recently introduced by Amazon that enable users to flag fake reviews.

### *1.4 Aim*

Implementation of Machine Learning Classifier to classify the Product Reviews given by people on social media platform like Amazon which helps marketer to know whether reviews are fake or genuine

### *1.5 Objectives*

1. To extract expressions of Product Review describing a target feature.
2. To classify the comments as Fake or Genuine.

## II. LITERATURE SURVEY

Title : ”Detection of fake online reviews using semi- supervised and supervised learning” Author: R. Hassan and

M. R. Islam, Year: 2019 IEEE

Authors have shown several semi-supervised and supervised text mining techniques for detecting fake online reviews in this research. They have combined features from several research works to create a better feature set.

Title :”Towards understanding and detecting fake reviews in app stores”

Author: Daniel Martens · Walid Maalej Year: 2019 Springer

Authors analyzed the market of fake review providers and their fake reviewing strategies and found that developers buy reviews to relatively expensive prices of a few dollars or deal with reviews in exchange portals

Title :”Detecting Review Manipulation on Online Platforms with Hierarchical Supervised Learn- ing”

Author: Naveen Kumar, Deepak Venugopal, Liangfei Qiu & Subodha Kumar

Year: 2018 Taylor Francis

Authors propose a novel hierarchical supervised-learning approach to increase the like- lihood of detecting anomalies by analyzing several user features and then characterizing their collective behavior in a unified manner.

Title : ”Semi-Supervised Learning Based Fake Review Detection,”

Author: H. Deng et al., Year: 2017 IEEE

Authors consider both the metadata features and content related features to construct a semi-supervised learning based fake review classifier. Firstly, they use the similarity characteristics of the text to determine a set of true negative cases or fake reviews and extract the characteristic vector from multiple aspects.

A pattern based approach for multiclass sentiment analysis in twitter

Authors have developed SENTA architecture, which concentrates on classifying of text into more classes. SENTA aims to achieve multi-class classification by providing an easy to-use graphical interface and also achieves good accuracy on classification with multiclass classification

## III. REQUIREMENT ANALYSIS

### *3.1 Functional Requirements*

Functional user requirements may be high-level statements of what the system should do but functional system requirements should also describe clearly about the system services in detail.

The following are the key fields, which should be part of the functional requirements:

- User :Execute the task

- Usability: This relates to how easily people can use your app. A measure of usability could be the time it takes for end users to become familiar with your app's functions, without training or help.
- Reliability: This is the percentage of time that your app works correctly to deliver the desired results, despite potential failures in its environment.
- Performance: This is essentially how fast your app works. A performance requirement for the app could be start in less than 20 seconds.
- Responsiveness: This requirement ensures that your app is ready to respond to a user's input or an external event no matter what it's doing currently

***3.2 Non Functional Requirements:***

***3.2.1 Performance Requirements***

- System can produce results faster on 4GB of RAM.
- It may take more time for peak loads at main node.
- The system will be available 100% of the time. Once there is a fatal error, the system will provide understandable feedback to the user.

***3.2.2 Safety Requirements***

- The system is designed in modules where errors can be detected and fixed easily.

***3.2.3 Security Requirements***

- The system is designed in modules where errors can be detected and fixed easily.

***3.2.4 Software Quality Attributes***

- Usability: This relates to how easily people can use your system. A measure of usability could be the time it takes for end users to become familiar with your system's functions, without training or help.
- Reliability: This is the percentage of time that your app works correctly to deliver the desired results, despite potential failures in its environment..
- Performance: This is essentially how fast your system works. A performance requirement for the sys- tem could be start in less than 20 seconds.
- Security : Say that your system saves all the previous QR code and lets you reuse a saved QR code.
- Responsiveness: This requirement ensures that your system is ready to respond to a user's input or an external event no matter what it's doing currently.

***3.3 External Interface Requirements***

***3.3.1 User Interfaces***

The user interface is Android app in which he need to register himself.

***3.3.2 Hardware Interfaces***

Since the application must run over the internet, all the hardware shall require to connect internet will be hardware interface for the system. As for e.g. Modem, WAN – LAN, Ethernet Cross-Cable.

***3.3.3 Software Interface Description***

This system is a multi-user, multi-tasking environment. It enables the user to interact with the server and attain interact with the server by sending the two locations for meeting and also leaves a record in the inbuilt database. It uses Java and android as the front end programming tool and MySQL as the back end application tool.

***3.3.4 Communication Interfaces***

The system shall use the HTTP protocol for communication over the internet and for the intranet communication will be through TCP/IP protocol suite.

### *3.4 System Requirement*

#### *3.4.1 Database Requirement*

- Dataset : Amazon Product Review

#### *3.4.2 Software Requirements*

1. Operating System: Windows
2. Programming Language: Python
3. IDE: Open CV

3.4.3 Hardware Requirements

1. Processor: Core Intel 5 or Higher
2. Hard Disk: 2100GB (min)
3. RAM: 8 GB or Higher

## IV. System Design and Architecture

### *4.1 System Architecture*

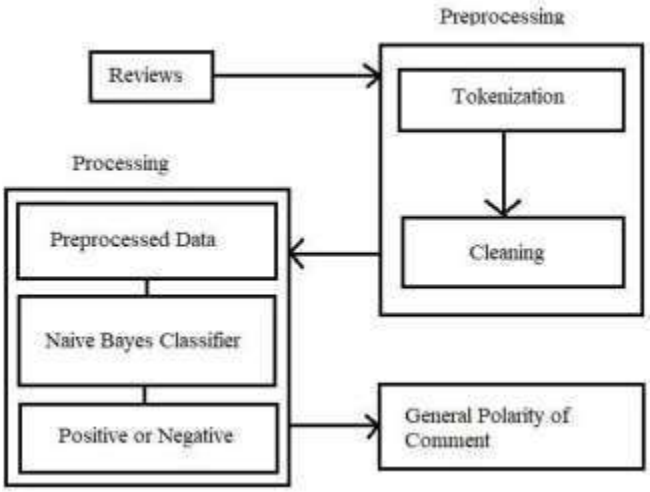



Figure 4.1: Architecture diagram

The proposed system is used to detect fake review. The first step of system is to capture the reviews given to specific product. These reviews go through process of preprocessing. The pre-processing covers the review to be tokenized and cleaned step wise. The next step is for main processing were the preprocessed data is collected and classifier is applied for same. In the system Naive Bayes classifier is used after analysis of three different classifier namely Random Forest , SVM and Naive Bayes as it gives more accuracy. After application of classifier the reviews are distributed as positive and negative. The next part of system covers to give the polarity for the comment. When three reviews are given from the same Ip address the Ip is blocked to top the fake reviews.

### *4.2 Data Flow Diagram*

A data flow diagram (DFD) is a graphical representation of the “flow” of data through an information system, modeling its process aspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated. DFDs can also be used for the visualization of data processing.

### *4.2.1 Level 0 Data Flow Diagram*

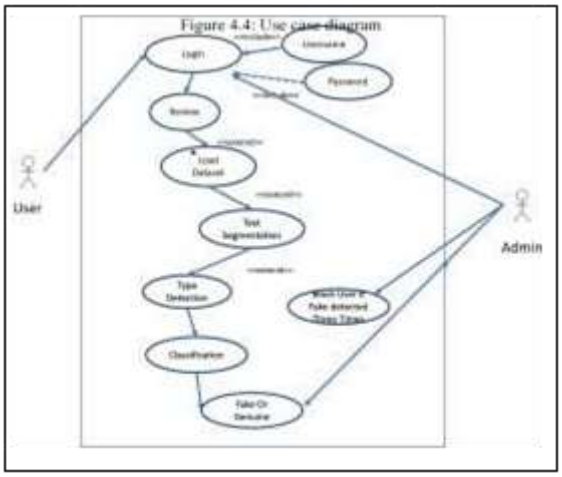
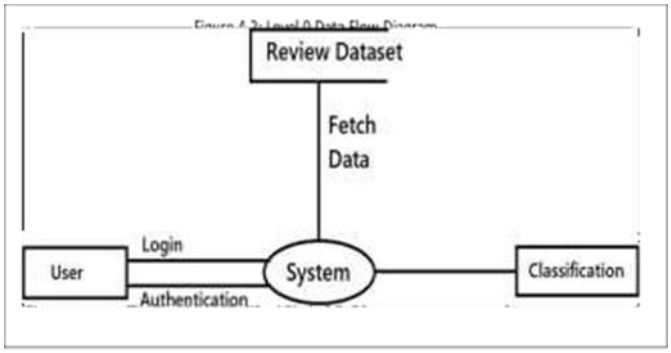


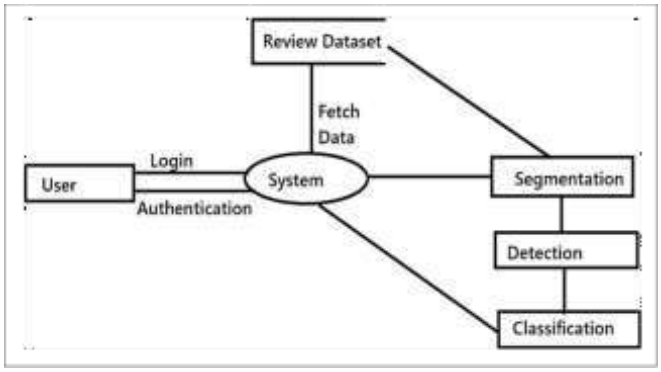

Figure 4.4: Use case diagram

Basic flow of data is through user and review data- set which is classified for fake and genuine review.User logs into the system to post review. Data set is fetched to be classified.




### *4.2.2. Level 1 Data Flow Diagram*

Classification if further divided for data flow to segment text, detect fake review and classify it to be genuine or fake.




### *4.3 UML*

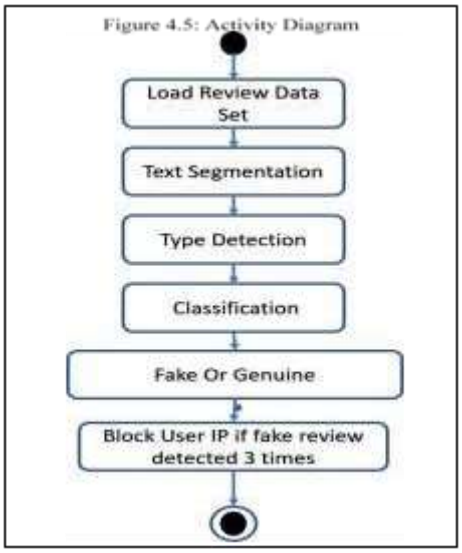
### *4.3.1 Use Case View*

Use Case Diagram. Example is given below It shows a set of use cases and actors (a special kind of class and their relationship).Use case diagrams address the static use case view of system. These diagrams are especially important in organizing and modeling the behavior of a system. The reviews are segmented and then classified by label to give polarity of fake reviews.

### *4.3.2 Activity Diagram*

Activity diagram focuses on flow of control from activity to activity. It shows work flow of our model. Above figure shows activity states, transitions, loops, decision nodes and concurrent activities use by our proposed system.

Activity starts with loading data-set and then for text segmentation to concept labelling and at last this helps to determine the polarity for tweets and determine fake reviews.




## V. Implementation Details

### *5.1 Methodology*

1. Review Acquisition : The review is captured.
2. Preprocessing : The review is tokenized and cleaned.
3. Processing:
   - Feature Extraction : Feature for fake reviews is extracted.
   - Classification : The Model classifies reviews into different category .
4. Polarity : Reviews are recognized.

### *5.2 Modules*

1. Data Preprocessing

A technique which is used to transform the raw data in a useful and efficient format. In System reviews are the Input

Data . For prepreprocessing , Cleaning and tokenization id done.

2. Data Classification

Classification is Data mining task of the predicting value of Categorical variable. This is done by building model depending on one or more attributes or features.

3. Training and Testing

Different datasets serve different purposes in preparing an algorithm to make pre- dictions and decisions based on real- world data.

**Training data:**

This type of data builds up the machine learning algorithm. The data scientist feeds the algorithm input data, which corresponds to an expected output. The model evaluates the data repeatedly to learn more about the data's behavior and then adjusts itself to serve its intended purpose.

**Test data:**

After the model is built, testing data once again validates that it can make accurate pre- dictions. If training and validation data include labels to monitor performance metrics of the model, the testing data should be unlabeled. Test data provides a final, real-world check of an unseen dataset to confirm that the ML algorithm was trained effectively.

### *5.3 Algorithm*

1. Logistic regression

Logistic regression algorithm, sometimes called logit model, is a common model for di- chotomous output variables and was extended for disease classification prediction. Sup- pose that there are p input variables where their values are indicated by x1, x2, ..., xp. Let z be a probability that an event will occur and 1-z be a probability that the event will not occur. The logistic regression model is given by

$-z \quad \log(1\ z) = \text{logit}(z) = \beta 0 + \beta 1x1 + \beta 2x2 + ... + \beta pxp$ or can be written by

Random Forest algorithm developed from trees algorithm and Bagging algorithm is mod- elled. The developed the algorithm found that it can potentially improve classification accuracy. It is also work well with a data set with large number of input variables. The algorithm is started by creating a combination of trees which each will vote for a class as shown in Fig. The figure presents how to model the Random Forest. Suppose that there are N data and M input variables in a data set where the real data used in this paper compose of data and input variables. Let k be the number of sampling groups, ni and mi be number of data and variables in group i where i is equal to 1, 2, ... and k. Each sampling group is as followed

a) ni data where ni is not greater than N are selected randomly from N.

b) mi variables where mi is not greater M are selected randomly from M. (c) A tree is grown and gives a prediction class.

After Step 1 to 3 was repeated for k times, these trees become a forest. Then the classifi- cation will be selected by a majority vote of all trees in the forest. Note that all data have to be returned to the data set before selecting a new sampling group. Therefore, there are thirteen models that will be evaluated in this process as shown in following Fig.

5.4 Tools and Technologies Python

1. Python is an interpreted high-level programming language for general-purpose program- ming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whites-pace.

$$z = \frac{e(\beta 0 + \beta 1 + \beta 2x2 + \cdots + \beta pxp)}{1e(\beta 0 + \beta 1x1 + \beta 2x2 + \cdots + \beta pxp)}$$

where 0 is the intercept and β0, β1, ..., βp are the regression coefficients.

2. Decision Tree

(a) Check if algorithm satisfies termination criteria

(b) Computer information-theoretic criteria for all attributes

(c) Choose best attribute according to the information- theoretic criteria

(d) Create a decision node based on the best attribute in step 3

(e) Induce (i.e. split) the dataset based on newly created decision node in step 4

(f) For all sub-dataset in step 5, call C4.5 algorithm to get a sub-tree (recursive call)

(g) Attach the tree obtained in step 6 to the decision node in step 4

(h) Return tree

3. Random Forest

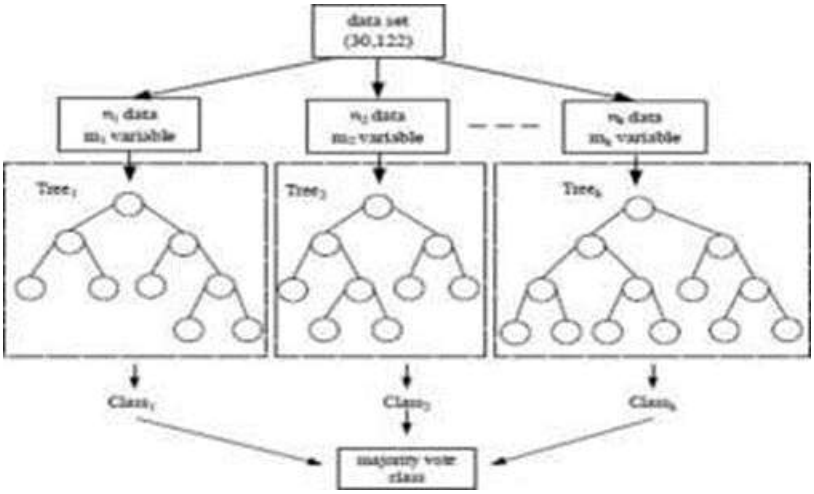



Figure 5.1: Random Forest

2. Python wrapper for it so that it can be used as Python modules. This gives two advan- tages: First, the code is as fast as original C/C++ code - since it is the actual C++ code working in background and Second, it is very easy to code in Python.
3. Python-OpenCV binding is an appropriate tool for fast prototyping of computer vision problems.

Python 7.3 Installation To install the software:

1. Point your web browser to the download page on the Python website.
2. Select the latest Windows x86 MSI Installer(python- 3.2.3.msi at the time of this writing) click on the link to download the .msi installer.
3. Run the installer (note: IE 9 will offer you this option when you click on the link).
4. Select Install for all users (the default option) and click the Next button.
5. Keep the default option as the destination directory and click Next button again.
6. Don't make any changes in the Customize Python 3.2.3 dialog, just click Next ¿ again.
7. Click Yes if asked if this program should be allowed to install software on your system.

Open CV

1. Open CV can be used for many image processing and computer vision application
2. OpenCV provide a set of image processing functions and computer vision applications. The functions are optimized for Intel architecture processors and are particularly effective with MMX technology
3. better use of up-to-date opportunities to apply computer vision in growing PC environ- ment and mobile plat-form. The library is open and has platform independent interface and supplied with whole C sources.

Tensor Flow

1. TensorFlow is an end-to-end open source platform for machine learning.
2. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in ML and developers easily build and de- ploy ML powered applications.
3. Build and train ML models easily using intuitive high-level APIs like Keras with eager execution, which makes for immediate model iteration and easy debugging.
4. Easily train and deploy models in the cloud, on-prem, in the browser, or on-device no matter what language you use. Keras
5. Keras is an API designed for human beings, not machines.
6. Keras follows best practices for reducing cognitive load: it offers consistent & simple APIs, it minimizes the number of user actions required for common use cases, and it provides clear & actionable error messages.
7. It also has extensive documentation and developer guides.

### *5.5 Software Engineering Methodology*

#### *5.5.1 Software Life-cycle used in this Project*

The Waterfall Model was first Process Model to be introduced. It is also referred to as a linear- sequential life cycle model. It is very simple to understand and use. In a waterfall model, each phase must be completed fully before the next phase can begin. This type of model is basically used for the for the project which is small and there are no uncertain requirements. At the end of each phase, a review takes place to determine if the project is on the right path and whether or not to continue or discard the project. In this model the testing starts only after the development is complete. In waterfall model phases do not overlap.

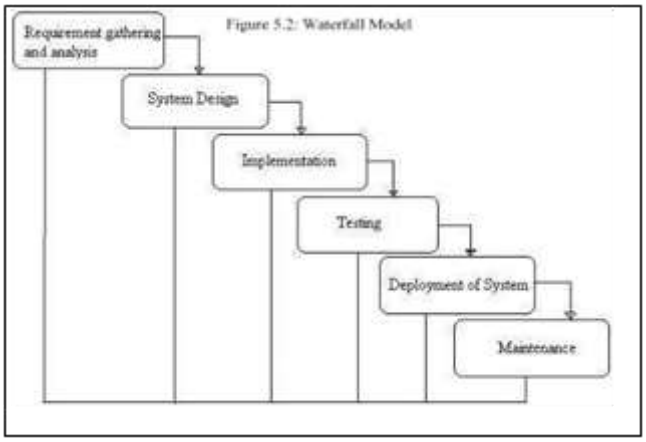


Figure 5.2: Waterfall Model

- Applications

1. This model is simple and easy to understand and use.
2. It is easy to manage due to the rigidity of the model - each phase has specific deliv- erables and a review process.
3. In this model phases are processed and completed one at a time. Phases do not overlap.
4. Waterfall model works well for smaller projects where requirements are very well understood.

**References**

[1] R. Hassan and M. R. Islam, ”Detection of fake online reviews using semi-supervised and supervised learning,” 2019 International Conference on Electrical, Computer and Com- munication Engineering (ECCE), 2019, pp. 1-5, doi: 10.1109/ECACE.2019.8679186.

[2] Martens, D., Maalej, W. Towards understanding and detecting fake reviews in app stores. Empir Software Eng 24, 3316–3355 (2019).

[3] Naveen Kumar, Deepak Venugopal, Liangfei Qiu & Subodha Kumar (2018) Detecting Review Manipulation on Online Platforms with Hierarchical Supervised Learning, Jour- nal of Management Information Systems, 35:1, 350-380, DOI: 10.1080/07421222.2018.1440758

[4] H. Deng et al., ”Semi-Supervised Learning Based Fake Review Detection,” 2017 IEEEInternational Symposium on Parallel and Distributed Processing with Applications and 2017 IEEE International Conference on Ubiquitous Computing and Communications (ISPA/IUCC), 2017, pp. 1278-1280, doi: 10.1109/ISPA/IUCC.2017.00195.

[5] W. Medhat, A. Hassan, and H. Korashy, “Sentiment analysis algorithms and applications: A survey,” Ain Shams Engineering Journal, vol. 5, no. 4, pp. 1093–1113, 2014.

[6] E. Cambria, “Affective computing and sentiment analysis,” IEEE Intel- ligent Systems, vol. 31, no. 2, pp. 102–107, 2016.

[7] S. Rosenthal, P. Nakov, S. Kiritchenko, S. Mohammad, A. Ritter, and

[8] 8V. Stoyanov, “Semeval-2015 task 10: Sentiment analysis in twitter,” in Proceedings of the 9th in- ternational workshop on semantic evaluation (SemEval 2015), pp. 451–463 , 2015.