



DEPRESSION DETECTION USING SOCIAL MEDIA FORUM

¹Ishmeet Singh Matharu, ²Ishika Singh, ³Dr. Pankaj Kumar

¹Student, ² Student, ³Professor

¹Bachelor of Technology (Computer Science and Engineering),

¹Shri Ramswaroop Memorial College of Engineering and Management, Faizabad Road, Lucknow, India

Abstract: As people become increasingly connected, through social media channels such as Facebook, Reddit, and Twitter, their digital personas begin to immerge. The present study explores the use of machine learning techniques to detect depression in users who use social media. Recent studies suggest social media usage is associated with increased depression. This can be accomplished by training and testing classifiers derived from a user's post to determine whether he or she is depressed or not. We created a web interface for blogs and can predict their posts with an accuracy of 71.37% using a trained model.

Keywords - Depression Detection, Machine Learning, Naive Bayes, SQLite, Python Reddit API Wrapper (PRAW).

I. INTRODUCTION

Worldwide, depression is a leading cause of disability and suicide, with over millions of people suffering from it. Every year, more than 300 million people are estimated to suffer from depression making it a worldwide cause for disability and death [1]. The general criteria for identifying if a person is suffering from depression is by clinical analysis by consulting to a doctor face to face. However, the early stages of depression, more than half of the population who is in fact suffering would not consult a doctor making the circumstances even worse in future [1]. Researchers and clinicians have begun using social media to detect, estimate, and track disease incidence trends [2]. The ubiquity of social media makes it possible to enrich the data available to mental health researchers and clinicians. The vast amount of information social media contains has allowed researchers to detect physical and mental illnesses [3]. Identified social media posts by individuals with major depressive order. The aim of our project is to identify such users by analyzing the posts, which they make on the social media platform. By collecting the data from the users' posts/ blogs and implying various techniques to identify whether the user is depressed or not. The data which is collected to train our model which we have deployed at the backend of the social media forum has been trained from Reddit posts that were collected using PRAW.

II. ABOUT SENTIMENT ANALYSIS

The sentiment analysis process is the process of determining the sentiment behind a particular statement or sentence. It is a classification technique that drives opinion from SocialBlog posts and formulated a sentiment, based on which sentiment classification is carried out. It refers to the process of determining the sentiment of a particular statement or sentence. This classification technique derives opinions from Blogs and formulates sentiments based on which sentiment classification is performed. Depending on the topic of interest, opinions are subjective. In order to create a sentiment, it is necessary to identify what features will determine the sentiment it embodies. Sentiment, as described in the programming model, refers to an entity class that someone conducting sentiment analysis wants to find in Posts. There can be a two-class Posts sentiment classification (positive and negative). There are two broad categories of approaches to sentiment analysis - lexicon-based and machine learning-based. On the other hand, machine learning involves identifying and extracting features from datasets and then training models based on them. To perform sentiment analysis, a number of basic steps must be followed, including data collection, pre-processing, feature extraction, selection of baseline features, sentiment detection, and subsequent classification, either using simple computation or machine learning.

III. METHODOLOGY

In order to perform sentiment analysis, we are required to collect data from the desired source (here Reddit). This data undergoes various steps of pre-processing which makes it more machine sensible than its previous form.

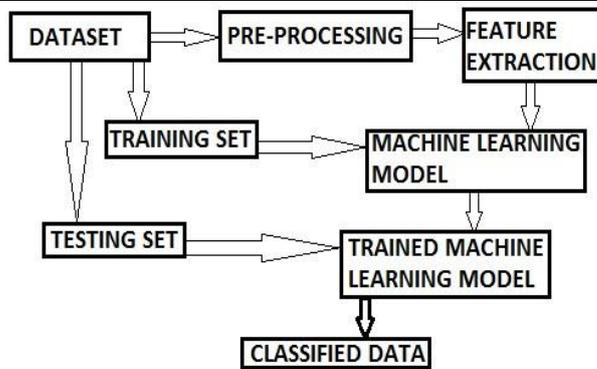


Fig. 1 – General Methodology for sentiment analysis

A. REDDIT POST COLLECTION

Reddit post-collection involves collecting relevant posts about the subject. The posts are collected using Python Reddit API Wrapper (PRAW) [5], or any other mining tool for the desired time period of analysis. Text retrieved from the internet is converted into a format depending on the convenience.

The dataset collected is crucial to achieving the efficiency of the model. The division of the dataset into training and testing sets is also a deciding factor for the efficiency of the model. The training set is the primary determinant of the results.

B. PRE-PROCESSING OF POSTS

The pre-processing of the data is a very important step as it decides the efficiency of the other steps down in line. It involves syntactical correction of the posts as desired. The steps involved should aim for making the data more machine-readable in order to reduce ambiguity in feature extraction. Below are a few steps used for pre-processing of posts -

- **Removal of re-posts.**
- **Converting upper case to lower case:** In case we are using case sensitive analysis, we might take two occurrences of same words as different due to their sentence case. It is important for an effective analysis not to provide such misgivings to the model.
- **Stop word removal:** Stop words that do not affect the meaning of the posts are removed (for example and, or, still etc.). [3] uses NLTK machine learning package for this purpose, which checks each word from the text against a dictionary [3]
- **Twitter feature removal:** Usernames and URLs are not important from the perspective of future processing; hence their presence is futile. All usernames and URLs are converted to generic tags [3] or removed.
- **Stemming:** Replacing words with their roots, reducing different types of words with similar meanings [3]. This helps in reducing the dimensionality of the feature set.
- **Special character and digit removal:** Digits and special characters don't convey any sentiment. Sometimes they are mixed with words, hence their removal can help in associating two words that were otherwise considered different.
- **Creating a dictionary to remove unwanted words and punctuation marks from the text [3].**
- **Lemmatization**

C. FEATURE EXTRACTION

A feature is a piece of information that can be used as a characteristic which can assist in solving a problem (like prediction [4]). The quality and quantity of features is very important as they are important for the results generated by the selected model. Selection of useful words from tweets is feature extraction.

- **Unigram features** – one word is considered at a time and decided whether it is capable of being a feature.
- **N-gram features** – more than one word is considered at a time.

Frequency analysis is a method to collect features with highest frequencies used in [1]. Further, they removed some of them due to the presence of words with similar sentiment (for example happy, joy, ecstatic etc.) and created a group of these words. Along with this affinity analysis is performed, which focuses on higher order n-grams in tweet feature representation.

Barnaghi et al [3], use unigrams and bigrams and apply Term Frequency Inverse Document Frequency (TF-IDF) to find the weight of a particular feature in a text and hence filter the features having the maximum weight. The TF-IDF is a very efficient approach and is widely used in text classification and data mining.

Bouazizi et al [6], propose an approach where they don't just rely on vocabulary used but also the expressions and sentence structure used in different conditions. They classified features into four classes: sentiment-based features, punctuation and syntax based features, unigram based features and pattern based features.

D. SENTIMENT CLASSIFIER

Naive Bayes (NB), and Decision Tree (DT) are some of the widely used algorithms in natural language processing tasks [17]. Of these, SVM-linear classifier demonstrates the best performance. As there is no one algorithm suited for all tasks, researchers tend to try various algorithms and enhance them for the problem of their interest [7].

1. NB:

NB is based on the “Bayes’ theorem” in probability. As a requirement of this theorem, NB can be applied only if the features are independent of each other [8].

$$P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)}$$

It is a prediction model that breaks the posterior possibilities of each class and the possible circumstances of the class for each feature. It is generally used in machine learning owing to its ability to efficiently merge the evidence from several features. Often, we know how frequently some evidence is observed, given a known outcome [17]. With the knowledge that certain evidence is observed provides us a conclusion. Although the NB classifier is considered the most straightforward method in the machine learning field, it is still competitive with SVM [9].

Reddit post-collection involves collecting relevant posts about the subject. The posts are collected using Python Reddit API Wrapper (PRAW) [5], or any other mining tool for the desired time period of analysis. Text retrieved from the internet is converted into a format depending on the convenience.

The dataset collected is crucial to achieving the efficiency of the model. The division of the dataset into training and testing sets is also a deciding factor for the efficiency of the model. The training set is the primary determinant of the results.

IV. INCORPORATED TECHNOLOGIES

The template issued to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin in this template measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

1. PRAW

PRAW (Python Reddit API Wrapper) is a Python module that provides a simple access to Reddit API. PRAW is easy to use in order to mine Post from different subreddit. For using PRAW first we have to create app instance by which we get a client-id and secret-key to initiate further process.

2. FLASK

Flask is a web application framework written in Python. Armin Ronacher, who leads an international group of Python enthusiasts named Pocco, develops it. It is based on Werkzeug WSGI toolkit and Jinja2 template engine[10].

It is considered more Pythonic web framework because in common situations the equivalent Flask web application is more explicit. Flask is also easy to get started with as a beginner because there is little boilerplate code for getting a simple app up and running.

3. NLTK

Natural Language toolkit (NLTK) is a library in python, which provides the base for text processing and classification. Operations such as tokenization, tagging, filtering, text manipulation can be performed with the use of NLTK.

The NLTK library also embodies various trainable classifiers (example – Naïve Bayes Classifier).

NLTK library is used for creating a bag-of words model, which is a type of unigram model for text. In this model, the number of occurrences of each word is counted. The data acquired can be used for training classifier models. The sentiment of the entire tweets is computed by assigning subjectivity score to each word using a sentiment lexicon.

4. Scikit-Learn

The Scikit-learn project started as scikits. learn, a Google Summer Code project by David Cournapeau. It is a powerful library that provides many machine learning classification algorithms, efficient tools for data mining and data analysis.

Below are various functions that can be performed using this library:

1. Classification: Identifying the category to which a particular object belongs.
2. Regression: Predicting a continuous-valued attribute associated with an object.
3. Clustering: Automatic grouping of similar objects into sets.
4. Dimension Reduction: Reducing the number of random variables under consideration.
5. Model selection: Comparing, validating and choosing parameters and models.
6. Preprocessing: Feature extraction and normalization in order to transform input data for use with machine learning algorithm.

V. PROPOSED WORK

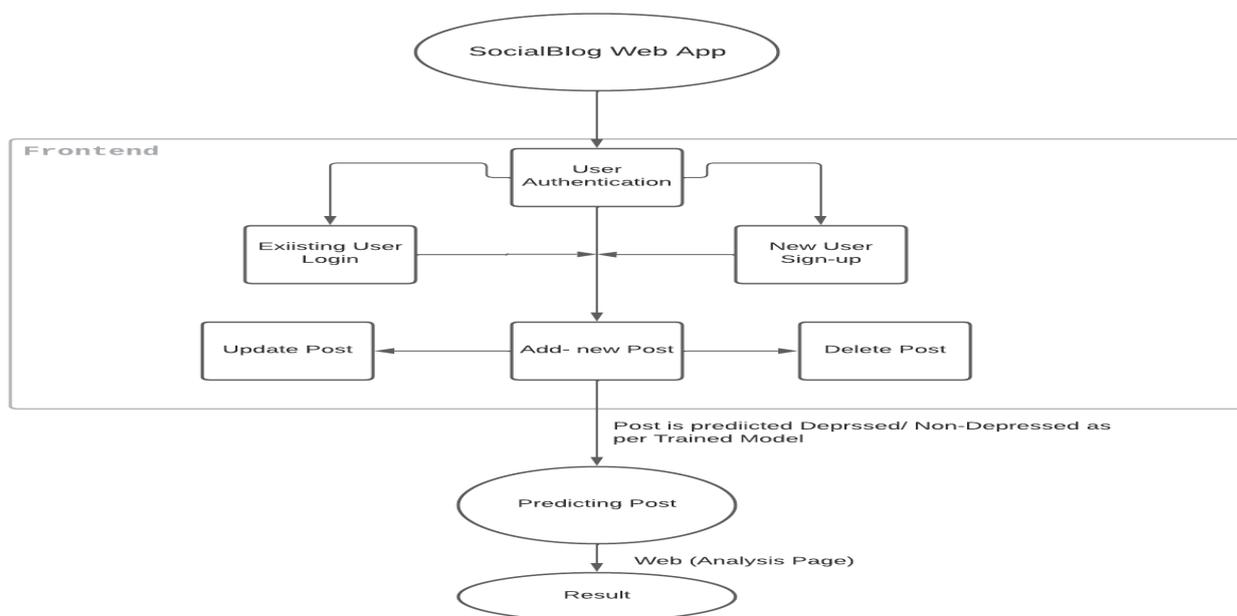


Fig-2- Social Blog Web Platform for User Depression Detection

A. SIGNUP-LOGIN MODULE:

```

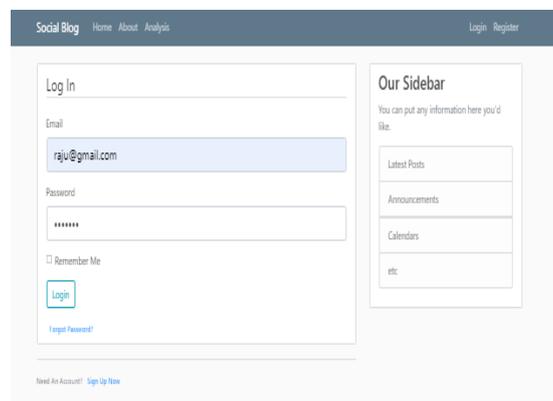
@app.route("/register", methods=['GET', 'POST'])
def register():
    if current_user.is_authenticated:
        return redirect(url_for('home'))
    form = RegistrationForm()
    if form.validate_on_submit():
        hashed_password = bcrypt.generate_password_hash(form.password.data).decode('utf-8')
        user = User(username=form.username.data, email=form.email.data, password=hashed_password)
        db.session.add(user)
        db.session.commit()
        flash('Your account has been created! You are now able to log in', 'success')
        return redirect(url_for('login'))
    return render_template('register.html', title='Register', form=form)
    
```

B. DATA COLLECTION USING PRAW:

```

import praw
reddit_posts = praw.Reddit(client_id = 'vs2VwfTEm8DtPacjHY0G6w',
    client_secret = 'Rv0JIAOABToSB-MxkVLcOv8ZEvlVQ',
    username='singhdevil123',
    passwd='*****',
    user_agent='Reddit_Posts')

subreddit_posts = reddit_posts.subreddit('depression_help')
hot_posts4 = subreddit_posts.hot(limit = 100000)
for submission in hot_posts4:
    dict["title"].append(submission.title)
    dict["subreddit"].append(submission.subreddit)
    dict["score"].append(submission.score)
    dict["id"].append(submission.id)
    dict["url"].append(submission.url)
    dict["comms_num"].append(submission.num_comments)
    dict["created"].append(submission.created)
    dict["body"].append(submission.selftext)
    
```



C. TRAINING DATA-SET:

```
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.3,random_state= 42)
x_train_vector = count_vect.transform(x_train)
x_test_vector = count_vect.transform(x_test)
model = naive_bayes.MultinomialNB()
model.fit(x_train_vector, y_train)
y_pred2 = model.predict(x_test_vector)
score = accuracy_score(y_test, y_pred2)
```

D. ADDING NEW POST:

```
@app.route("/post/new", methods=['GET', 'POST'])
@login_required
def new_post():
    form = PostForm()
    if form.validate_on_submit():
        post = Post(title=form.title.data, content=form.content.data,\
            author=current_user,depression_check = predictRoute(form.content.data) )
        db.session.add(post)
        db.session.commit()
        flash('Your post has been created!', 'success')
        return redirect(url_for('home'))
    return render_template('create_post.html', title='New Post',
        form=form, legend='New Post')
```

E. RESULT:

Analysis of depression of users with the cumulative result of posts that user post for example if user posts 10 blogs and out of which 3 are classified depressed then it will Output the mental health as depressed for that user.

```
@app.route("/analysis")
def admin():

    users = [user.to_dict() for user in User.query]
    posts = [post.to_dict() for post in Post.query]
    dict = {}
    for user in users:
        for post in posts:
            if user['id'] == post['user_id']:
                try:
                    dict[user['username']].append(post['depression_check'])
                except KeyError:
                    dict[user['username']] = [post['depression_check']]
```

Analysis

Name	Mental Health
Raju	Non-Depressed
Shyam	Depressed

VI. Conclusion

Social Blog sentiment analysis comes under the category of text and opinion mining. It focuses on analyzing the sentiments of the blogs and feeding the data to a machine learning model in order to train it and then check its accuracy, so that we can use this model for future use according to the results. It comprises of steps like data collection, text pre-processing, sentiment detection, sentiment classification, training and testing the model. This research topic has evolved during the last decade with models reaching the efficiency of almost 72%. But it still lacks the dimension of diversity in the data. Along with this it has a lot of application issues with the slang used and the short forms of words. Many analyzers don't perform well when the number of classes are increased. Also it's still not tested that how accurate the model will be for topics other than the one in consideration. Hence sentiment analysis has a very bright scope of development in future.

VII. ACKNOWLEDGMENT

We would like to thank our guide Dr. Pankaj Kumar for helping throughout this research and for the continuous guidance during the development of this platform.

REFERENCES

- [1] Shen, G., Jia, J., Nie, L., Feng, F., Zhang, C., Hu, T., Chua, T.-S., & Zhu, W. Depression detection via harvesting social media: A multimodal dictionary learning solution in *IJCAI* 3838-3844 (2017).
- [2] Varsha Sahayak, Vijaya Shete and Apashabi Pathan, "Sentiment Analysis on Twitter Data", (IJIRAE) ISSN: 2349-2163, January 2015
- [3] Peiman Barnaghi, John G. Breslin and Parsa Ghaffari, "Opinion Mining and Sentiment Polarity on Twitter and Correlation between Events and Sentiment", 2016 IEEE Second International Conference on Big Data Computing Service an Applications
- [4] Feature engineering, Wikipedia, https://en.wikipedia.org/wiki/Feature_engineering
- [5] PRAW, GeeksforGeeks , [Python | PRAW - Python Reddit API Wrapper - GeeksforGeeks](#)
- [6] Mondher Bouazizi and Tomoaki Ohtsuki, "Sentiment Analysis: from Binary to Multi-Class Classification", IEEE ICC 2016 SAC Social Networking, ISBN 978-14799-6664-6.
- [7] Jamil, Z. Monitoring tweets for depression to detect at-risk users. Université d'Ottawa/University of Ottawa, 2017
- [8] Khan, A., Baharudin, B., Lee, L. H., & Khan, K. A review of machine learning algorithms for text-documents classification. *J. of Adv. in Inf. Technol.* **1** (1), 4-20 (2010).
- [9] Nadeem, M. Identifying depression on Twitter. Preprint at arXiv:1607.07384 (2016).
- [10] Flask, [Flask Tutorial \(tutorialspoint.com\)](#)

