HATE SPEECH FINDER: A PRAGMATIC APPROACH TO COLLECT HATEFUL AND OFFENSIVE EXPRESSIONS (TEXT)

Submitted in partial fulfillment of the requirements for the degree of

Bachelor of Technology

In

CSE and CSE with specialization in Bioinformatics (BCB)

School of Computer Science & Engineering

By

Aashis Panjiyar
17BCB0142

Rajan Sahani
17BCE2318

Shivam Sah
17BCE2386

Under the guidance of

Dr. MARGRET ANOUNCIA S
School of Computer Science and Engineering
Department of Software Systems
VIT, Vellore

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

June, 2021

DECLARATION

I hereby declare that the thesis entitled "**HATE SPEECH FINDER: A PRAGMATIC APPROACH TO COLLECT HATEFUL AND OFFENSIVE EXPRESSIONS (TEXT)"** submitted by me, for the award of the degree of *Bachelor of Technology in CSE with specialization in Bioinformatics* to VIT is a record of bonafide work carried out by me under the supervision of Dr. MARGRET ANOUNCIA S.

I further declare that the work reported in this thesis has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Place : Vellore
Date :09/06/2021

GHANI R.. Shivam Sah

Signature of the Candidate

CERTIFICATE

This is to certify that the thesis entitled "**HATE SPEECH FINDER: A PRAGMATIC APPROACH TO COLLECT HATEFUL AND OFFENSIVE EXPRESSIONS (TEXT)**" submitted by **AASHIS PANJIYAR (17BCB0142)**, **RAJAN SAHANI (17BCE2318), SHIVAM SAH (17BCE2386) School of Computer Science & Engineering**, VIT, for the award of the degree of *Bachelor of Technology in CSE and CSE with specialization in Bioinformatics*, is a record of bonafide work carried out by him under my supervision during the period, 19. 12. 2020 to 09.06.2021, as per the VIT code of academic and research ethics.

The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university. The thesis fulfills the requirements and regulations of the University and in my opinion meets the necessary standards for submission.

Place : Vellore
Date : 09/06/2021

Signature of the Guide

Internal Examiner

External Examiner

Dr. PRIYA G, Dr. VAIRAMUTHU S
Head of the Department (HOD)
Department of Analytics

ACKNOWLEDGEMENTS

It is my privilege to express my sincere regards to my Guide **Dr. MARGRET ANOUNCIA S** for her valuable inputs, able guidance, encouragement, wholehearted cooperation and constructive criticism throughout my Capstone Project. I deeply express my sincere thanks to my Head of Department, Proctor and Dean for encouraging and allowing me to work and complete my Capstone project. I take this opportunity to thank all my lecturers who have directly or indirectly helped me throughout the journey. I pay respects and love to my parents and all other family members along with my classmates, team members and friends for their love and encouragement.

Aashis Panjiyar, Rajan Sahani, Shivam Sah
Student Name

Abstract

All the Organizations and Social Networking sites perform and find their struggle to fight against aggressive, hate speech, insulting and vulgar words in social Networking sites as abuse and misuse their freedom of communication with others while commenting or making any posts. Making an allowance for the number of internets and social sites user and consumer in this world and the fight initiated by aggressive, vulgar, and insulting content present in posts is a need to find and develop aggressive and hate content recognition for their comments and posts. This project utilizes a logistic regression model for arranging the words as non-hostile or negative words. The difficulties are simple grammar and word shortened form used in social media. Henceforth, there are noise elimination and normalization cycles to address these difficulties. This model can help the public authority uphold the data and electronic exchange law and diminishes the number of questions because of yearning opportunity abuse in social media. This project utilizes a logistic regression model in differentiation of the words whether they are as offensive or not through the words. The difficulties are simple grammar along with the text contraction which people use in the Networking media for interaction. Henceforth, the project has some offensive word detection and removal of cycles to address all kind of difficulties. The prototype will help the public authority uphold the data and electrical exchange law and diminishes the number of debates because of desire opportunity misuse in networking sites. In this project, it is developed a social blog to demonstrate this entire process, showing promising results. The testing is done on the basis that whether the post contains offensive content or not at the time of posting itself.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

2G	Second generation
5G	Fifth generation
ML	Machine learning
SVM	Support vector machine
LSF	Lexical synaptic feature
POS	Point of Sale
CNN	Convolutional neural network
RNN	Recurrent neural networks
TF-IDF	Term frequency-inverse document frequency
WEKA	Waikato Environment for Knowledge Analysis
TP	True positive
FP	False positive
BOW	Bag of words
UML	Unified Modeling Language
GB	Gigabyte
OLID	Offensive Language Identification Dataset
NLP	Natural Language Processing
LTC	Latent Topic Clustering
LDA	Latent Dirichlet Allocation
BERT	Bidirectional Encoder Representations from Transformers
ROC	Receiver operating characteristic curve
AUC	Area under the Curve

SYMBOLS AND NOTATIONS

100K	Hundred Thousand
%	Percentage
H/W	Hardware
S/W	Software

1. INTRODUCTION

The aggressive material through web-based collaborating media stages may be hostile, jokey, one-sided, and vulgar. The unpleasant material will cause the client to vary the others individuals' contemplations in the misconception among the others will happen and can provoke conflict in various social classes through their inconsiderate Comments on the internet and social life. As indicated by individuals, individuals can talk anything they wish, and to posts whatever they feel, individuals utilize these internets based life regions outstandingly. As a result of the expanding thought of disagreeable substance bit by bit by means of web-based communicating media objections, it is much harder to manage or to recognize that word in the blog and to find the aggressive terms concerning their possible client who start the utilization of the aggressive relations within speech. This will be a programmatic approach to stop the harassment which is going over internet with all the precautions taken to not let anyone post the bad comment or any kind of post for the betterment of the society

1.1. Theoretical Background

Recently, Online Social Network has demonstrated to be a viable vehicle for individuals to convey what needs be uninhibitedly. The clients can, without much of a stretch, impart among themselves utilizing talk errand people and offer or include posts, pictures, texts, and so on. On other client's profile. A portion of these messages possibly thought to be hostile by certain clients.

In the UK, an overview was led. Its insights show 28% of the youngsters matured somewhere in the range of 11 and 16 with a profile on a person-to-person communication webpage have encountered something disquieting on that site, of which 18% have encountered vicious language and 3% were urged to hurt themselves. Individuals are permitted to banner such remarks; however, there is no distinct answer for this issue. As the quantities of clients on informal communication have expanded quickly because of the expanded access to the web, it is demonstrating to be a test for the current frameworks to arrange such messages successfully. Therefore this report consists of a framework, which identifies such observations. Client remarks from different destinations are extricated, prepared, and investigated. The text order will be finished utilizing a short text classifier that actualizes Machine Learning calculation to construct a prescient model.

1.2. Motivation

Different scientists have successfully attempted to recognize this aggressive material and channel the mocking words by using various Filtering approaches of Machine Learning and can dismiss these words from the definite posts or dialogue. So in order of the span of the aggressive material increasing step-wise on the online interacting locales, it is increasingly tough to channel the aggressive material in a mechanized way. They have also seen in the current situation that everyone is using an aggressive sentence to comment or post about a particular topic, therefor it has also been designed with a website that detects the post and will not let the user post the content if it is aggressive. Additionally, the current framework knows the aggressive description idea in an all-out attack style word existing in the dictionary, yet it disregards to find the sentence which is by all explanations awful, yet it's a decent line or Comment in a unique way. To defeat the problem of the current framework which is logistic regression, the planned framework keep up the over-all order and can predict the all sort of aggressive material in the precise discussion and can find the possible client by means for whom the specific hostile language is spread in the discourse. To arrange the talk, the Logistic Regression calculation of information mining is utilized.

1.3. Aim of the Proposed Work

The Logistic Regression is a directed characterization strategy mostly used to group the information and in the proposed framework it can use as a text order for arrangement of the hostile term. The Logistic Regression utilizes the library SKLEARN for model choice through cross-approval and forecast. It gives the precise arrangement utilizing the forecast dependent on the preparation information. As the ubiquity of the intuitive media expanding hugely, seeing heinous Comment on person to person communication site turns into a sweeping and imperative research territory. Long-range interpersonal communication site where a lot of individuals can do dialogue all-around with anybody on the planet. The discourse has regularly done as remarks, input, audit and other structure, which might be sure or negative.

1.4. Objectives of the Proposed Work

Moreover, an overview uncovers that the real purpose for the noteworthy number of suicides submitted by teenagers is the negative Comments to them in online life. These remarks result in expanded disappointment to them, which confine them from cooperating with friends and celebrating. Aside from this, it frequently brings about passionate injury for the prey. On systems administration locales like Myspace, a generally young person have uncovered the irritation as it makes an excessive amount of enthusiastic pressure. People are given contend free hand concerning what they can post when on the web. They are likewise enabled to post hostile comments or pictures paying little mind to what lamentable ramifications it might make. With the improvement in speed nature of web use from 2G to 5G, presently in a portion of time, data can be spread far and wide.

These Interactive media likewise flops in sifting these antagonistic remarks and status which can be transferred in open stages, however, they are furnished with detailing frameworks, which empowers the client to report the substance as maltreatment, and such substance would then be able to be expelled from the social stage. For models, Facebook has the number of representatives taking a shot at the substance, which are being transferred day by day on client's walls, profiles, remarks and so forth. They can physically confirm the announced oppressive substance. Twitter additionally demands their clients not to pursue individuals if they found the substance of that client as hostile. Be that as it may, nothing unless there are other options destinations provides security systems on the server side to confine the negative remarks and the resulting harm it can acquire.

2. Literature Survey

“Since the textual substance on online internet based life is profoundly unstructured, casual, and frequently incorrectly spelt, existing exploration on message-level hostile language identification can't precisely identify a hostile substance. The used method is Lexical Syntactic Feature (LSF) which is used to differentiate aggressive material and identify potential aggressive clients in social life”. Prevalent online person to person communication destinations applies a few instruments to screen hostile substance. On Facebook, clients can add comma-isolated watchwords to the "Control Blacklist". When people include refused slogans in a post or possibly a remark on a page, the element will be accordingly documented as spam and in this manner it will be projected. By and large, most well-known internet based life utilize straightforward vocabulary based way to deal with channel hostile substance. Lexical highlights treat each word and expression as an element. Word examples, for example, the appearance of specific catchphrases and their frequencies are regularly used to speak to the language model.

Different researchers investigated character representations and their capacity contrasted along the word-stages of representations. In predictable ML techniques has been depending on feature building, experts have proposed neural-based models in association of greater directories. CNN and RNN have been linked to recognize the unforgiving language, and they have outflanked predictable ML differentiators, for instance, LR and Support Vector Machine. In any case, no examinations are discovering the ability of part methods in significant of too much directories more than the 100 thousand data.

Survey Methodologies:

In section signifies the implementations happening common ML differentiators along with the neural system based models in point. Moreover, the project represents additional points and difference models inspected. Conservative Machine Learning methods it will be represent five-element scheming founded ML differentiators which are frequently utilized aimed at unpleasant linguistic finder. Info pre-processing, text bunches are modified over into a gathering of arguments representations and consistent in TF-IDF ethics. It is to find disparate ways regarding word-level points applying n-grams spreading from to 3 then parts features from 3 to 8-grams. Every differentiators is represented through related realities:

- **Naive Bayes (NB):** Multinomial NB with added substance smoothing constant 1

- **Logistic Regression (LR):** Logistic LR with L2 regularization constant 1 and restricted memory BFGS optimization

- **Logistic Regression:** Logistic Regression with L2 regularization constant 1 and logistic misfortune function

- **Random Forests (RF):** Averaging probabilistic predictions of 10 randomized decision trees

- **Gradient Boosted Trees (GBT):** Tree boosting with learning rate 1 and logistic misfortune function

- **Neural Network-based Models:** Together with conventional machine learning techniques, which inspect neural system based models to survey their ability inside an unrivaled dataset. In particular, researchers investigated CNN, RNN, and their variation models. A pre-prepared (Pennington et al., 2014) depiction is used for word-based features. CNN: Everyone embraced Kim's (2014) execution as the check. The word-level CNN models have 3 convolutional channels of different sizes [1, 2, 3] with ReLU actuation and a most extreme pooling layer. For the character-level CNN, system uses 6 convolutional channels of different sizes [3, 4, 5, 6, 7, 8], by then include max-pooling layers sought after by 1 totally connected layer with a component of 1024. Park and Fung (2017) proposed a

HybridCNN model which beat both word-level and character-level CNNs in cruel language recognition.

To quantify the HybridCNN in this dataset, project connected the yield of max-pooled levels from word-based and character-based CNN and feed this vector to a totally related layer to anticipate the yield. All of the three CNN models (word-level, character level, and hybrid) use cross-entropy with softmax as their misfortune limit and Adam (Kingma and Ba, 2014) as the enhancer. RNN: System uses bidirectional RNN (Schuster and Paliwal, 1997) as the pattern, executing a GRU (Cho et al., 2014) cell for every standard unit.

After comprehensive boundary look for tests, the system picked 1 indoctrination layer with 50-D covered conditions and an information dropout probability of 0.3. Now for the RNN methods use cross-entropy through use in the misfortune limit and Adam as the enhancer. For a conceivable improvement, the method asks to apply a self-coordinating consideration component on RNN pattern models (Wang et al., 2017) with the objective that they may better comprehend the information by recovering content arrangements twice. System additionally examine a presented procedure, Latent Topic Clustering (LTC) (Yoon et al., 2018). The LTC technique eliminates latent theme information from the concealed conditions of RNN with utilizations along with additional details in ordering the content information.

Existing Methodology

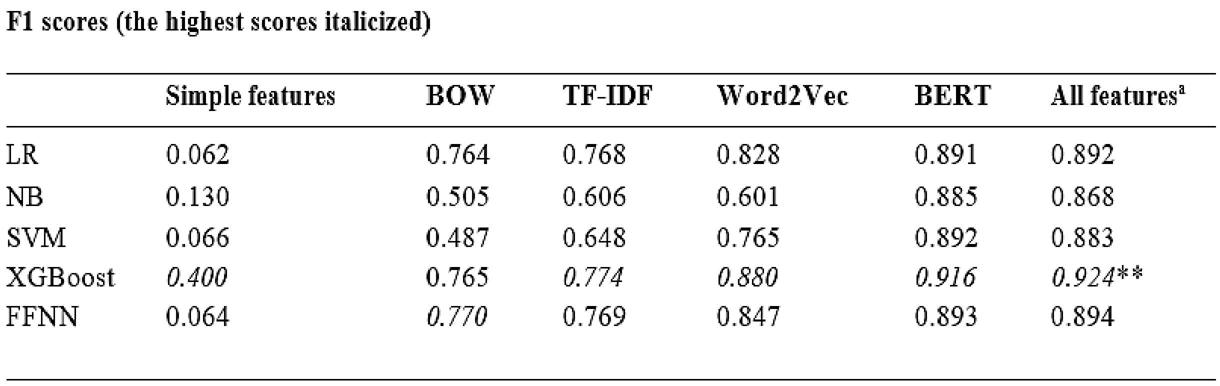
In the previous techniques, they utilized ordinary pre-processing strategies and irrelevant terms from the documented information. They investigated with three classifiers NB, Support Vector Machine along with the use of Decision Tree authentication. They have then isolated the component planetary hooked on general highlights and name natural highlights.

Researchers used to show their experiments and projects to spot also to examine the aloofness speech in communal mass media. Thus, it separated their work into 4 portions. Part A, in the systematic pre-processing strategies it utilized 3 systems in the project specifically unigram, unigram + bigram and POS colored unigram + bigram. Then in categorization mission it had different methods for the cross authentication using the application. To a limited Part B, so the scientist's job they utilized the comparative differentiator in Part A with 10-overlay authentication then additionally changed in the top methods having 5-crease authentication through grid search. Aimed at classifying individual sign job into own gatherings it utilized the called object credit with skilled CRF along with Support Vector Machine independently with ten-overlap cross authentication. Now part C, which made a similar component symbol, differentiators and limitation tuning concerning the previous 2 subtasks with 10-overlap cross authentication. It utilized LDA just as its variation implication implementation in their investigative apparatus to determine the connected subjects in harassment follow in part D.

Researcher’s likewise utilized three groups for teaching for the digital bullying differentiator in this it was found that it was having meaning, digital victimization formation and client based highlights. In pre-processing researchers eliminated every stop words along with the functional stopping in the directory. It was SVM to classify harassment remarks also good authentication. For managing the issue of disdain in contradiction of African American community the researchers in trained a Naive Bayes classifier to ready to classify the new tweet as bigoted. They pre-handled the dataset by removing the links, references, alongside the researchers established with 86% comments and posts which are bigoted simply as it had the aggressive arguments in which supported unigram prototypical which include the exercise information.

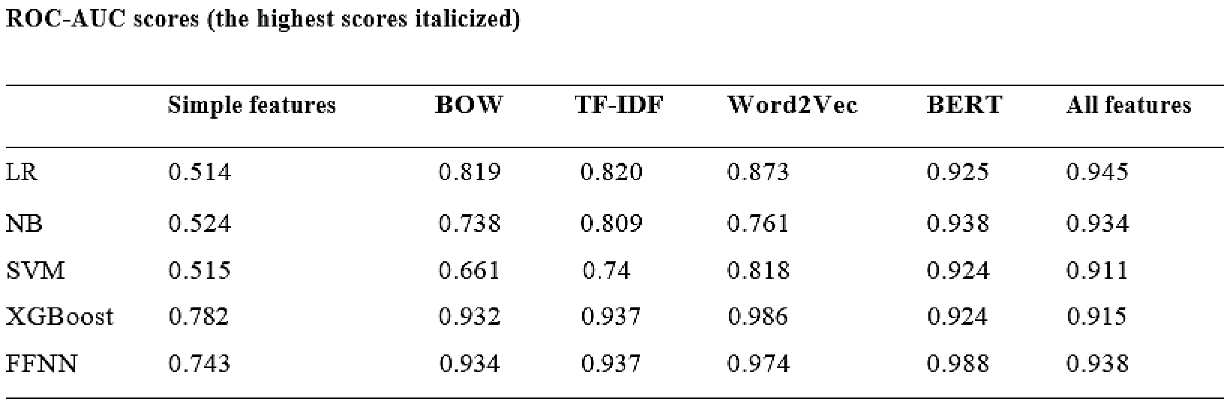
In some of the techniques after applying the standard pre-processing methods, they separated their task into 2 parts for the finding of hate speech from the user’s posts and comments. First they hired paragraph to vec to learn the circulated demonstration of comments and terms using the neural language model of the continuous BOW. This formed a little dimensional embedding’s where the semantically similar comments exist in in the similar part of the space. Then a logistic regression classifier was used on these embedding’s to categorize the type of user comment as hateful or good.

Some researchers used the Vowpal Wabbit’s regression model to find the dissimilar aspect of the consumer comments using NLP features. They separated their features into 4 groups which were N-grams, Linguistics, Syntactic and Distributional Semantics. Because of the noise found in the data they did about mild-preprocessing for the first 3 features but did not performed any normalization for the 4th feature.
From all other above mentioned surveys and all the methodologies used these are the summery of the methods used earlier



F1 scores (the highest scores italicized)

	Simple features	BOW	TF-IDF	Word2Vec	BERT	All features[a]
LR	0.062	0.764	0.768	0.828	0.891	0.892
NB	0.130	0.505	0.606	0.601	0.885	0.868
SVM	0.066	0.487	0.648	0.765	0.892	0.883
XGBoost	*0.400*	0.765	*0.774*	*0.880*	*0.916*	*0.924***
FFNN	0.064	*0.770*	0.769	0.847	0.893	0.894



ROC-AUC scores (the highest scores italicized)

	Simple features	BOW	TF-IDF	Word2Vec	BERT	All features
LR	0.514	0.819	0.820	0.873	0.925	0.945
NB	0.524	0.738	0.809	0.761	0.938	0.934
SVM	0.515	0.661	0.74	0.818	0.924	0.911
XGBoost	0.782	0.932	0.937	0.986	0.924	0.915
FFNN	0.743	0.934	0.937	0.974	0.988	0.938

Differentiating these element techniques, it is noticed a direct tendency having presentation in the differentiators at the time they are upgrading from the normal format to some advance format matching separate highlights. Although the TF-IDF in course of using the BOW highlights accomplish a lot of more regrettable, the presentation is significantly difficult from any chance supposition. In point while using the TF-IDF methods are essentially marginally improved from the function called bag of words models designates as the TF be situated basic in forecasts. Probability is that the most considerable data gain originates from the existence of certain terms like "fuck", which is noticed by BOW strategy additionally by TF-IDF highlights.

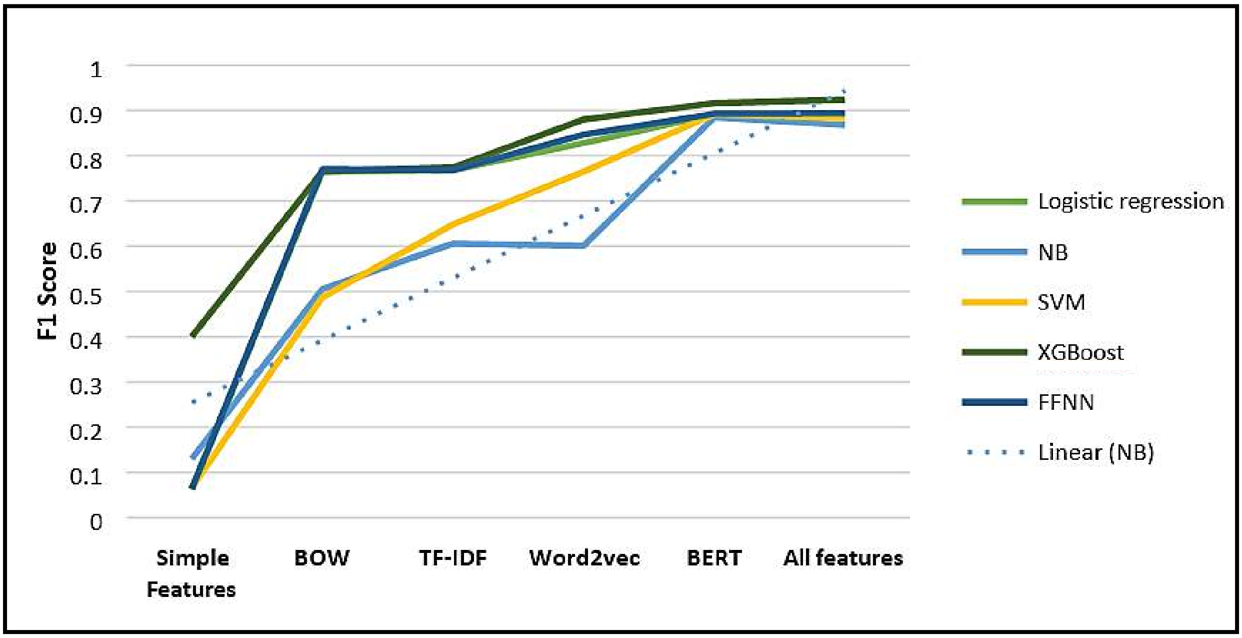


Fig.1: Comparison graph of different algorithm

2.1. Tabular format of Existing Models/Work

Sr. No.	Recognition of aggressive comments	Gaps and Limitation
1.	This is the report of Dewier Yin . which was published in 2009 The administered knowledge was utilized for recognizing provocation. The procedure utilizes substance highlights, conclusion highlights, and contextual highlights of reports with noteworthy upgrades more than a few baselines, including the Term Frequency-Inverse Document Frequency (TFIDF) approach.	The tests were finished utilizing regulated techniques. The worldly or client data was not completely used.
2.	In 2012 Ravi who is the researcher of this machine learning algorithms executed to distinguish remarks that might be hostile or offending on a long-range interpersonal communication stage. WEKA machine learning toolbox was connected and got an exactness of 82% on the dataset.	The marked datasets are not utilized in a particular area. Ready to discover just 78.86% poisonous quality.
3.	In 2012 Chen Y did a research in the Lexical Syntactic Feature (LSF) design is utilized to distinguish hostile substance and recognize the potential hostile clients in intelligent media. Thus, the LSF system performed altogether more to anything which is already there in the other techniques in aggressive substance recognition as accomplished accuracy of 98.24% and appraisal of 94.34% in line of aggressive location, just as the accuracy of 77.9% and examination of 77.8% in client hostile identification.	The main concentrated was on the classifier with the most noteworthy precision as opposed to danger of remarks.
4.	"A fire locator model which recover the composed notes of the clients on person to person communication destinations and distinguish the flaring words and compute the force level of those words". Was the result of the research in 2013 by prof. shukla	Thus, some imperative in blending two dialects like "bookon" in Urdu appears in English as "books" their tagger disregard such sort of hostile word.
5.	In 2012 Xiang wrote about the semi-managed method was connected for distinguishing foulness connected hostile substance on twitter utilizing (ML) algorithms. "In the test, the genuine	The concentrated was on word level circulation and 860,071 Tweets. Not ready to

	positive rate was 75.1% more than 4029 testing tweets utilizing Logistic Regression, fundamentally beating the well-known catchphrase coordinating benchmark, which has a TP of 69.7%, while keeping the bogus positive rate (FP) at a similar level as the gauge at about 3.77%”.	adapt up to the unpredictable component, complex weighting instrument and with more information.
6.	In this work which was written by Hirschberg and warner in 2012, the creators demonstrate an approach to perform supposition examination in blog information by utilizing the strategy for auxiliary correspondence learning. This technique suits the different issues with blog information, for example, spelling varieties, content distinction, design exchanging. By contrasting and English and Urdu dialects.	The force of the fire is distinguished however not expelled.
7.	A programmed fire identification technique, which concentrates highlights at various theoretical levels and applies staggered grouping for fire discovery. By Razvi in 2015	Necessity to progress the recognition precision for the hostile remarks.
8.	An improved cyberbullying framework which arranges the clients' remarks on YouTube utilizing content-based, cyberbullying-explicit and client based highlights by applying bolster vector machine”. This program was done by	The more semantic data isn't removed by pre-processing the terms and the context to which every starter flares.

2.2. Summary and the Gaps identified in the following Survey

- As a result of their utilization of straightforward dictionary-based programmed separating way to deal with square the hostile words and sentences, these frameworks have low exactness and may create numerous bogus positive cautions.

- The logistic regression is more supported than different systems. The LR gives worldwide information arrangement and anticipate the high exactness result than the Naive Bayes and another characterization strategy. The paper by Author portrays the less computational unpredictability of the calculation and expressed that the paper can deal with the enormous dataset than existing scale-up techniques.

- Also, when these frameworks rely upon clients and overseers to recognize and report hostile substance, they frequently neglect to take activities in an auspicious manner.

- Hostile language recognizable proof in web-based social networking is a troublesome assignment because the textual substance in such condition is regularly unstructured, casual, and even incorrectly spelt.

- Albeit lexical highlights perform well in identifying hostile elements, without thinking about the linguistic structure of the entire sentence, they neglect to recognize sentences' unpleasantness which contains the same words yet in various requests.

- Early McEnery et al utilized Bag-of-Words (BOW) in repulsiveness discovery. The disadvantage of this framework is that it has a high false-positive rate. N-gram perhaps utilized as an elective way to deal with Bag of Words and it yields better outcomes. Climbed et al identify hostile tweets utilizing machine learning algorithms which accomplish a positive pace of 75.1% utilizing strategic relapse.

- This offensive content can make the user disagree the others people idea based on the words and sentence they use in their conversation.

- The non-appearance of specific words in the lexicon would yield mistaken outcomes. Kontosta et al use guideline based on correspondence to follow and sort online predators. The framework marks and breaks down talk transcripts to recognize ruthless and non-savage discourse. Spertus proposed a fire acknowledgement framework called Smokey that includes syntactic builds. It fabricates highlight vectors dependent on the language structure and semantics of each sentence inside each message.

3. Overview of the Proposed System

3.1. Introduction and Related Concepts

The planned outline can use the Logistic regression to precisely arrange and differentiate the offensive and the defensive sentence with high exactness or precision of 91.75%. The proposed system can recognize the possible customer by strategies for whosoever the offensive verbal is used. The system direct the principal near investigation of different learning models on Dislike and Insulting Language in social media website and talk about the probability of utilizing additional features and context information for updates. This undertaking applies machine learning strategies to perform electronic hostile language identification. Hostile language can be characterized as communicating preposterous subjectivity and this investigation generally centers around two classes 'sexual' and 'bigot'. From the outset, system is to be checked which algorithm which would be smarter to train with the dataset, it was found that logistic regression to be most appropriate in view of its capacity to be viable with extremely enormous datasets while algorithm like naive Bayes and SVM would not be reasonable for the given dataset.

Some related concepts are:

•KNN: K-Nearest Neighbor is unassuming and typically utilized time-based knowledge process, it helps in together for organization also reversion. The term which is not parametric is named in that the algorithm doesn't have the information additionally don't brand any possibilities identified with fundamental information stream, although time-based is known at all through the preparation stage this procedure doesn't utilize the preparation models for any outline. The preparation stage is quick and the preparation models then utilized in the analysis stage for classifying a hidden example. Though KNN will be the straightforward process, without any expectations of information to put on to together prearrangement then reversion effort, it is exorbitant in recollection then is an essential in for stock entirely preparation information and likewise in interval to link respectively fresh example to all the preparation examples and figure out resemblance.

In the wake of taking the worth of k, here the system will stack all training data and store them in memory. All through the testing stage, for each new model, the algorithm analyses the change of this instance to each training instance through some distance metric and classifications the results in ascending request. The algorithm saves the top k detachments and amasses the most frequent mark amongst these consequences. Additionally to the above algorithms, Scikit Learn and Weka were utilized for applying KNN. Albeit the precision, review and f-measure in equivalent libraries were alike, Weka application was significantly more expensive when it ways to deal with biggest dataset.

- **Logistic regression:** Logistic Regression can likewise be named as the factual methodology which gets utilized from the info explained (for example offensive or non-offensive words). In chance of yield is considered by through the mixture from some of the terms. The algorithm will widespread for using in the different terms ordering of multiple factors, termed as Multinomial L R. It is called subsequently the center purpose used, now some terms that is characterized by the shape called s bend which assigns the terms as binary numbers from 0, 1 for every one of the genuine numbers. Expression used in logistic for the characterization of the system like the LR part. Component esteems are united linearly with related burdens to assess. Result of a LR is the probability that an input vector is assigned with an obtainable class/name. Now to transform the planned possibilities into 0 or 1. It is attained by using the logistic task which confines the assortment of the yield esteems between 0 and 1 as previously expressed. Logistic regression is well supported in the part of the aggressive word finder portal as the algorithm it provides is very precise and it has three types mainly called as Binary, multinomial and ordinal and this is a very unique approach to be used. It uses lee time to be trained for the project and gives the best result for the use. Its performance in consideration to other systems and methods is comparatively very good.

- **Naive Bayes:** It is basic classifier, which relays on the theorem of the Bayesian measurements. This is also called as the effortless differentiator by confined presentation because of the not depending on other assumption identified with the highlights, disregarding some likely connections in them. Now differentiation apportions single mark in a time which is required to show the terms previously present and have been shown. It is utilized as classifier as a baseline to some model. For each accessible discussion, NB finds the statistics to find the lesson to show the term X of the values in the terms x1, x2, . . . , xn, for example p(Ck|x1, x2, . . .

, xn). For managing a significant number of highlights, the term is determined which is built on Bayes method:

$$p(C_k|\mathbf{x}) = \frac{p(C_k)p(\mathbf{x}|C_k)}{p(\mathbf{x})}$$

Equ.1: Naïve Bayes method

NB normally executes very nicely linked in the new more troublesome differentiators, having the same standard. The supposition improves on the teaching and taking the teachings procedure, which is eliminated for prototypical knowledge the characteristics independently, endlessly dropping time intricacy with huge information.

•**Neural Networks:** Also known as Artificial Neural Networks (ANNs) they get their learnings by living natural nervous systems by the way they are organized and methodology data. ANNs were never considered as the new thought yet they previously performed in the year of 1940. In any case, the investigation on this part were freeze because of confined PC limit. Currently, the PC technologies has seriously have too much of capacity in them to execute so they came back due to the personal computers.

Since they are replicating organic neural networks, their progression is likewise alike. They are not set with task-explicit rubrics, yet they learn by test similarly as humans learn. Besides, artificial Neural Networks are ready of many profoundly unified neurons in different layers where every connection conveys a signal to the next part from the neurons.

Boundaries that join neurons to be considered on their values, also the individual neurons have precise limit defining if the part of it is set off. The weight and inclination (point) values are tuned through the learning process. The technique utilized by the system to learn is called Backpropagation.

Previously clarifying how Backpropagation works, it is essential to depict the possibility of a neuron and the design of an Artificial Neural Network. A self-made part is considered by 2 stages, the preparation and using steps. Throughout the preparation stage, founded within the input shapes, learns if to actuate. For hidden inputs, in the consuming stage, the neuron determines the numbers with values which are bias and it also determines the weight which the system taught them in the time of training.

ANNs are ready of different layers. The main layer organizes the data vector, and every part from this layer signifies a particular element significance. Layer which is at the end has only a single part that provides the data of the construction in the given data. Among the 2 parts, it is probable to have any unseen layers that likewise contain self-made parts. The neurons of the individual part are linked to every one of the part from the other valued sides. They have effectively stated, every part is branded by a limit esteem. Hence, the purpose of controlling the parts are enthusiastic and is established by loads of the edges that join this with parts of the previous layer and limit esteem.

•**Random Forest:** This process can be utilized for administered learning for classification as well as for the regression work. Foremost thought is that Random Forest (RF) makes a collaborative of result trees in the time of the training stage, also every tree releases the mark of the giving in example. Frequently, profound trees tend to over fit while preparation. To defeat this matter is the capturing or bootstrap combining technique, which keeps away from the formation of connected trees and diminishes the worth of the alteration of the system. Related trees are generally reason of the systems which over fits the preparation dataset.

RF processes each component's position on forecast result by searching at what forecast mistake increases in every nodule which utilizes the particular part. Also the next vital portion given by RF is the closeness quantity, i.e., a vicinity framework.

• **TD-IDF:** It is a factual measure that assesses how relevant a word is to a document in collecting documents.

•**CHI-SQUARE:** The main point about CHI-SQUARE is that it is the calculated graphical test that is independent from both the terms and any event in the info of two variable. So, the more is the value the response on the dependence is more on the return. Also it is used for training the model.

•**F1 Score:** The calculation of the precision and recall's average is termed as F1 score. That is the reason it accepts both false negative and false positive into the consideration.

•**Precision and recall:** These two are numbers that are used together to find the reassurance of the data and to find the user interaction. Precision can be named as the segment of significant models among every recuperated instance. Recall now and again alluded to as 'sensitivity, is the part of recovered models amongst every single applicable example.

3.2. Framework, Architecture or Module for the Proposed System

System Architecture:

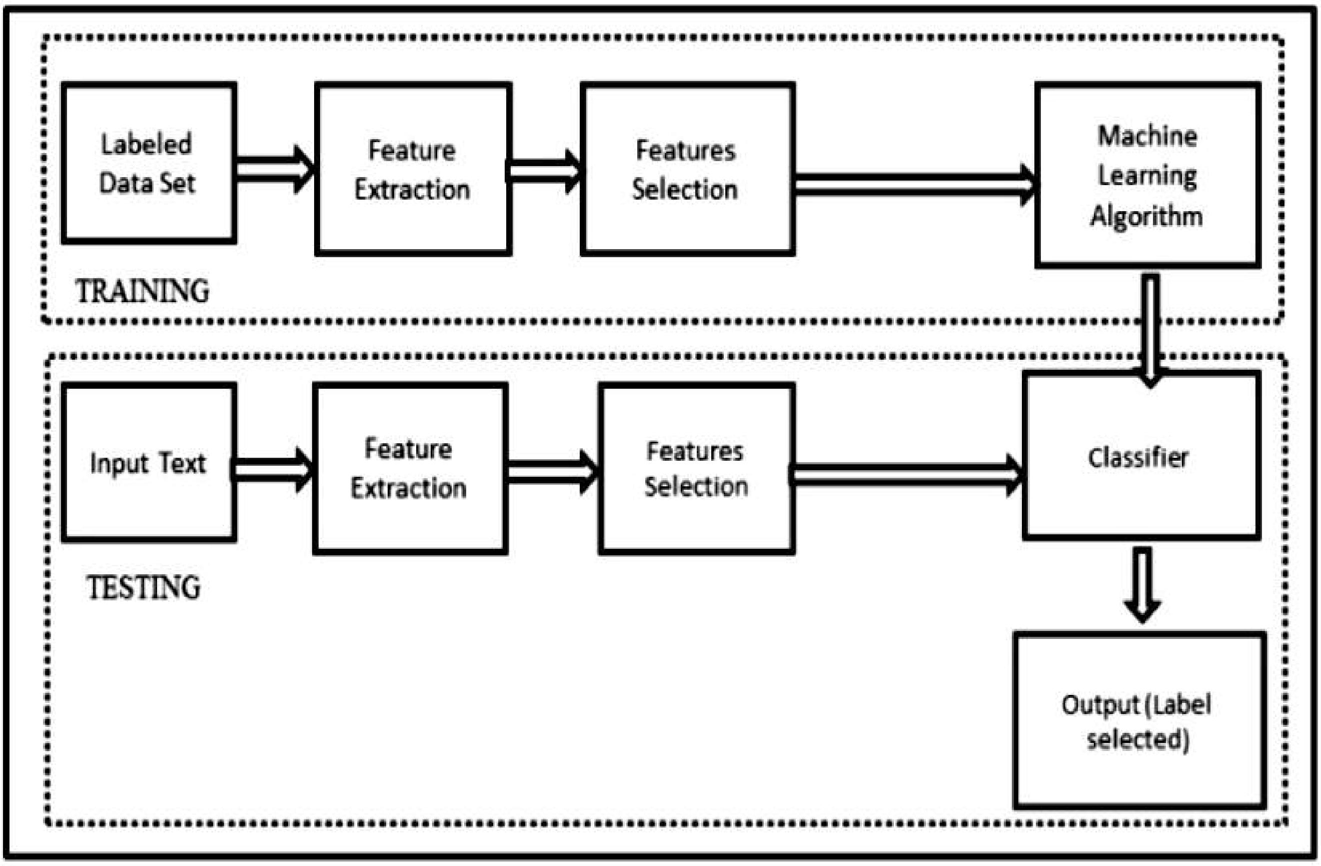


Fig. 2: Architecture of training and testing data

A. Highlight Extraction: It includes the extraction of highlights utilizing techniques like n-gram, skip-gram and term recurrence opposite archive recurrence (TF-IDF) from sentences that have been pre-handled to expel stemming, stop words and rectifying slang words. This improves the proficiency of the framework.

B. Highlight Selection: This module includes the determination of suitable highlights utilizing Chi-square.

C. Machine learning-based Algorithm: It is utilized to prepare the classifier to manufacture a proactive model.

D. Classifier: It will almost certainly characterize if the given sentence is annoying or non-annoying

3.3. Proposed System Model

ER Diagram

ER stands for Entity Relationship Diagram, a representation that shows the relation form the entity groups put away in a databank. This assists with explaining the logical design of databanks. For instance, in the following chart, the user can make their profile and the user id and name will be sent to the Admin. Then the uses can login to the website and make a post if the post is offensive it won't be posted and it isn't posted

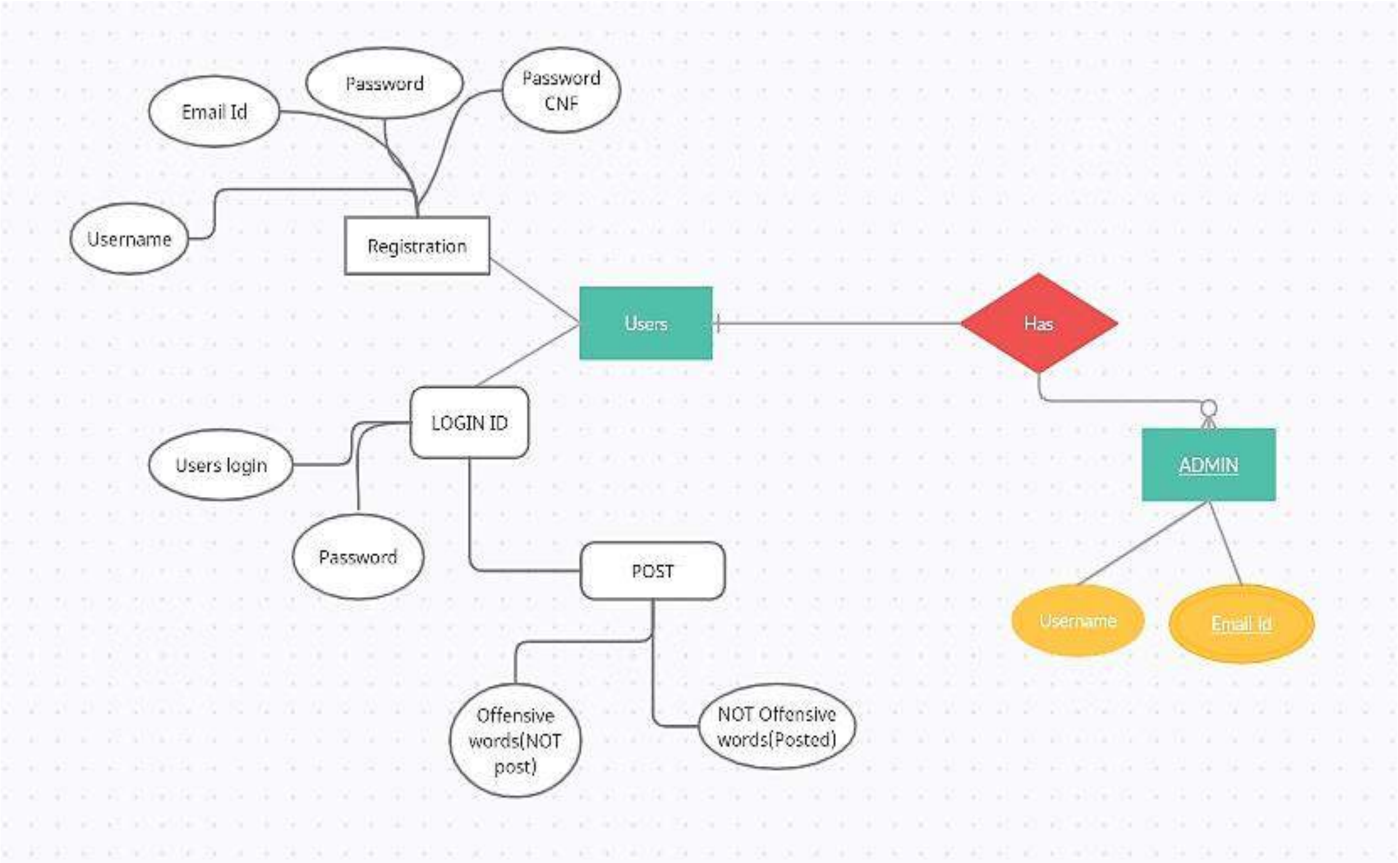


Fig 3: ER diagram of website

Use Case Diagram

The use case diagram shows what user has direct relation and interactions with entities in the system, the user can register then login in the system, and use its function to post anything which will be checked by a filter in the system, and assuming the user has been permitted to post, the user is likewise ready to see that content and can logout on the off chance that they want to leave the system.

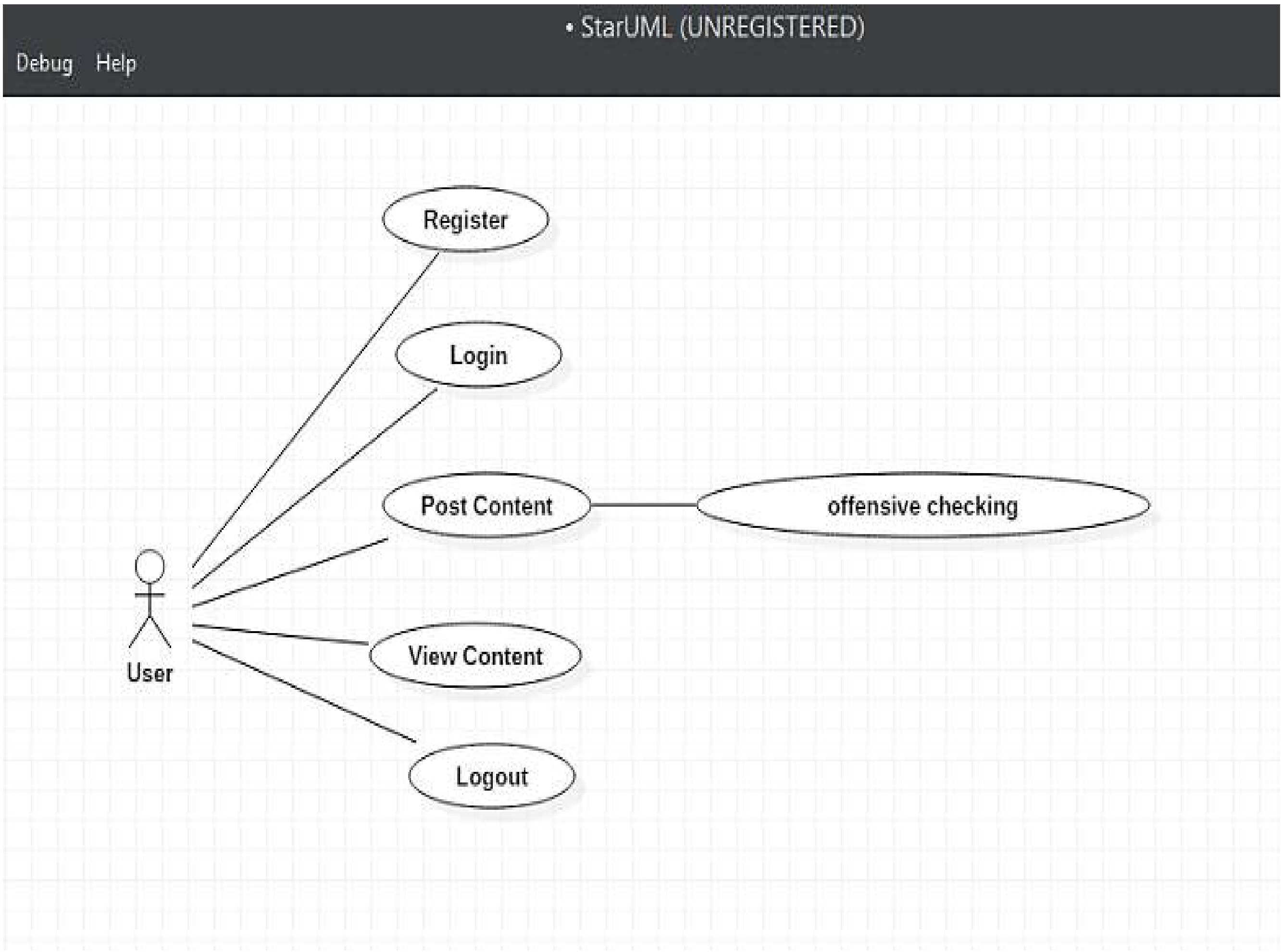


Fig. 4: Use Case Diagram for users

Sequence Diagram

This is an interaction diagram it portrays how and in what order entities work together. It additionally assists us with understanding the requirements for a given system and what are functions it every user and system can execute. Combination outline from link will be kind of association chart which gives the outcome that how strategies task in one another then in demand. "It is a form of a Message Sequence Chart. Succession outlines are now and again called occasion charts, occasion situations, and timi ng diagrams".

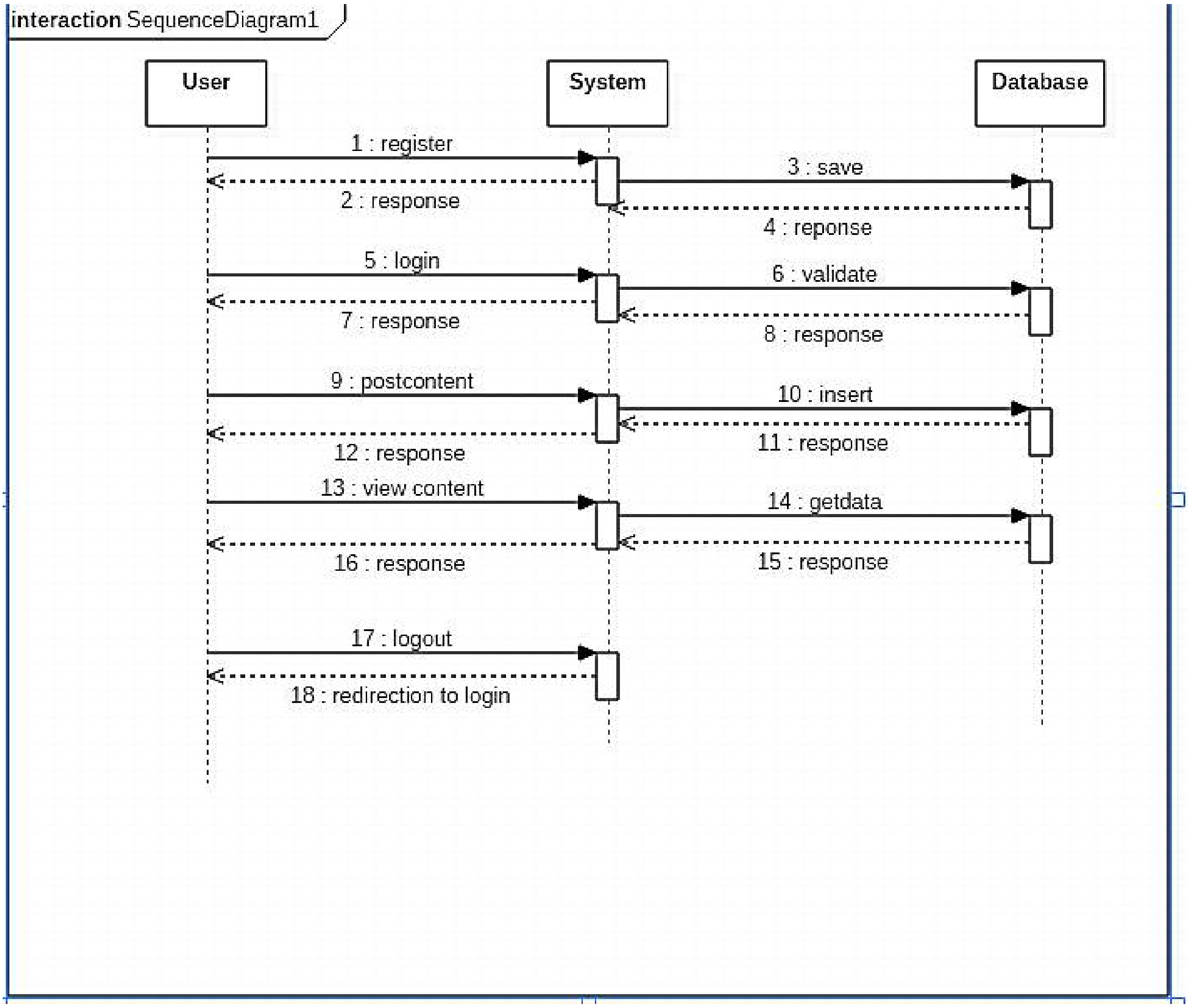


Fig.5: Sequence Diagram for website

Activity Diagram

The activity diagram offers a series of control stream in a system presenting control-stream of the construction and steps to execute the essential function in the system. It can likewise be defined in the form of process of the network. Control stream is produced using single process to alternative process. The design will be branched consecutive, or coexisting. It has been used to manage all kind of stream control in using other elements. For instance, the activity diagram for the hate speech recognition system is given beneath.

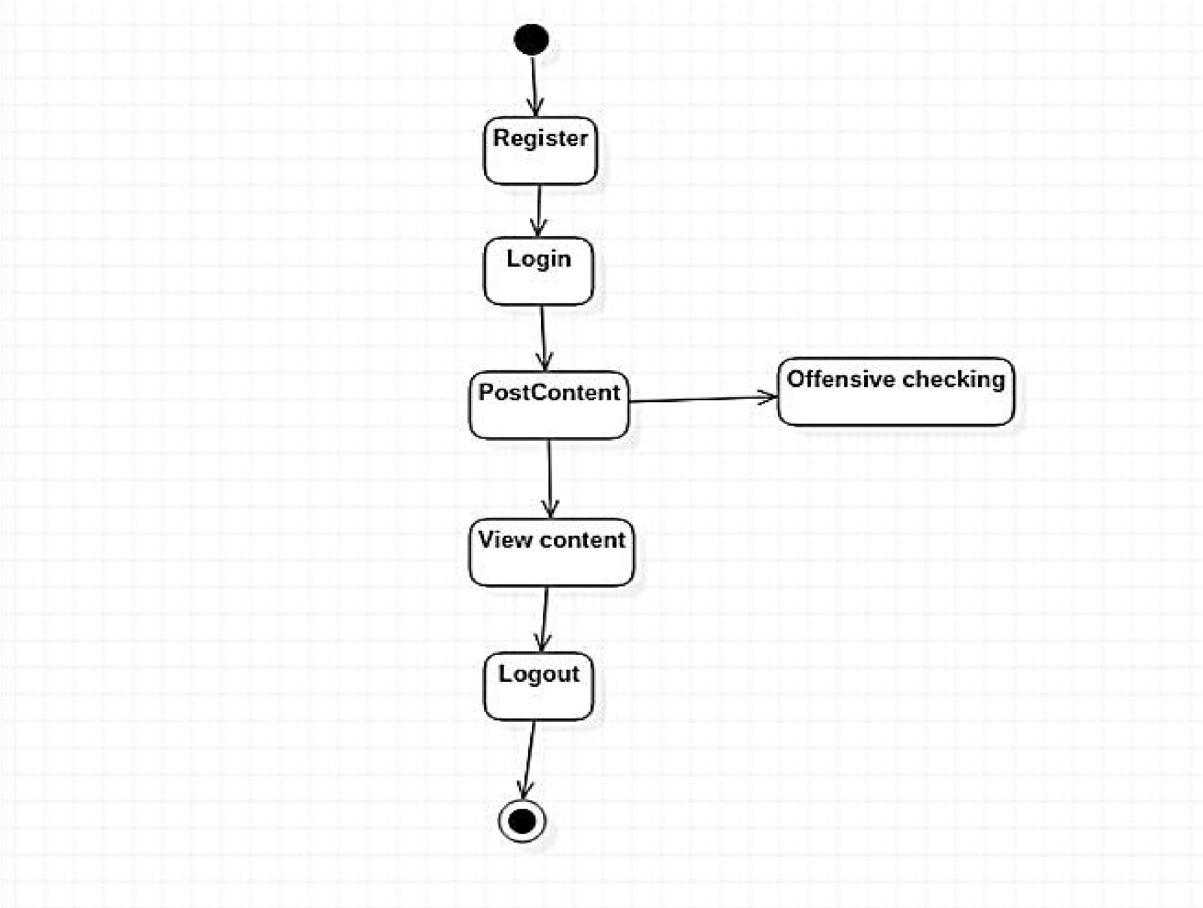


Fig.6: Activity Diagram for user website

Class Diagram

In software structure, a class chart in the link is kind of statistical design diagram that depicts the construction of a "framework by demonstrating the framework's classes, their traits, activities (or techniques), and the connections among the classes. It explains which class contains data". A class diagram is showing the conceptual modeling of the given system.

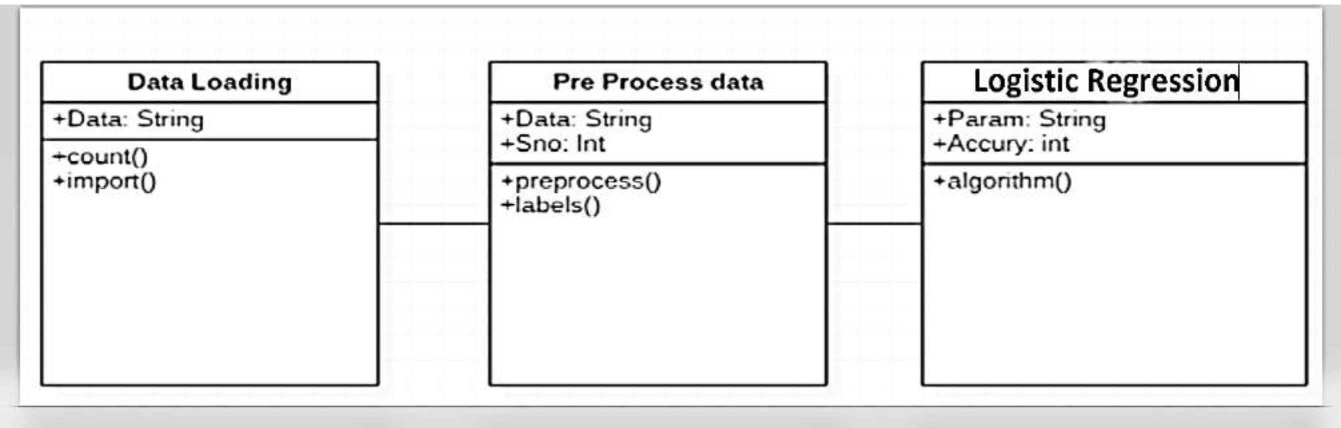


Fig.7: Class Diagram of data processing

Deployment Diagram

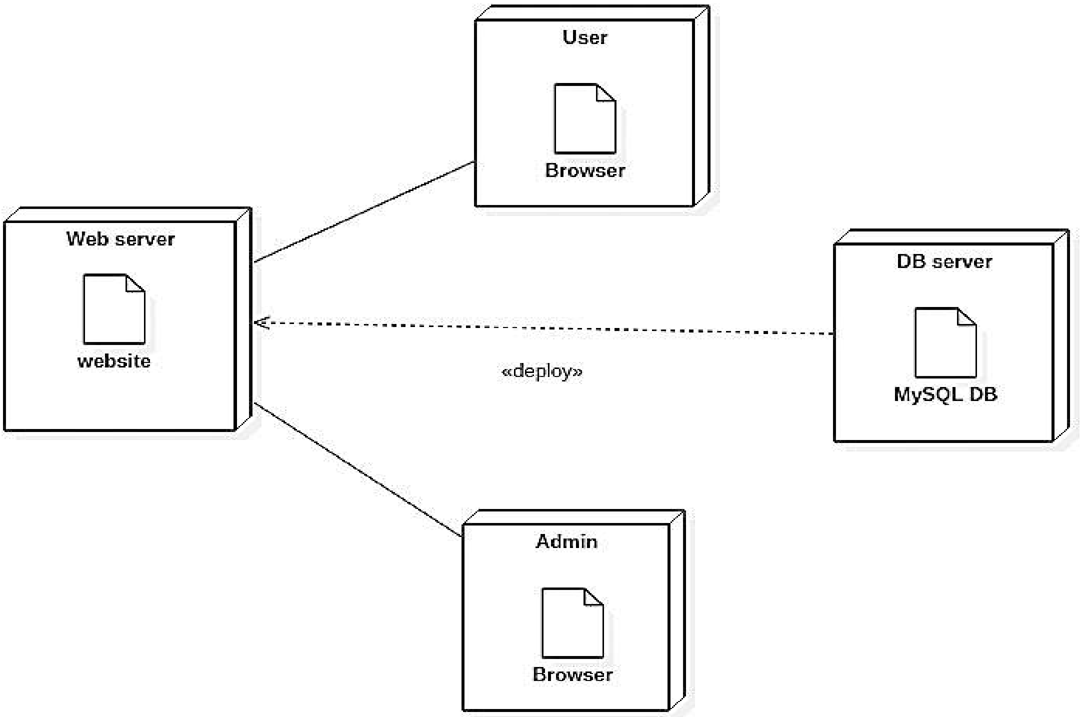


Fig. 8: Deployment Diagram for data storing

4. Proposed System Analysis and Design

4.1. Introduction

In this proposed system uses the Logistic regression to precisely order and distinguishes the offensive and the defensive sentence with high exactness or precision of 91.75%. The proposed framework can recognize the possible user by strategies for which the offensive language is used. The system is directed with the principal near investigation of different learning models on Hate and Abusive Speech on a social media website and examine the likelihood of utilizing additional features and context information for overhauls. This task applies machine learning techniques to perform computerized hostile language identification. Negative language can be characterized as communicating preposterous subjectivity and this investigation generally centers around two classes 'sexual' and 'bigot'.

4.2. Requirement Analysis

The requirements for the following system to work is given below:

4.2.1. Functional Requirements

4.2.1.1. Product Perspective

Data collection: The data collection is the process of having different kinds of information to gather for the project to use in the system. So the data for this can be collected from different online places where the use of comments posts or communication is done. So for the time being the data for the following system is collected from the sources like YouTube, Facebook, and twitter. The amount of aggressive words in these platforms are too much so it will be good and for the further process of the given method.

Data pre-processing: The data pre-process instantly the directory part is given. "As the online social framework data is profoundly unstructured, there is a need to pre-process the data before it related with the genuine characterization model. Notwithstanding, to pre-process the data the stop words it contains can be cleared and the slang words are planned to their one-of-a-kind construction".

The mentioned steps are the sequence of text pre-processing used concerning in the dataset to improve version of the post to be checked.

- **Tokenization:** Tokenization is known as splitting sentences hooked on particular parts that are singular words. That is chief advance also it will provide the info of visions with the sense in the content through the mining of terms present within. This can provide us with the all-out word total or separate word occurrence.

- **Stemming:** Changed types of a term is molded by increasing in the affixes at improper word. It will help us in converting the changed connected techniques for a comment to its base structure by detecting the addition from a word. It isn't needed in the base term shaped with a little of literal sense.

- **Lemmatization**: Lemmatization converts the changed type in the term in its real form or structure by eliminating the ends. Be that as it may, this root structure made is the word data type of a term named Lemma. This profits for account the sense of expression present in the line which is to change the term hooked on its original type. The system takes the assistance of a jargon and does different morphological examination of terms.

- **Lower Case Conversion**: The technique is for the most part changing every one of the words in the content to lowercase structure. It is done to standardize the written data.

- **Stop Words Removal**: Whenever in the sentence the system has to make essential styles from the content, then the stop words with no sense. They are typically and frequently happening words from the content that are basically the articles, this is done to find the calculation time by dropping the scope of the words in use.

- **Punctuation Removal:** These are signs that are used for inscription for explaining about the sense in the content by extrication of the whole line and the components. The punctuations are basically of no use to bring the sense of text then only utilized in writing the great clarity to eliminate them out of the dataset.

• **User and URL Mentions Removal**: After having the printed data brimming with USER and URL references in the file it doesn't transfer any sense with the file, so it will be withdrawn in the presence of preparation as well as from the test information.

Feature Extraction: Likely sequence work of best part in the given file will be banished so much, "that it can increase the overall execution. In feature extraction, considering some counter amount, the element can be separated. To clear the component, the data mining strategies like tokenization, term recurrence, and Inverse term recurrence can discover alongside the regression system".

4.2.1.2. Product features

- Give customers a pre-arranged to-use, expressive visual exhibiting language so they can make and exchange significant models.
- If someone posts an offensive word, then the website will know that it is an official word. Then after that, the admin does not allow to post

4.2.1.3. User characteristics

Register: A signup page (also known as a registration page) enables users and organizations to register and gain access to the system independently. It is common to have multiple signup pages depending on the types of people and organizations anybody wants to register.

Login: A login page is a web page or an entry page to a website that requires user identification and authentication, regularly performed by entering a username and password combination. Logins may provide access to an entire site or part of a website. Logging in not only provides site access for the user but also allows the website to track user actions and behavior.

Logout: To end access to a computer system or a website. Logging out informs the computer or website that the current user wishes to end the login session. Log out is also known as log-off, sign off or sign out.

4.2.1.4. Assumption & Dependencies

- The project is a developed a social blog to demonstrate this entire process and it shows promising results. The system tests whether the post contains offensive content or not at the time of posting itself.

- The structure ought to be accessible consistently, which means that the user will get to it using a web browser, just confined with the stoppage in the main part which is responsible or the system to work.

4.2.1.5. User Requirements

Register: A signup page (also known as a registration page) enables users and organizations to register and gain access to the system independently. It is common to have multiple signup pages depending on the types of people and organizations anyone wants to register.

Users' login: A login page is a web page or an entry page to a website that requires user identification and authentication, regularly performed by entering a username and password combination. Logins may provide access to an entire site or part of a website. Logging in not only provides site access for the user but also allows the website to track user actions and behavior.

Create Post: First of all, the post has to be created, after the post system will find out whether it is an official word or not.

Post Content: If someone posts an offensive word, then the website will know that it is an official word. Then after that, the admin does not allow to post.

Logout: To end access to a computer system or a website. Logging out informs the computer or website that the current user wishes to complete the login session. Log out is also known as log-off, sign off or sign out.

4.2.2. Non Functional Requirements

4.2.2.1. Product Requirements

This project requires designing mind and straightforward collaboration like python ide, windows Operating System, Hard Disk: Min 160 GB, Ram: Min 4GB and PyCharm etc.

4.2.2.1.1. Efficiency (in terms of Time and Space)

When the system was evaluated, then the results received was having 95% efficiency (F1-score=0.95 &Accuracy 0.95)

4.2.2.1.2. Reliability

The capacity of the structure to act dependably in a user-satisfactory method after the operation inside the setting in that the method is planned.

4.2.2.1.3. Portability

According to the terms of the portability the system is portable enough because it needs only laptop to perform the task and it can also perform in different platforms as well as PC so the requirements are only the necessary links and codes which can be taken easily also all the frameworks are easy to access.

4.2.2.1.4. Usability

The system is not difficult to use by whoever wants to use it for finding hate speech or offensive words in their Comment or post. The constraint of a framework to give a condition to its clients to play out the errands safely, enough, and beneficially while appreciating the experience

4.2.2.2. Organizational Requirements

4.2.2.2.1. Implementation Requirements

Execution contains different advancements utilized (python ide, PyCharm, Django's, CAD application Fandango), the establishment of required programming and libraries, design graph of the undertaking, engineering charts of different models, calculation of the superb model utilized, and test coding of the venture will be executed on the compiler in windows. The system should be available at all times, meaning the user can access it using a web browser, only restricted by the downtime of the server on which the system runs. Crude information assortment and pre-processing will happen first, at that point, Feature creation and mark age, and subsequently system will carry out the ML Algorithm to track down the negative words.

4.2.2.2.2. Engineering Standard Requirements

Secure access of confidential data (user information). It essentially suggests securing a website or any app is distinguishing, prevention, with the reaction of computerized risks. This crucial partition from IT Safety is fundamental to the assurance of locales, web applications, and web administrations. The system ought to be efficient that it won't get hang if substantial traffic. The proportion of how well a site does what it should prepare. Supposing to be that the utility and suitability objective is fulfilled, efficiency is the following convenience objective to contemplate. Proficiency within the use of gadgets transported with the website is correspondingly pretty much which is as important as the apparatuses' presence. Data in the database of the system ought not lose or harm. The data on amplifying of the user personal prosperity with the safety chances with the presence of their information which are not be shared and using the internet also for the security of someone doing anything wrong in the technical ground all things considered.

4.2.2.3. Operational Requirements

- **Economic:** In the part of the economic as an operational requirement it is the part which describes about the total cost of making the project as well as how a person will be able to use this in a manner of the price so we did the project to define the justification of the economic management and also this project is completely new with the user to provide the finder of offensive words in the post so it only needs minimum amount of things to run through and this is not at all expensive as it only requires basic thing to run through and this is very cost effective to use it makes good run for the user it requires less time to work on and it is fast system to work with, the build quality of this website as well as the codes are good to work within the system. There is no negligence in the making of this program as well as the website for the user to use it without any hindrance.

- **Social:** The project is mainly based for the welfare of the social health of the clients as people nowadays don't think about anybody's mental health before using hate speech or offensive words, so people have to understand and feel the people's psychological and physical health so, with the help of Machine learning. It was decided to check for offensive words.

- **Political:** There are a lot of people waiting for their leaders to comment something and to deny the fact as they are not being benefitted with that plan so they start using offensive words and start passing comments which are harmful to the society as everyone starts to do it and unfollow the rules so this project will help in a political point of view too.

- **Ethical:** This tool can be used to disengage hate from people's perception, make sure no bullying is done in society, hence saving many current generation students and making sure the debate can become neutral and polite so that people can respond to things with patience and calm.

- **Legality:** As this project is related to finding hate speech or offensive words, it will help people find the person’s perception of thinking and how they react to a thing, so it will be completely legal and authentic to be used as a project.

- **Inspectability:** As this project, it is continuously inspecting through the offensive words in the post, the website's inspection and the post from the admin side will always be done to make less trouble for uses to use the project smoothly.

- **Technical feasibility:** In this project, the particular necessities of the structure are termed as the technical feasibility. "Any structure made should not have a serious interest on the available specific resources. This will provoke levels of prevalence on the available specific resources. This will provoke levels of prevalence being put on the client. The made structure should have a modest need, as unimportant or invalid changes are needed for completing this system.

4.2.3. System Requirements

4.2.3.1. H/W Requirements

- Processor: Intel i5 or more
- Hard Disk: Min 16 GB
- Ram: Min 4GB
- Mouse: Optical Mouse
- Monitor: LED/LCD
- Motherboard: Intel® Chipset Motherboard.
- Cache: 512 KB
- Speed: 2.7GHZ and more

4.2.3.2. S/W Requirements

- Windows operating System
- Python
- PyCharm IDE

5 Results and Discussion

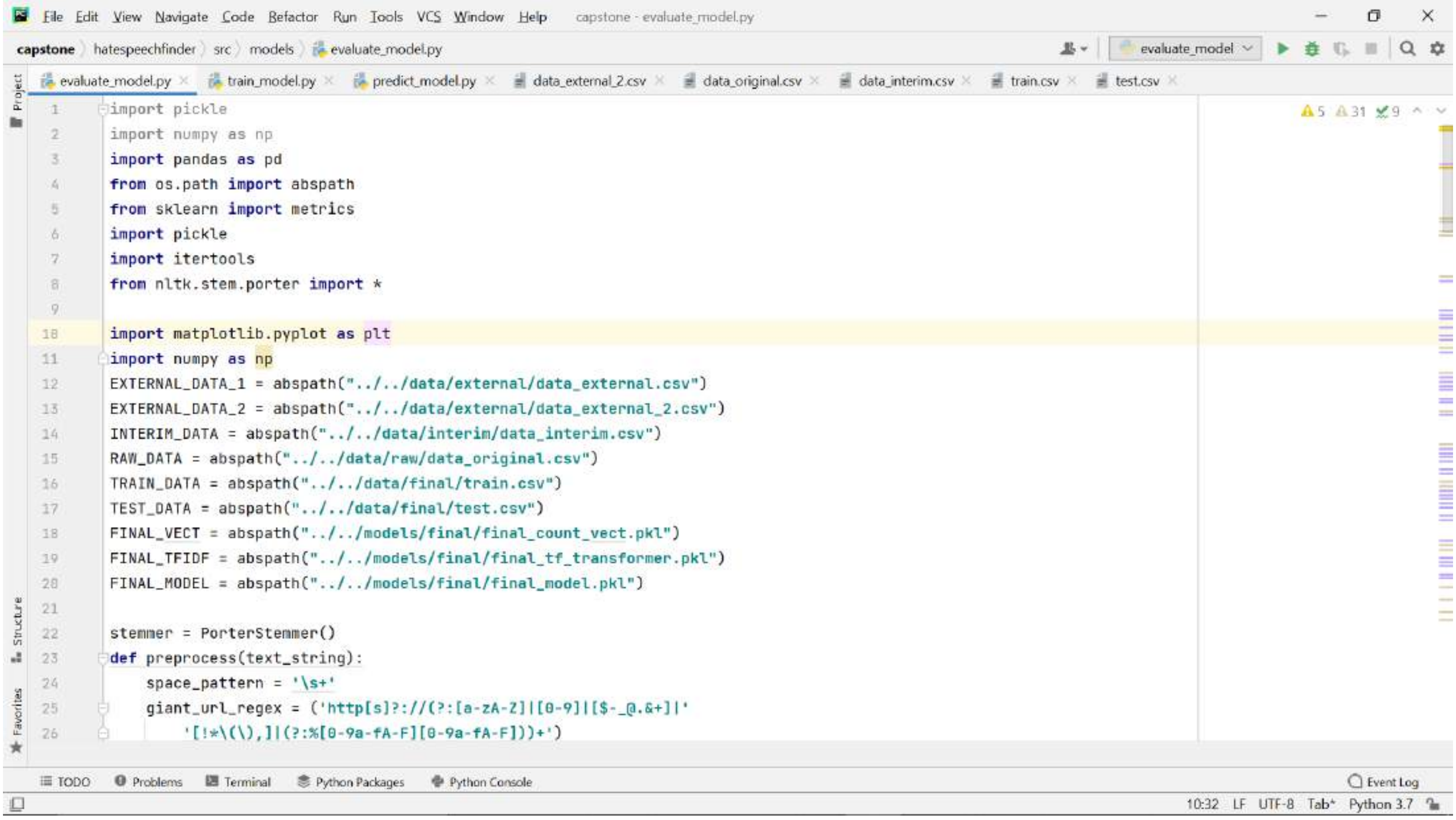
The system is a working social blog, a social network for users to come and share and interpret their views or anything of their interest; in a filtered manner, the system results in avoiding large no. It is because of conflicts within current social media, like online bullying, cutting off the discussion by saying offensive words and help their users to behave in subtle approach to other's people viewpoints which eventually will help people psyche to be calm and open to new ideas.

The user has to the first signup, then log in for posting the Comment then the system checks if this Comment is subtle or clean for conversation or not, if it chooses the comments to be offensive, it will alert the user, they cannot post abusive posts in this platform, and if the system deems it to be okay, then the user can post their Comment successfully.

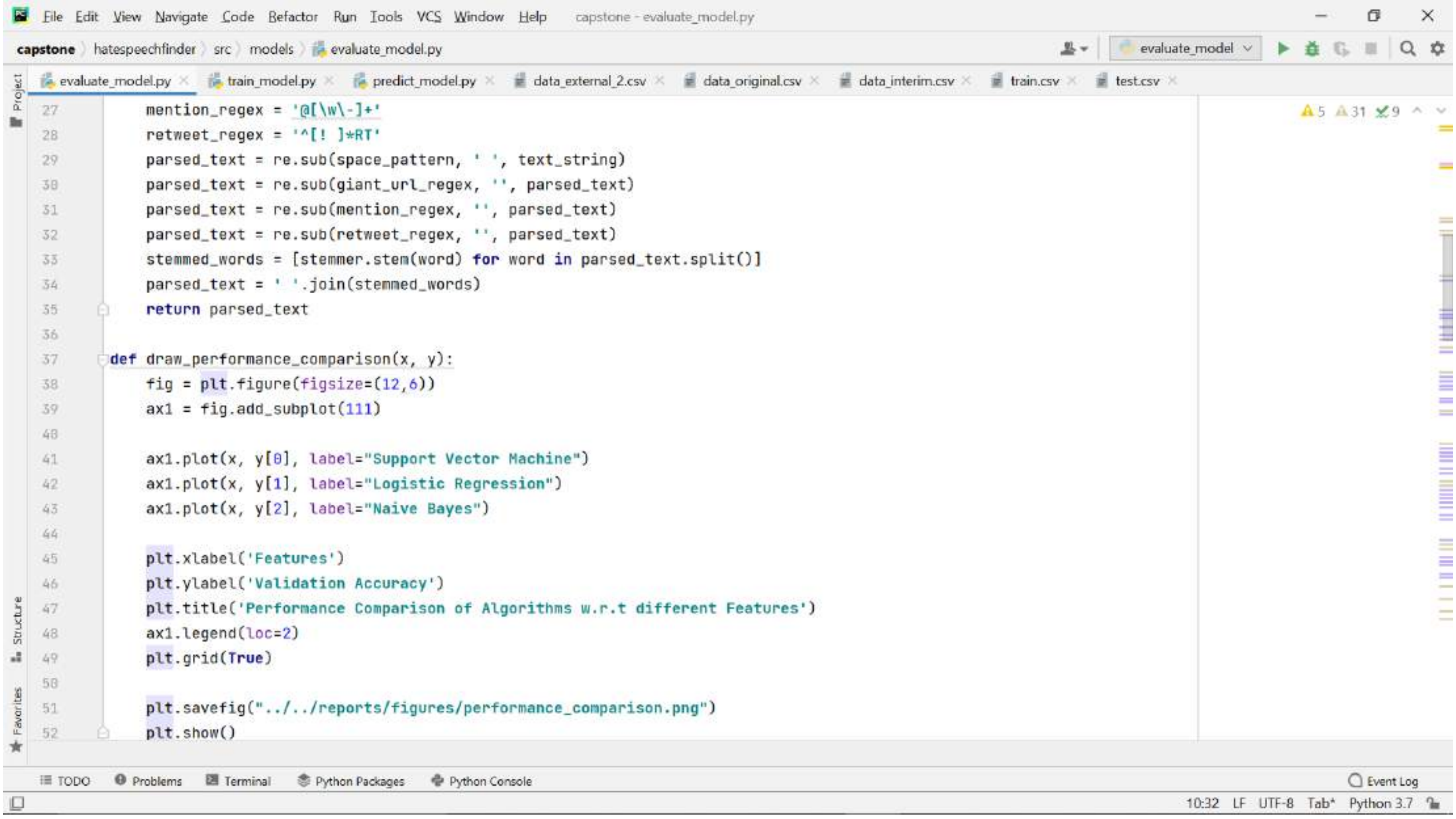
Below are the steps of implementation which will be describing the system.

Codes:

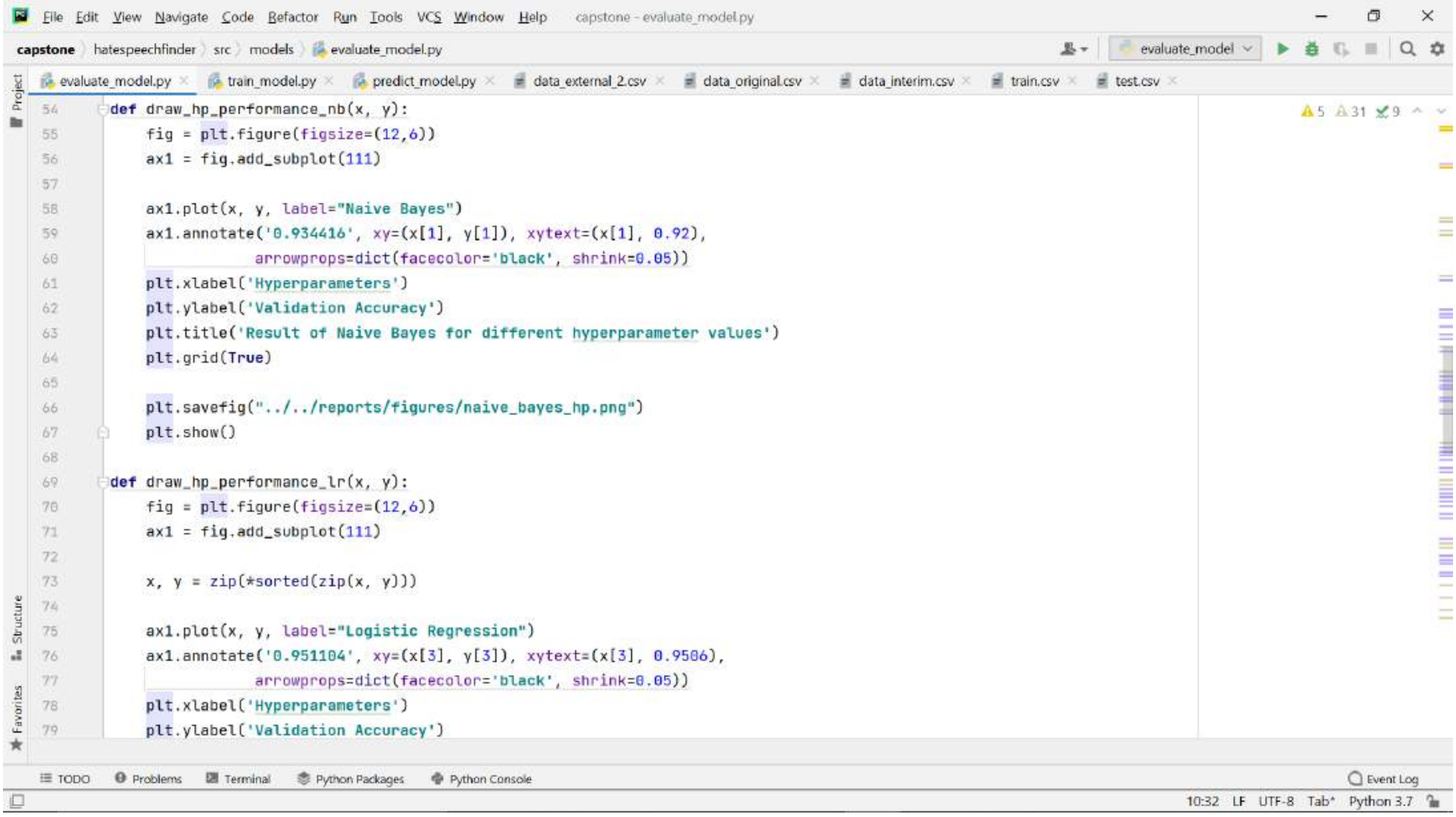
Evaluation of model



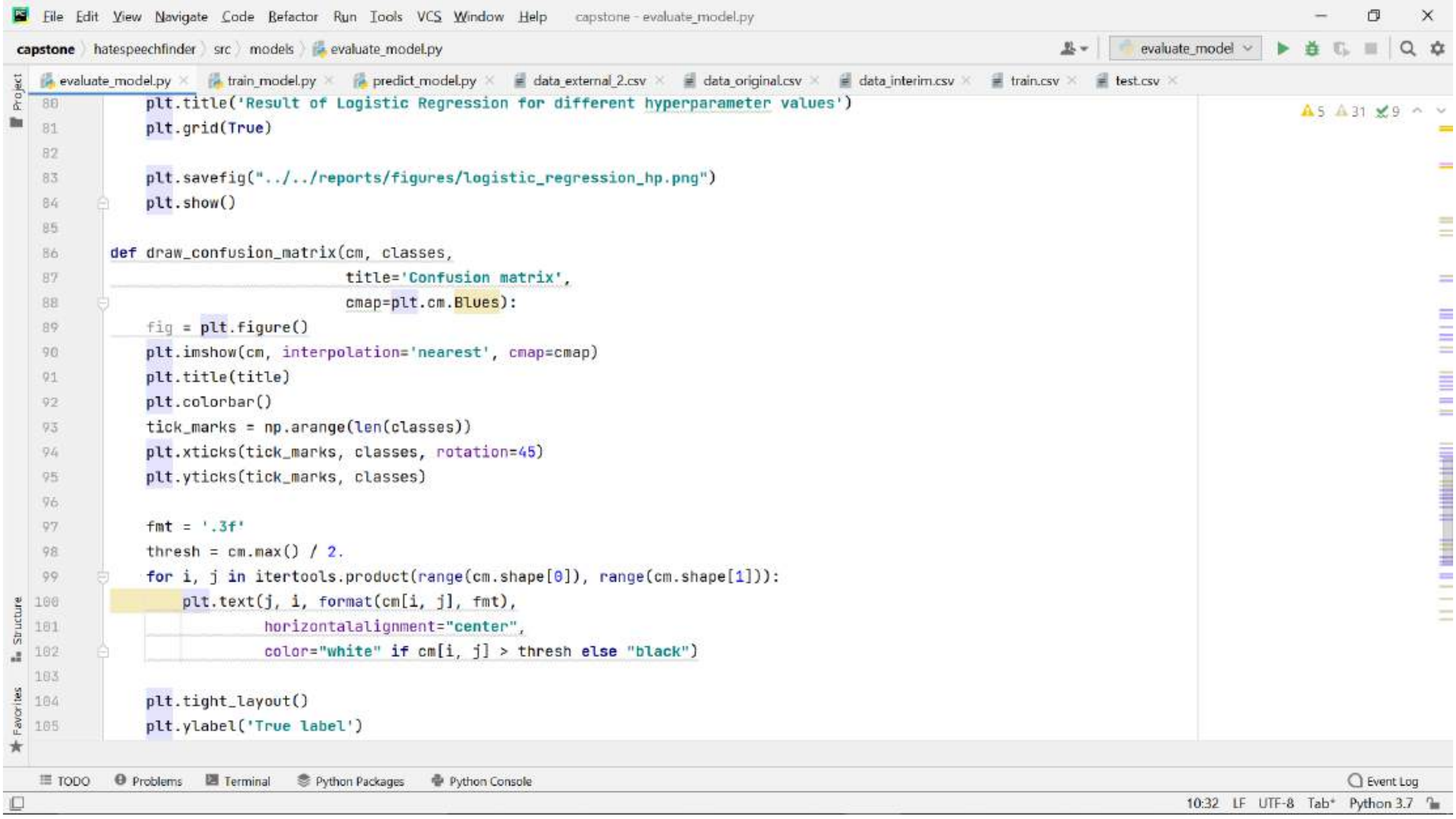
```python
import pickle
import numpy as np
import pandas as pd
from os.path import abspath
from sklearn import metrics
import pickle
import itertools
from nltk.stem.porter import *

import matplotlib.pyplot as plt
import numpy as np
EXTERNAL_DATA_1 = abspath("../../data/external/data_external.csv")
EXTERNAL_DATA_2 = abspath("../../data/external/data_external_2.csv")
INTERIM_DATA = abspath("../../data/interim/data_interim.csv")
RAW_DATA = abspath("../../data/raw/data_original.csv")
TRAIN_DATA = abspath("../../data/final/train.csv")
TEST_DATA = abspath("../../data/final/test.csv")
FINAL_VECT = abspath("../../models/final/final_count_vect.pkl")
FINAL_TFIDF = abspath("../../models/final/final_tf_transformer.pkl")
FINAL_MODEL = abspath("../../models/final/final_model.pkl")

stemmer = PorterStemmer()
def preprocess(text_string):
    space_pattern = '\s+'
    giant_url_regex = ('http[s]?://(?:[a-zA-Z]|[0-9]|[$-_@.&+]|'
        '[!*\(\),]|(?:%[0-9a-fA-F][0-9a-fA-F]))+')
```



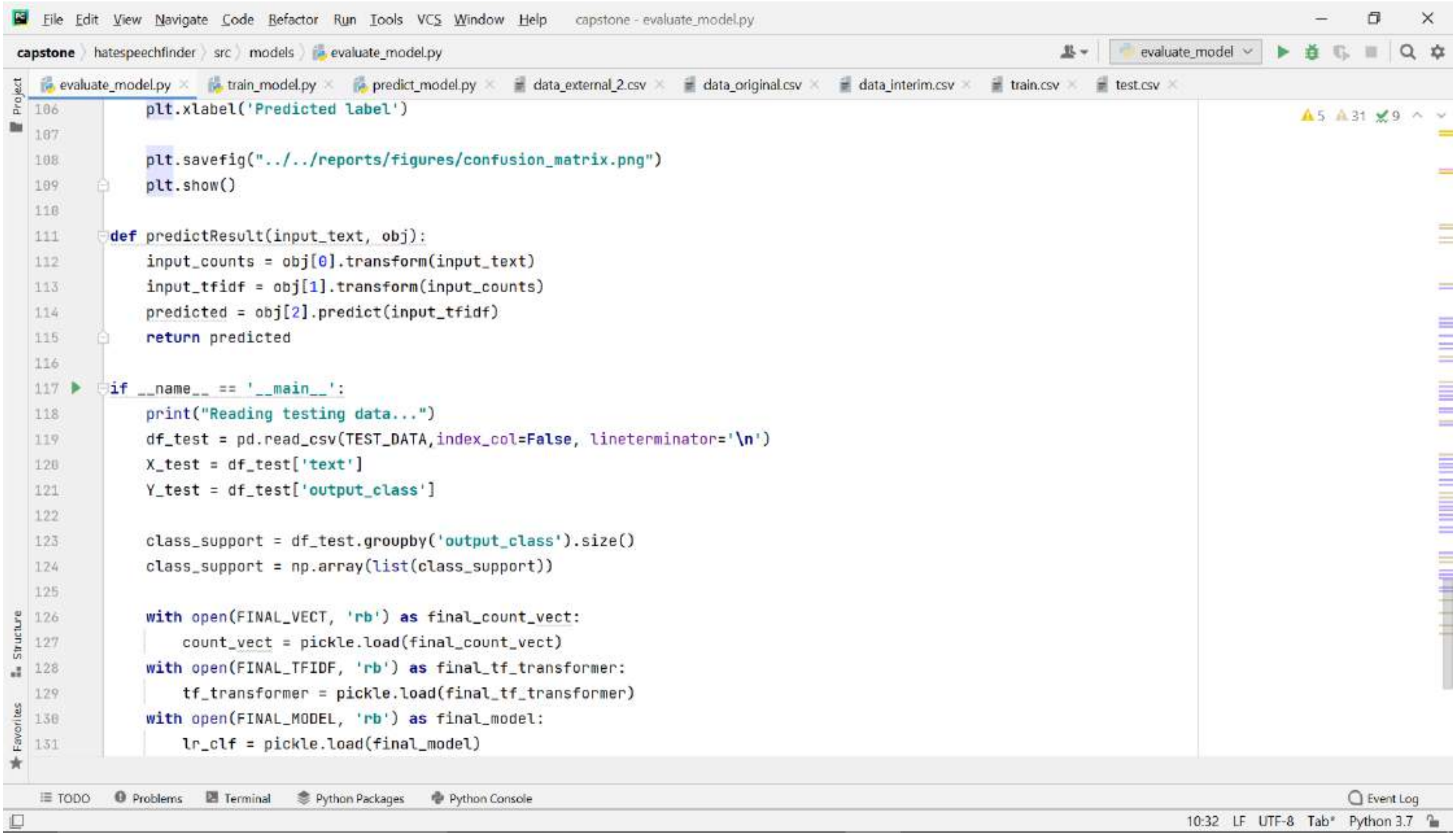
```python
    mention_regex = '@[\w\-]+'
    retweet_regex = '^[! ]*RT'
    parsed_text = re.sub(space_pattern, ' ', text_string)
    parsed_text = re.sub(giant_url_regex, '', parsed_text)
    parsed_text = re.sub(mention_regex, '', parsed_text)
    parsed_text = re.sub(retweet_regex, '', parsed_text)
    stemmed_words = [stemmer.stem(word) for word in parsed_text.split()]
    parsed_text = ' '.join(stemmed_words)
    return parsed_text

def draw_performance_comparison(x, y):
    fig = plt.figure(figsize=(12,6))
    ax1 = fig.add_subplot(111)

    ax1.plot(x, y[0], label="Support Vector Machine")
    ax1.plot(x, y[1], label="Logistic Regression")
    ax1.plot(x, y[2], label="Naive Bayes")

    plt.xlabel('Features')
    plt.ylabel('Validation Accuracy')
    plt.title('Performance Comparison of Algorithms w.r.t different Features')
    ax1.legend(loc=2)
    plt.grid(True)

    plt.savefig("../../reports/figures/performance_comparison.png")
    plt.show()
```



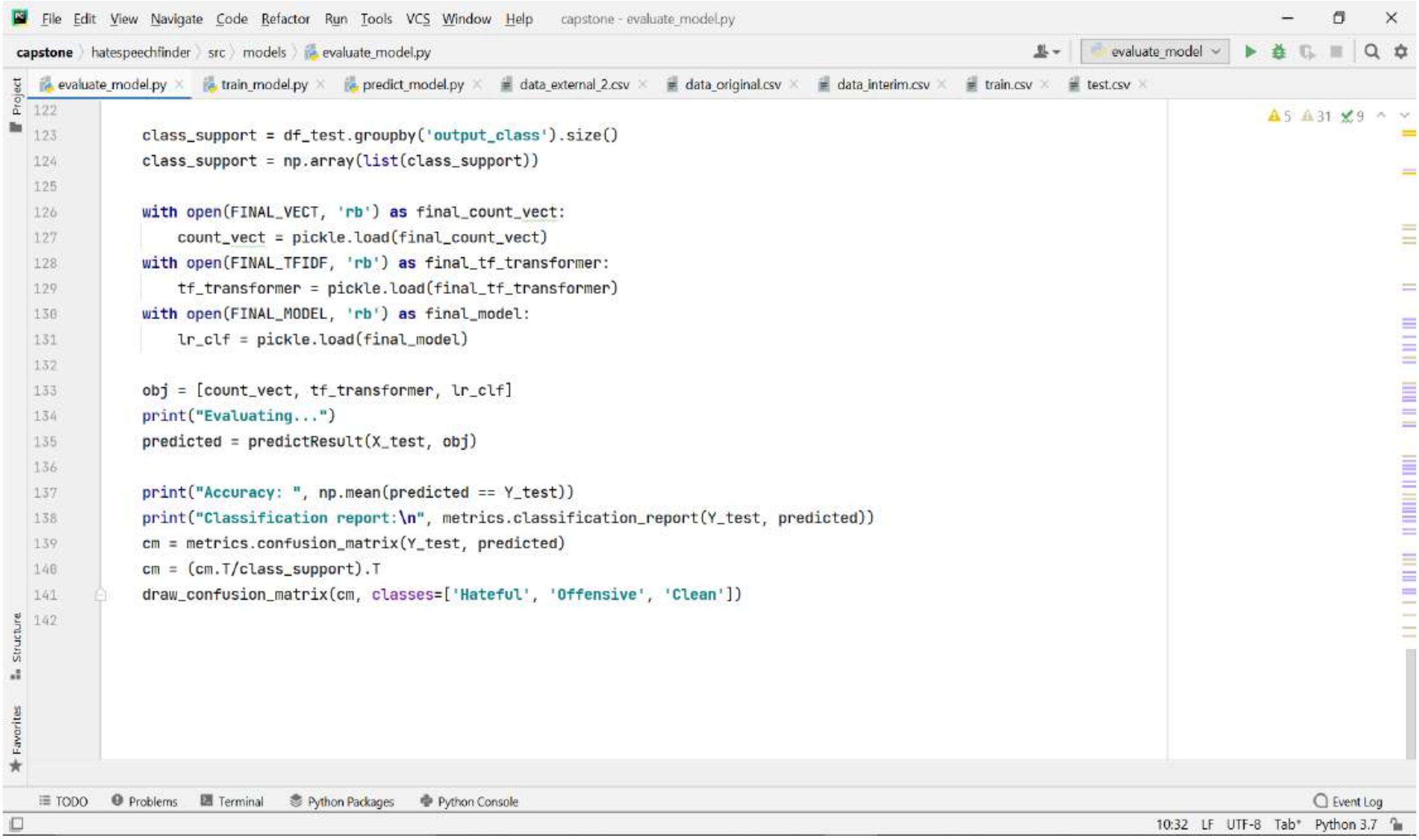
```
def draw_hp_performance_nb(x, y):
    fig = plt.figure(figsize=(12,6))
    ax1 = fig.add_subplot(111)

    ax1.plot(x, y, label="Naive Bayes")
    ax1.annotate('0.934416', xy=(x[1], y[1]), xytext=(x[1], 0.92),
                 arrowprops=dict(facecolor='black', shrink=0.05))
    plt.xlabel('Hyperparameters')
    plt.ylabel('Validation Accuracy')
    plt.title('Result of Naive Bayes for different hyperparameter values')
    plt.grid(True)

    plt.savefig("../../reports/figures/naive_bayes_hp.png")
    plt.show()

def draw_hp_performance_lr(x, y):
    fig = plt.figure(figsize=(12,6))
    ax1 = fig.add_subplot(111)

    x, y = zip(*sorted(zip(x, y)))

    ax1.plot(x, y, label="Logistic Regression")
    ax1.annotate('0.951104', xy=(x[3], y[3]), xytext=(x[3], 0.9506),
                 arrowprops=dict(facecolor='black', shrink=0.05))
    plt.xlabel('Hyperparameters')
    plt.ylabel('Validation Accuracy')
```



```
    plt.title('Result of Logistic Regression for different hyperparameter values')
    plt.grid(True)

    plt.savefig("../../reports/figures/logistic_regression_hp.png")
    plt.show()

def draw_confusion_matrix(cm, classes,
                          title='Confusion matrix',
                          cmap=plt.cm.Blues):
    fig = plt.figure()
    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.title(title)
    plt.colorbar()
    tick_marks = np.arange(len(classes))
    plt.xticks(tick_marks, classes, rotation=45)
    plt.yticks(tick_marks, classes)

    fmt = '.3f'
    thresh = cm.max() / 2.
    for i, j in itertools.product(range(cm.shape[0]), range(cm.shape[1])):
        plt.text(j, i, format(cm[i, j], fmt),
                 horizontalalignment="center",
                 color="white" if cm[i, j] > thresh else "black")

    plt.tight_layout()
    plt.ylabel('True label')
```

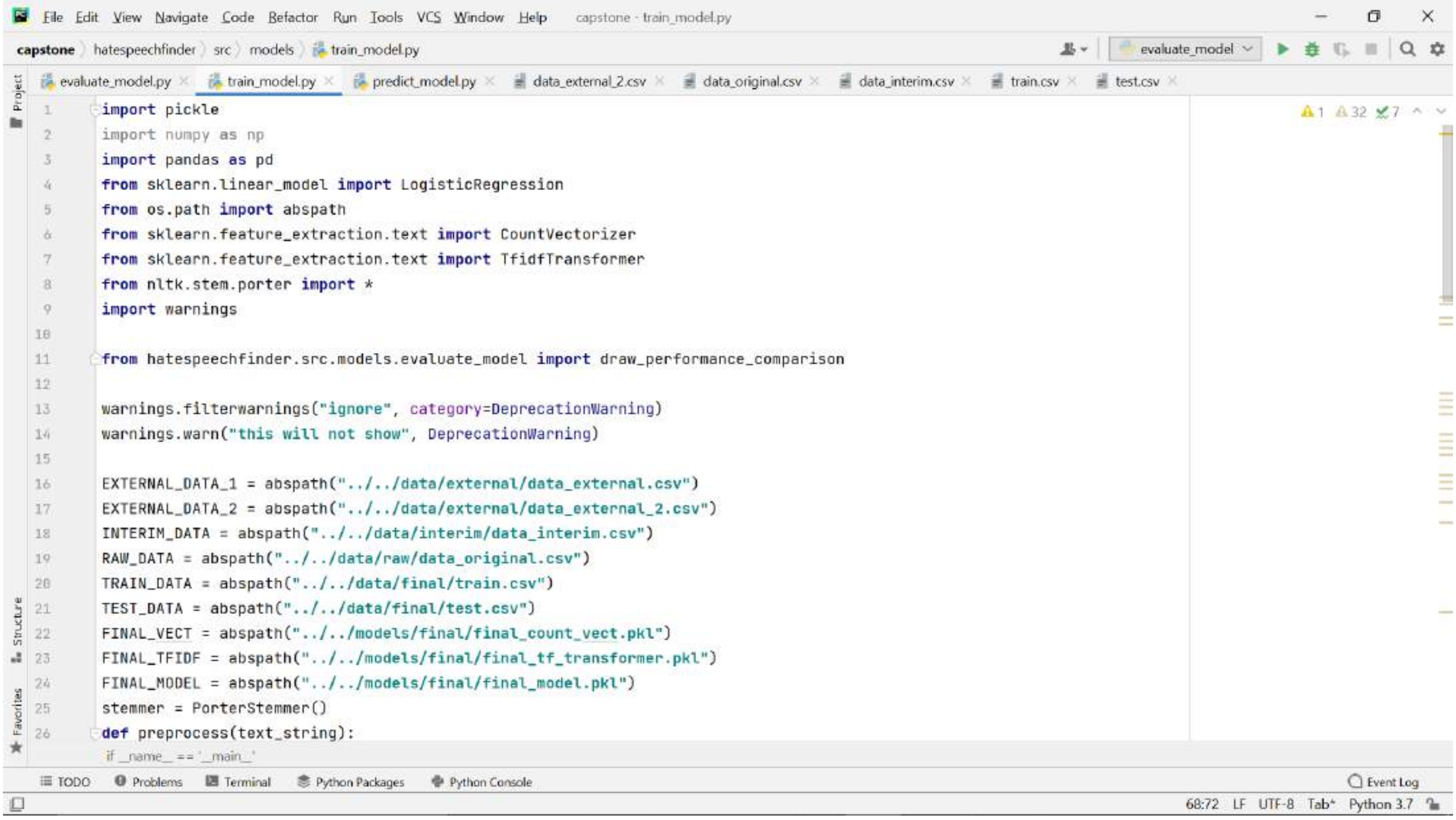


```python
    plt.xlabel('Predicted label')

    plt.savefig("../../reports/figures/confusion_matrix.png")
    plt.show()

def predictResult(input_text, obj):
    input_counts = obj[0].transform(input_text)
    input_tfidf = obj[1].transform(input_counts)
    predicted = obj[2].predict(input_tfidf)
    return predicted

if __name__ == '__main__':
    print("Reading testing data...")
    df_test = pd.read_csv(TEST_DATA,index_col=False, lineterminator='\n')
    X_test = df_test['text']
    Y_test = df_test['output_class']

    class_support = df_test.groupby('output_class').size()
    class_support = np.array(list(class_support))

    with open(FINAL_VECT, 'rb') as final_count_vect:
        count_vect = pickle.load(final_count_vect)
    with open(FINAL_TFIDF, 'rb') as final_tf_transformer:
        tf_transformer = pickle.load(final_tf_transformer)
    with open(FINAL_MODEL, 'rb') as final_model:
        lr_clf = pickle.load(final_model)
```

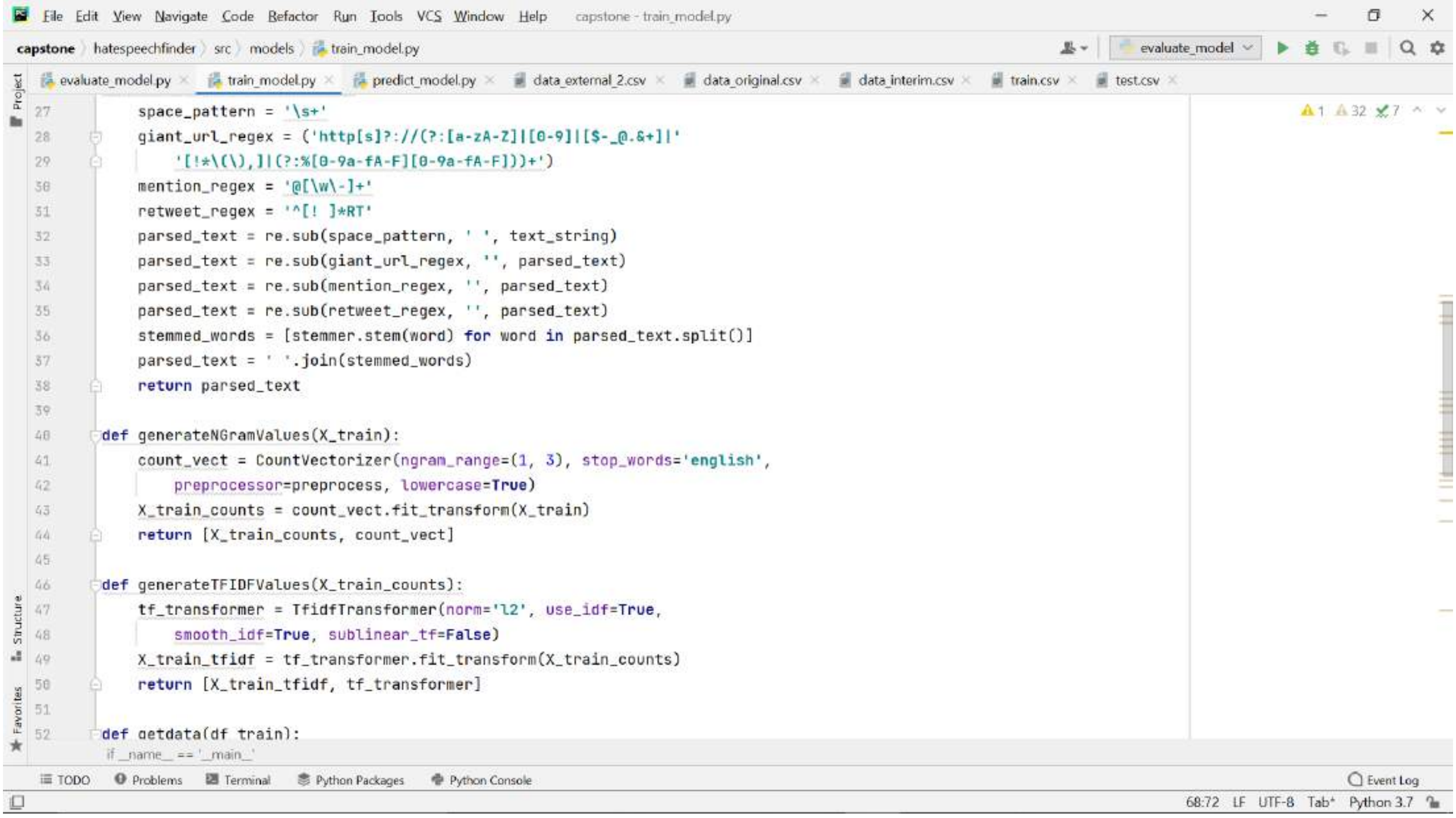


```python

    class_support = df_test.groupby('output_class').size()
    class_support = np.array(list(class_support))

    with open(FINAL_VECT, 'rb') as final_count_vect:
        count_vect = pickle.load(final_count_vect)
    with open(FINAL_TFIDF, 'rb') as final_tf_transformer:
        tf_transformer = pickle.load(final_tf_transformer)
    with open(FINAL_MODEL, 'rb') as final_model:
        lr_clf = pickle.load(final_model)

    obj = [count_vect, tf_transformer, lr_clf]
    print("Evaluating...")
    predicted = predictResult(X_test, obj)

    print("Accuracy: ", np.mean(predicted == Y_test))
    print("Classification report:\n", metrics.classification_report(Y_test, predicted))
    cm = metrics.confusion_matrix(Y_test, predicted)
    cm = (cm.T/class_support).T
    draw_confusion_matrix(cm, classes=['Hateful', 'Offensive', 'Clean'])

```

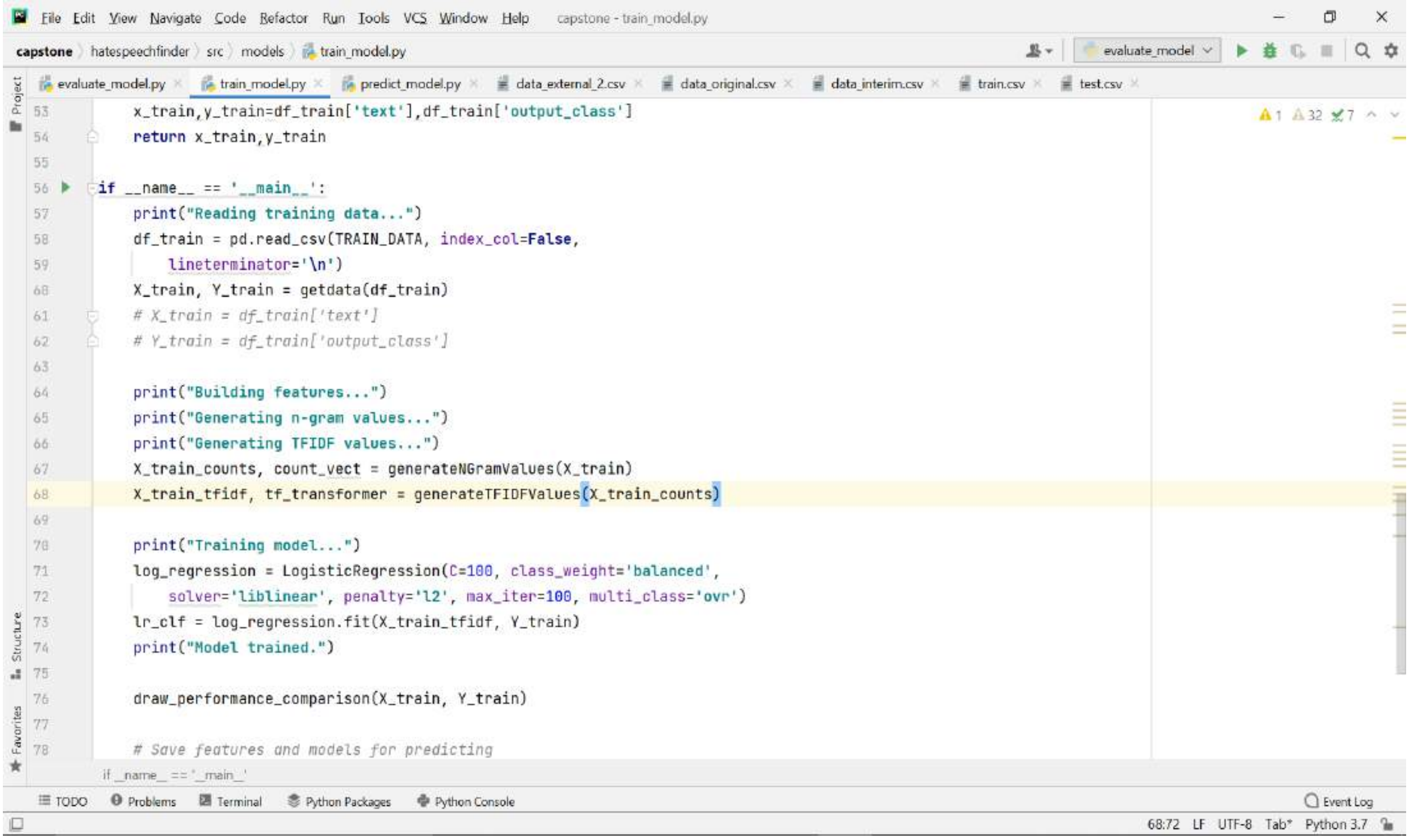
Train model.py


File Edit View Navigate Code Refactor Run Tools VCS Window Help capstone - train_model.py
capstone hatespeechfinder src models train_model.py
evaluate_model
evaluate_model.py train_model.py predict_model.py data_external_2.csv data_original.csv data_interim.csv train.csv test.csv
if __name__ == '__main__'
TODO Problems Terminal Python Packages Python Console Event Log
68:72 LF UTF-8 Tab* Python 3.7

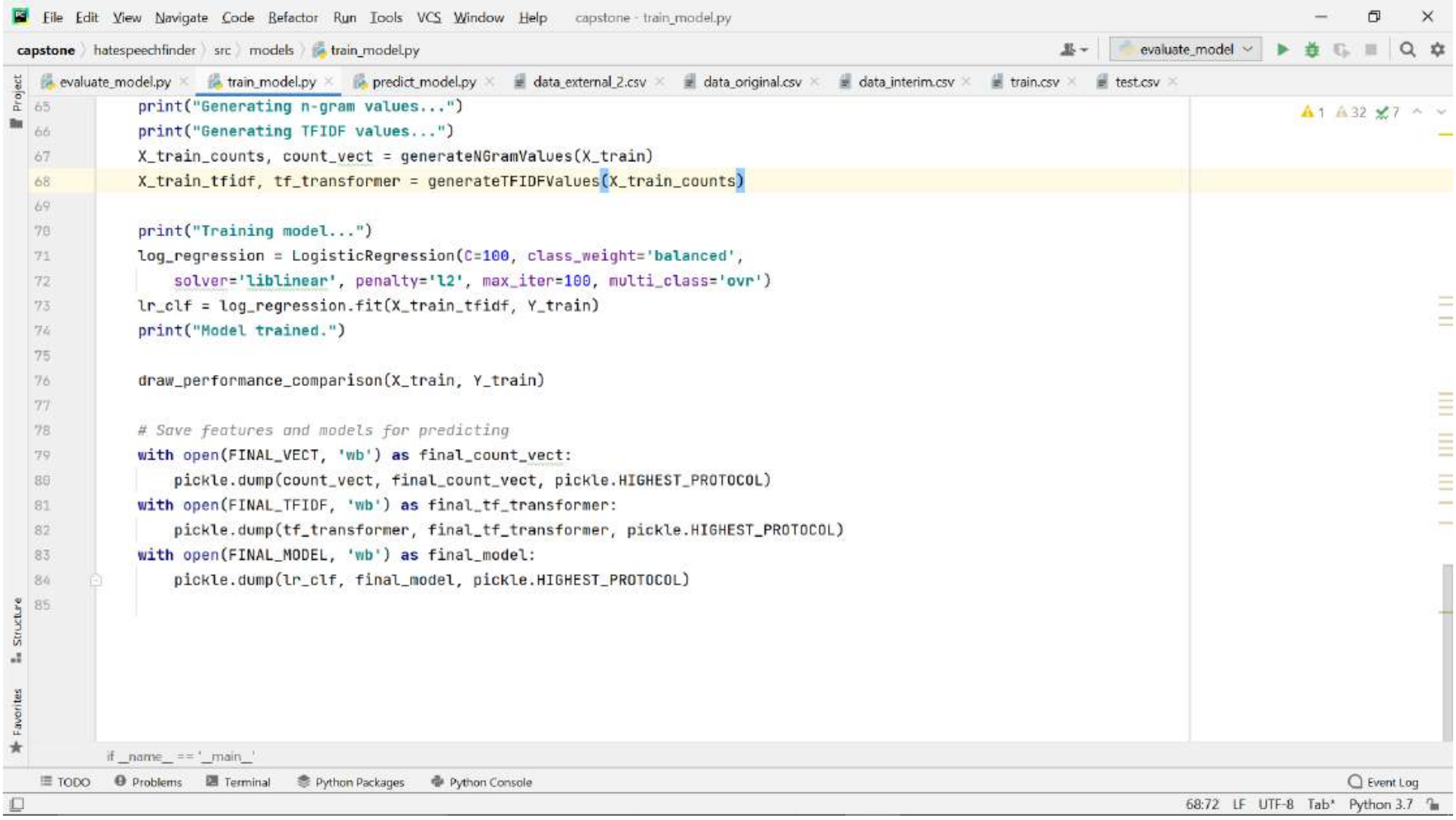
```python
import pickle
import numpy as np
import pandas as pd
from sklearn.linear_model import LogisticRegression
from os.path import abspath
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfTransformer
from nltk.stem.porter import *
import warnings

from hatespeechfinder.src.models.evaluate_model import draw_performance_comparison

warnings.filterwarnings("ignore", category=DeprecationWarning)
warnings.warn("this will not show", DeprecationWarning)

EXTERNAL_DATA_1 = abspath("../../data/external/data_external.csv")
EXTERNAL_DATA_2 = abspath("../../data/external/data_external_2.csv")
INTERIM_DATA = abspath("../../data/interim/data_interim.csv")
RAW_DATA = abspath("../../data/raw/data_original.csv")
TRAIN_DATA = abspath("../../data/final/train.csv")
TEST_DATA = abspath("../../data/final/test.csv")
FINAL_VECT = abspath("../../models/final/final_count_vect.pkl")
FINAL_TFIDF = abspath("../../models/final/final_tf_transformer.pkl")
FINAL_MODEL = abspath("../../models/final/final_model.pkl")
stemmer = PorterStemmer()
def preprocess(text_string):
```


File Edit View Navigate Code Refactor Run Tools VCS Window Help capstone - train_model.py
capstone hatespeechfinder src models train_model.py
evaluate_model
evaluate_model.py train_model.py predict_model.py data_external_2.csv data_original.csv data_interim.csv train.csv test.csv
if __name__ == '__main__'
TODO Problems Terminal Python Packages Python Console Event Log
68:72 LF UTF-8 Tab* Python 3.7

```python
    space_pattern = '\s+'
    giant_url_regex = ('http[s]?://(?:[a-zA-Z]|[0-9]|[$-_@.&+]|'
        '[!*\(\),]|(?:%[0-9a-fA-F][0-9a-fA-F]))+')
    mention_regex = '@[\w\-]+'
    retweet_regex = '^[! ]*RT'
    parsed_text = re.sub(space_pattern, ' ', text_string)
    parsed_text = re.sub(giant_url_regex, '', parsed_text)
    parsed_text = re.sub(mention_regex, '', parsed_text)
    parsed_text = re.sub(retweet_regex, '', parsed_text)
    stemmed_words = [stemmer.stem(word) for word in parsed_text.split()]
    parsed_text = ' '.join(stemmed_words)
    return parsed_text

def generateNGramValues(X_train):
    count_vect = CountVectorizer(ngram_range=(1, 3), stop_words='english',
        preprocessor=preprocess, lowercase=True)
    X_train_counts = count_vect.fit_transform(X_train)
    return [X_train_counts, count_vect]

def generateTFIDFValues(X_train_counts):
    tf_transformer = TfidfTransformer(norm='l2', use_idf=True,
        smooth_idf=True, sublinear_tf=False)
    X_train_tfidf = tf_transformer.fit_transform(X_train_counts)
    return [X_train_tfidf, tf_transformer]

def getdata(df_train):
```

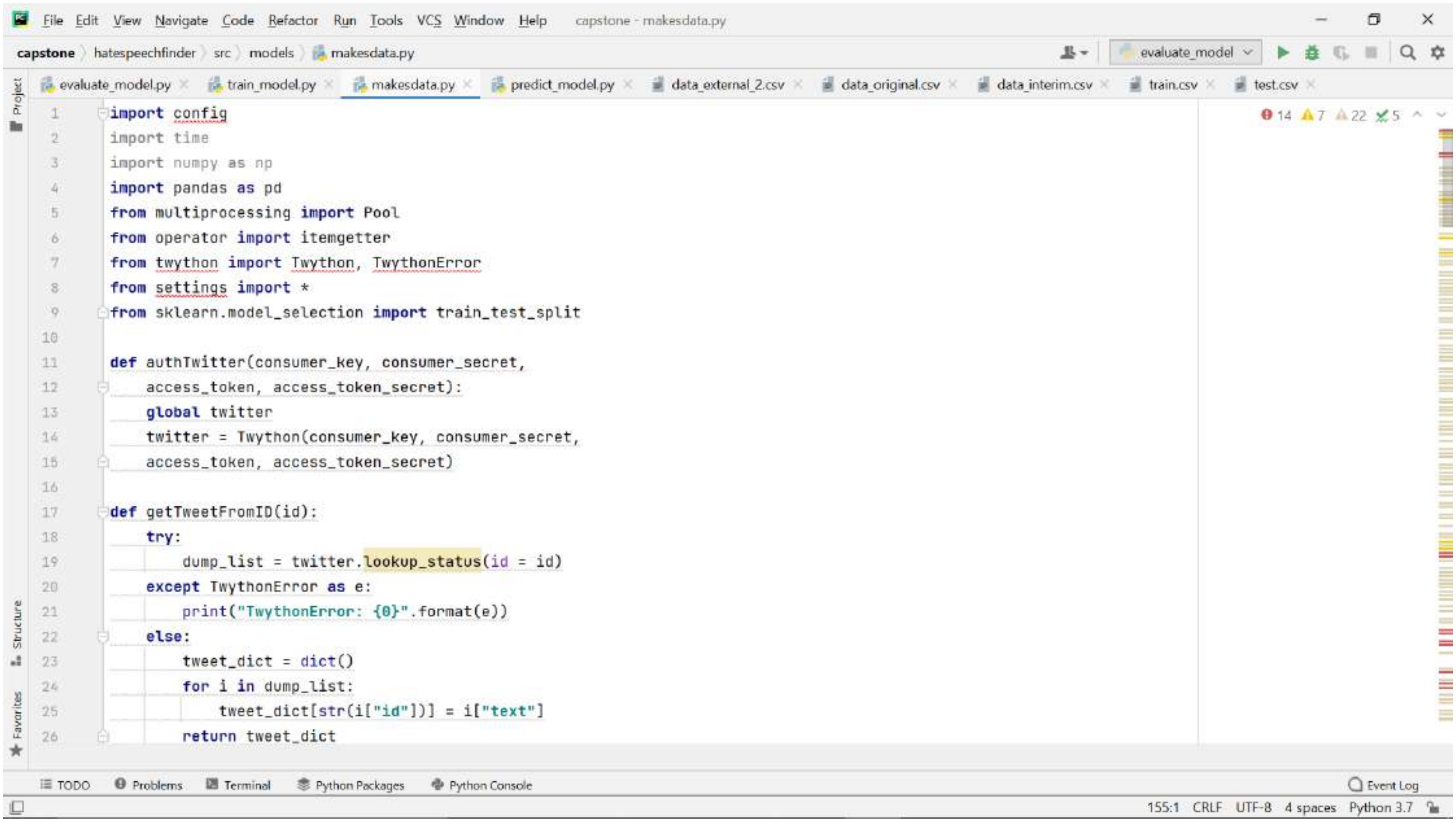

File Edit View Navigate Code Refactor Run Tools VCS Window Help capstone - train_model.py
capstone 〉 hatespeechfinder 〉 src 〉 models 〉 train_model.py
evaluate_model
evaluate_model.py
train_model.py
predict_model.py
data_external_2.csv
data_original.csv
data_interim.csv
train.csv
test.csv
if __name__ == '__main__'
TODO
Problems
Terminal
Python Packages
Python Console
Event Log
68:72 LF UTF-8 Tab* Python 3.7

```
    x_train,y_train=df_train['text'],df_train['output_class']
    return x_train,y_train

if __name__ == '__main__':
    print("Reading training data...")
    df_train = pd.read_csv(TRAIN_DATA, index_col=False,
        lineterminator='\n')
    X_train, Y_train = getdata(df_train)
    # X_train = df_train['text']
    # Y_train = df_train['output_class']

    print("Building features...")
    print("Generating n-gram values...")
    print("Generating TFIDF values...")
    X_train_counts, count_vect = generateNGramValues(X_train)
    X_train_tfidf, tf_transformer = generateTFIDFValues(X_train_counts)

    print("Training model...")
    log_regression = LogisticRegression(C=100, class_weight='balanced',
        solver='liblinear', penalty='l2', max_iter=100, multi_class='ovr')
    lr_clf = log_regression.fit(X_train_tfidf, Y_train)
    print("Model trained.")

    draw_performance_comparison(X_train, Y_train)

    # Save features and models for predicting
```

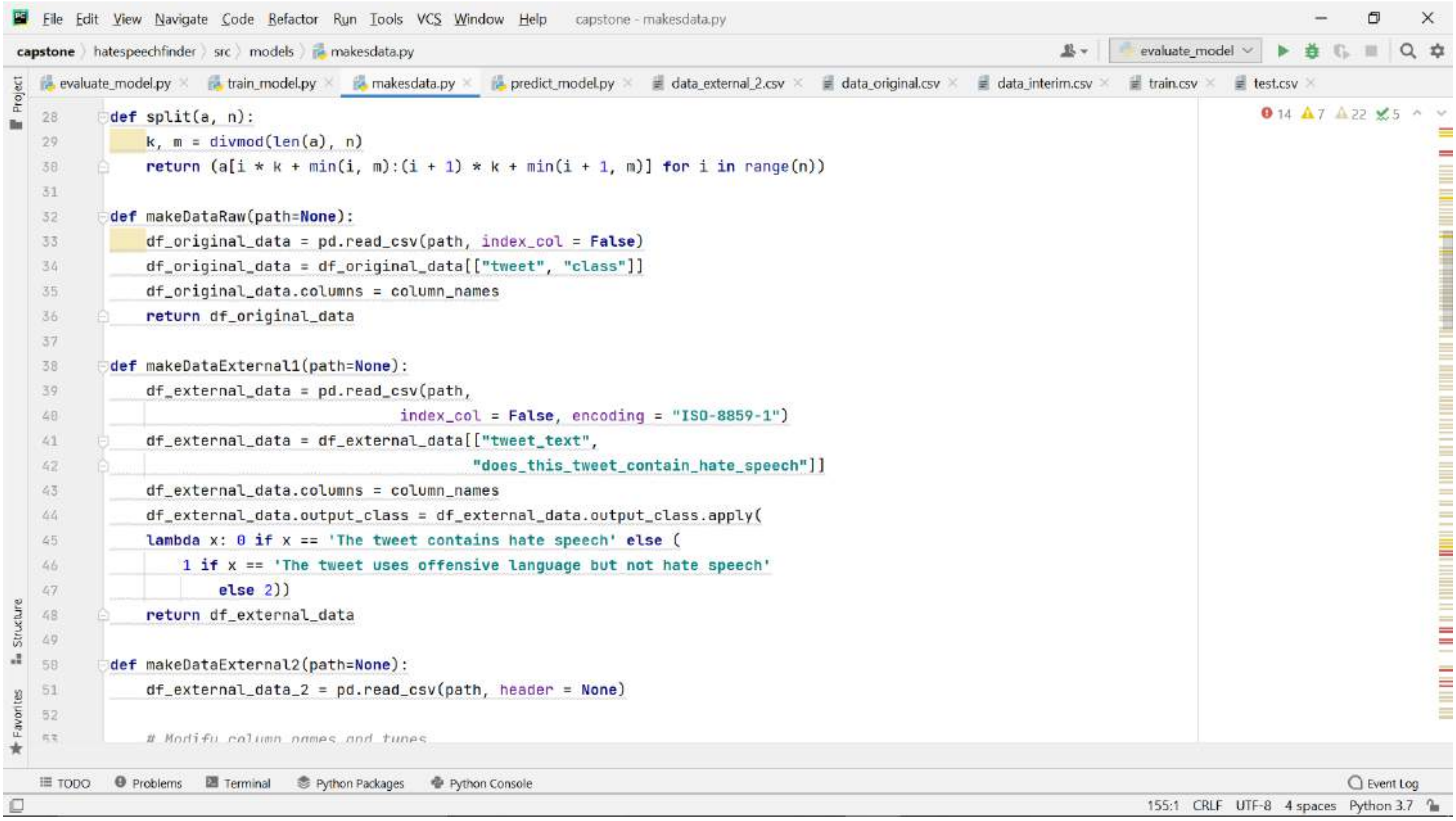

File Edit View Navigate Code Refactor Run Tools VCS Window Help capstone - train_model.py
capstone 〉 hatespeechfinder 〉 src 〉 models 〉 train_model.py
evaluate_model
evaluate_model.py
train_model.py
predict_model.py
data_external_2.csv
data_original.csv
data_interim.csv
train.csv
test.csv
if __name__ == '__main__'
TODO
Problems
Terminal
Python Packages
Python Console
Event Log
68:72 LF UTF-8 Tab* Python 3.7

```
    print("Generating n-gram values...")
    print("Generating TFIDF values...")
    X_train_counts, count_vect = generateNGramValues(X_train)
    X_train_tfidf, tf_transformer = generateTFIDFValues(X_train_counts)

    print("Training model...")
    log_regression = LogisticRegression(C=100, class_weight='balanced',
        solver='liblinear', penalty='l2', max_iter=100, multi_class='ovr')
    lr_clf = log_regression.fit(X_train_tfidf, Y_train)
    print("Model trained.")

    draw_performance_comparison(X_train, Y_train)

    # Save features and models for predicting
    with open(FINAL_VECT, 'wb') as final_count_vect:
        pickle.dump(count_vect, final_count_vect, pickle.HIGHEST_PROTOCOL)
    with open(FINAL_TFIDF, 'wb') as final_tf_transformer:
        pickle.dump(tf_transformer, final_tf_transformer, pickle.HIGHEST_PROTOCOL)
    with open(FINAL_MODEL, 'wb') as final_model:
        pickle.dump(lr_clf, final_model, pickle.HIGHEST_PROTOCOL)
```

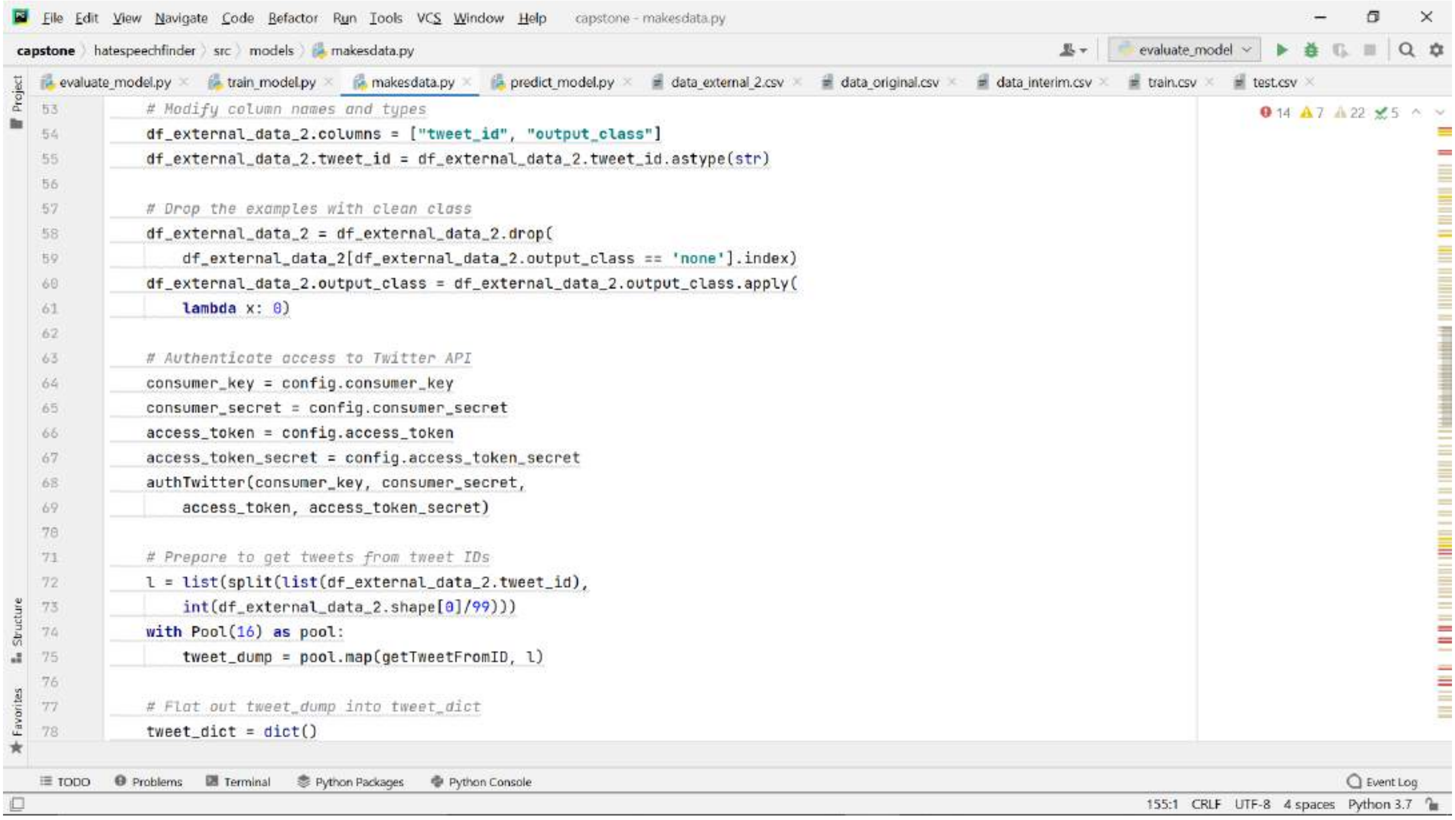
Makesdata.py


File Edit View Navigate Code Refactor Run Tools VCS Window Help capstone - makesdata.py
capstone hatespeechfinder src models makesdata.py
evaluate_model
evaluate_model.py train_model.py makesdata.py predict_model.py data_external_2.csv data_original.csv data_interim.csv train.csv test.csv
14 7 22 5
TODO Problems Terminal Python Packages Python Console Event Log
155:1 CRLF UTF-8 4 spaces Python 3.7
Project Structure Favorites

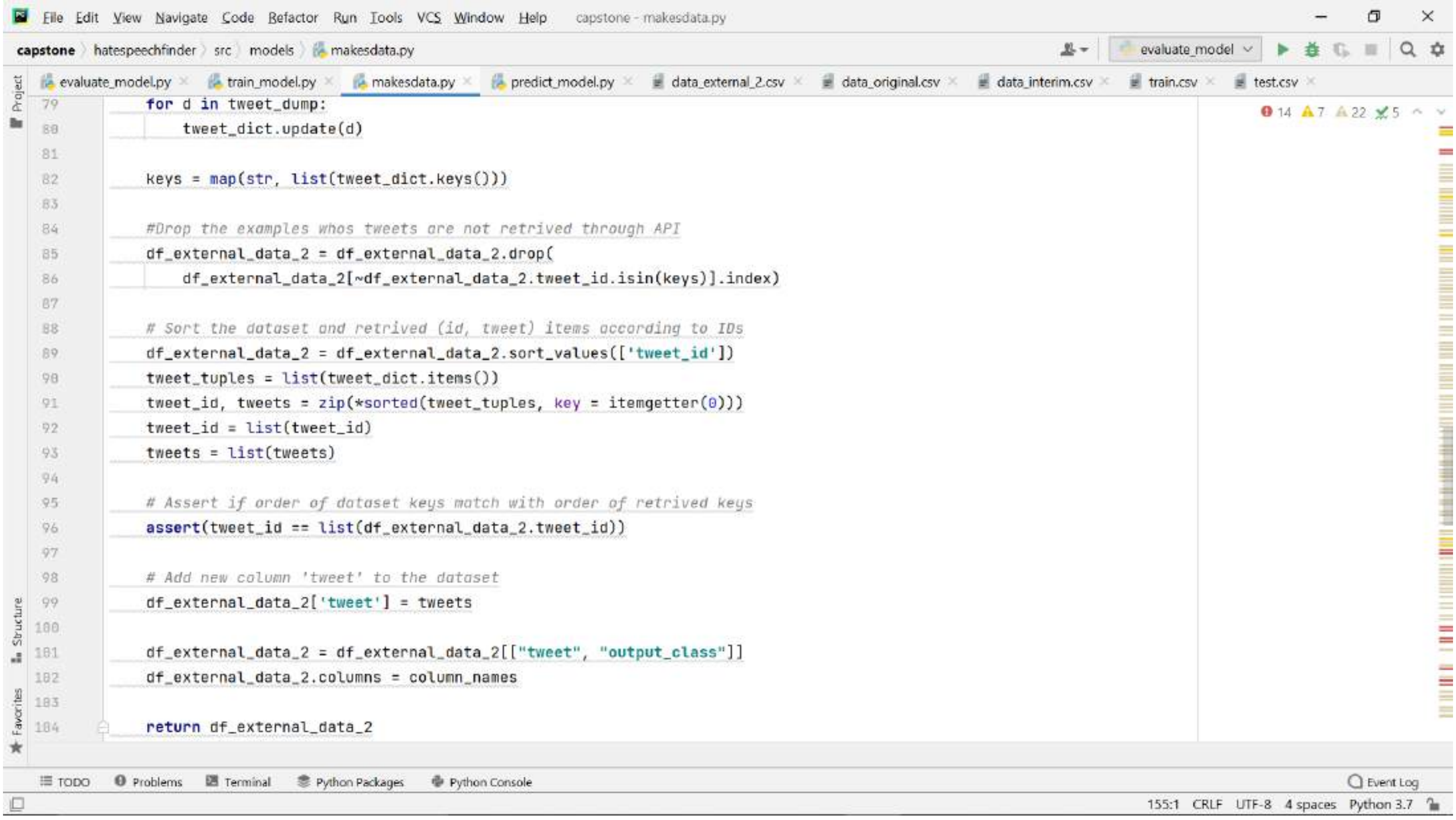
```python
import config
import time
import numpy as np
import pandas as pd
from multiprocessing import Pool
from operator import itemgetter
from twython import Twython, TwythonError
from settings import *
from sklearn.model_selection import train_test_split

def authTwitter(consumer_key, consumer_secret,
    access_token, access_token_secret):
    global twitter
    twitter = Twython(consumer_key, consumer_secret,
    access_token, access_token_secret)

def getTweetFromID(id):
    try:
        dump_list = twitter.lookup_status(id = id)
    except TwythonError as e:
        print("TwythonError: {0}".format(e))
    else:
        tweet_dict = dict()
        for i in dump_list:
            tweet_dict[str(i["id"])] = i["text"]
        return tweet_dict
```


File Edit View Navigate Code Refactor Run Tools VCS Window Help capstone - makesdata.py
capstone hatespeechfinder src models makesdata.py
evaluate_model
evaluate_model.py train_model.py makesdata.py predict_model.py data_external_2.csv data_original.csv data_interim.csv train.csv test.csv
14 7 22 5
TODO Problems Terminal Python Packages Python Console Event Log
155:1 CRLF UTF-8 4 spaces Python 3.7
Project Structure Favorites

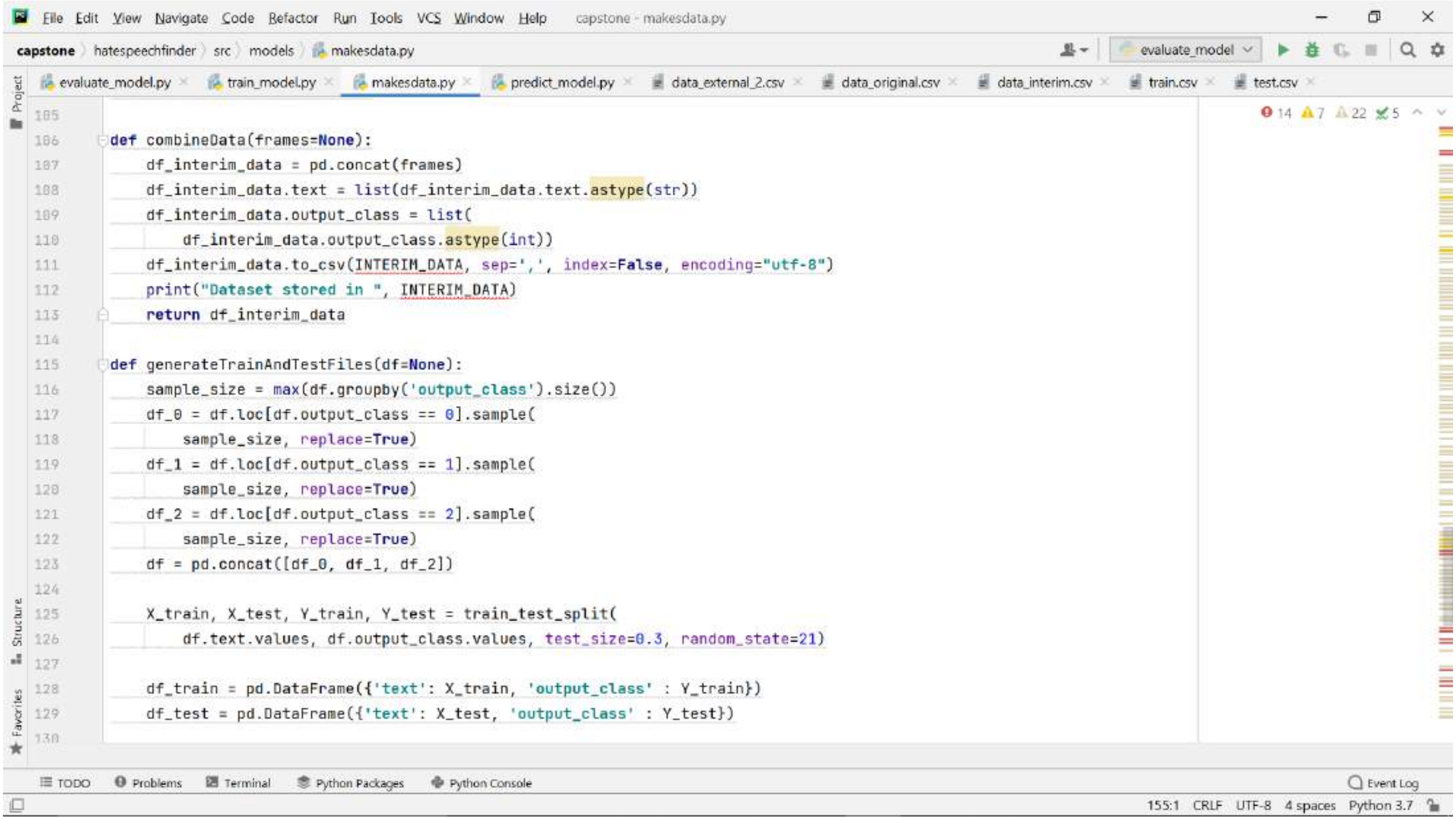
```python
def split(a, n):
    k, m = divmod(len(a), n)
    return (a[i * k + min(i, m):(i + 1) * k + min(i + 1, m)] for i in range(n))

def makeDataRaw(path=None):
    df_original_data = pd.read_csv(path, index_col = False)
    df_original_data = df_original_data[["tweet", "class"]]
    df_original_data.columns = column_names
    return df_original_data

def makeDataExternal1(path=None):
    df_external_data = pd.read_csv(path,
                                   index_col = False, encoding = "ISO-8859-1")
    df_external_data = df_external_data[["tweet_text",
                                         "does_this_tweet_contain_hate_speech"]]
    df_external_data.columns = column_names
    df_external_data.output_class = df_external_data.output_class.apply(
    lambda x: 0 if x == 'The tweet contains hate speech' else (
        1 if x == 'The tweet uses offensive language but not hate speech'
            else 2))
    return df_external_data

def makeDataExternal2(path=None):
    df_external_data_2 = pd.read_csv(path, header = None)

    # Modify column names and types
```



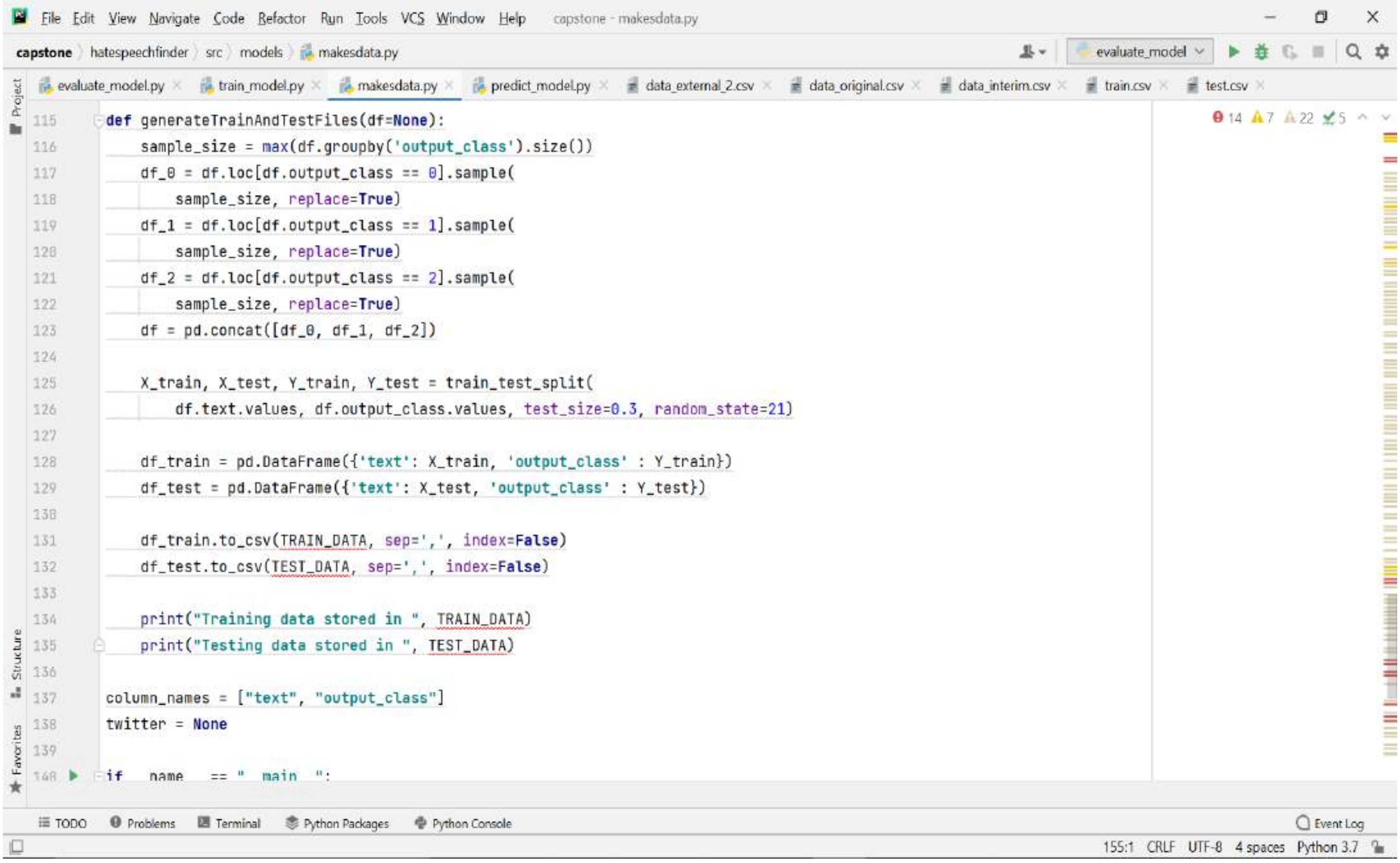
```
    # Modify column names and types
    df_external_data_2.columns = ["tweet_id", "output_class"]
    df_external_data_2.tweet_id = df_external_data_2.tweet_id.astype(str)

    # Drop the examples with clean class
    df_external_data_2 = df_external_data_2.drop(
        df_external_data_2[df_external_data_2.output_class == 'none'].index)
    df_external_data_2.output_class = df_external_data_2.output_class.apply(
        lambda x: 0)

    # Authenticate access to Twitter API
    consumer_key = config.consumer_key
    consumer_secret = config.consumer_secret
    access_token = config.access_token
    access_token_secret = config.access_token_secret
    authTwitter(consumer_key, consumer_secret,
        access_token, access_token_secret)

    # Prepare to get tweets from tweet IDs
    l = list(split(list(df_external_data_2.tweet_id),
        int(df_external_data_2.shape[0]/99)))
    with Pool(16) as pool:
        tweet_dump = pool.map(getTweetFromID, l)

    # Flat out tweet_dump into tweet_dict
    tweet_dict = dict()
```



```
    for d in tweet_dump:
        tweet_dict.update(d)

    keys = map(str, list(tweet_dict.keys()))

    #Drop the examples whos tweets are not retrived through API
    df_external_data_2 = df_external_data_2.drop(
        df_external_data_2[~df_external_data_2.tweet_id.isin(keys)].index)

    # Sort the dataset and retrived (id, tweet) items according to IDs
    df_external_data_2 = df_external_data_2.sort_values(['tweet_id'])
    tweet_tuples = list(tweet_dict.items())
    tweet_id, tweets = zip(*sorted(tweet_tuples, key = itemgetter(0)))
    tweet_id = list(tweet_id)
    tweets = list(tweets)

    # Assert if order of dataset keys match with order of retrived keys
    assert(tweet_id == list(df_external_data_2.tweet_id))

    # Add new column 'tweet' to the dataset
    df_external_data_2['tweet'] = tweets

    df_external_data_2 = df_external_data_2[["tweet", "output_class"]]
    df_external_data_2.columns = column_names

    return df_external_data_2
```

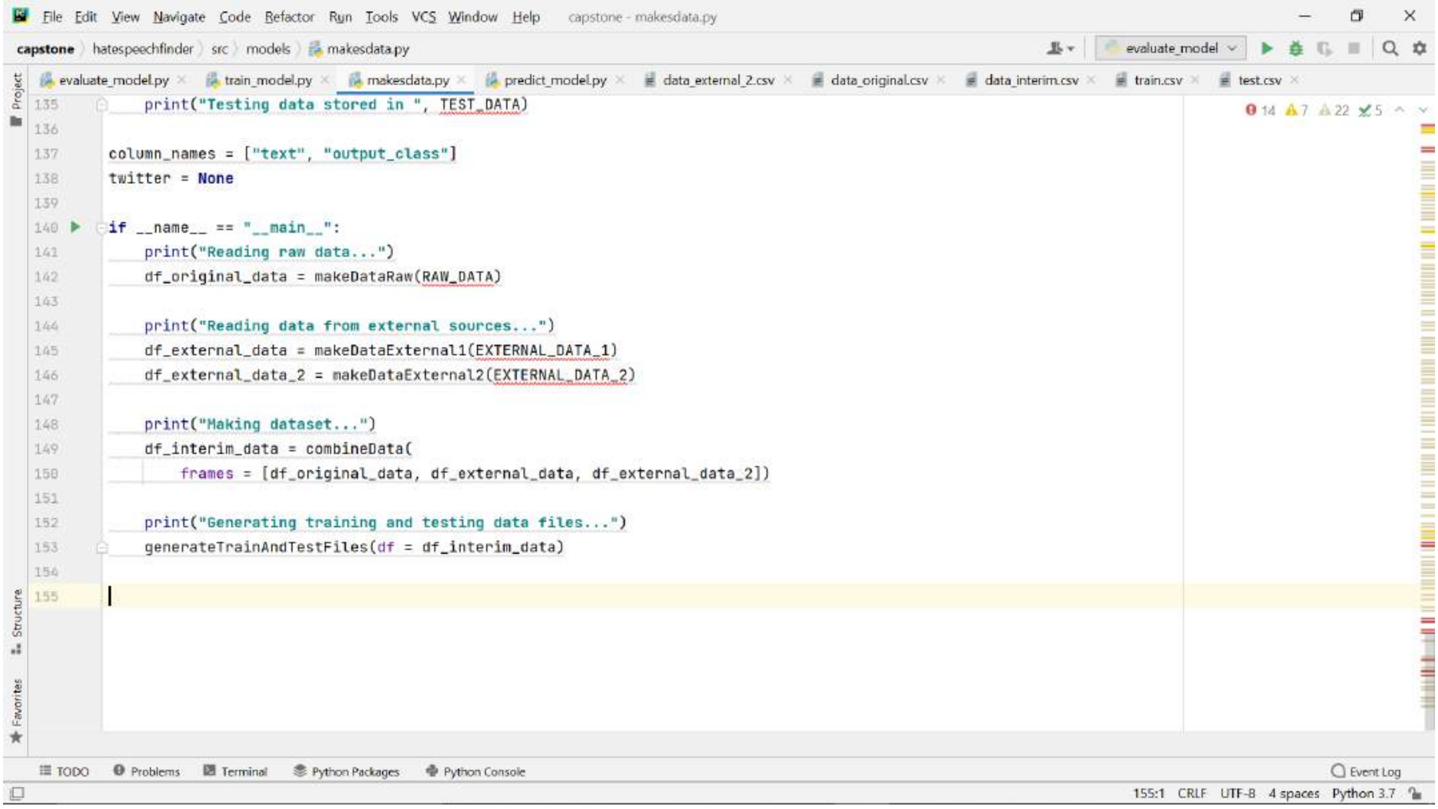


```

def combineData(frames=None):
    df_interim_data = pd.concat(frames)
    df_interim_data.text = list(df_interim_data.text.astype(str))
    df_interim_data.output_class = list(
        df_interim_data.output_class.astype(int))
    df_interim_data.to_csv(INTERIM_DATA, sep=',', index=False, encoding="utf-8")
    print("Dataset stored in ", INTERIM_DATA)
    return df_interim_data

def generateTrainAndTestFiles(df=None):
    sample_size = max(df.groupby('output_class').size())
    df_0 = df.loc[df.output_class == 0].sample(
        sample_size, replace=True)
    df_1 = df.loc[df.output_class == 1].sample(
        sample_size, replace=True)
    df_2 = df.loc[df.output_class == 2].sample(
        sample_size, replace=True)
    df = pd.concat([df_0, df_1, df_2])

    X_train, X_test, Y_train, Y_test = train_test_split(
        df.text.values, df.output_class.values, test_size=0.3, random_state=21)

    df_train = pd.DataFrame({'text': X_train, 'output_class' : Y_train})
    df_test = pd.DataFrame({'text': X_test, 'output_class' : Y_test})
```



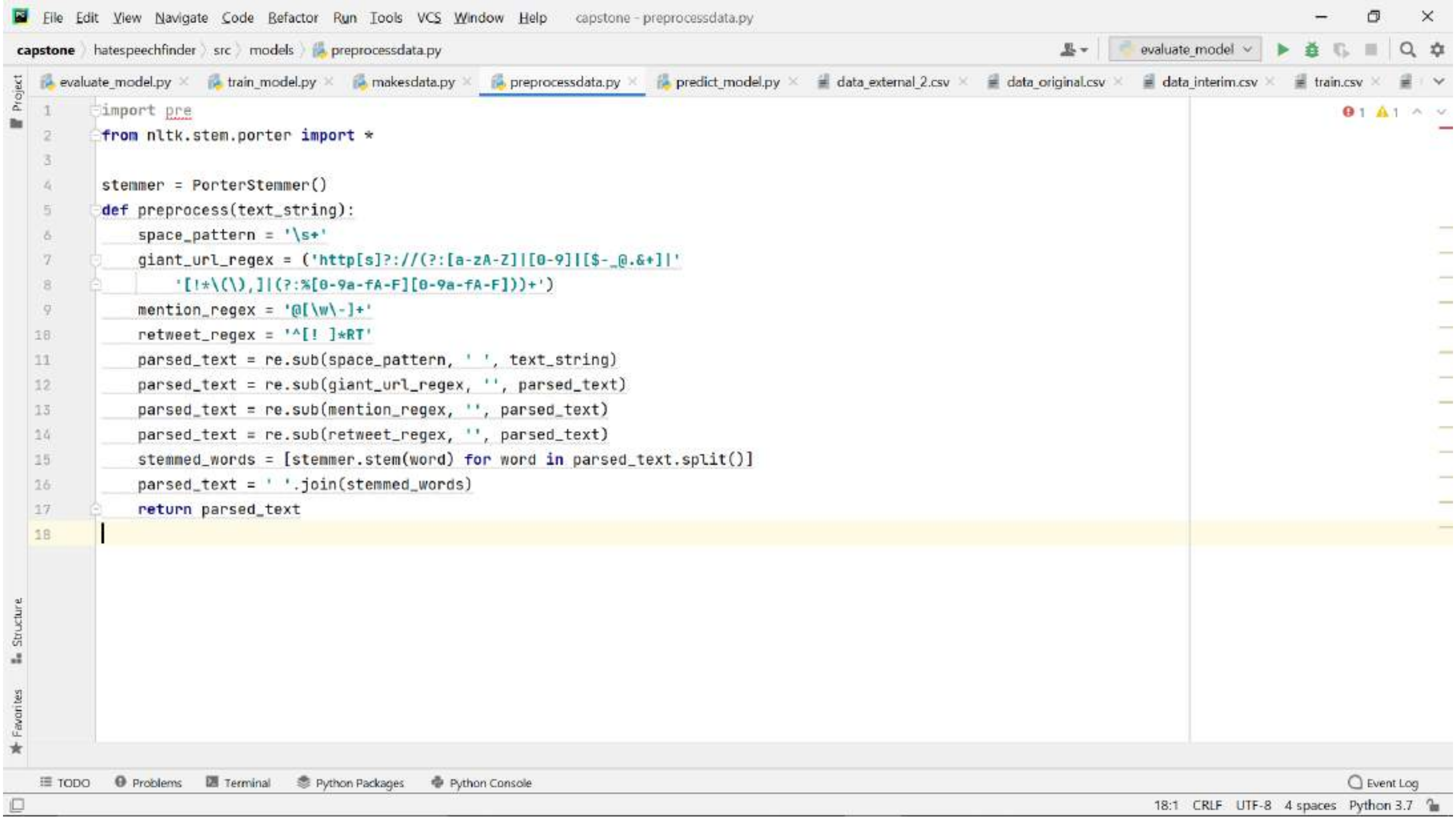
```
def generateTrainAndTestFiles(df=None):
    sample_size = max(df.groupby('output_class').size())
    df_0 = df.loc[df.output_class == 0].sample(
        sample_size, replace=True)
    df_1 = df.loc[df.output_class == 1].sample(
        sample_size, replace=True)
    df_2 = df.loc[df.output_class == 2].sample(
        sample_size, replace=True)
    df = pd.concat([df_0, df_1, df_2])

    X_train, X_test, Y_train, Y_test = train_test_split(
        df.text.values, df.output_class.values, test_size=0.3, random_state=21)

    df_train = pd.DataFrame({'text': X_train, 'output_class' : Y_train})
    df_test = pd.DataFrame({'text': X_test, 'output_class' : Y_test})
```

```

    df_train.to_csv(TRAIN_DATA, sep=',', index=False)
    df_test.to_csv(TEST_DATA, sep=',', index=False)

    print("Training data stored in ", TRAIN_DATA)
    print("Testing data stored in ", TEST_DATA)

column_names = ["text", "output_class"]
twitter = None

if __name__ == "__main__":
```



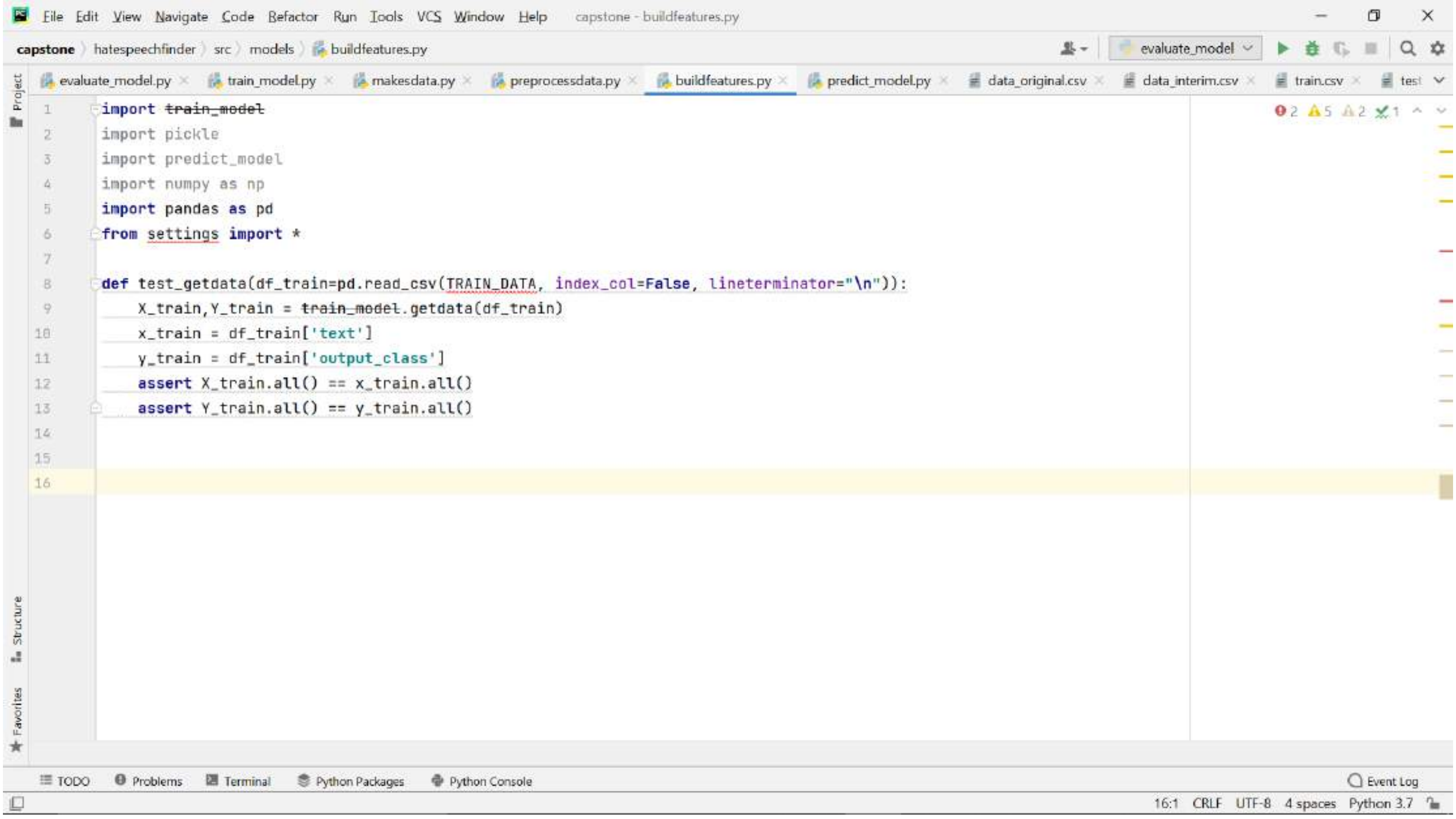
```
    print("Testing data stored in ", TEST_DATA)

column_names = ["text", "output_class"]
twitter = None

if __name__ == "__main__":
    print("Reading raw data...")
    df_original_data = makeDataRaw(RAW_DATA)

    print("Reading data from external sources...")
    df_external_data = makeDataExternal1(EXTERNAL_DATA_1)
    df_external_data_2 = makeDataExternal2(EXTERNAL_DATA_2)

    print("Making dataset...")
    df_interim_data = combineData(
        frames = [df_original_data, df_external_data, df_external_data_2])

    print("Generating training and testing data files...")
    generateTrainAndTestFiles(df = df_interim_data)

```

Preprocessdata.py



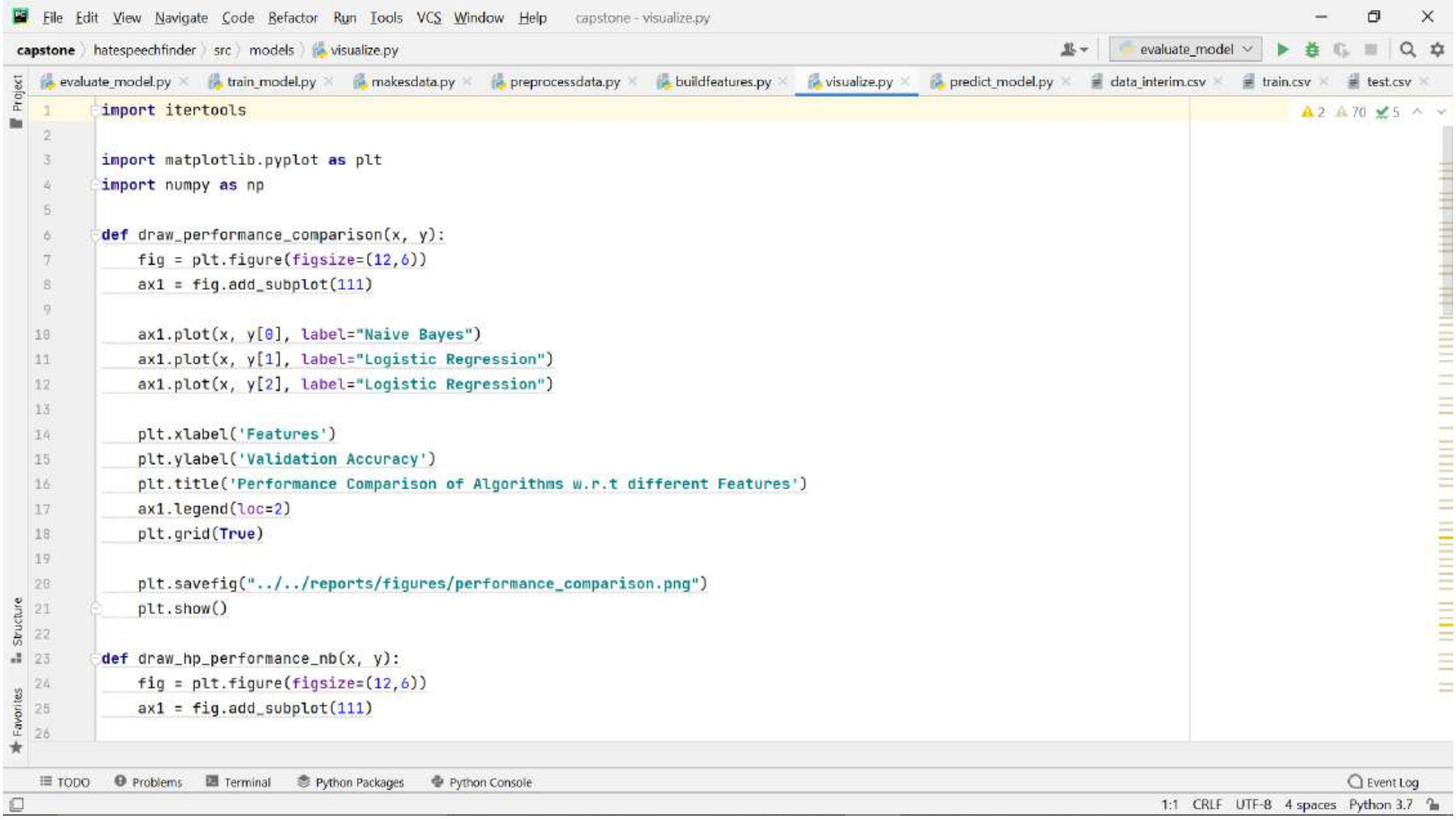
```python
import pre
from nltk.stem.porter import *

stemmer = PorterStemmer()
def preprocess(text_string):
    space_pattern = '\s+'
    giant_url_regex = ('http[s]?://(?:[a-zA-Z]|[0-9]|[$-_@.&+]|'
        '[!*\(\),]|(?:%[0-9a-fA-F][0-9a-fA-F]))+')
    mention_regex = '@[\w\-]+'
    retweet_regex = '^[! ]*RT'
    parsed_text = re.sub(space_pattern, ' ', text_string)
    parsed_text = re.sub(giant_url_regex, '', parsed_text)
    parsed_text = re.sub(mention_regex, '', parsed_text)
    parsed_text = re.sub(retweet_regex, '', parsed_text)
    stemmed_words = [stemmer.stem(word) for word in parsed_text.split()]
    parsed_text = ' '.join(stemmed_words)
    return parsed_text

```

Buildfeatures.py

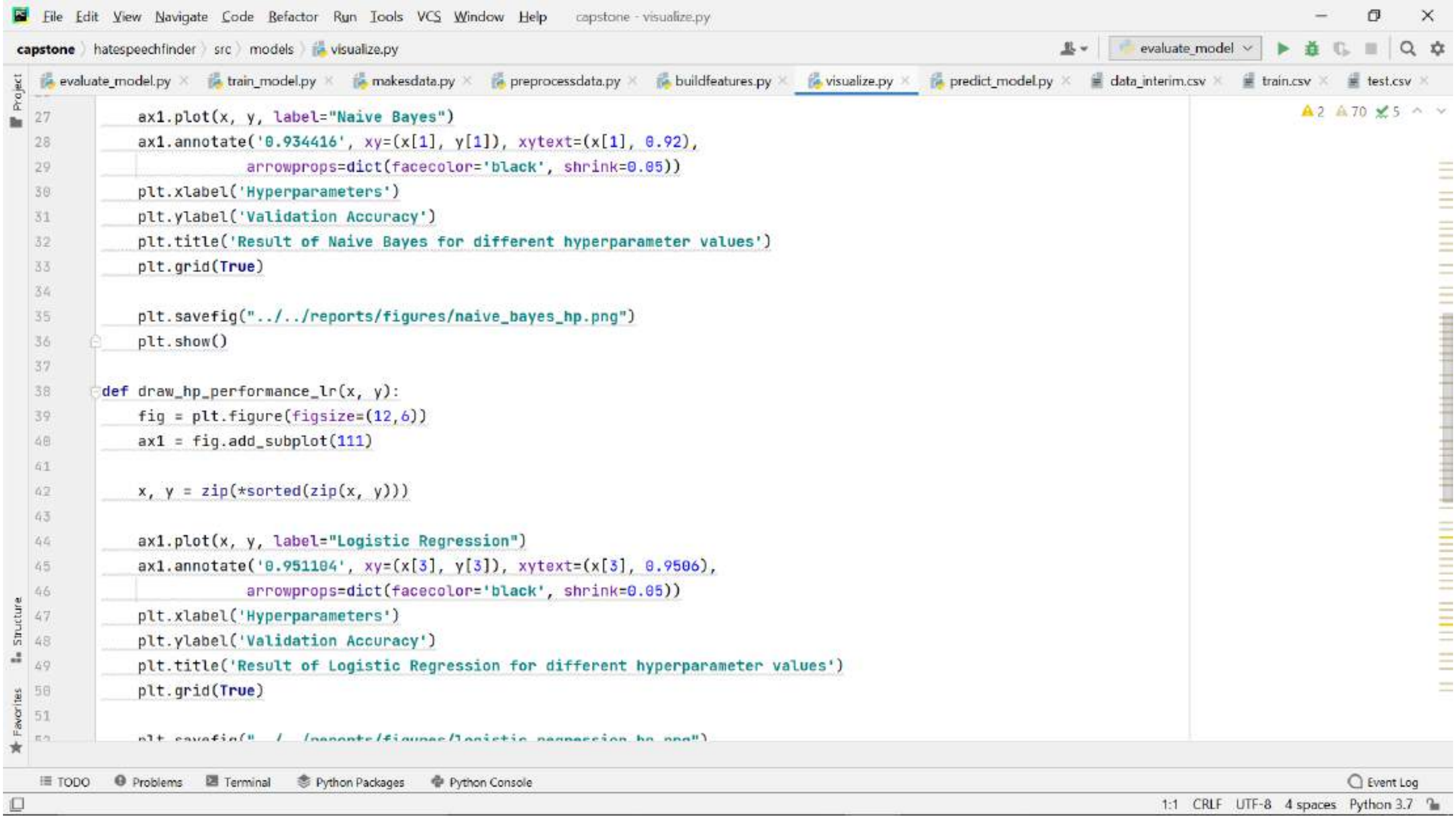


```python
import ~~train_model~~
import pickle
import predict_model
import numpy as np
import pandas as pd
from settings import *

def test_getdata(df_train=pd.read_csv(TRAIN_DATA, index_col=False, lineterminator="\n")):
    X_train,Y_train = ~~train_model~~.getdata(df_train)
    x_train = df_train['text']
    y_train = df_train['output_class']
    assert X_train.all() == x_train.all()
    assert Y_train.all() == y_train.all()


```

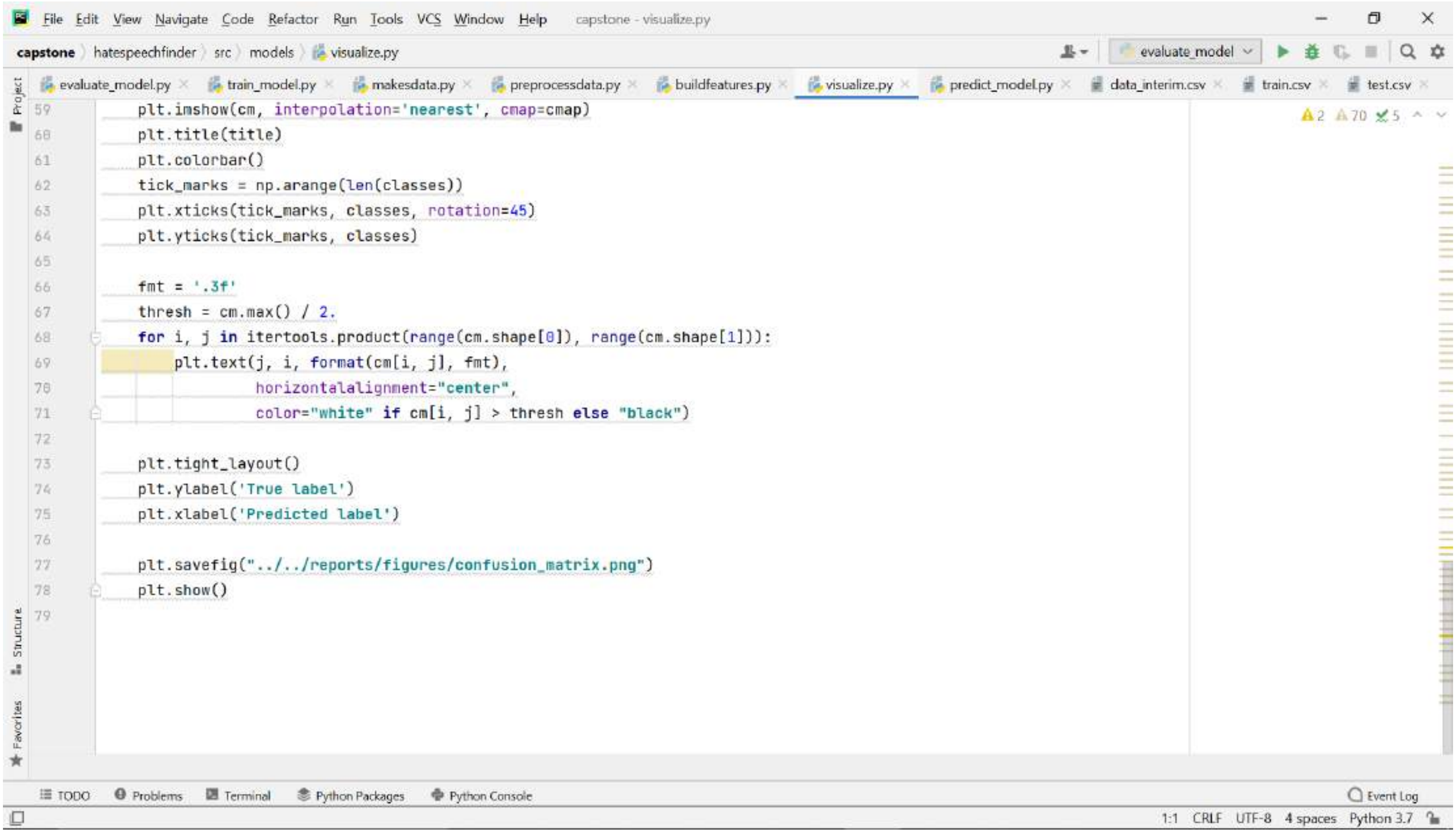
Visualize.py


File Edit View Navigate Code Refactor Run Tools VCS Window Help
capstone - visualize.py
capstone hatespeechfinder src models visualize.py
evaluate_model
evaluate_model.py train_model.py makesdata.py preprocessdata.py buildfeatures.py visualize.py predict_model.py data_interim.csv train.csv test.csv
TODO Problems Terminal Python Packages Python Console
Event Log
1:1 CRLF UTF-8 4 spaces Python 3.7

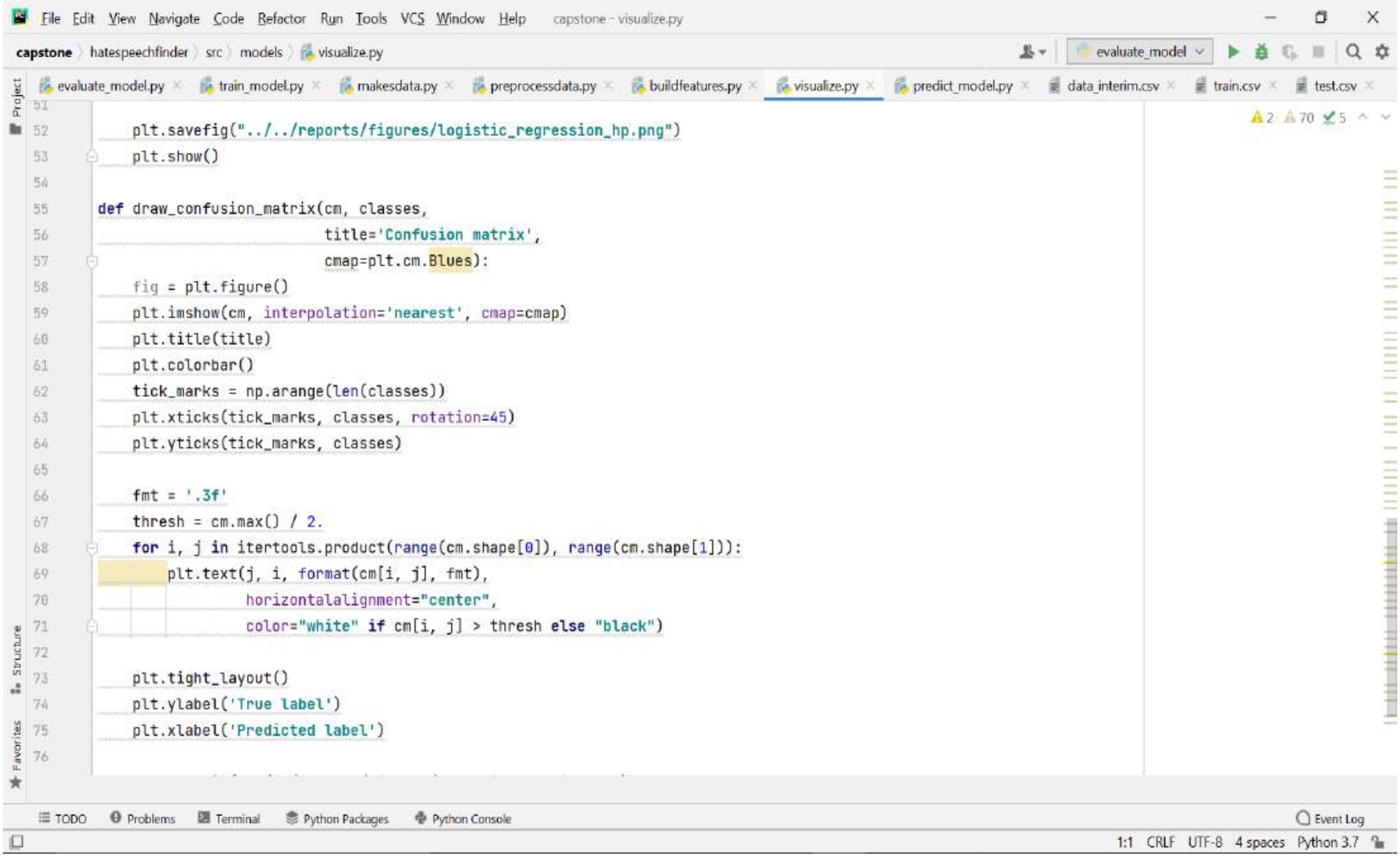
```python
import itertools

import matplotlib.pyplot as plt
import numpy as np

def draw_performance_comparison(x, y):
    fig = plt.figure(figsize=(12,6))
    ax1 = fig.add_subplot(111)

    ax1.plot(x, y[0], label="Naive Bayes")
    ax1.plot(x, y[1], label="Logistic Regression")
    ax1.plot(x, y[2], label="Logistic Regression")

    plt.xlabel('Features')
    plt.ylabel('Validation Accuracy')
    plt.title('Performance Comparison of Algorithms w.r.t different Features')
    ax1.legend(loc=2)
    plt.grid(True)

    plt.savefig("../../reports/figures/performance_comparison.png")
    plt.show()

def draw_hp_performance_nb(x, y):
    fig = plt.figure(figsize=(12,6))
    ax1 = fig.add_subplot(111)

```


File Edit View Navigate Code Refactor Run Tools VCS Window Help
capstone - visualize.py
capstone hatespeechfinder src models visualize.py
evaluate_model
evaluate_model.py train_model.py makesdata.py preprocessdata.py buildfeatures.py visualize.py predict_model.py data_interim.csv train.csv test.csv
TODO Problems Terminal Python Packages Python Console
Event Log
1:1 CRLF UTF-8 4 spaces Python 3.7

```python
    ax1.plot(x, y, label="Naive Bayes")
    ax1.annotate('0.934416', xy=(x[1], y[1]), xytext=(x[1], 0.92),
                 arrowprops=dict(facecolor='black', shrink=0.05))
    plt.xlabel('Hyperparameters')
    plt.ylabel('Validation Accuracy')
    plt.title('Result of Naive Bayes for different hyperparameter values')
    plt.grid(True)

    plt.savefig("../../reports/figures/naive_bayes_hp.png")
    plt.show()

def draw_hp_performance_lr(x, y):
    fig = plt.figure(figsize=(12,6))
    ax1 = fig.add_subplot(111)

    x, y = zip(*sorted(zip(x, y)))

    ax1.plot(x, y, label="Logistic Regression")
    ax1.annotate('0.951104', xy=(x[3], y[3]), xytext=(x[3], 0.9506),
                 arrowprops=dict(facecolor='black', shrink=0.05))
    plt.xlabel('Hyperparameters')
    plt.ylabel('Validation Accuracy')
    plt.title('Result of Logistic Regression for different hyperparameter values')
    plt.grid(True)

    plt.savefig("../../reports/figures/logistic_regression_hp.png")
```

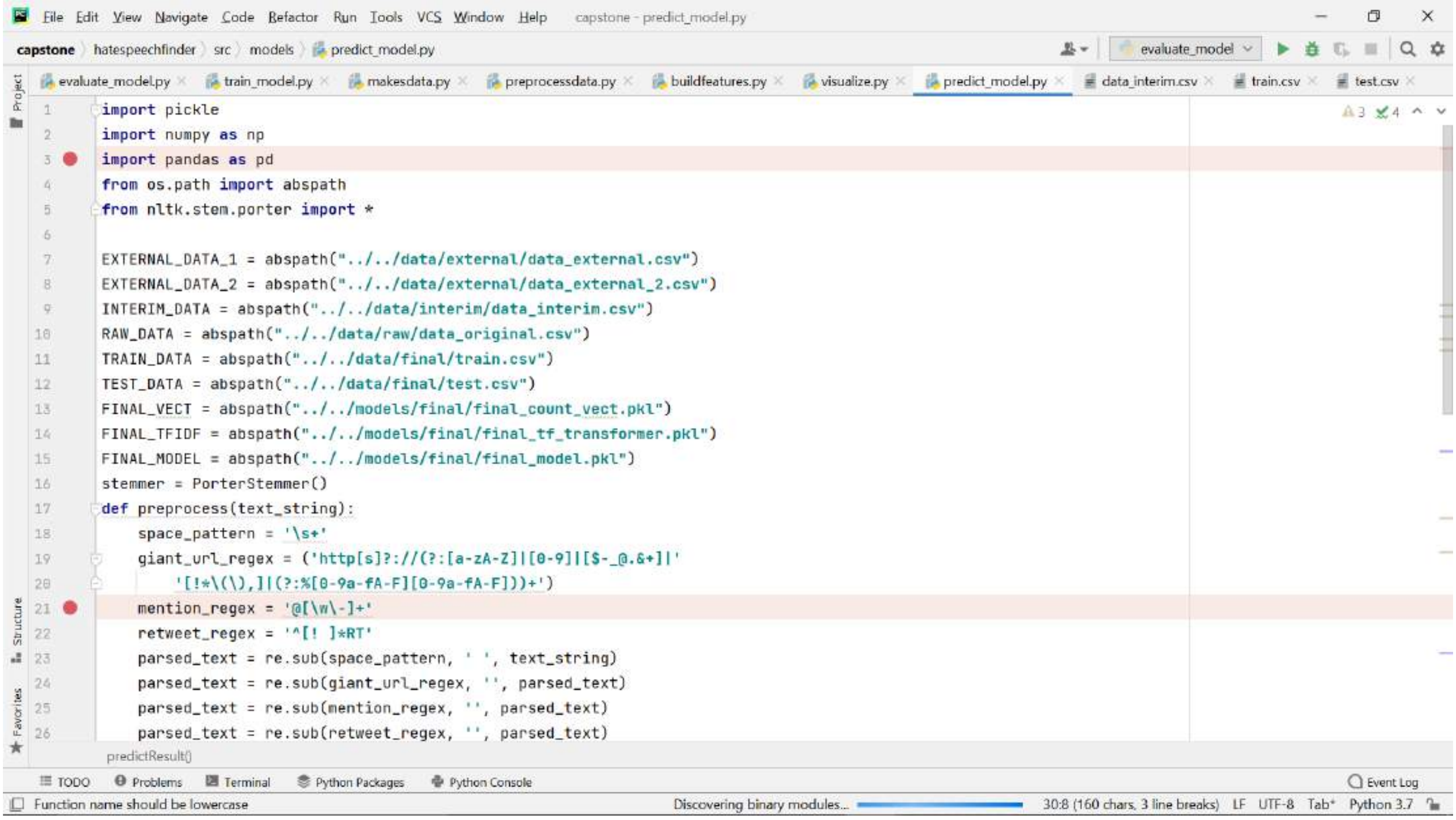

File Edit View Navigate Code Refactor Run Tools VCS Window Help capstone - visualize.py
capstone hatespeechfinder src models visualize.py
evaluate_model
evaluate_model.py train_model.py makesdata.py preprocessdata.py buildfeatures.py visualize.py predict_model.py data_interim.csv train.csv test.csv
Project Structure Favorites
TODO Problems Terminal Python Packages Python Console Event Log
1:1 CRLF UTF-8 4 spaces Python 3.7

```
    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.title(title)
    plt.colorbar()
    tick_marks = np.arange(len(classes))
    plt.xticks(tick_marks, classes, rotation=45)
    plt.yticks(tick_marks, classes)

    fmt = '.3f'
    thresh = cm.max() / 2.
    for i, j in itertools.product(range(cm.shape[0]), range(cm.shape[1])):
        plt.text(j, i, format(cm[i, j], fmt),
                 horizontalalignment="center",
                 color="white" if cm[i, j] > thresh else "black")

    plt.tight_layout()
    plt.ylabel('True label')
    plt.xlabel('Predicted label')

    plt.savefig("../../reports/figures/confusion_matrix.png")
    plt.show()
```


File Edit View Navigate Code Refactor Run Tools VCS Window Help capstone - visualize.py
capstone hatespeechfinder src models visualize.py
evaluate_model
evaluate_model.py train_model.py makesdata.py preprocessdata.py buildfeatures.py visualize.py predict_model.py data_interim.csv train.csv test.csv
Project Structure Favorites
TODO Problems Terminal Python Packages Python Console Event Log
1:1 CRLF UTF-8 4 spaces Python 3.7

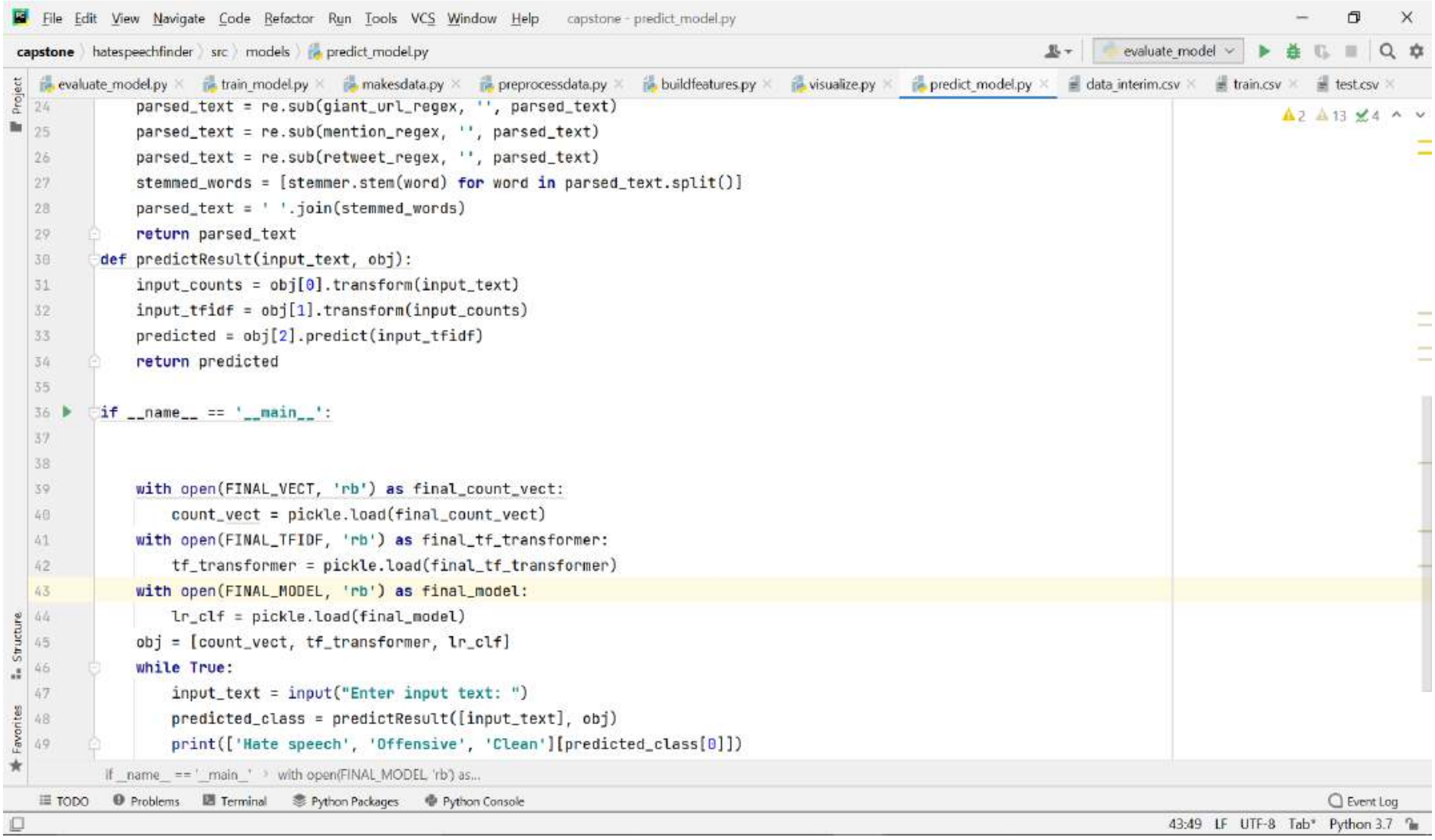
```
    plt.savefig("../../reports/figures/logistic_regression_hp.png")
    plt.show()

def draw_confusion_matrix(cm, classes,
                          title='Confusion matrix',
                          cmap=plt.cm.Blues):
    fig = plt.figure()
    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.title(title)
    plt.colorbar()
    tick_marks = np.arange(len(classes))
    plt.xticks(tick_marks, classes, rotation=45)
    plt.yticks(tick_marks, classes)

    fmt = '.3f'
    thresh = cm.max() / 2.
    for i, j in itertools.product(range(cm.shape[0]), range(cm.shape[1])):
        plt.text(j, i, format(cm[i, j], fmt),
                 horizontalalignment="center",
                 color="white" if cm[i, j] > thresh else "black")

    plt.tight_layout()
    plt.ylabel('True label')
    plt.xlabel('Predicted label')
```

Predict model.py

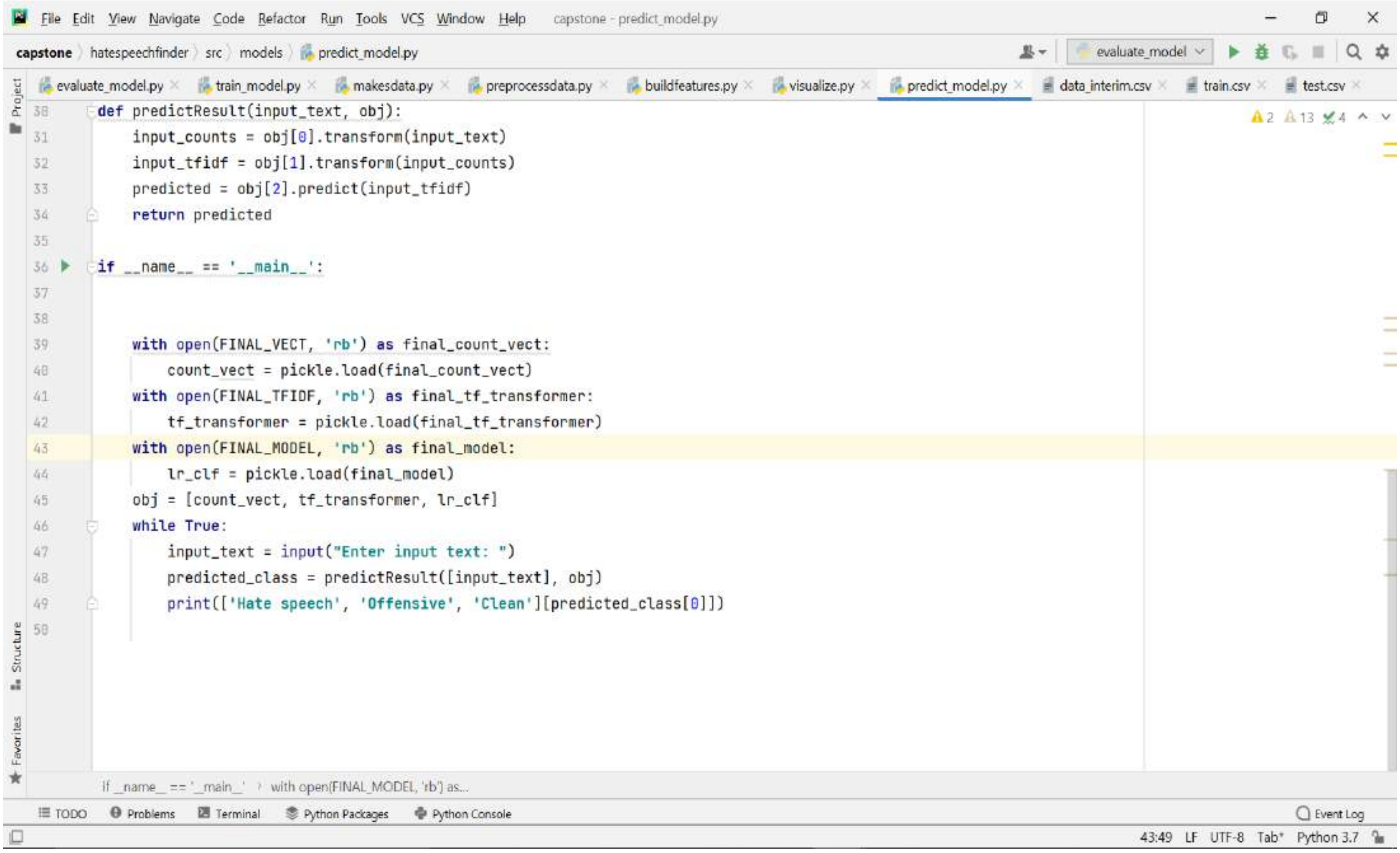


```
import pickle
import numpy as np
import pandas as pd
from os.path import abspath
from nltk.stem.porter import *

EXTERNAL_DATA_1 = abspath("../../data/external/data_external.csv")
EXTERNAL_DATA_2 = abspath("../../data/external/data_external_2.csv")
INTERIM_DATA = abspath("../../data/interim/data_interim.csv")
RAW_DATA = abspath("../../data/raw/data_original.csv")
TRAIN_DATA = abspath("../../data/final/train.csv")
TEST_DATA = abspath("../../data/final/test.csv")
FINAL_VECT = abspath("../../models/final/final_count_vect.pkl")
FINAL_TFIDF = abspath("../../models/final/final_tf_transformer.pkl")
FINAL_MODEL = abspath("../../models/final/final_model.pkl")
stemmer = PorterStemmer()
def preprocess(text_string):
    space_pattern = '\s+'
    giant_url_regex = ('http[s]?://(?:[a-zA-Z]|[0-9]|[$-_@.&+]|'
        '[!*\(\),]|(?:%[0-9a-fA-F][0-9a-fA-F]))+')
    mention_regex = '@[\w\-]+'
    retweet_regex = '^[! ]*RT'
    parsed_text = re.sub(space_pattern, ' ', text_string)
    parsed_text = re.sub(giant_url_regex, '', parsed_text)
    parsed_text = re.sub(mention_regex, '', parsed_text)
    parsed_text = re.sub(retweet_regex, '', parsed_text)
```

File Edit View Navigate Code Refactor Run Tools VCS Window Help
capstone - predict_model.py
capstone › hatespeechfinder › src › models › predict_model.py
evaluate_model
evaluate_model.py
train_model.py
makesdata.py
preprocessdata.py
buildfeatures.py
visualize.py
predict_model.py
data_interim.csv
train.csv
test.csv
predictResult()
TODO
Problems
Terminal
Python Packages
Python Console
Event Log
Function name should be lowercase
Discovering binary modules...
30:8 (160 chars, 3 line breaks) LF UTF-8 Tab* Python 3.7


File Edit View Navigate Code Refactor Run Tools VCS Window Help
capstone - predict_model.py
capstone › hatespeechfinder › src › models › predict_model.py
evaluate_model
evaluate_model.py
train_model.py
makesdata.py
preprocessdata.py
buildfeatures.py
visualize.py
predict_model.py
data_interim.csv
train.csv
test.csv
Project
Structure
Favorites
if __name__ == '__main__' › with open(FINAL_MODEL, 'rb') as...
TODO
Problems
Terminal
Python Packages
Python Console
Event Log
43:49 LF UTF-8 Tab* Python 3.7

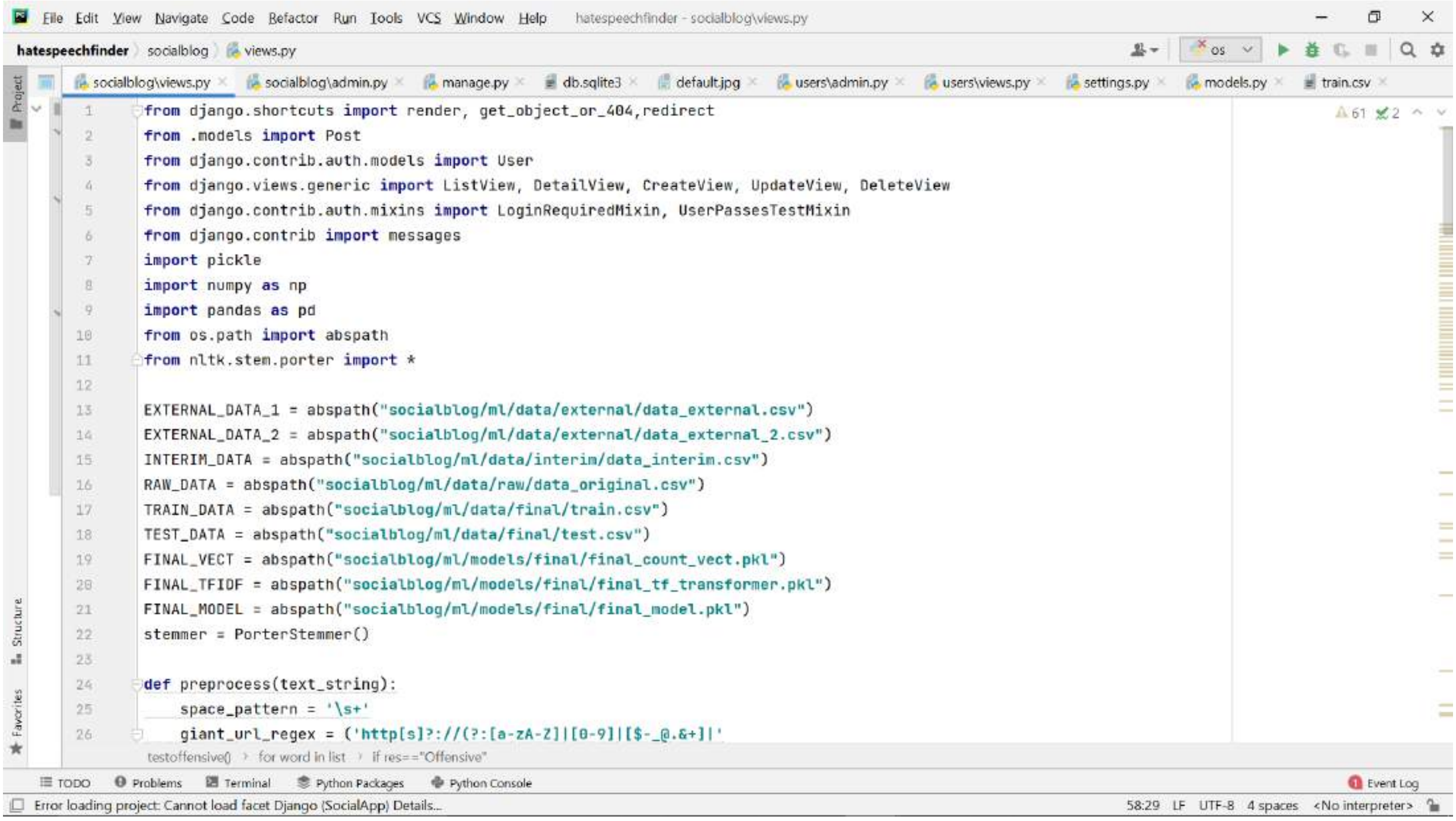
```python
    parsed_text = re.sub(giant_url_regex, '', parsed_text)
    parsed_text = re.sub(mention_regex, '', parsed_text)
    parsed_text = re.sub(retweet_regex, '', parsed_text)
    stemmed_words = [stemmer.stem(word) for word in parsed_text.split()]
    parsed_text = ' '.join(stemmed_words)
    return parsed_text
def predictResult(input_text, obj):
    input_counts = obj[0].transform(input_text)
    input_tfidf = obj[1].transform(input_counts)
    predicted = obj[2].predict(input_tfidf)
    return predicted

if __name__ == '__main__':


    with open(FINAL_VECT, 'rb') as final_count_vect:
        count_vect = pickle.load(final_count_vect)
    with open(FINAL_TFIDF, 'rb') as final_tf_transformer:
        tf_transformer = pickle.load(final_tf_transformer)
    with open(FINAL_MODEL, 'rb') as final_model:
        lr_clf = pickle.load(final_model)
    obj = [count_vect, tf_transformer, lr_clf]
    while True:
        input_text = input("Enter input text: ")
        predicted_class = predictResult([input_text], obj)
        print(['Hate speech', 'Offensive', 'Clean'][predicted_class[0]])
```


File Edit View Navigate Code Refactor Run Tools VCS Window Help
capstone - predict_model.py
capstone › hatespeechfinder › src › models › predict_model.py
evaluate_model
evaluate_model.py
train_model.py
makesdata.py
preprocessdata.py
buildfeatures.py
visualize.py
predict_model.py
data_interim.csv
train.csv
test.csv
Project
Structure
Favorites
if __name__ == '__main__' › with open(FINAL_MODEL, 'rb') as...
TODO
Problems
Terminal
Python Packages
Python Console
Event Log
43:49 LF UTF-8 Tab* Python 3.7

```python
def predictResult(input_text, obj):
    input_counts = obj[0].transform(input_text)
    input_tfidf = obj[1].transform(input_counts)
    predicted = obj[2].predict(input_tfidf)
    return predicted

if __name__ == '__main__':


    with open(FINAL_VECT, 'rb') as final_count_vect:
        count_vect = pickle.load(final_count_vect)
    with open(FINAL_TFIDF, 'rb') as final_tf_transformer:
        tf_transformer = pickle.load(final_tf_transformer)
    with open(FINAL_MODEL, 'rb') as final_model:
        lr_clf = pickle.load(final_model)
    obj = [count_vect, tf_transformer, lr_clf]
    while True:
        input_text = input("Enter input text: ")
        predicted_class = predictResult([input_text], obj)
        print(['Hate speech', 'Offensive', 'Clean'][predicted_class[0]])

```

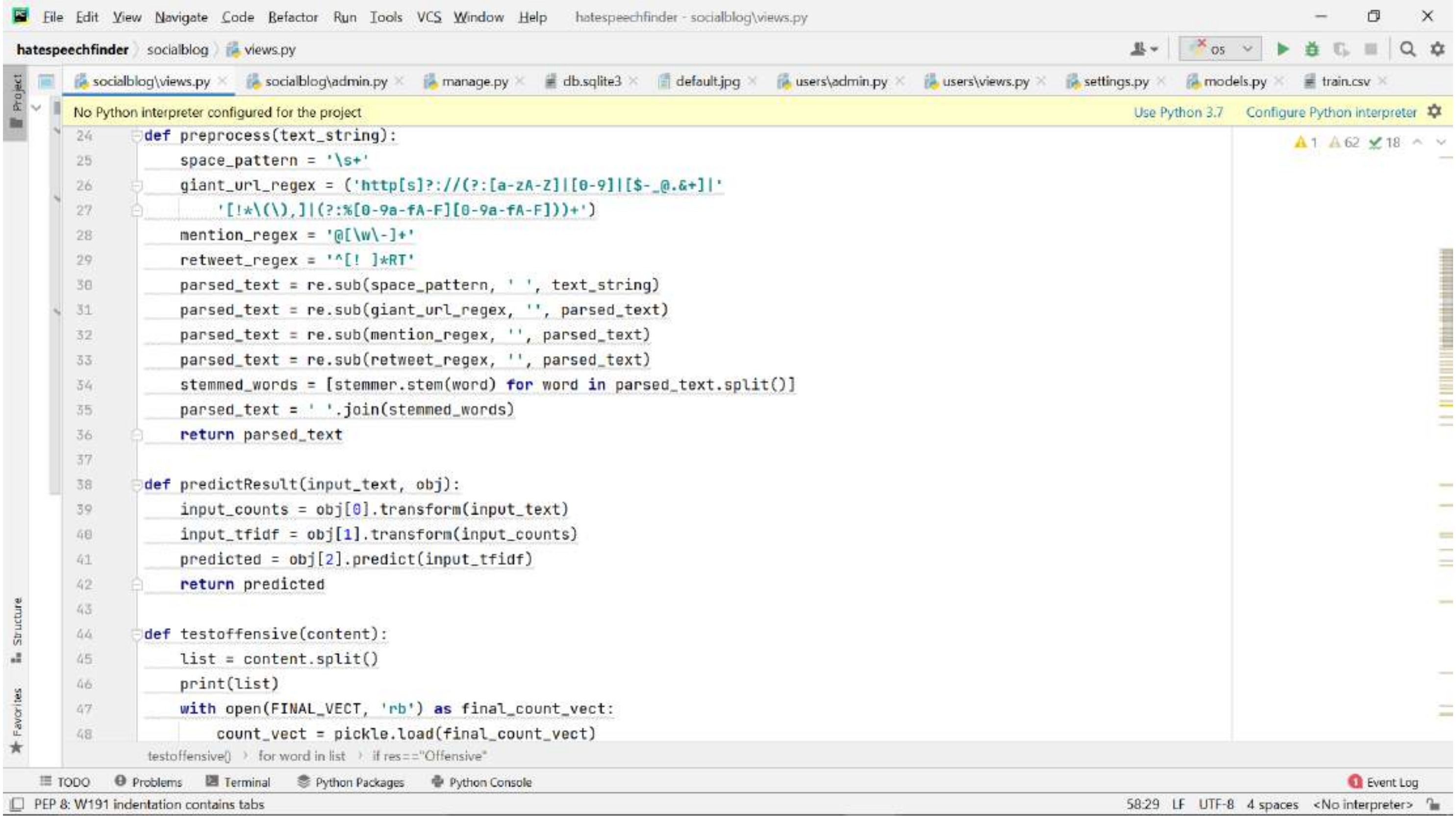
Core codes integrated in django

Socialblog\Views.py


File Edit View Navigate Code Refactor Run Tools VCS Window Help
hatespeechfinder - socialblog\views.py
hatespeechfinder › socialblog › views.py
os
socialblog\views.py
socialblog\admin.py
manage.py
db.sqlite3
default.jpg
users\admin.py
users\views.py
settings.py
models.py
train.csv
Project
Structure
Favorites
⚠61 ✓2
testoffensive() › for word in list › if res=="Offensive"
TODO
Problems
Terminal
Python Packages
Python Console
Event Log
58:29 LF UTF-8 4 spaces <No interpreter>

```python
from django.shortcuts import render, get_object_or_404,redirect
from .models import Post
from django.contrib.auth.models import User
from django.views.generic import ListView, DetailView, CreateView, UpdateView, DeleteView
from django.contrib.auth.mixins import LoginRequiredMixin, UserPassesTestMixin
from django.contrib import messages
import pickle
import numpy as np
import pandas as pd
from os.path import abspath
from nltk.stem.porter import *

EXTERNAL_DATA_1 = abspath("socialblog/ml/data/external/data_external.csv")
EXTERNAL_DATA_2 = abspath("socialblog/ml/data/external/data_external_2.csv")
INTERIM_DATA = abspath("socialblog/ml/data/interim/data_interim.csv")
RAW_DATA = abspath("socialblog/ml/data/raw/data_original.csv")
TRAIN_DATA = abspath("socialblog/ml/data/final/train.csv")
TEST_DATA = abspath("socialblog/ml/data/final/test.csv")
FINAL_VECT = abspath("socialblog/ml/models/final/final_count_vect.pkl")
FINAL_TFIDF = abspath("socialblog/ml/models/final/final_tf_transformer.pkl")
FINAL_MODEL = abspath("socialblog/ml/models/final/final_model.pkl")
stemmer = PorterStemmer()

def preprocess(text_string):
    space_pattern = '\s+'
    giant_url_regex = ('http[s]?://(?:[a-zA-Z]|[0-9]|[$-_@.&+]|'
```

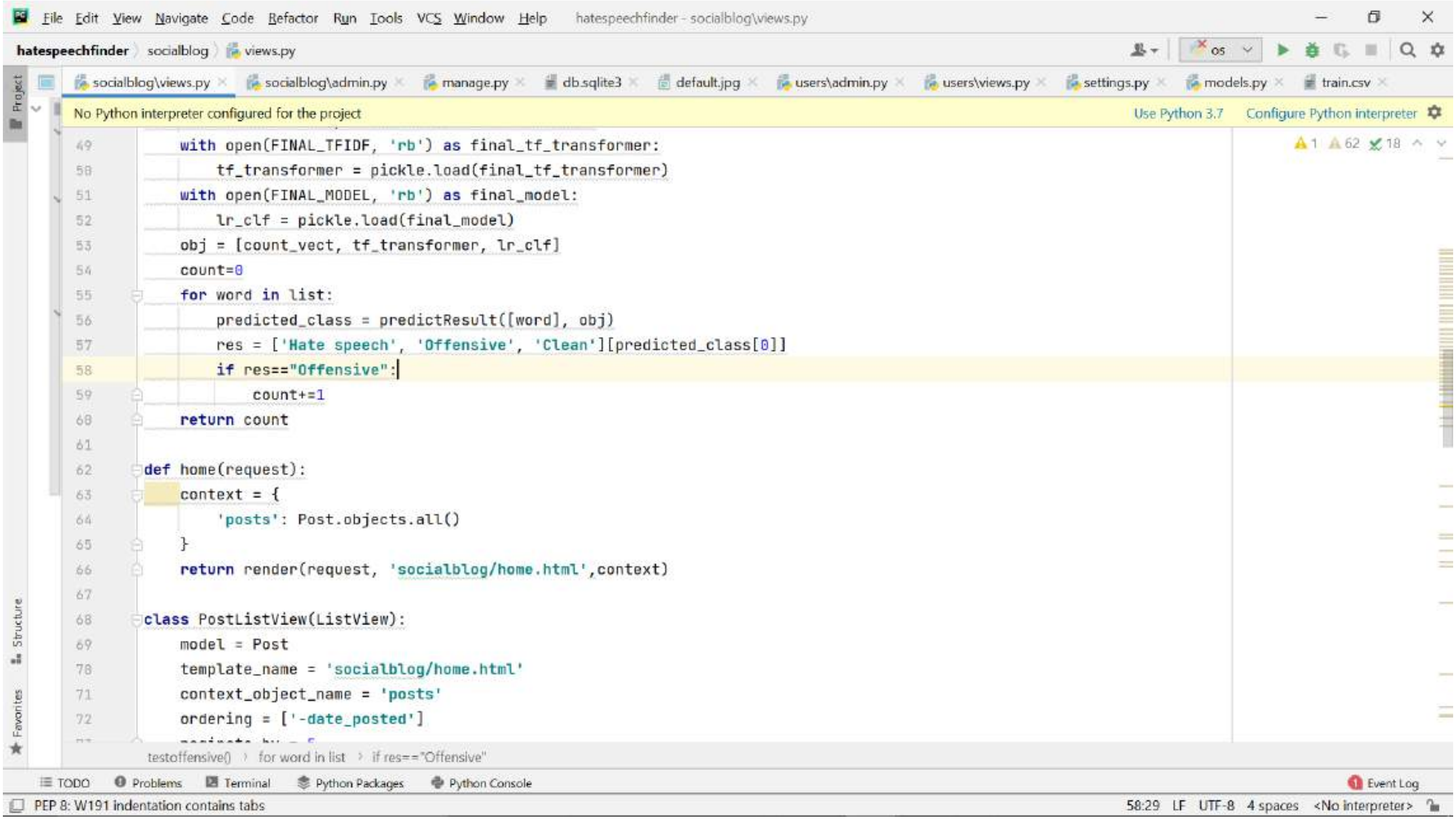
Error loading project: Cannot load facet Django (SocialApp) Details...


File Edit View Navigate Code Refactor Run Tools VCS Window Help
hatespeechfinder - socialblog\views.py
hatespeechfinder › socialblog › views.py
os
socialblog\views.py
socialblog\admin.py
manage.py
db.sqlite3
default.jpg
users\admin.py
users\views.py
settings.py
models.py
train.csv
Use Python 3.7
Configure Python interpreter
Project
Structure
Favorites
⚠1 ⚠62 ✓18
testoffensive() › for word in list › if res=="Offensive"
TODO
Problems
Terminal
Python Packages
Python Console
Event Log
58:29 LF UTF-8 4 spaces <No interpreter>

No Python interpreter configured for the project

```python
def preprocess(text_string):
    space_pattern = '\s+'
    giant_url_regex = ('http[s]?://(?:[a-zA-Z]|[0-9]|[$-_@.&+]|'
        '[!*\(\),]|(?:%[0-9a-fA-F][0-9a-fA-F]))+')
    mention_regex = '@[\w\-]+'
    retweet_regex = '^[! ]*RT'
    parsed_text = re.sub(space_pattern, ' ', text_string)
    parsed_text = re.sub(giant_url_regex, '', parsed_text)
    parsed_text = re.sub(mention_regex, '', parsed_text)
    parsed_text = re.sub(retweet_regex, '', parsed_text)
    stemmed_words = [stemmer.stem(word) for word in parsed_text.split()]
    parsed_text = ' '.join(stemmed_words)
    return parsed_text

def predictResult(input_text, obj):
    input_counts = obj[0].transform(input_text)
    input_tfidf = obj[1].transform(input_counts)
    predicted = obj[2].predict(input_tfidf)
    return predicted

def testoffensive(content):
    list = content.split()
    print(list)
    with open(FINAL_VECT, 'rb') as final_count_vect:
        count_vect = pickle.load(final_count_vect)
```

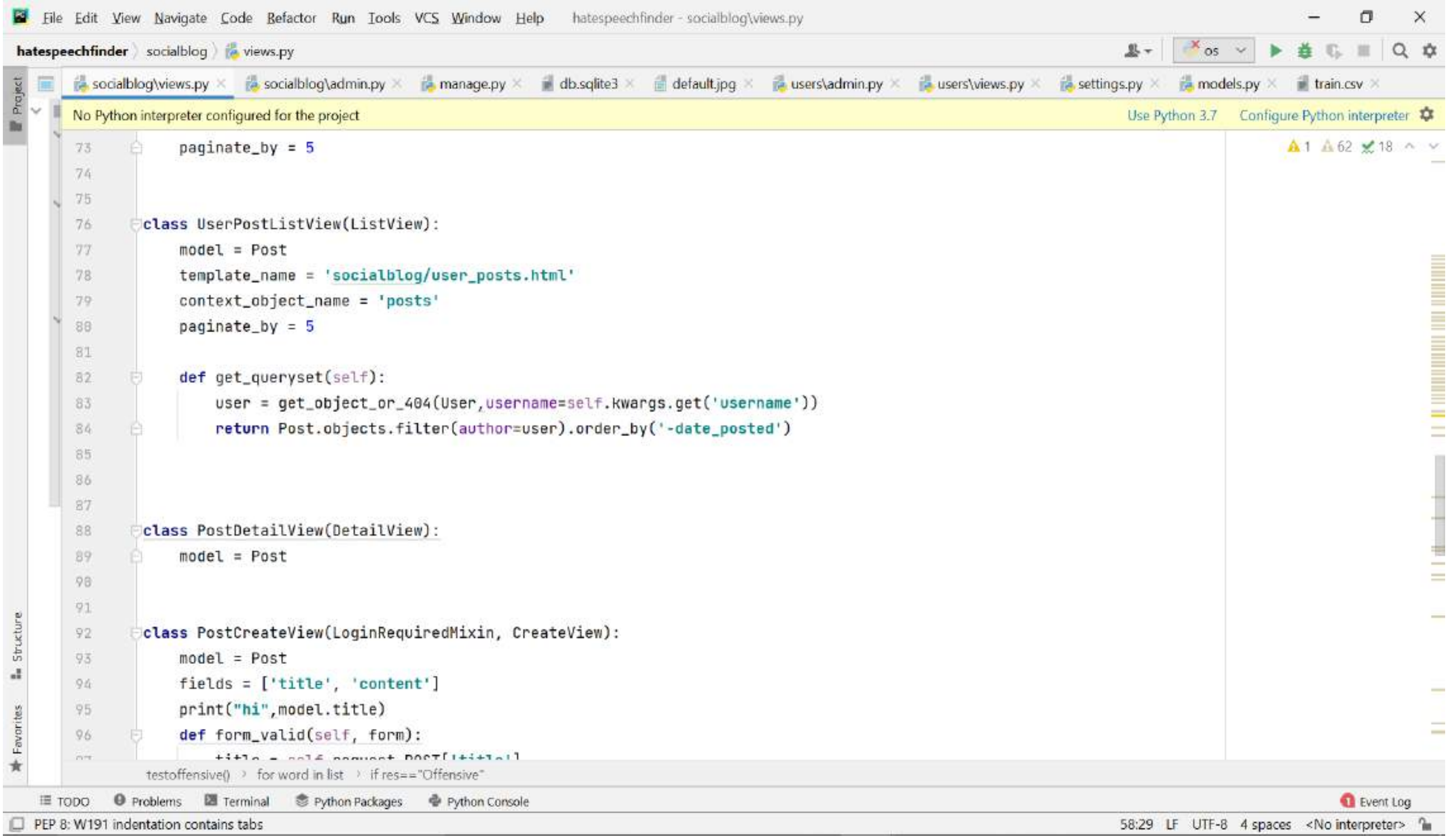
PEP 8: W191 indentation contains tabs


File Edit View Navigate Code Refactor Run Tools VCS Window Help
hatespeechfinder - socialblog\views.py
hatespeechfinder 〉 socialblog 〉 views.py
os
socialblog\views.py
socialblog\admin.py
manage.py
db.sqlite3
default.jpg
users\admin.py
users\views.py
settings.py
models.py
train.csv
Use Python 3.7
Configure Python interpreter
Project
Structure
Favorites
1
62
18
testoffensive() 〉 for word in list 〉 if res=="Offensive"
TODO
Problems
Terminal
Python Packages
Python Console
Event Log
58:29 LF UTF-8 4 spaces <No interpreter>

No Python interpreter configured for the project

```
    with open(FINAL_TFIDF, 'rb') as final_tf_transformer:
        tf_transformer = pickle.load(final_tf_transformer)
    with open(FINAL_MODEL, 'rb') as final_model:
        lr_clf = pickle.load(final_model)
    obj = [count_vect, tf_transformer, lr_clf]
    count=0
    for word in list:
        predicted_class = predictResult([word], obj)
        res = ['Hate speech', 'Offensive', 'Clean'][predicted_class[0]]
        if res=="Offensive":
            count+=1
    return count

def home(request):
    context = {
        'posts': Post.objects.all()
    }
    return render(request, 'socialblog/home.html',context)

class PostListView(ListView):
    model = Post
    template_name = 'socialblog/home.html'
    context_object_name = 'posts'
    ordering = ['-date_posted']
    paginate_by = 5
```

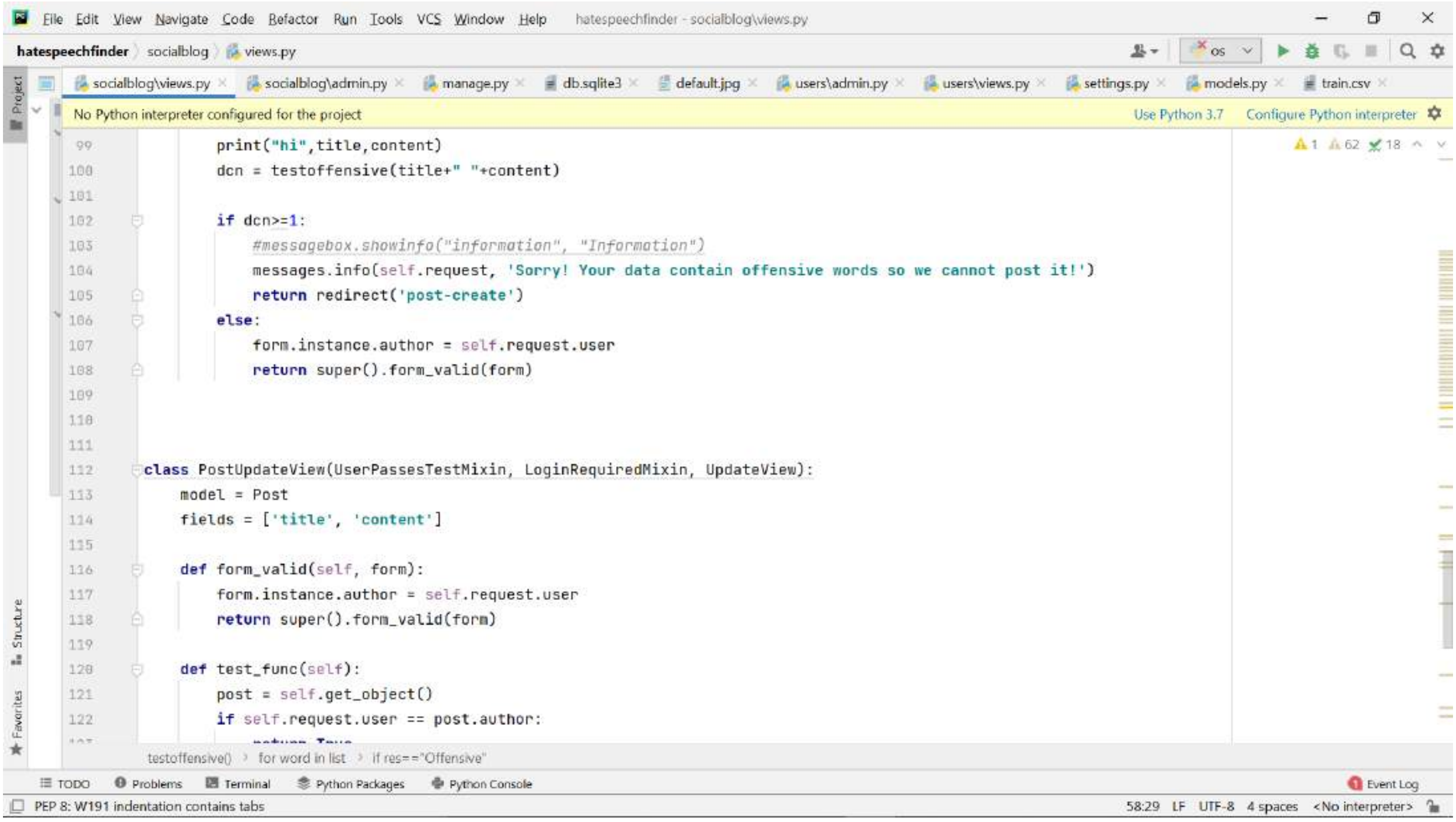
PEP 8: W191 indentation contains tabs


File Edit View Navigate Code Refactor Run Tools VCS Window Help
hatespeechfinder - socialblog\views.py
hatespeechfinder 〉 socialblog 〉 views.py
os
socialblog\views.py
socialblog\admin.py
manage.py
db.sqlite3
default.jpg
users\admin.py
users\views.py
settings.py
models.py
train.csv
Use Python 3.7
Configure Python interpreter
Project
Structure
Favorites
1
62
18
testoffensive() 〉 for word in list 〉 if res=="Offensive"
TODO
Problems
Terminal
Python Packages
Python Console
Event Log
58:29 LF UTF-8 4 spaces <No interpreter>

No Python interpreter configured for the project

```
    paginate_by = 5


class UserPostListView(ListView):
    model = Post
    template_name = 'socialblog/user_posts.html'
    context_object_name = 'posts'
    paginate_by = 5

    def get_queryset(self):
        user = get_object_or_404(User,username=self.kwargs.get('username'))
        return Post.objects.filter(author=user).order_by('-date_posted')



class PostDetailView(DetailView):
    model = Post


class PostCreateView(LoginRequiredMixin, CreateView):
    model = Post
    fields = ['title', 'content']
    print("hi",model.title)
    def form_valid(self, form):
        title = self.request.POST['title']
```

PEP 8: W191 indentation contains tabs



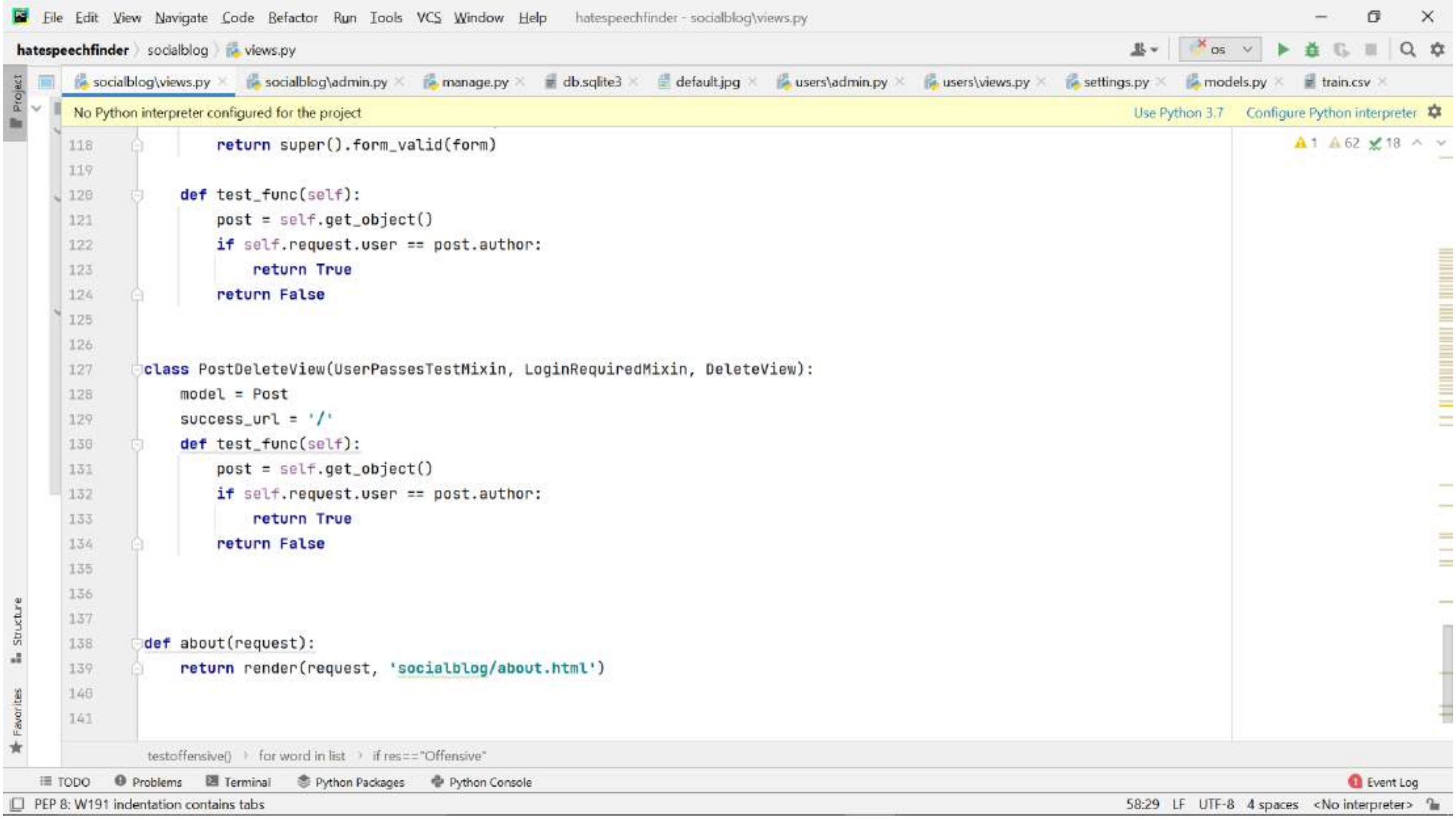
No Python interpreter configured for the project

```python
        print("hi",title,content)
        dcn = testoffensive(title+" "+content)

        if dcn>=1:
            #messagebox.showinfo("information", "Information")
            messages.info(self.request, 'Sorry! Your data contain offensive words so we cannot post it!')
            return redirect('post-create')
        else:
            form.instance.author = self.request.user
            return super().form_valid(form)



class PostUpdateView(UserPassesTestMixin, LoginRequiredMixin, UpdateView):
    model = Post
    fields = ['title', 'content']

    def form_valid(self, form):
        form.instance.author = self.request.user
        return super().form_valid(form)

    def test_func(self):
        post = self.get_object()
        if self.request.user == post.author:
            return True
```

testoffensive() › for word in list › if res=="Offensive"

PEP 8: W191 indentation contains tabs

58:29 LF UTF-8 4 spaces <No interpreter>



No Python interpreter configured for the project

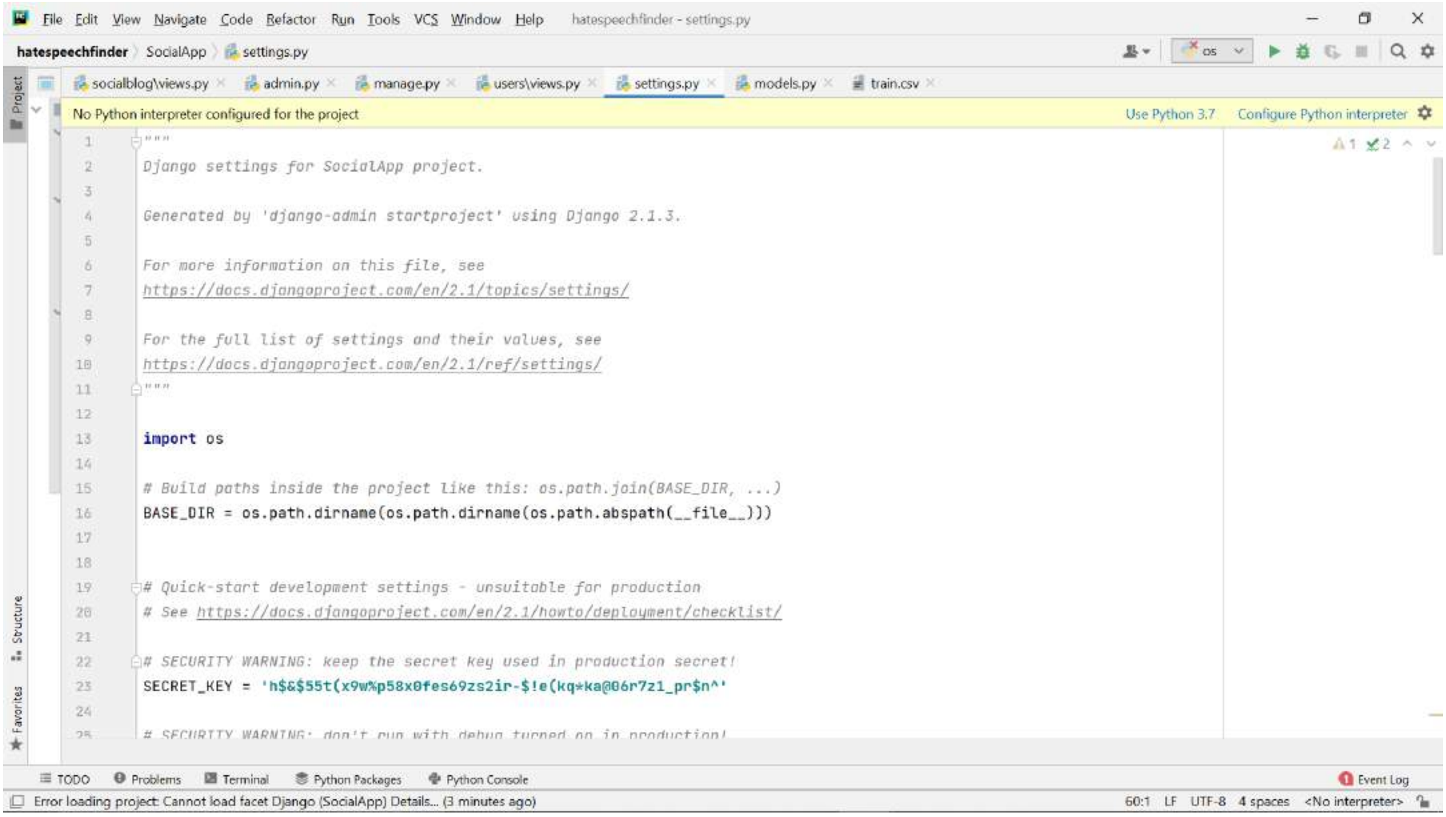
```python
        return super().form_valid(form)

    def test_func(self):
        post = self.get_object()
        if self.request.user == post.author:
            return True
        return False


class PostDeleteView(UserPassesTestMixin, LoginRequiredMixin, DeleteView):
    model = Post
    success_url = '/'
    def test_func(self):
        post = self.get_object()
        if self.request.user == post.author:
            return True
        return False



def about(request):
    return render(request, 'socialblog/about.html')


```

testoffensive() › for word in list › if res=="Offensive"

PEP 8: W191 indentation contains tabs

58:29 LF UTF-8 4 spaces <No interpreter>

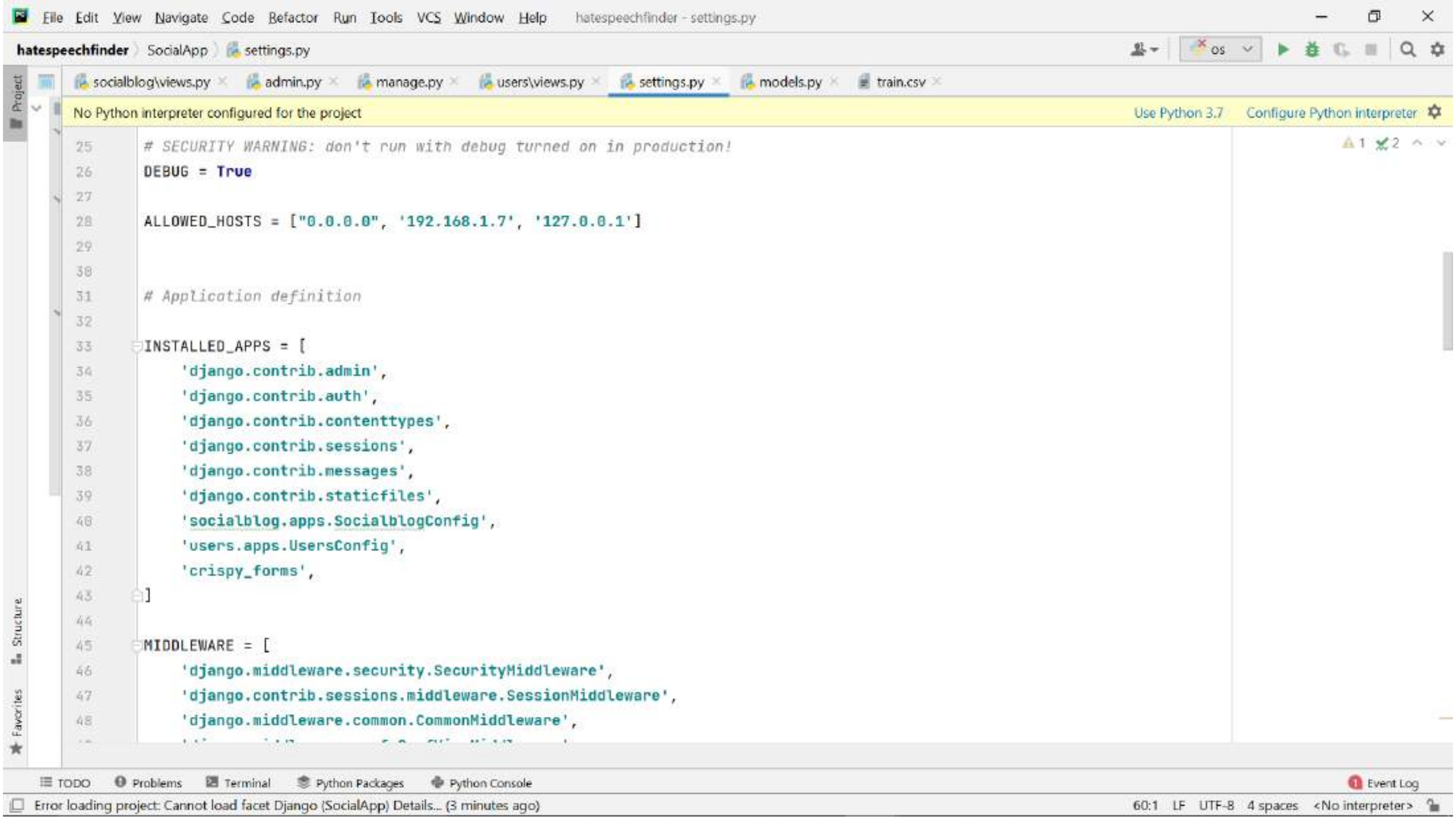
Settings.py


File Edit View Navigate Code Refactor Run Tools VCS Window Help hatespeechfinder - settings.py
hatespeechfinder 〉 SocialApp 〉 settings.py
os
socialblog\views.py
admin.py
manage.py
users\views.py
settings.py
models.py
train.csv
Project
Structure
Favorites
TODO
Problems
Terminal
Python Packages
Python Console
Event Log
60:1 LF UTF-8 4 spaces <No interpreter>

No Python interpreter configured for the project   Use Python 3.7   Configure Python interpreter

```
"""
Django settings for SocialApp project.

Generated by 'django-admin startproject' using Django 2.1.3.

For more information on this file, see
https://docs.djangoproject.com/en/2.1/topics/settings/

For the full list of settings and their values, see
https://docs.djangoproject.com/en/2.1/ref/settings/
"""

import os

# Build paths inside the project like this: os.path.join(BASE_DIR, ...)
BASE_DIR = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))


# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/2.1/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = 'h$&$55t(x9w%p58x0fes69zs2ir-$!e(kq*ka@06r7z1_pr$n^'

# SECURITY WARNING: don't run with debug turned on in production!
```

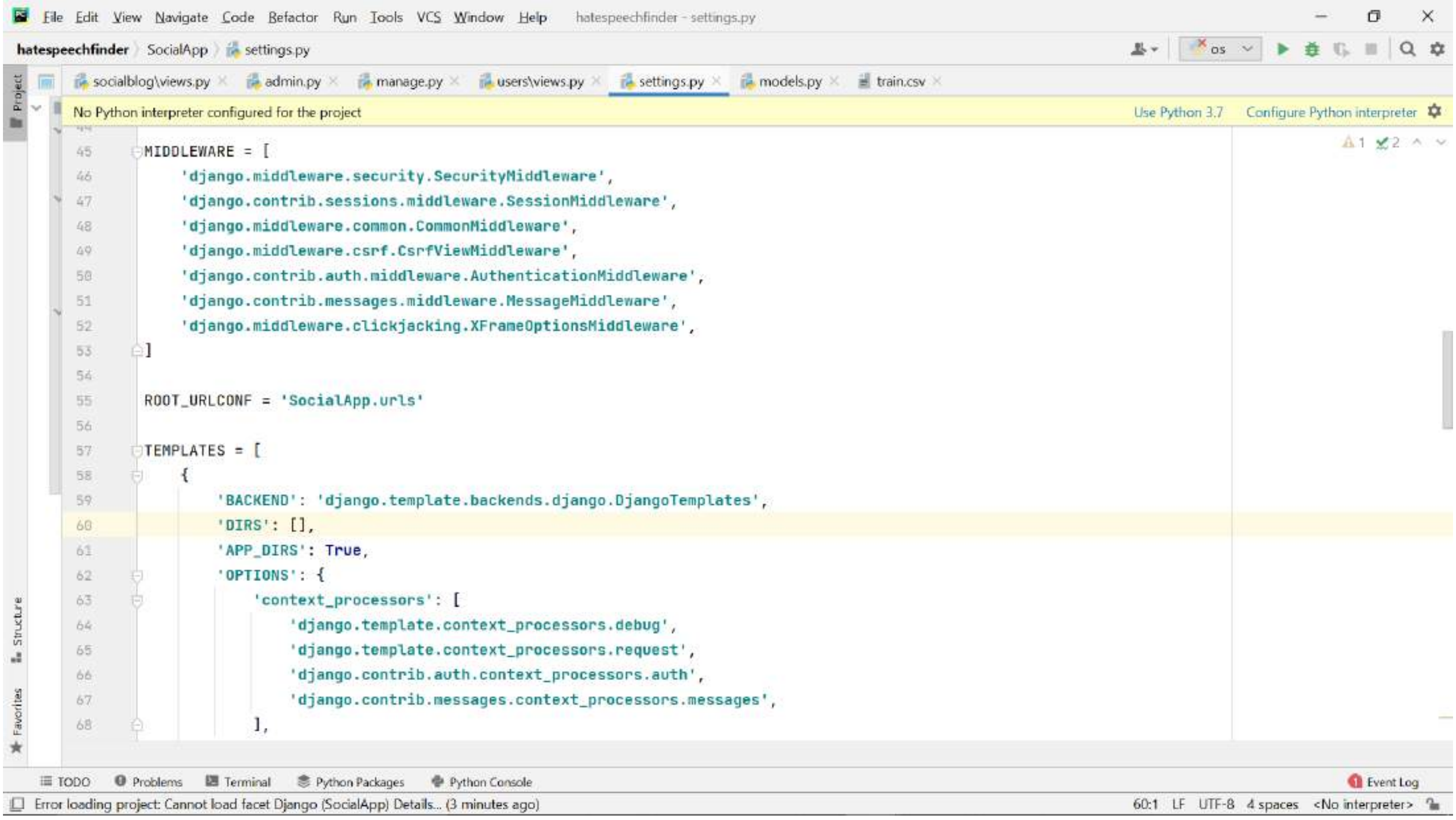
Error loading project: Cannot load facet Django (SocialApp) Details... (3 minutes ago)


File Edit View Navigate Code Refactor Run Tools VCS Window Help hatespeechfinder - settings.py
hatespeechfinder 〉 SocialApp 〉 settings.py
os
socialblog\views.py
admin.py
manage.py
users\views.py
settings.py
models.py
train.csv
Project
Structure
Favorites
TODO
Problems
Terminal
Python Packages
Python Console
Event Log
60:1 LF UTF-8 4 spaces <No interpreter>

No Python interpreter configured for the project   Use Python 3.7   Configure Python interpreter

```
# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = True

ALLOWED_HOSTS = ["0.0.0.0", '192.168.1.7', '127.0.0.1']


# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'socialblog.apps.SocialblogConfig',
    'users.apps.UsersConfig',
    'crispy_forms',
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
```

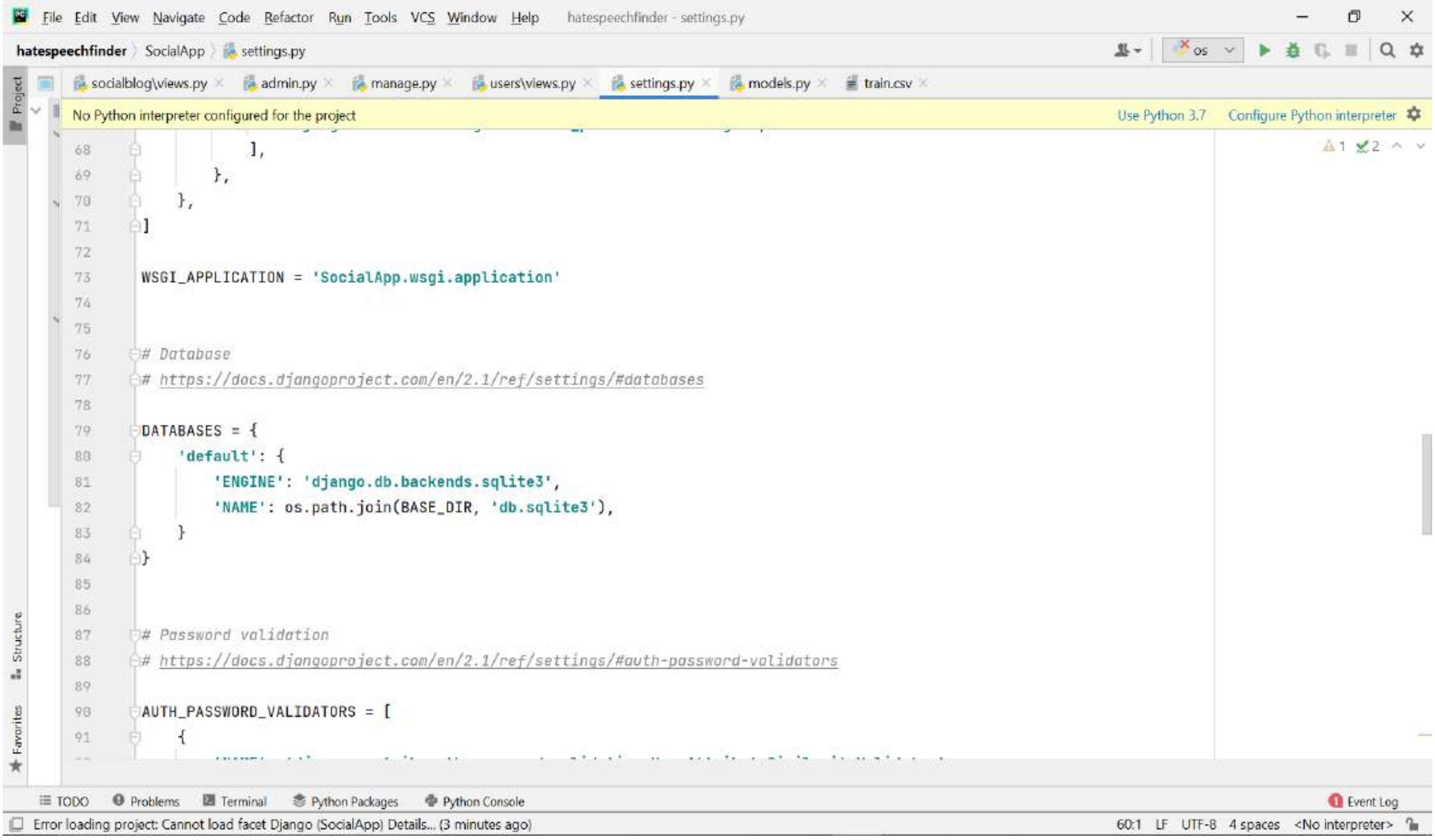
Error loading project: Cannot load facet Django (SocialApp) Details... (3 minutes ago)


File Edit View Navigate Code Refactor Run Tools VCS Window Help
hatespeechfinder - settings.py
hatespeechfinder SocialApp settings.py
socialblog\views.py admin.py manage.py users\views.py settings.py models.py train.csv
Use Python 3.7 Configure Python interpreter
TODO Problems Terminal Python Packages Python Console Event Log
60:1 LF UTF-8 4 spaces <No interpreter>

No Python interpreter configured for the project

```
MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

ROOT_URLCONF = 'SocialApp.urls'

TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [],
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages',
            ],
```

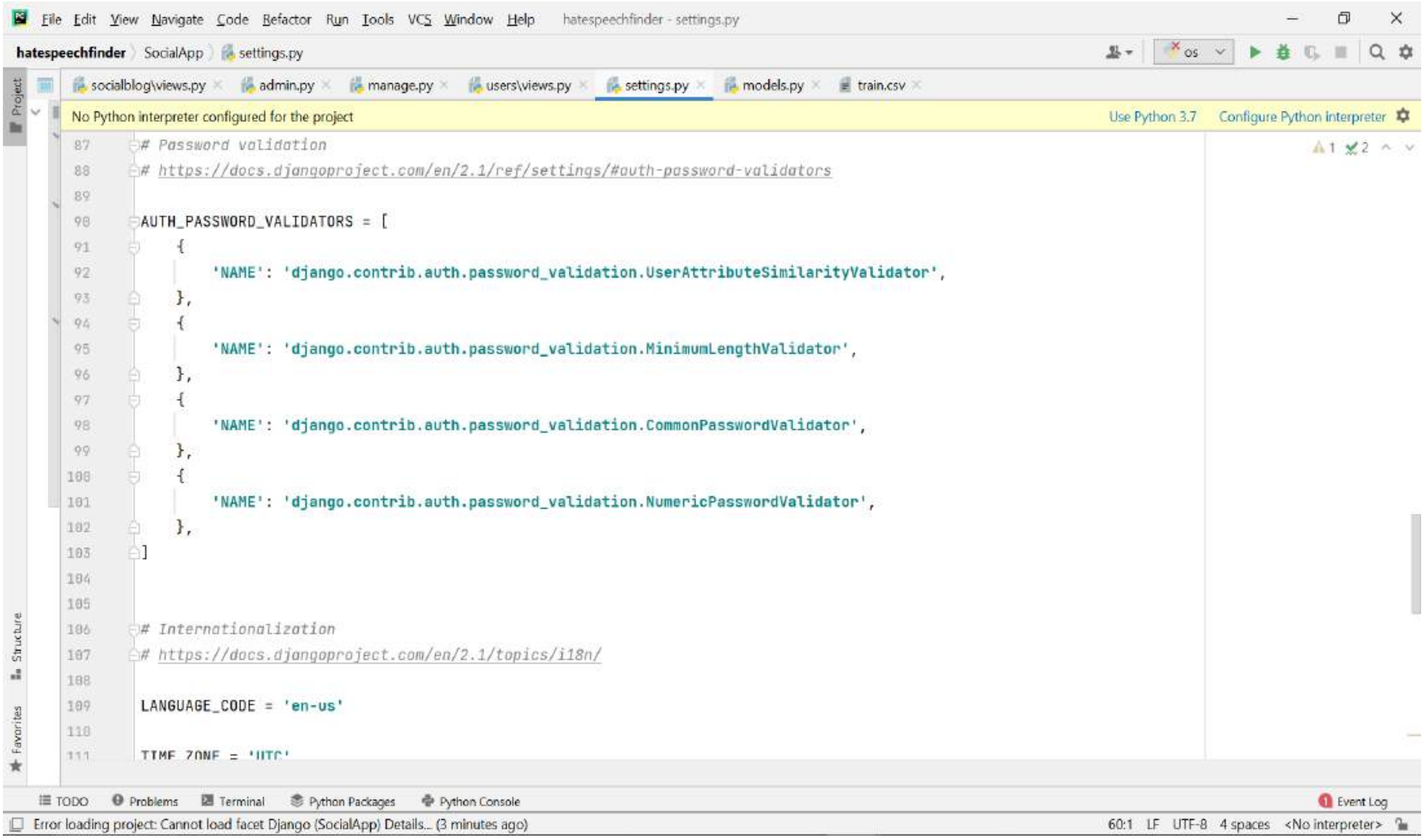
Error loading project: Cannot load facet Django (SocialApp) Details... (3 minutes ago)


File Edit View Navigate Code Refactor Run Tools VCS Window Help
hatespeechfinder - settings.py
hatespeechfinder SocialApp settings.py
socialblog\views.py admin.py manage.py users\views.py settings.py models.py train.csv
Use Python 3.7 Configure Python interpreter
TODO Problems Terminal Python Packages Python Console Event Log
60:1 LF UTF-8 4 spaces <No interpreter>

No Python interpreter configured for the project

```
            ],
        },
    },
]

WSGI_APPLICATION = 'SocialApp.wsgi.application'


# Database
# https://docs.djangoproject.com/en/2.1/ref/settings/#databases

DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.sqlite3',
        'NAME': os.path.join(BASE_DIR, 'db.sqlite3'),
    }
}


# Password validation
# https://docs.djangoproject.com/en/2.1/ref/settings/#auth-password-validators

AUTH_PASSWORD_VALIDATORS = [
    {
```

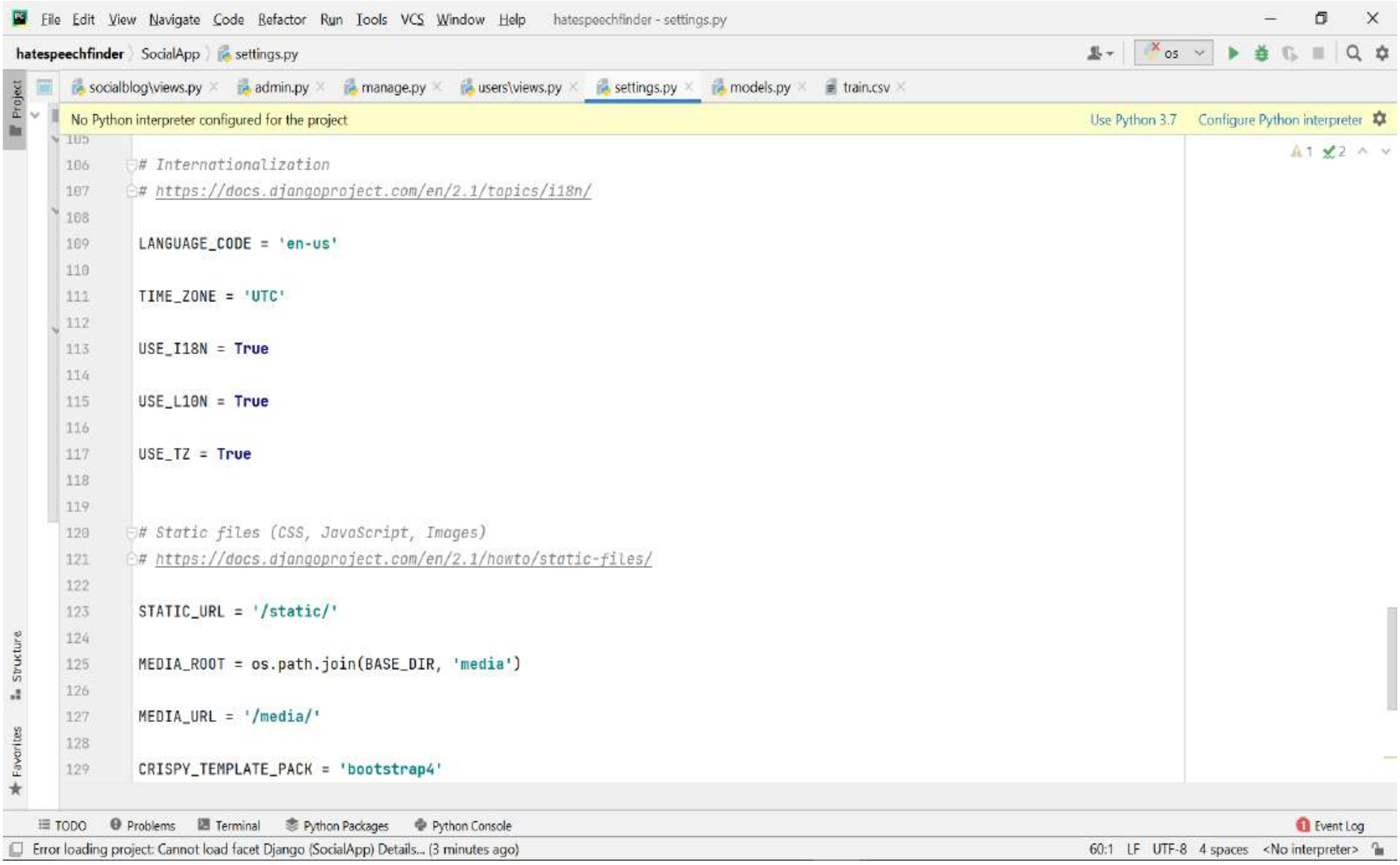
Error loading project: Cannot load facet Django (SocialApp) Details... (3 minutes ago)


File Edit View Navigate Code Refactor Run Tools VCS Window Help
hatespeechfinder - settings.py
hatespeechfinder › SocialApp › settings.py
socialblog\views.py
admin.py
manage.py
users\views.py
settings.py
models.py
train.csv
Use Python 3.7
Configure Python interpreter
TODO
Problems
Terminal
Python Packages
Python Console
Event Log
60:1 LF UTF-8 4 spaces <No interpreter>

No Python interpreter configured for the project

```
# Password validation
# https://docs.djangoproject.com/en/2.1/ref/settings/#auth-password-validators

AUTH_PASSWORD_VALIDATORS = [
    {
        'NAME': 'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.MinimumLengthValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.CommonPasswordValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.NumericPasswordValidator',
    },
]


# Internationalization
# https://docs.djangoproject.com/en/2.1/topics/i18n/

LANGUAGE_CODE = 'en-us'

TIME_ZONE = 'UTC'
```

Error loading project: Cannot load facet Django (SocialApp) Details... (3 minutes ago)


File Edit View Navigate Code Refactor Run Tools VCS Window Help
hatespeechfinder - settings.py
hatespeechfinder › SocialApp › settings.py
socialblog\views.py
admin.py
manage.py
users\views.py
settings.py
models.py
train.csv
Use Python 3.7
Configure Python interpreter
TODO
Problems
Terminal
Python Packages
Python Console
Event Log
60:1 LF UTF-8 4 spaces <No interpreter>

No Python interpreter configured for the project

```
# Internationalization
# https://docs.djangoproject.com/en/2.1/topics/i18n/

LANGUAGE_CODE = 'en-us'

TIME_ZONE = 'UTC'

USE_I18N = True

USE_L10N = True

USE_TZ = True


# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/2.1/howto/static-files/

STATIC_URL = '/static/'

MEDIA_ROOT = os.path.join(BASE_DIR, 'media')

MEDIA_URL = '/media/'

CRISPY_TEMPLATE_PACK = 'bootstrap4'
```

Error loading project: Cannot load facet Django (SocialApp) Details... (3 minutes ago)

SYSTEM TESTING

Testing gives an approach to check the usefulness of parts, sub congregations, gatherings or potentially a completed item it is the way toward practicing software with the plan of guaranteeing that the Software framework lives up to its prerequisites and client desires and does not bomb in a wrong way. The project is working in a desired manner or not and also what kind of experience is given to all the user while using the system. So it is mentioned and used some of the methods to determine the system testing part. There are different kinds of tests".

Types of Testing:

There are different types of testing for the system some of them are as follows:

Unit Testing:

It includes the "plan of experiments that support that the inward program rationale is working fittingly and that program information sources produce substantial yields. Every single decision branch and interior code stream should be supported. It is the trying of individual software units of the application. It is done after the finishing of an individual unit prior to joining. This is secondary testing that depends on the information of its development and is prominent. Unit tests perform necessary tests at part level and test a specific business strategy, application, and framework design. Unit tests guarantee that every novel method of a business methodology performs precisely to the revealed points of interest and contains unmistakably characterized wellsprings of info and anticipated results".

Integration Testing:

Organization examinations ware planned to examine combined "software segments to decide whether they keep running as one program. Testing is occasion-driven and is increasingly stressed over the real consequence of screens or fields. Coordination tests display that even however the segments were only fulfillment, as showed up by successfully unit testing, the blend of segments is correct and dependable". Coordination testing is unequivocally planned in the removal of all the problems which th system uses while the system tends to do the part by part testing.

Functional examination is focused on the supplementary stuffs:

Real input: recognized lessons from important information should be acknowledged.

Capacities: recognized capacities should be worked out.

Invalid input: Well-known lessons in inacceptable information should be dismissed.

Produce: It predictable lessons of use yields should be functioned.

Procedures: Systems should be summoned.

The Suggestion along with the arrangement in the useful examinations are "centered on prerequisites, key capacities, or uncommon experiments. Moreover, efficient inclusion relating to recognizing Business strategy streams, information fields, pre-defined procedures, and advanced procedures should be considered for testing. Before functional testing is finished, additional tests are recognized, and the possible estimation of current tests is settled".

The system test

The designed system should be able to meet the framework which is made by the creators as mentioned above and it should meet the users prospective as they will be the end user of software outline which encounters fundamentals. "It tests a design to guarantee known and unsurprising results. A case of framework testing is the design arranged framework reconciliation test". Context analysis rest on on system depictions and transfers, underscoring already driven strategy influences and combination efforts.

White Box Testing

It is a "trying wherein in which the software analyzer knows about the inward operations, design and language of the software, or if nothing else its motivation. It is the reason. It is used to test regions that can't become to from a discovery level".

Block Testing

Detection Analysis is using the software through amount info within the module's interior processes, design, or linguistic existence pushed. Detection examinations is termed as dissimilar kinds of examinations, should remain collected of a definitive foundation record, instance, specific report, for instance, detail archive wherein the software in examination is managed, with detection .

Unit Testing:

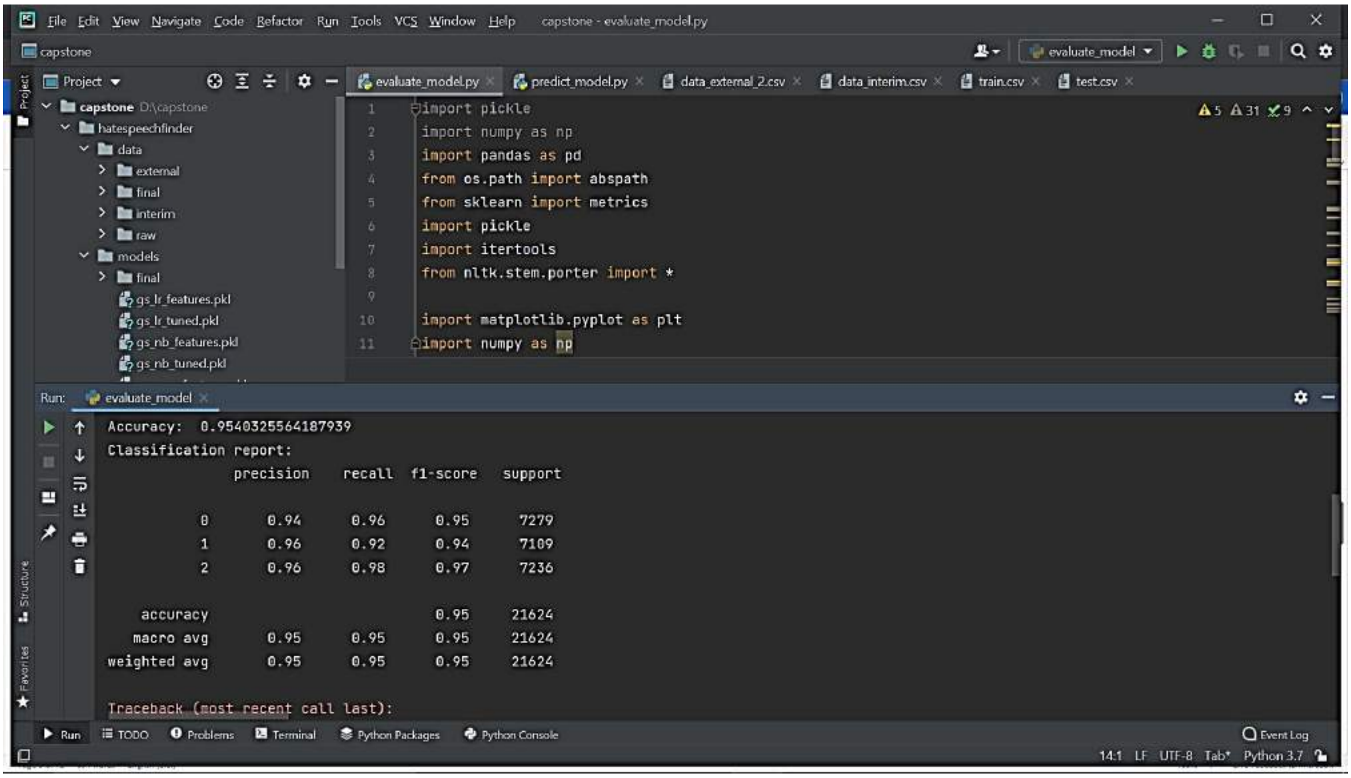
"Unit testing is normally led as a feature of a joined code and unit test period of the software lifecycle, even though it isn't remarkable for coding and unit testing to be directed as two unmistakable stages."

These are the modules on which system worked in the following project:

- User Authentication
- Blog Posting and Viewing
- Implement the ML Algorithm, i.e., Logistic Regression, to find the offensive words
- Validate the results

Screenshots

The code Accuracy, F1 score, precision and recall score with relation to all the datasets used for the system.

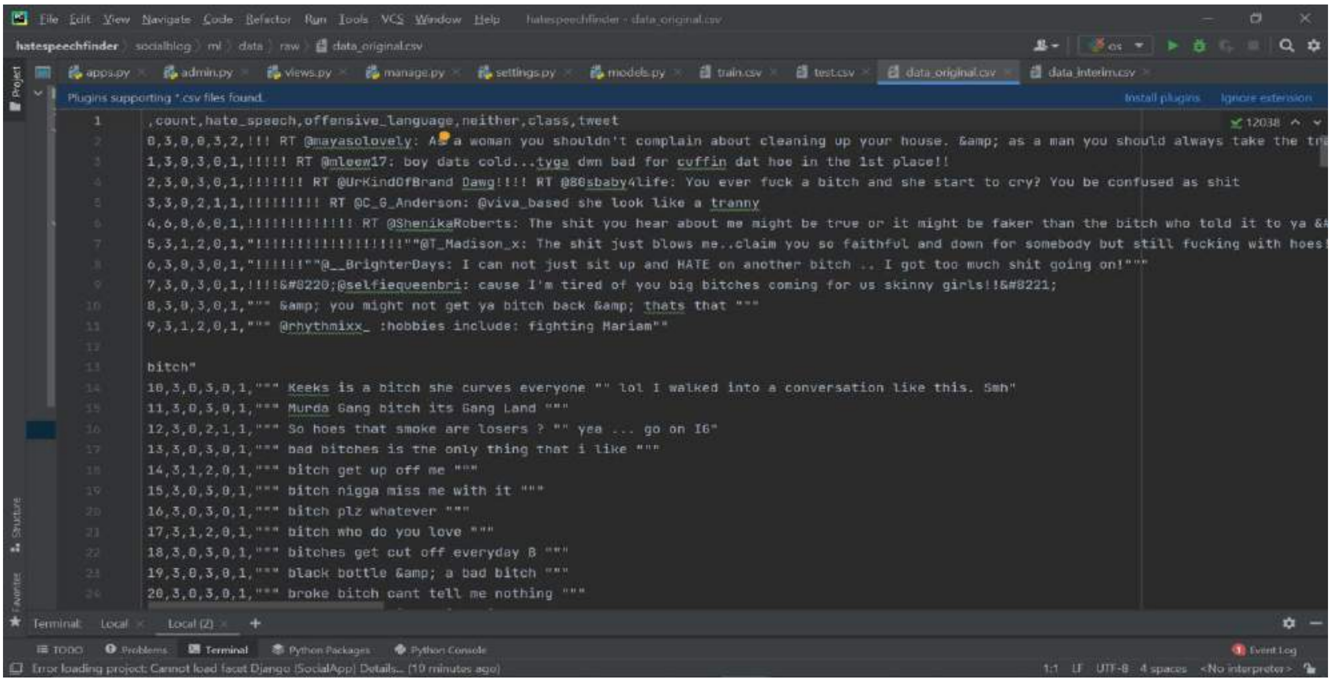


```
import pickle
import numpy as np
import pandas as pd
from os.path import abspath
from sklearn import metrics
import pickle
import itertools
from nltk.stem.porter import *

import matplotlib.pyplot as plt
import numpy as np
```

```
Accuracy:  0.9540325564187939
Classification report:
               precision    recall  f1-score   support

           0       0.94      0.96      0.95      7279
           1       0.96      0.92      0.94      7109
           2       0.96      0.98      0.97      7236

    accuracy                           0.95     21624
   macro avg       0.95      0.95      0.95     21624
weighted avg       0.95      0.95      0.95     21624

Traceback (most recent call last):
```

Fig.9: Accuracy of F1 score

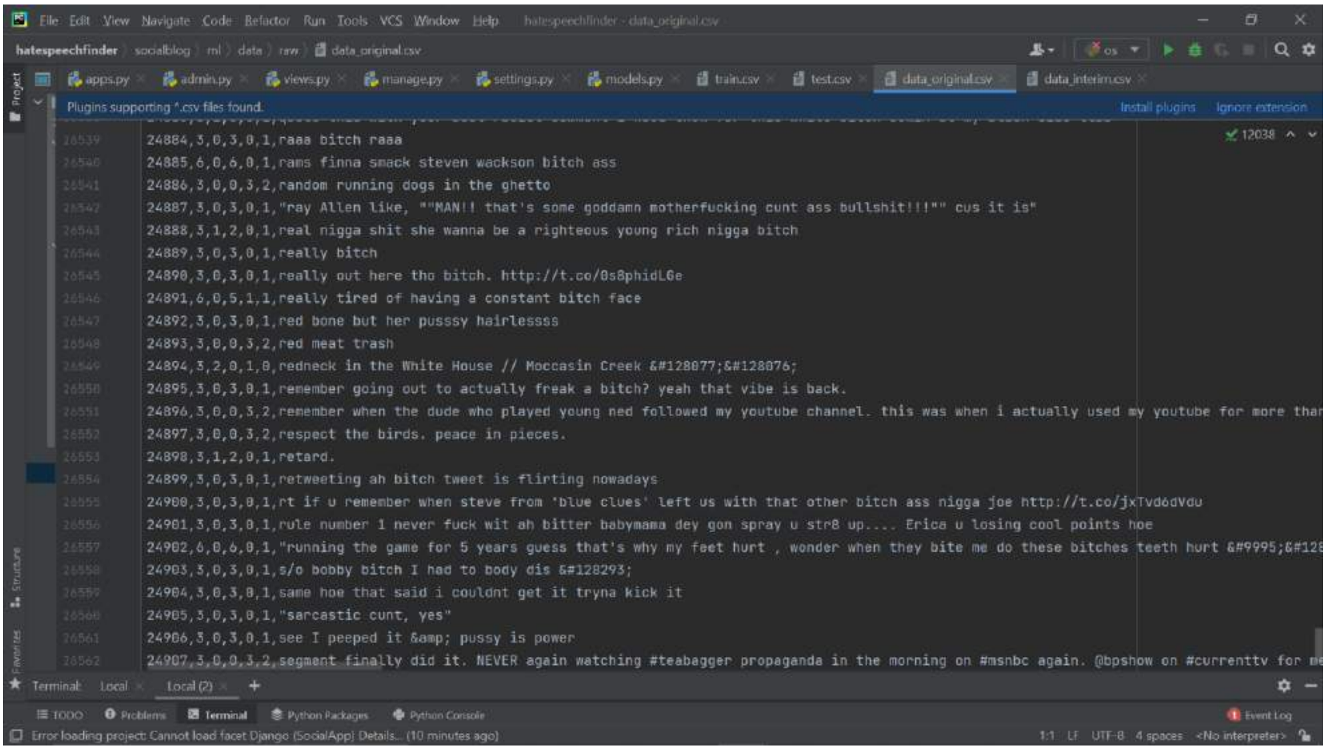
Raw Data used



File Edit View Navigate Code Refactor Run Tools VCS Window Help hatespeechfinder - data_original.csv
hatespeechfinder socialblog ml data raw data_original.csv
apps.py admin.py views.py manage.py settings.py models.py train.csv test.csv data_original.csv data_interim.csv
Plugins supporting *.csv files found.
Install plugins
Ignore extension
12038
Terminal: Local Local (2)
TODO Problems Terminal Python Packages Python Console Event Log
1:1 LF UTF-8 4 spaces <No interpreter>

```
,count,hate_speech,offensive_language,neither,class,tweet
0,3,0,0,3,2,!!! RT @mayasolovely: As a woman you shouldn't complain about cleaning up your house. &amp; as a man you should always take the tr
1,3,0,3,0,1,!!!!! RT @mleew17: boy dats cold...tyga dwn bad for cuffin dat hoe in the 1st place!!
2,3,0,3,0,1,!!!!!!! RT @UrKindOfBrand Dawg!!!! RT @80sbaby4life: You ever fuck a bitch and she start to cry? You be confused as shit
3,3,0,2,1,1,!!!!!!!!! RT @C_G_Anderson: @viva_based she look like a tranny
4,6,0,6,0,1,!!!!!!!!!!!!! RT @ShenikaRoberts: The shit you hear about me might be true or it might be faker than the bitch who told it to ya &
5,3,1,2,0,1,"!!!!!!!!!!!!!!!!!""@T_Madison_x: The shit just blows me..claim you so faithful and down for somebody but still fucking with hoes
6,3,0,3,0,1,"!!!!!!""@__BrighterDays: I can not just sit up and HATE on another bitch .. I got too much shit going on!"""
7,3,0,3,0,1,!!!!&#8220;@selfiequeenbri: cause I'm tired of you big bitches coming for us skinny girls!!&#8221;
8,3,0,3,0,1,""" &amp; you might not get ya bitch back &amp; thats that """
9,3,1,2,0,1,""" @rhythmixx_ :hobbies include: fighting Mariam""

bitch"
10,3,0,3,0,1,""" Keeks is a bitch she curves everyone "" lol I walked into a conversation like this. Smh"
11,3,0,3,0,1,""" Murda Gang bitch its Gang Land """
12,3,0,2,1,1,""" So hoes that smoke are losers ? "" yea ... go on IG"
13,3,0,3,0,1,""" bad bitches is the only thing that i like """
14,3,1,2,0,1,""" bitch get up off me """
15,3,0,3,0,1,""" bitch nigga miss me with it """
16,3,0,3,0,1,""" bitch plz whatever """
17,3,1,2,0,1,""" bitch who do you love """
18,3,0,3,0,1,""" bitches get cut off everyday B """
19,3,0,3,0,1,""" black bottle &amp; a bad bitch """
20,3,0,3,0,1,""" broke bitch cant tell me nothing """
```

Error loading project: Cannot load facet Django (SocialApp) Details... (10 minutes ago)



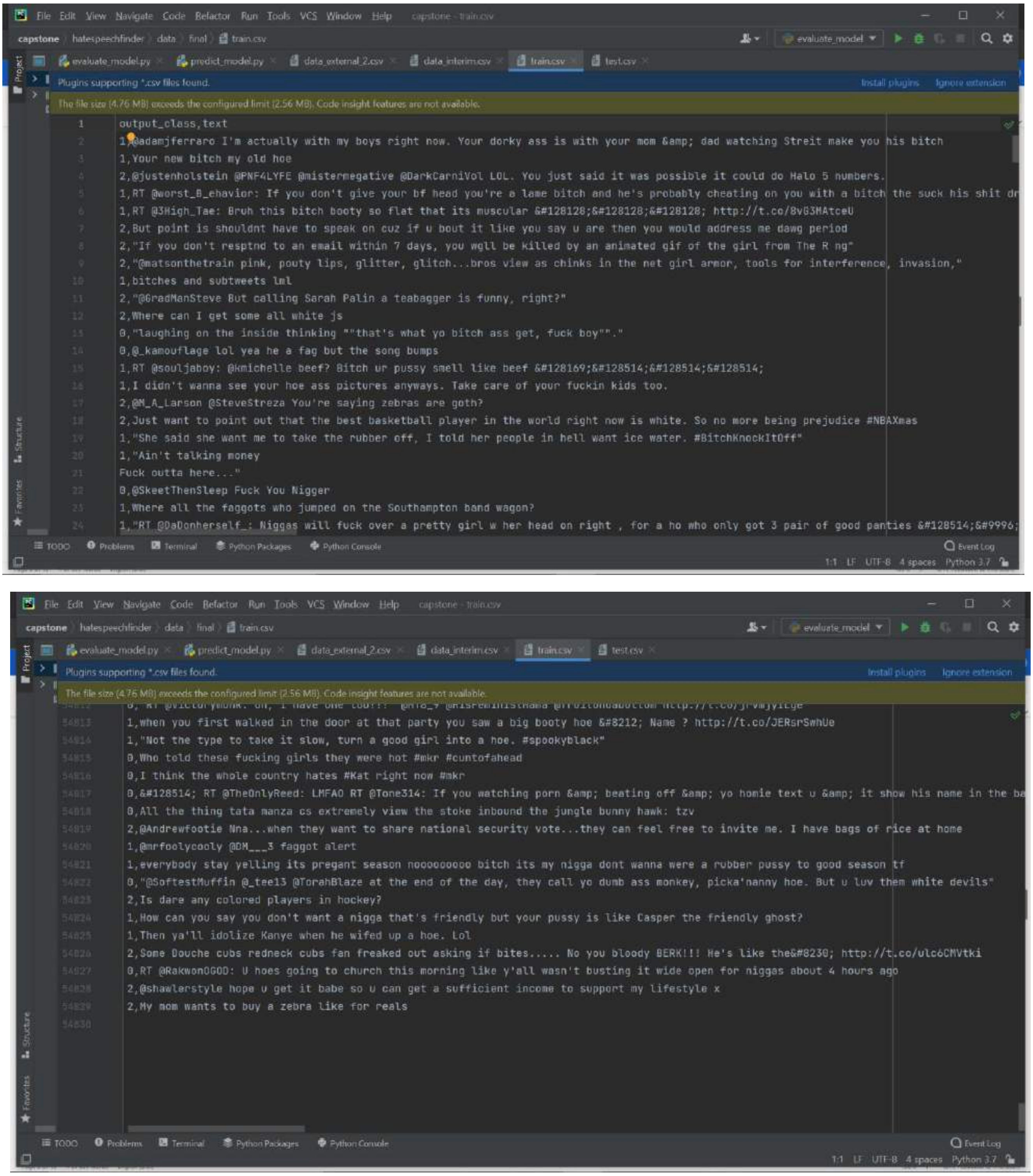
File Edit View Navigate Code Refactor Run Tools VCS Window Help hatespeechfinder - data_original.csv
hatespeechfinder socialblog ml data raw data_original.csv
apps.py admin.py views.py manage.py settings.py models.py train.csv test.csv data_original.csv data_interim.csv
Plugins supporting *.csv files found.
Install plugins
Ignore extension
12038
Terminal: Local Local (2)
TODO Problems Terminal Python Packages Python Console Event Log
1:1 LF UTF-8 4 spaces <No interpreter>

```
24884,3,0,3,0,1,raaa bitch raaa
24885,6,0,6,0,1,rams finna smack steven wackson bitch ass
24886,3,0,0,3,2,random running dogs in the ghetto
24887,3,0,3,0,1,"ray Allen like, ""MAN!! that's some goddamn motherfucking cunt ass bullshit!!!"" cus it is"
24888,3,1,2,0,1,real nigga shit she wanna be a righteous young rich nigga bitch
24889,3,0,3,0,1,really bitch
24890,3,0,3,0,1,really out here tho bitch. http://t.co/0s8phidL6e
24891,6,0,5,1,1,really tired of having a constant bitch face
24892,3,0,3,0,1,red bone but her pusssy hairlessss
24893,3,0,0,3,2,red meat trash
24894,3,2,0,1,0,redneck in the White House // Moccasin Creek &#128077;&#128076;
24895,3,0,3,0,1,remember going out to actually freak a bitch? yeah that vibe is back.
24896,3,0,0,3,2,remember when the dude who played young ned followed my youtube channel. this was when i actually used my youtube for more thar
24897,3,0,0,3,2,respect the birds. peace in pieces.
24898,3,1,2,0,1,retard.
24899,3,0,3,0,1,retweeting ah bitch tweet is flirting nowadays
24900,3,0,3,0,1,rt if u remember when steve from 'blue clues' left us with that other bitch ass nigga joe http://t.co/jxTvd6dVdu
24901,3,0,3,0,1,rule number 1 never fuck wit ah bitter babymama dey gon spray u str8 up.... Erica u losing cool points hoe
24902,6,0,6,0,1,"running the game for 5 years guess that's why my feet hurt , wonder when they bite me do these bitches teeth hurt &#9995;&#128
24903,3,0,3,0,1,s/o bobby bitch I had to body dis &#128293;
24904,3,0,3,0,1,same hoe that said i couldnt get it tryna kick it
24905,3,0,3,0,1,"sarcastic cunt, yes"
24906,3,0,3,0,1,see I peeped it &amp; pussy is power
24907,3,0,0,3,2,segment finally did it. NEVER again watching #teabagger propaganda in the morning on #msnbc again. @bpshow on #currenttv for me
```

Error loading project: Cannot load facet Django (SocialApp) Details... (10 minutes ago)

Fig. 10: raw data used in the system

Training datasets with 50,000+lines web scraped



Plugins supporting *.csv files found.

The file size (4.76 MB) exceeds the configured limit (2.56 MB). Code insight features are not available.

```
output_class,text
1,@adamjferraro I'm actually with my boys right now. Your dorky ass is with your mom &amp; dad watching Streit make you his bitch
1,Your new bitch my old hoe
2,@justenholstein @PNF4LYFE @mistermegative @DarkCarniVol LOL. You just said it was possible it could do Halo 5 numbers.
1,RT @worst_B_ehavior: If you don't give your bf head you're a lame bitch and he's probably cheating on you with a bitch the suck his shit dr
1,RT @3High_Tae: Bruh this bitch booty so flat that its muscular &#128128;&#128128;&#128128; http://t.co/8vG3MAtceU
2,But point is shouldnt have to speak on cuz if u bout it like you say u are then you would address me dawg period
2,"If you don't resptnd to an email within 7 days, you wqll be killed by an animated gif of the girl from The R ng"
2,"@matsonthetrain pink, pouty lips, glitter, glitch...bros view as chinks in the net girl armor, tools for interference, invasion,"
1,bitches and subtweets lml
2,"@GradManSteve But calling Sarah Palin a teabagger is funny, right?"
2,Where can I get some all white js
0,"laughing on the inside thinking ""that's what yo bitch ass get, fuck boy""."
0,@_kamouflage lol yea he a fag but the song bumps
1,RT @souljaboy: @kmichelle beef? Bitch ur pussy smell like beef &#128169;&#128514;&#128514;&#128514;
1,I didn't wanna see your hoe ass pictures anyways. Take care of your fuckin kids too.
2,@M_A_Larson @SteveStreza You're saying zebras are goth?
2,Just want to point out that the best basketball player in the world right now is white. So no more being prejudice #NBAXmas
1,"She said she want me to take the rubber off, I told her people in hell want ice water. #BitchKnockItOff"
1,"Ain't talking money
Fuck outta here..."
0,@SkeetThenSleep Fuck You Nigger
1,Where all the faggots who jumped on the Southampton band wagon?
1,"RT @DaDonherself_: Niggas will fuck over a pretty girl w her head on right , for a ho who only got 3 pair of good panties &#128514;&#9996;
```

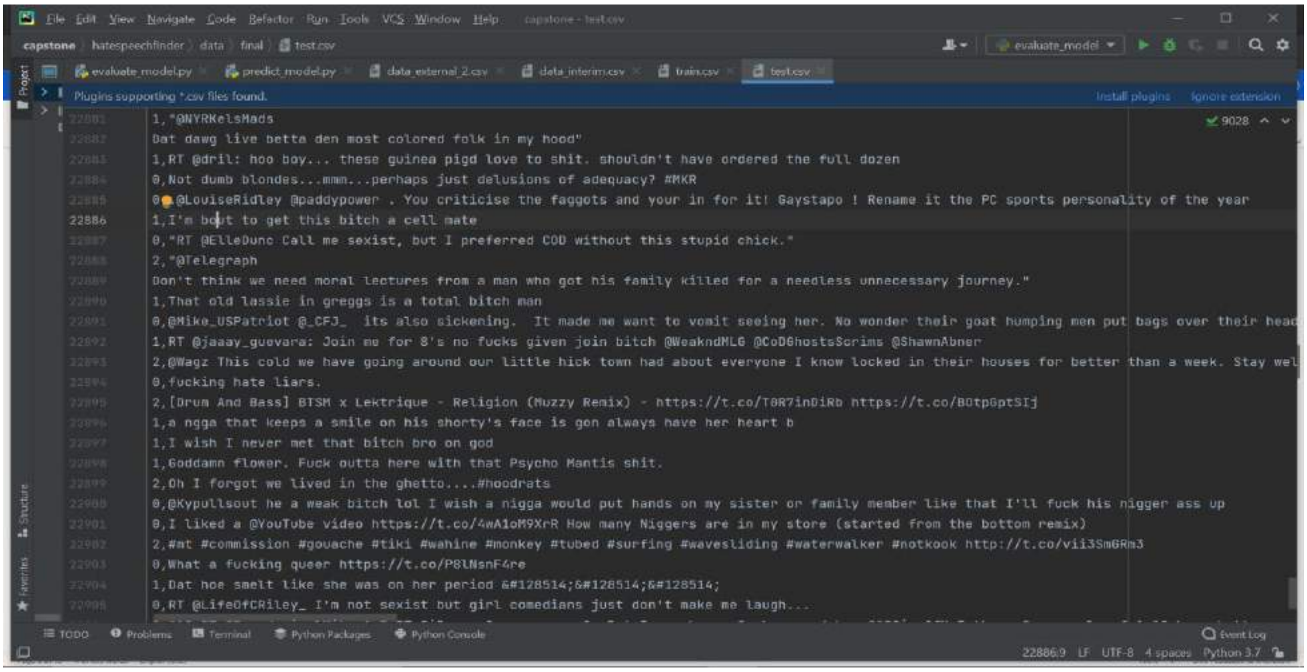
Plugins supporting *.csv files found.

The file size (4.76 MB) exceeds the configured limit (2.56 MB). Code insight features are not available.

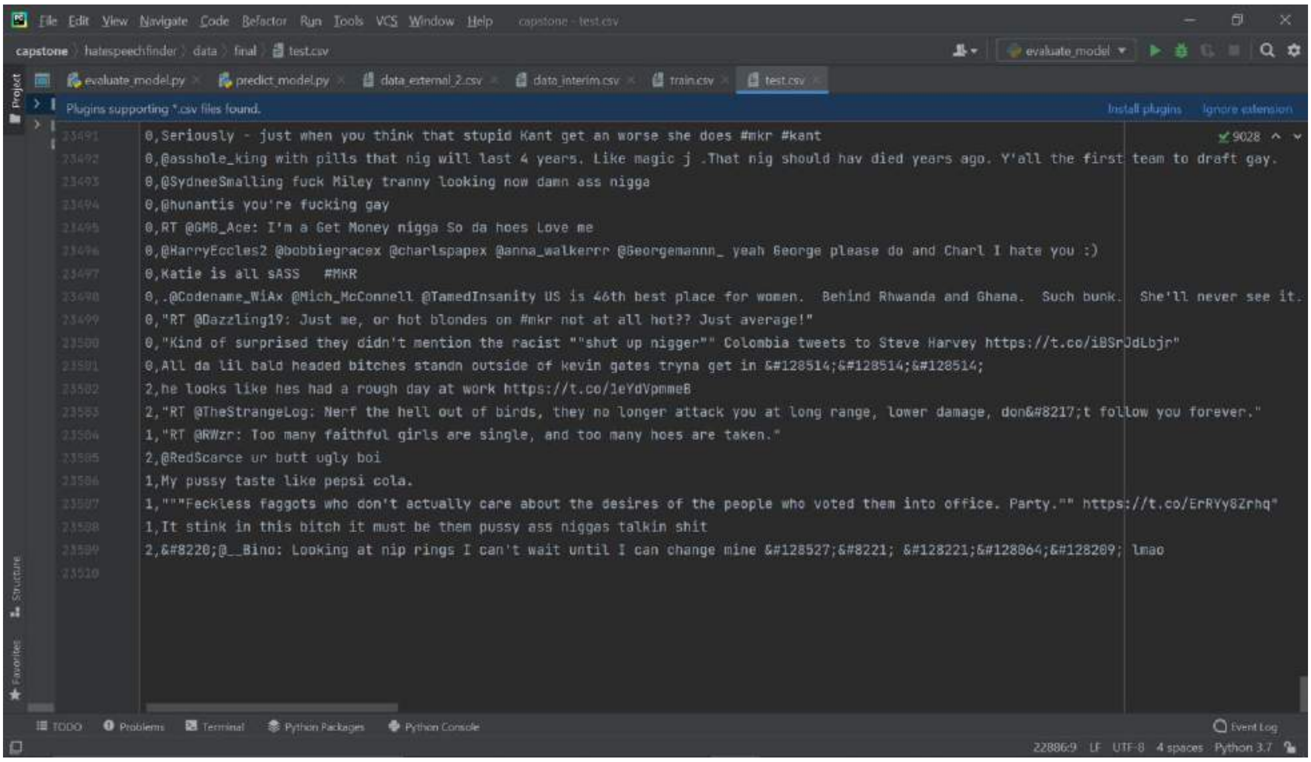
```
1,when you first walked in the door at that party you saw a big booty hoe &#8212; Name ? http://t.co/JERsrSwhUe
1,"Not the type to take it slow, turn a good girl into a hoe. #spookyblack"
0,Who told these fucking girls they were hot #mkr #cuntofahead
0,I think the whole country hates #Kat right now #mkr
0,&#128514; RT @TheOnlyReed: LMFAO RT @Tone314: If you watching porn &amp; beating off &amp; yo homie text u &amp; it show his name in the ba
0,All the thing tata manza cs extremely view the stoke inbound the jungle bunny hawk: tzv
2,@Andrewfootie Nna...when they want to share national security vote...they can feel free to invite me. I have bags of rice at home
1,@mrfoolycooly @DM___3 faggot alert
1,everybody stay yelling its pregant season noooooooooo bitch its my nigga dont wanna were a rubber pussy to good season tf
0,"@SoftestMuffin @_tee13 @TorahBlaze at the end of the day, they call yo dumb ass monkey, picka'nanny hoe. But u luv them white devils"
2,Is dare any colored players in hockey?
1,How can you say you don't want a nigga that's friendly but your pussy is like Casper the friendly ghost?
1,Then ya'll idolize Kanye when he wifed up a hoe. Lol
2,Some Douche cubs redneck cubs fan freaked out asking if bites..... No you bloody BERK!!! He's like the&#8230; http://t.co/ulc6CMVtki
0,RT @RakwonOGOD: U hoes going to church this morning like y'all wasn't busting it wide open for niggas about 4 hours ago
2,@shawlerstyle hope u get it babe so u can get a sufficient income to support my lifestyle x
2,My mom wants to buy a zebra like for reals
```

Fig.11: Training datasets with 50,000+lines

Testing datasets with 20,000 line plus


File Edit View Navigate Code Refactor Run Tools VCS Window Help capstone - test.csv
capstone hatespeechfinder data final test.csv
evaluate_model
evaluate_model.py predict_model.py data_external_2.csv data_interim.csv train.csv test.csv
Plugins supporting *.csv files found.
Install plugins Ignore extension
9028
TODO Problems Terminal Python Packages Python Console
Event Log
22886:9 LF UTF-8 4 spaces Python 3.7

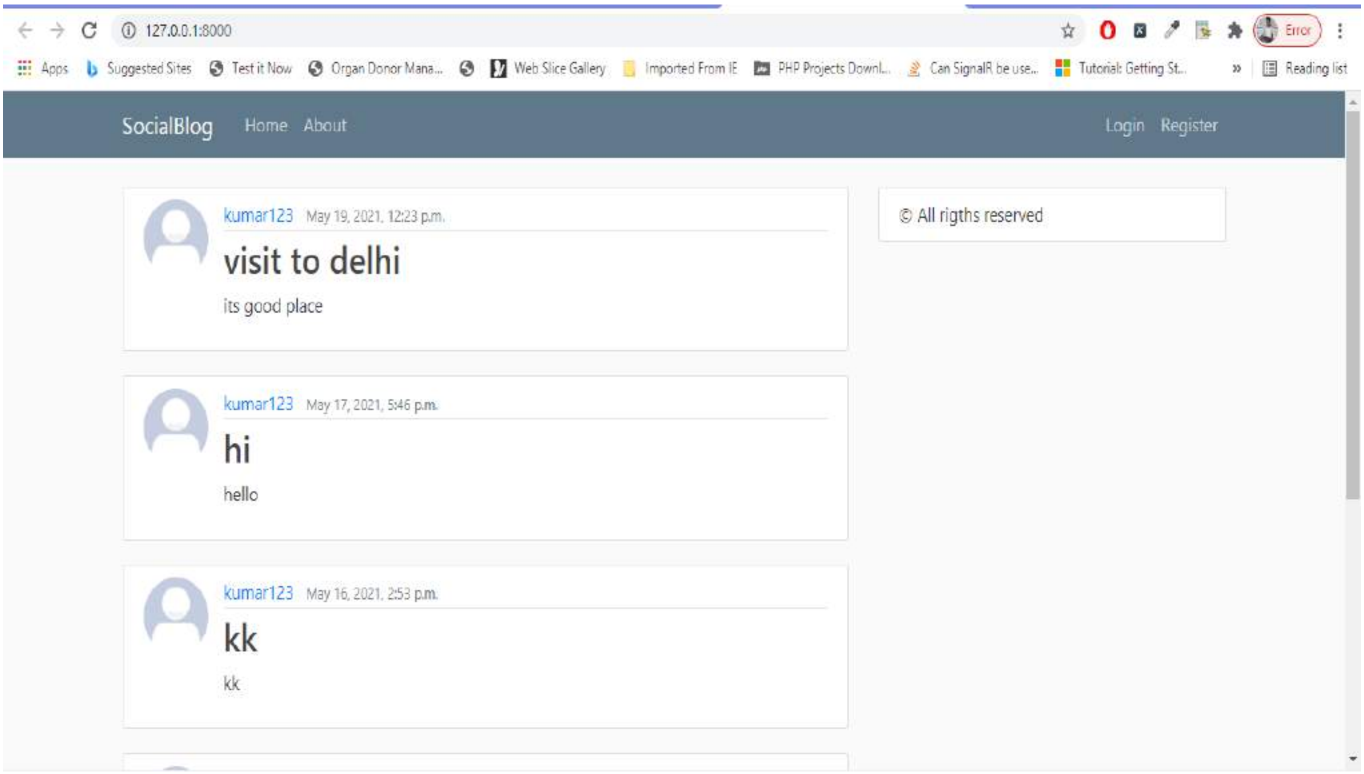
```
1,"@NYRKelsMads
Dat dawg live betta den most colored folk in my hood"
1,RT @dril: hoo boy... these guinea pigd love to shit. shouldn't have ordered the full dozen
0,Not dumb blondes...mmm...perhaps just delusions of adequacy? #MKR
0,@LouiseRidley @paddypower . You criticise the faggots and your in for it! Gaystapo ! Rename it the PC sports personality of the year
1,I'm bout to get this bitch a cell mate
0,"RT @ElleDunc Call me sexist, but I preferred COD without this stupid chick."
2,"@Telegraph
Don't think we need moral lectures from a man who got his family killed for a needless unnecessary journey."
1,That old lassie in greggs is a total bitch man
0,@Mike_USPatriot @_CFJ_  its also sickening.  It made me want to vomit seeing her. No wonder their goat humping men put bags over their head
1,RT @jaaay_guevara: Join me for 8's no fucks given join bitch @WeakndMLG @CoDGhostsScrims @ShawnAbner
2,@Wagz This cold we have going around our little hick town had about everyone I know locked in their houses for better than a week. Stay wel
0,fucking hate liars.
2,[Drum And Bass] BTSM x Lektrique - Religion (Muzzy Remix) - https://t.co/T0R7inDiRb https://t.co/B0tpGptSIj
1,a ngga that keeps a smile on his shorty's face is gon always have her heart b
1,I wish I never met that bitch bro on god
1,Goddamn flower. Fuck outta here with that Psycho Mantis shit.
2,Oh I forgot we lived in the ghetto....#hoodrats
0,@Kypullsout he a weak bitch lol I wish a nigga would put hands on my sister or family member like that I'll fuck his nigger ass up
0,I liked a @YouTube video https://t.co/4wA1oM9XrR How many Niggers are in my store (started from the bottom remix)
2,#mt #commission #gouache #tiki #wahine #monkey #tubed #surfing #wavesliding #waterwalker #notkook http://t.co/vii3SmGRm3
0,What a fucking queer https://t.co/P8lNsnF4re
1,Dat hoe smelt like she was on her period &#128514;&#128514;&#128514;
0,RT @LifeOfCRiley_ I'm not sexist but girl comedians just don't make me laugh...
```


File Edit View Navigate Code Refactor Run Tools VCS Window Help capstone - test.csv
capstone hatespeechfinder data final test.csv
evaluate_model
evaluate_model.py predict_model.py data_external_2.csv data_interim.csv train.csv test.csv
Plugins supporting *.csv files found.
Install plugins Ignore extension
9028
TODO Problems Terminal Python Packages Python Console
Event Log
22886:9 LF UTF-8 4 spaces Python 3.7

```
0,Seriously - just when you think that stupid Kant get an worse she does #mkr #kant
0,@asshole_king with pills that nig will last 4 years. Like magic j .That nig should hav died years ago. Y'all the first team to draft gay.
0,@SydneeSmalling fuck Miley tranny looking now damn ass nigga
0,@hunantis you're fucking gay
0,RT @GMB_Ace: I'm a Get Money nigga So da hoes Love me
0,@HarryEccles2 @bobbiegracex @charlspapex @anna_walkerrr @Georgemannn_ yeah George please do and Charl I hate you :)
0,Katie is all sASS   #MKR
0,.@Codename_WiAx @Mich_McConnell @TamedInsanity US is 46th best place for women.  Behind Rhwanda and Ghana.  Such bunk.  She'll never see it.
0,"RT @Dazzling19: Just me, or hot blondes on #mkr not at all hot?? Just average!"
0,"Kind of surprised they didn't mention the racist ""shut up nigger"" Colombia tweets to Steve Harvey https://t.co/iBSrJdLbjr"
0,All da lil bald headed bitches standn outside of kevin gates tryna get in &#128514;&#128514;&#128514;
2,he looks like hes had a rough day at work https://t.co/1eYdVpmmeB
2,"RT @TheStrangeLog: Nerf the hell out of birds, they no longer attack you at long range, lower damage, don&#8217;t follow you forever."
1,"RT @RWzr: Too many faithful girls are single, and too many hoes are taken."
2,@RedScarce ur butt ugly boi
1,My pussy taste like pepsi cola.
1,"""Feckless faggots who don't actually care about the desires of the people who voted them into office. Party."" https://t.co/ErRYy8Zrhq"
1,It stink in this bitch it must be them pussy ass niggas talkin shit
2,&#8220;@__Bino: Looking at nip rings I can't wait until I can change mine &#128527;&#8221; &#128221;&#128064;&#128209; lmao
```

Fig.12: Testing datasets with 20,000 lines

- The website front view



kumar123 May 19, 2021, 12:23 p.m.

visit to delhi

its good place

kumar123 May 17, 2021, 5:46 p.m.

hi

hello

kumar123 May 16, 2021, 2:53 p.m.

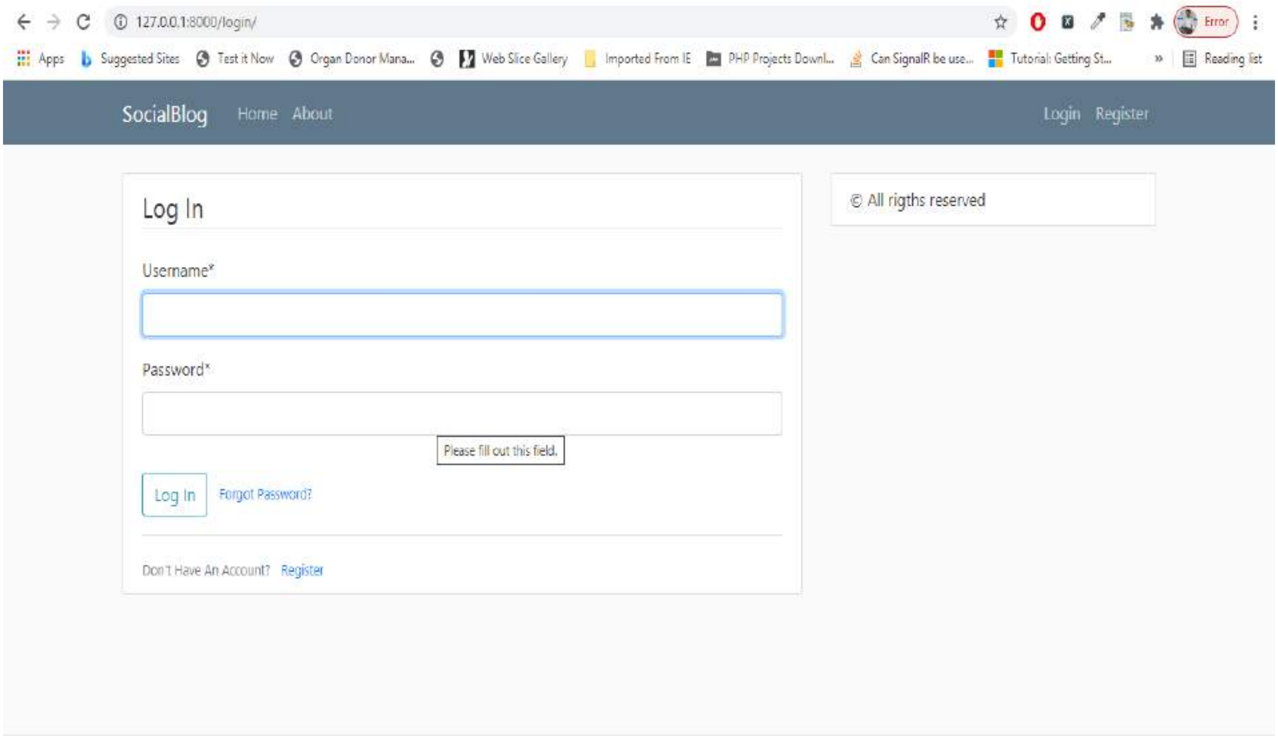
kk

kk

© All rigths reserved

Fig.13: website front view

- Login Page



Log In

Username*

Password*

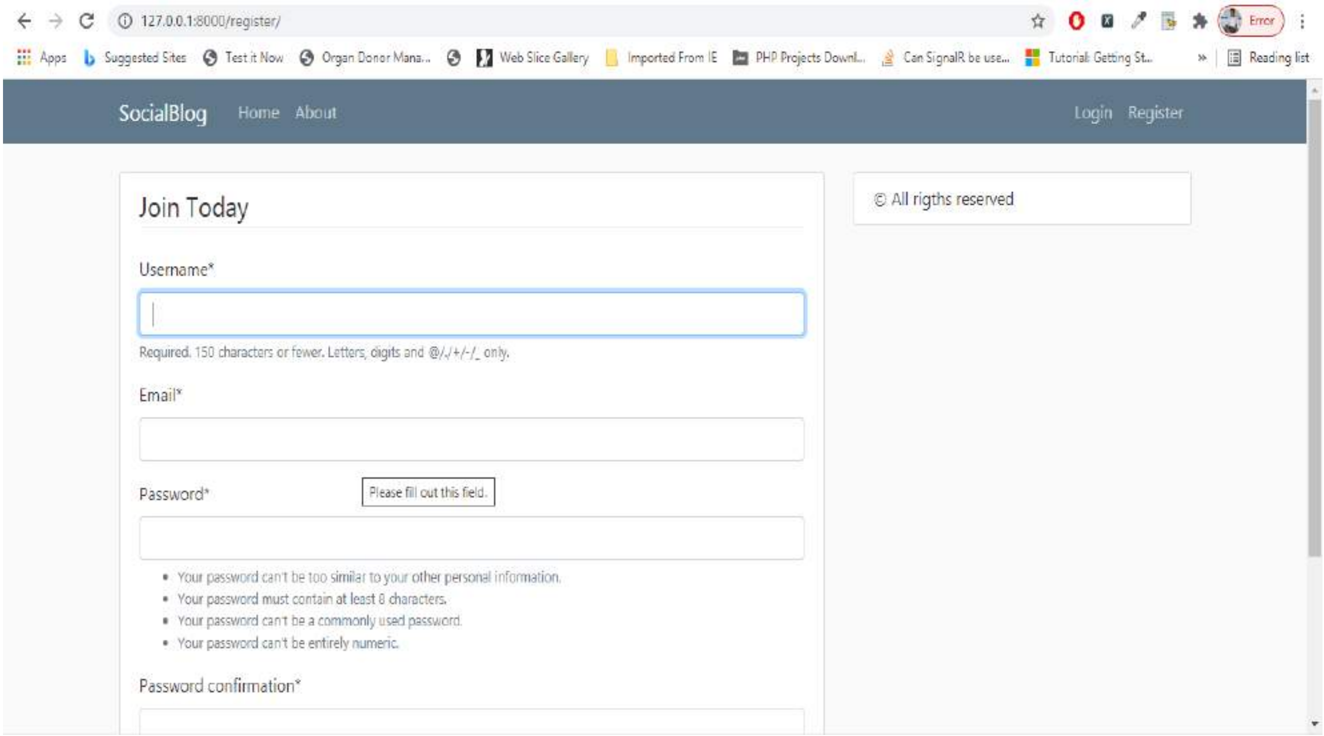
Please fill out this field.

Don't Have An Account? Register

© All rigths reserved

Fig.14: login page of website

- Register page for user



Join Today

Username*

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Email*

Password*

Please fill out this field.

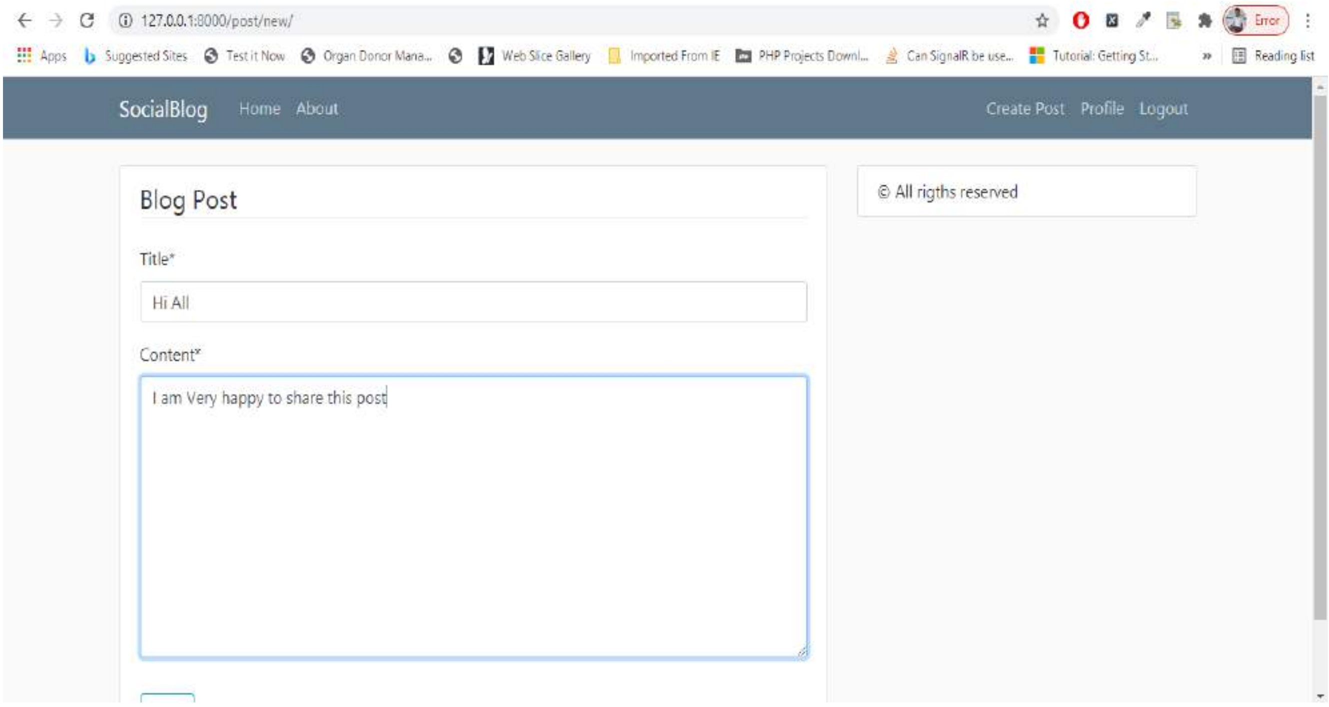
- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

Password confirmation*

© All rigths reserved

Fig.15: register page for website

- Post content



Blog Post

Title*

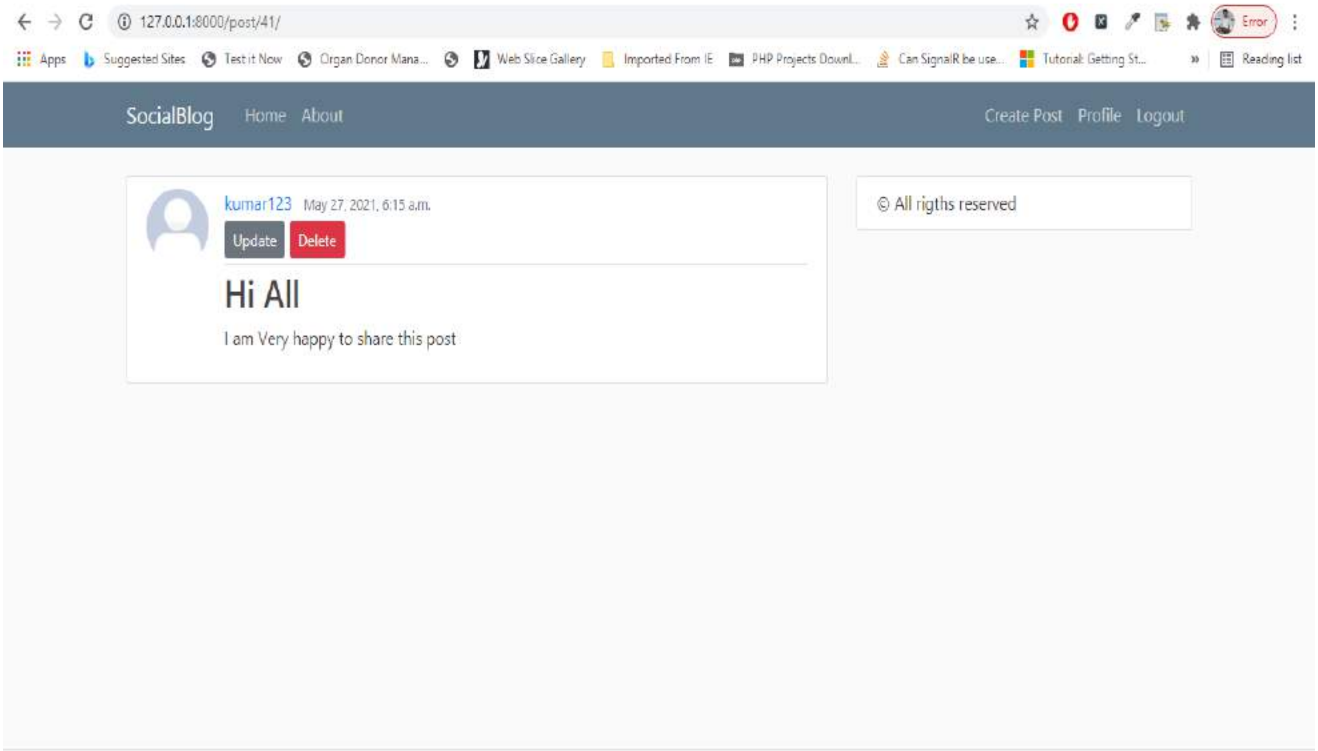
Hi All

Content*

I am Very happy to share this post

© All rigths reserved

Fig.16: Posting a comment



kumar123 May 27, 2021, 6:15 a.m.

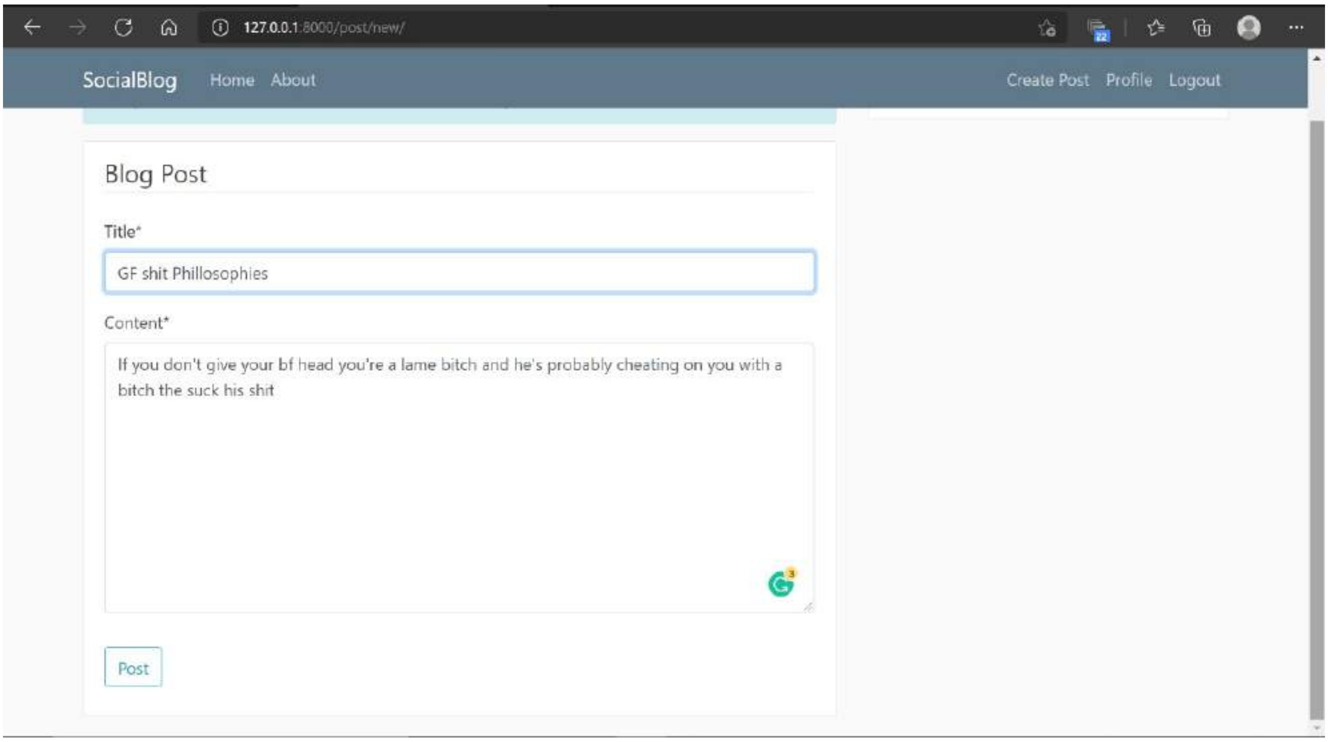
Hi All

I am Very happy to share this post

© All rigths reserved

Fig.17: Non-offensive content posted

- Trying Offensive post



Blog Post

Title*

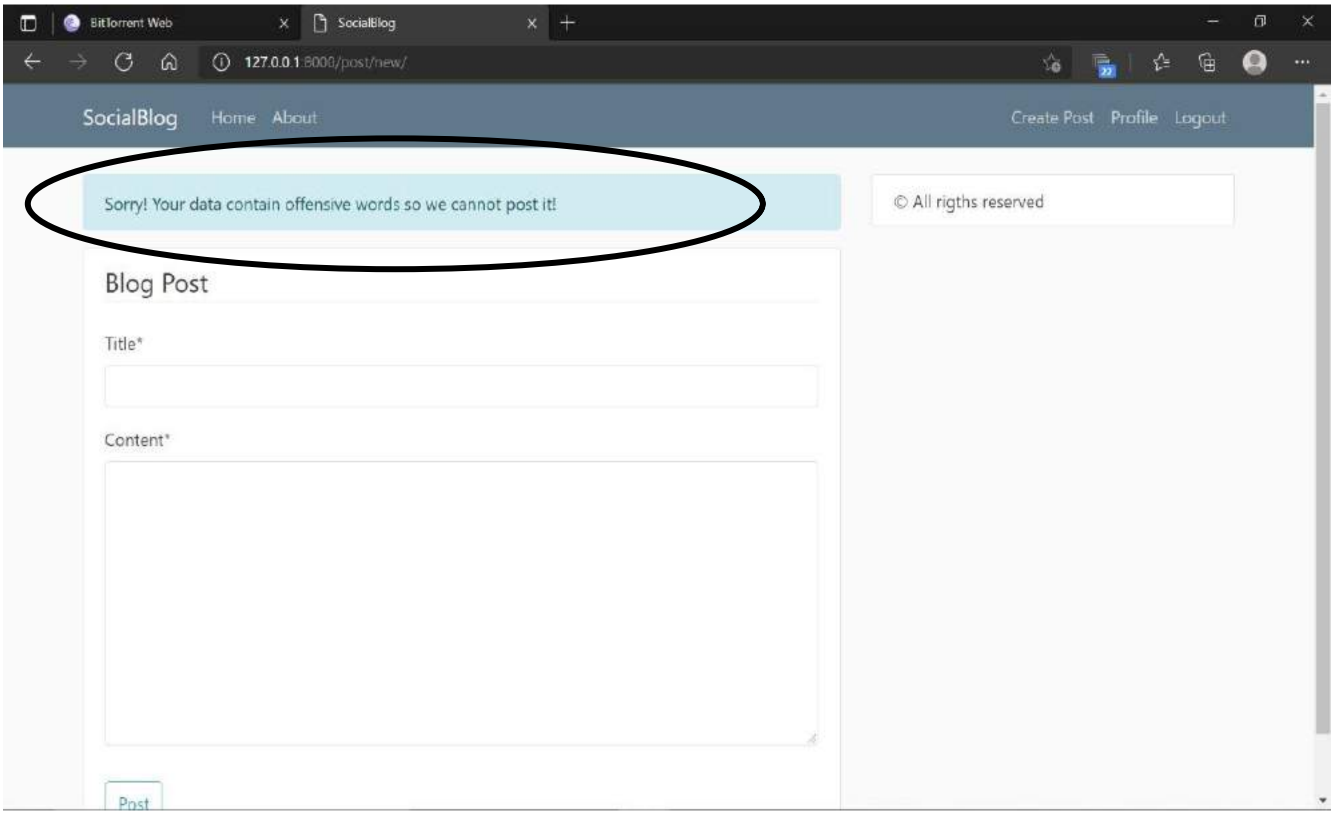
GF shit Phillosophies

Content*

If you don't give your bf head you're a lame bitch and he's probably cheating on you with a bitch the suck his shit

Post

fig.18: Trying to post offensive content



Sorry! Your data contain offensive words so we cannot post it!

© All rigths reserved

Blog Post

Title*

Content*

Post

Fig.19: Denied posting offensive content

- Database stored in SQL for the Admin

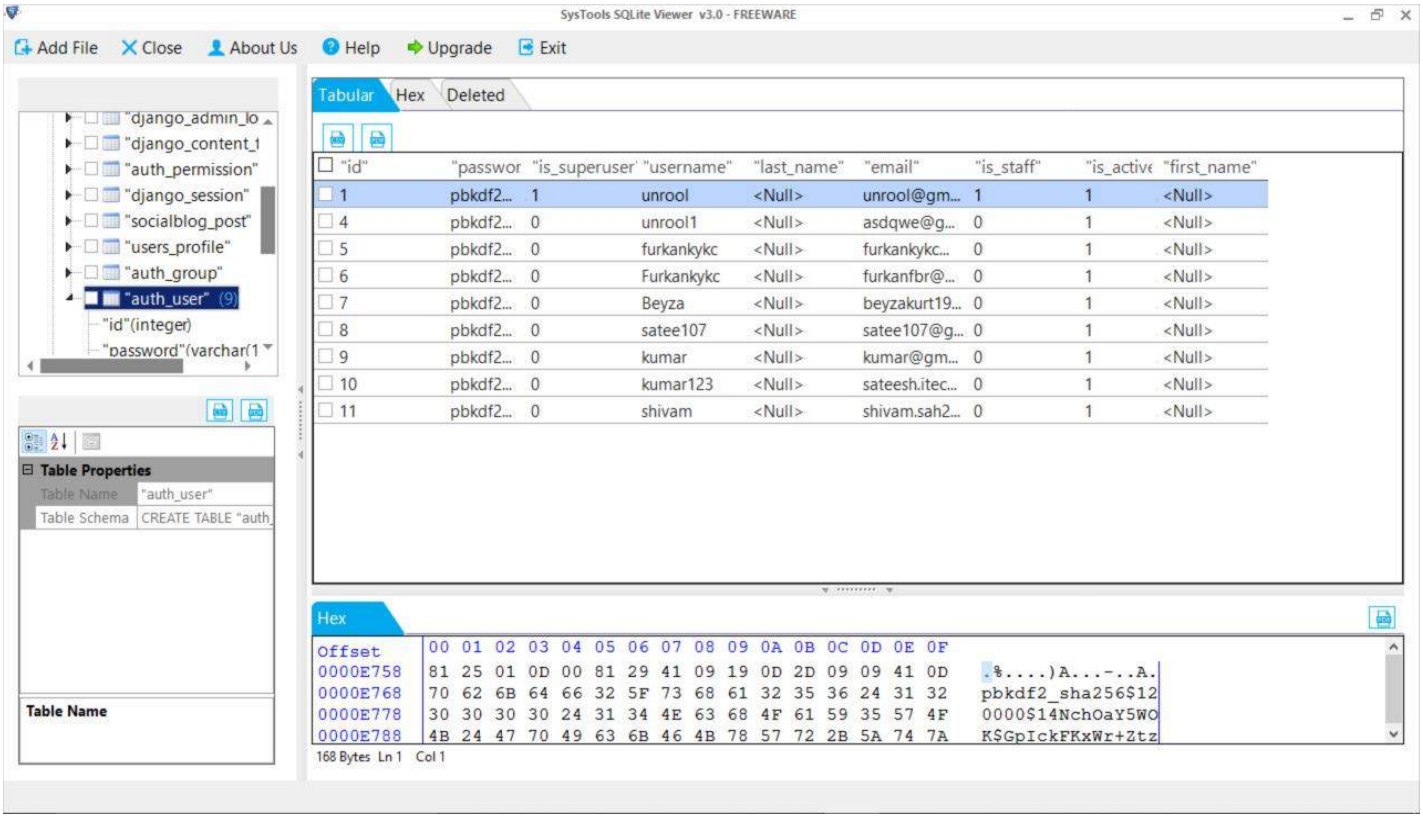


Table Properties	
Table Name	"auth_user"
Table Schema	CREATE TABLE "auth_

Table Name

"id"	"passwor	"is_superuser'	"username"	"last_name"	"email"	"is_staff"	"is_active	"first_name"
1	pbkdf2...	1	unrool	<Null>	unrool@gm...	1	1	<Null>
4	pbkdf2...	0	unrool1	<Null>	asdqwe@g...	0	1	<Null>
5	pbkdf2...	0	furkankykc	<Null>	furkankykc...	0	1	<Null>
6	pbkdf2...	0	Furkankykc	<Null>	furkanfbr@...	0	1	<Null>
7	pbkdf2...	0	Beyza	<Null>	beyzakurt19...	0	1	<Null>
8	pbkdf2...	0	satee107	<Null>	satee107@g...	0	1	<Null>
9	pbkdf2...	0	kumar	<Null>	kumar@gm...	0	1	<Null>
10	pbkdf2...	0	kumar123	<Null>	sateesh.itec...	0	1	<Null>
11	pbkdf2...	0	shivam	<Null>	shivam.sah2...	0	1	<Null>

Hex

```
Offset    00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
0000E758  81 25 01 0D 00 81 29 41 09 19 0D 2D 09 09 41 0D   .%....)A...-..A.
0000E768  70 62 6B 64 66 32 5F 73 68 61 32 35 36 24 31 32   pbkdf2_sha256$12
0000E778  30 30 30 30 24 31 34 4E 63 68 4F 61 59 35 57 4F   0000$14NchOaY5WO
0000E788  4B 24 47 70 49 63 6B 46 4B 78 57 72 2B 5A 74 7A   K$GpIckFKxWr+Ztz
168 Bytes  Ln 1   Col 1
```

Fig.20: Database stored in SQL for the Admin

User profile:

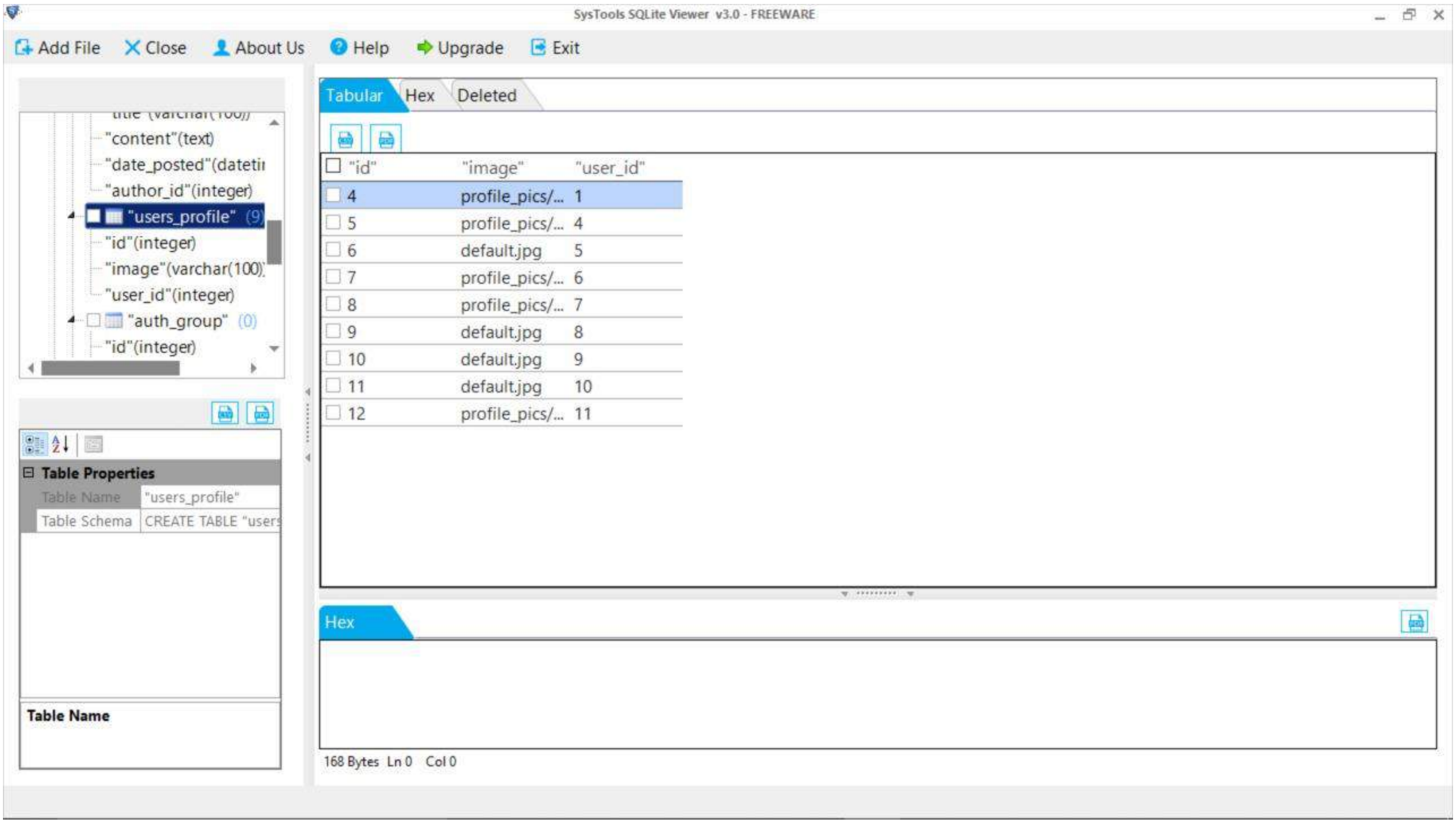


Table Properties	
Table Name	"users_profile"
Table Schema	CREATE TABLE "users

Table Name

"id"	"image"	"user_id"
4	profile_pics/...	1
5	profile_pics/...	4
6	default.jpg	5
7	profile_pics/...	6
8	profile_pics/...	7
9	default.jpg	8
10	default.jpg	9
11	default.jpg	10
12	profile_pics/...	11

168 Bytes Ln 0 Col 0

Fig.20: User profile

Posts by users:

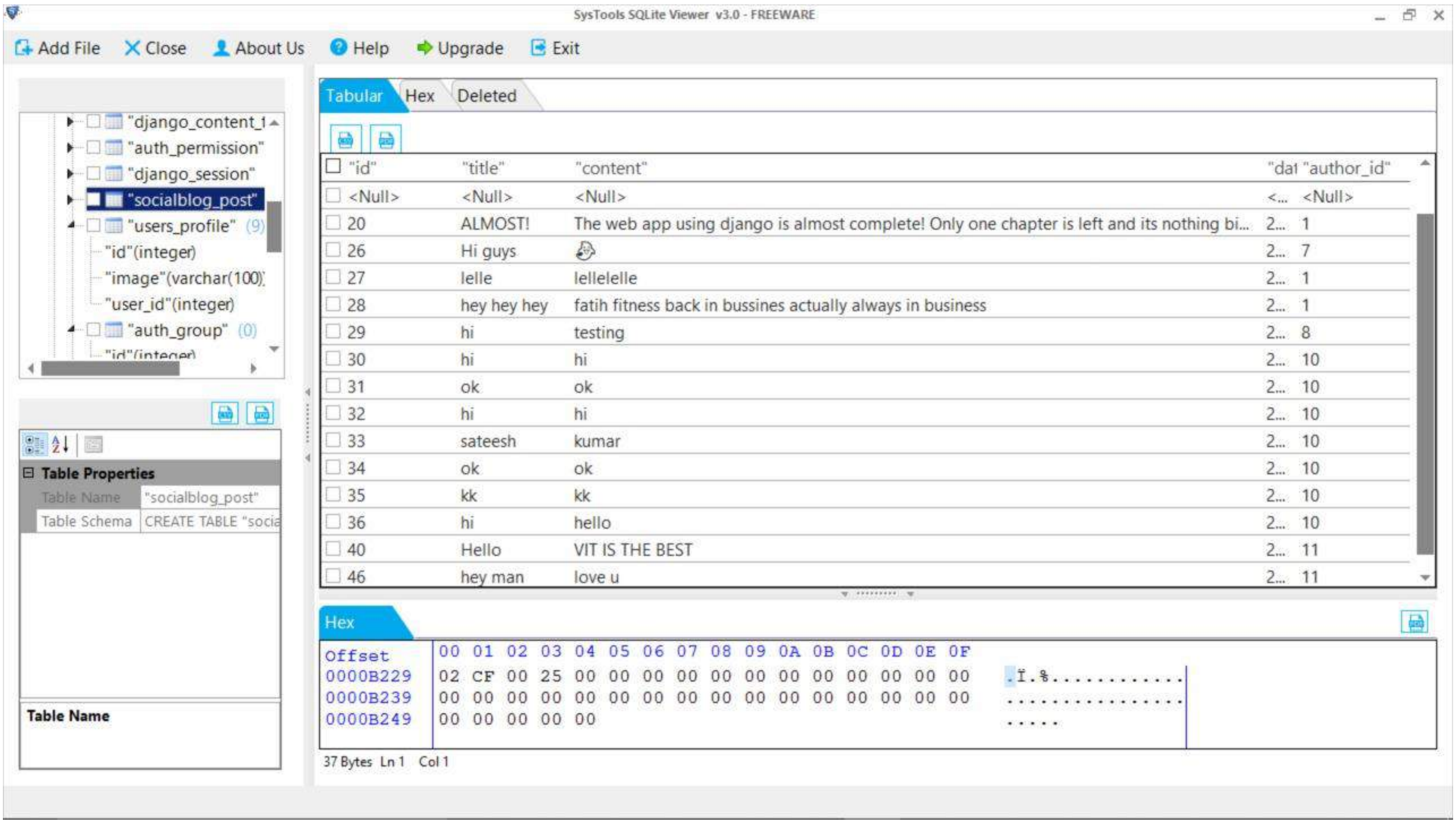


Table Name	"socialblog_post"
Table Schema	CREATE TABLE "socia

Table Name

□ "id"	"title"	"content"	"dat	"author_id"
□ <Null>	<Null>	<Null>	<...	<Null>
□ 20	ALMOST!	The web app using django is almost complete! Only one chapter is left and its nothing bi...	2...	1
□ 26	Hi guys	😊	2...	7
□ 27	lelle	lellelelle	2...	1
□ 28	hey hey hey	fatih fitness back in bussines actually always in business	2...	1
□ 29	hi	testing	2...	8
□ 30	hi	hi	2...	10
□ 31	ok	ok	2...	10
□ 32	hi	hi	2...	10
□ 33	sateesh	kumar	2...	10
□ 34	ok	ok	2...	10
□ 35	kk	kk	2...	10
□ 36	hi	hello	2...	10
□ 40	Hello	VIT IS THE BEST	2...	11
□ 46	hey man	love u	2...	11

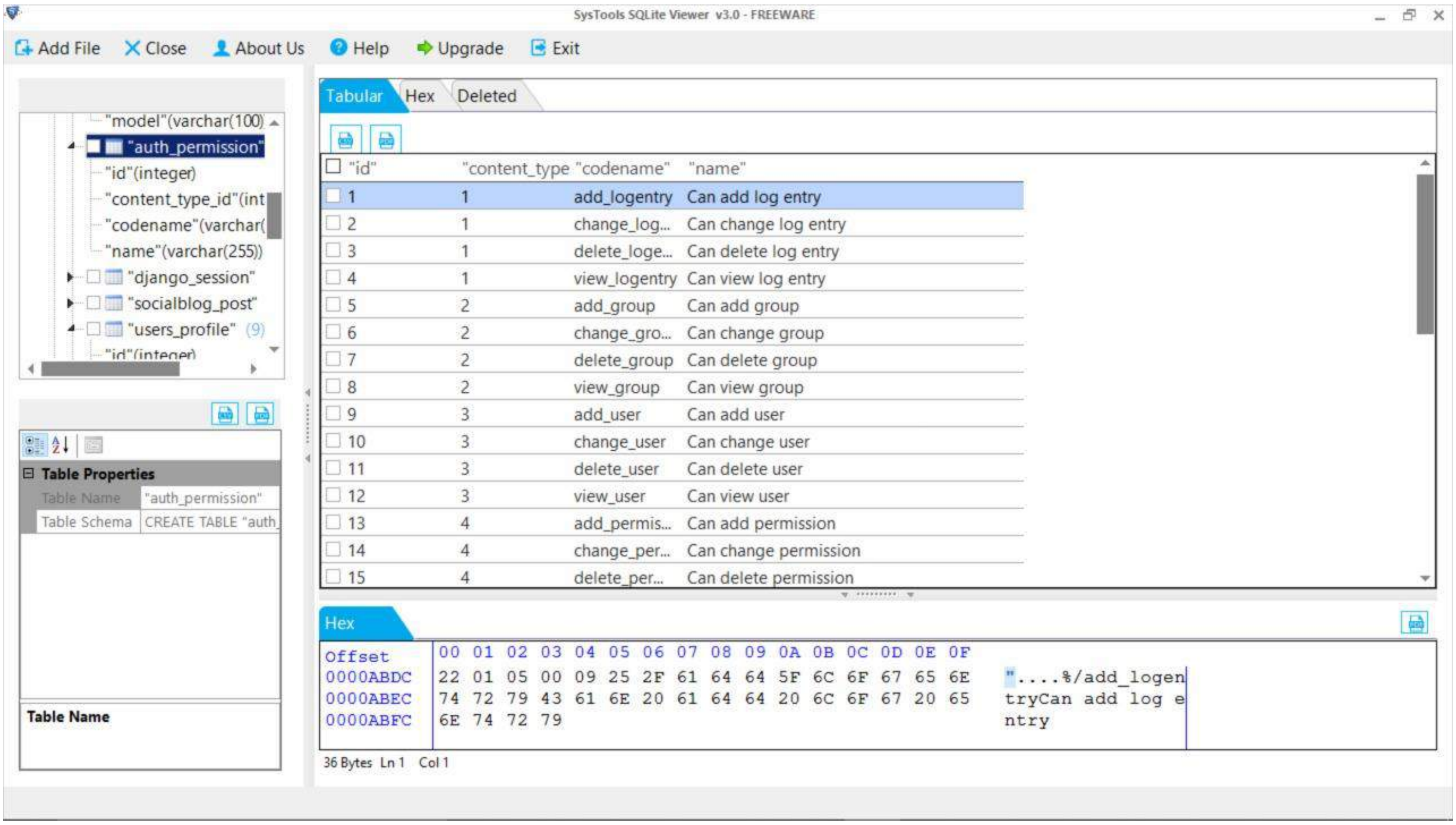
Hex

```
Offset    00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
0000B229  02 CF 00 25 00 00 00 00 00 00 00 00 00 00 00 00   .Ï.%............
0000B239  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
0000B249  00 00 00 00 00                                    .....
```

37 Bytes Ln 1 Col 1

Fig. 22: Posts by users

Permission from the admin



"id"	"content_type	"codename"	"name"
□ 1	1	add_logentry	Can add log entry
□ 2	1	change_log...	Can change log entry
□ 3	1	delete_loge...	Can delete log entry
□ 4	1	view_logentry	Can view log entry
□ 5	2	add_group	Can add group
□ 6	2	change_gro...	Can change group
□ 7	2	delete_group	Can delete group
□ 8	2	view_group	Can view group
□ 9	3	add_user	Can add user
□ 10	3	change_user	Can change user
□ 11	3	delete_user	Can delete user
□ 12	3	view_user	Can view user
□ 13	4	add_permis...	Can add permission
□ 14	4	change_per...	Can change permission
□ 15	4	delete_per...	Can delete permission

Table Properties

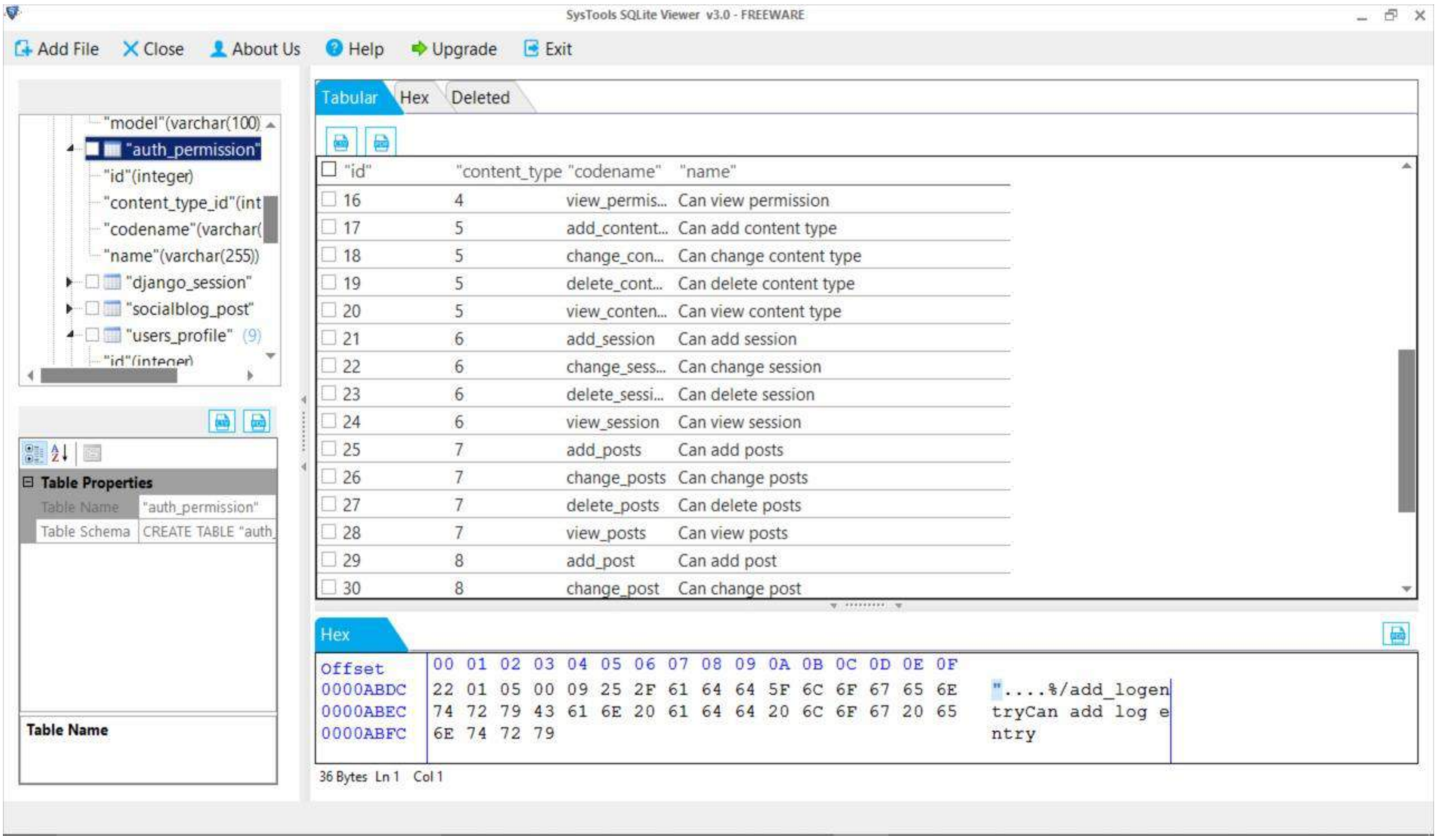
Table Name	"auth_permission"
Table Schema	CREATE TABLE "auth_

Table Name

Hex

```
Offset    00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
0000ABDC  22 01 05 00 09 25 2F 61 64 64 5F 6C 6F 67 65 6E   "....%/add_logen
0000ABEC  74 72 79 43 61 6E 20 61 64 64 20 6C 6F 67 20 65   tryCan add log e
0000ABFC  6E 74 72 79                                       ntry
```

36 Bytes Ln 1 Col 1



"id"	"content_type	"codename"	"name"
□ 16	4	view_permis...	Can view permission
□ 17	5	add_content...	Can add content type
□ 18	5	change_con...	Can change content type
□ 19	5	delete_cont...	Can delete content type
□ 20	5	view_conten...	Can view content type
□ 21	6	add_session	Can add session
□ 22	6	change_sess...	Can change session
□ 23	6	delete_sessi...	Can delete session
□ 24	6	view_session	Can view session
□ 25	7	add_posts	Can add posts
□ 26	7	change_posts	Can change posts
□ 27	7	delete_posts	Can delete posts
□ 28	7	view_posts	Can view posts
□ 29	8	add_post	Can add post
□ 30	8	change_post	Can change post

Table Properties

Table Name	"auth_permission"
Table Schema	CREATE TABLE "auth_

Table Name

Hex

```
Offset    00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
0000ABDC  22 01 05 00 09 25 2F 61 64 64 5F 6C 6F 67 65 6E   "....%/add_logen
0000ABEC  74 72 79 43 61 6E 20 61 64 64 20 6C 6F 67 20 65   tryCan add log e
0000ABFC  6E 74 72 79                                       ntry
```

36 Bytes Ln 1 Col 1

fig.23: Permission from the admin

Test cases and Test data

TEST ID	TEST MODULE	TEST DESCRIPTION	EXPECTED OUTCOME	ACTUAL OUTCOME	TEST DATA
1	Registration page	If provided with wrong or incorrect information, such as your email address, password.	not registered due to incorrect format for mail	user not registered	PASS
2	Registration page	The e-mail and password feature should adhere following: Your password can’t be too similar to your other personal information. Your password must contain at least 8 characters. Your password can’t be a commonly used password. Your password can’t be entirely numeric.	not registered due to incorrect format for password	user not registered	PASS
3	Login Page	User can directly login to a site using correct mail id and password.	logged in with correct registered mail and password	logged in into the system	PASS
4	Home Page	Home page consists of various features like Create post, profile, About website tab, which is successfully done.	Create posts, user profile, about website are available.	User can create posts, check profile and visit about page.	PASS
5.	About US Page	Page should have information about website.	The page should tell the features used for the website.	the page does not display features of website	FAIL
6.	PROFILE	Should have editing and updating options for profile picture, email id, username.	this page should let users change their username, email-id , reset password, and have ability to change their profile pic	user’s username, email-id , reset password and profile pic changed	PASS

7.	CREATE POSTS	Should let users have write title and content IF the posts are detected to be non-offensive.	this page let’s user to post their words in clean and subtle manner only	the posts can be written	PASS
8	CREATE POSTS	Should NOT detect about the title and content if the posts are detected to be in other language than in English.	the page let’s user to post their words in clean and subtle manner only and give warning to user for making abusive posts while pre-posting.	the test failed the post was in another language other than english and not detected by system	PASS
9	CREATE POST	Should not detect the title and content if the letters are in English alphabets but word’s meaning is in some other language.	the page let’s user to post their words in clean and subtle manner only and give warning to user for making abusive posts while pre-posting.	warning from the system showing the system has ability to detect abusive language while the post was read by the system (pre-posting).	FAIL
10	CREATE POSTS	Should let users to post title and content if the posts are detected to be offensive.	this page let’s the user to post their words in clean and subtle manner only and give warning to user for making abusive posts while pre-posting.	warning from the system showing the system has ability to detect abusive language while the post was read by the system (pre-posting).	PASS

6 Conclusions:

In this examination, it was researched that the existing text-mining approaches in recognizing hostile elements in the test or the posts for securing online wellbeing. Differentiate harmful material in online networking and further predict a client's chance to convey aggressive substance. “The examination has a few commitments. In the first place, for all aims and purposes concept the thought of aggressive online substance, and further recognize the commitment of pejoratives/obscenities and obscenities in deciding hostile substance, and present hand creating syntactic principles in distinguishing verbally abusing badgering”. Secondly, it has improved the standard ML techniques by utilizing logistic regression highlights to recognize hostile dialects and fusing style highlights, structure highlights, and context-explicit highlights to foresee a client's probability of conveying aggressive substance in all the more likely internet-based life.

Future Scope:

These are some of the future works which can be added in this project:

a. It is known that human language is way too much varied, and some of the posts may not look violent from the exterior, but essentially they may have contained hate speech when analyzed by humans. Therefore, in the future, the system will examine the detention of the syntactic and semantic features with their mixture and other pre-trained structures, which will help us improve the working of this project.

b. It is observed that to tackle imbalance, the logistic regression has performed the best with the dataset. But at the same time the big problem with this approach is that system have lost a good amount of significant information which might be useful in the training of the model. This approach have reduced the training data to a great amount. In future researchers may come up with the suitable approach and exploration that may tackle this problem of huge imbalance in the dataset efficiently without the reduction in training data.

c. In the future, researchers can implement more classifications of hate speech. And also, researchers can implement it in other languages like Hindi, Tamil, etc. they can still improve the algorithm's performance and can implement artificial intelligence for the automatic detection of hate speech.

d. As per the examination, it can clearly be seen that the methods of deep learning are not performing that accurately because of less depth in their designs. So, in the future, the researchers can aim to increase the difficulty of these models by increasing the layers and the configuration settings given the computational resources and the time.

e. The system which is designed is only designed to test for the textual content so in the future it can be designed with some method in which the offensive words can be determined and examined form the voice an images with the image and voice processing tools which will warn the user to not use the offensive word in the communication.

7 References

[1] Pinkesh Badjatiya, Shashank Gupta, Manish Gupta, and Vasudeva Varma. 2017. Deep learning for hate speech detection in tweets. In Proceedings of the 26th International Conference on World Wide Web Companion, pages 759–760.

[2] David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. Journal of machine learning research, 3(Jan):993–1022.

[3] Peter Burnap and Matthew Leighton Williams. 2014. Hate speech, machine classification and statistical modelling of information flows on twitter: Interpretation and communication for policy decision making.

[4] Despoina Chatzakou, Nicolas Kourtellis, Jeremy Blackburn, Emiliano De Cristofaro, Gianluca Stringhini, and Athena Vakali. 2017. Mean birds: Detecting aggression and bullying on twitter. In Proceedings of the 2017 ACM, pages 13–22. ACM.

[5] Kyunghyun Cho, Bart Van Merri¨enboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation.

[6] Nemanja Djuric, Jing Zhou, Robin Morris, Mihajlo Grbovic, Vladan Radosavljevic, and Narayan Bhamidipati. 2015. Hate speech detection with comment embeddings, pages 29–30. ACM.

[7] Maeve Duggan. 2017. Online harassment 2017. Pew Research Center; 2018.

[8] Antigoni Founta, Constantinos Djouvas, Despoina Chatzakou, Ilias Leontiadis, Jeremy Blackburn, Gianluca Stringhini, Athena Vakali, Michael Sirivianos, and Nicolas Kourtellis. 2018. Large scale crowdsourcing and characterization of twitter abusive behavior.

[9] Jennifer Golbeck, Zahra Ashktorab, Rashad O Banjo, Alexandra Berlinger, Siddharth Bhagwan, Cody Buntain, Paul Cheakalos, Alicia A Geller, 2017.

[10] https://www.javatpoint.com/logistic-regression-in-machine-learning