



Natural Language Processing (NLP) Course-End Project

Help Twitter Combat Hate Speech Using NLP and Machine Learning

Objectives

- Use NLP and ML to make a model to identify hate speech (racist or sexist tweets) on Twitter
- Use cleaned-up tweets to build a classification model using NLP techniques, cleanup-specific data for all tweets, regularization, and hyperparameter tuning using stratified k-fold and cross-validation to get the best model



Prerequisites

- Sklearn
- Grid search
- Stratified K-fold



Industry Relevance



- **Sklearn:** It includes several effective methods of statistical modeling and machine learning, such as classification, regression, clustering, and dimensionality reduction.
- **Grid search:** It is a procedure that thoroughly explores a manually chosen portion of the targeted algorithm's hyperparameter space.
- **Stratified K-fold:** This cross validation is an extension of the cross-validation technique used for classification problems.

Dataset Description



Variable	-	Description
id	-	Identifier number of the comment
comment_text	-	The text in the comment
toxic	-	Status of toxicity with 0 for <i>nontoxic</i> and 1 for <i>toxic</i>

Problem Statement



Twitter is a massive platform where anybody can have their views heard. Some people on twitter spread hate and negativity. Twitter is wary of its platform being used as a medium to spread hate.

You are a data scientist at Twitter and have been asked to identify and remove the tweets with hate speech. You will need to use NLP techniques, perform specific cleanup for the data on tweets, and make a robust model.

Task to Perform



1. Load the tweets file using the **read_csv** function from the **Pandas** package
2. Upload the tweets into a list for easy text cleanup and manipulation
3. Apply the following steps to clean up the tweets:
 - a) Normalize the casing
 - b) Use regular expressions and remove user handles that begin with @
 - c) Use regular expressions, and remove URLs
 - d) Use **TweetTokenizer** from NLTK to tokenize the tweets into individual terms
 - e) Remove stop words
 - f) Remove redundant terms like *amp* and *rt*
 - g) Remove # from the tweets while retaining the text that follows it

Task to Perform



4. Use the cleanup process to remove terms with a length of 1
5. Check the top terms in the tweets:
 - a) First, get all tokenized terms into one list
 - b) Use the counter, and find the 10 most common terms
6. Format the data for predictive modeling:
 - a) Join the tokens back to form strings, which will be required for the vectorizers
 - b) Assign x and y
 - c) Perform **train_test_split** using **sklearn**

Task to Perform



7. Use **TF-IDF** values for the terms as a feature to get into a vector space model
 - a) Import **TF-IDF vectorizer** from **sklearn**
 - b) Instantiate the model with a maximum of 5000 terms in your vocabulary
 - c) Fit and apply the vector space model on the train set
 - d) Apply the model on the test set
8. Model building: ordinary logistic regression
 - a) Instantiate **logistic regression** from **sklearn** with default parameters
 - b) Fit model on the train data
 - c) Make predictions for the train and the test sets

Task to Perform



9. Model evaluation: accuracy, recall, and f_1 score
 - a) Report the accuracy of the train set
 - b) Report the recall on the train set: decent, high, or low
 - c) Get the f_1 score on the train set
10. Adjust the class imbalance, if any
 - a) Adjust the appropriate class in the logistic regression model
11. Train the model again with the adjustment and evaluate
 - a) Train the model on the train set
 - b) Evaluate the predictions on the train set: accuracy, recall, and f_1 score

Task to Perform



12. Regularization and hyperparameter tuning:
 - a) Import **GridSearch** and **StratifiedKFold**
 - b) Choose for **C** and **penalty** parameters under the parameters grid
 - c) Use a balanced class weight while instantiating the logistic regression
13. Find the parameters with the best recall in cross-validation
 - a) Choose **recall** as the metric for scoring
 - b) Choose a stratified four-fold cross-validation scheme
 - c) Fit it on the train set
14. List the best parameters

Task to Perform



15. Predict and evaluate parameters using the best estimator
- a) Use the best estimator from the grid search to make predictions on the test set
 - b) Find the recall on the test set for the toxic comments
 - c) Find the f_1 score

Project Outcome



- Create a model to recognize hate speech (racist or sexist tweets) on Twitter using NLP and ML
- Use NLP approaches, the cleanup data for tweets, regularization, and hyperparameter tuning using stratified k-fold and cross-validation to find the optimal model and develop a classification model once the tweets have been cleaned up.

Submission Process



1. Complete the project in the Simplilearn lab
2. Complete each task listed in the problem statement
3. Take screenshots of the results for each question and the corresponding code
4. Save it as a document, and submit this using the **Assessment** tab
5. Tap the **Submit** button (this will present you with three choices)
6. Attach the three files, and then click on **Submit**

Note: Be sure to include screenshots of the output

Thank You