

CS116.N21.KHCL

Sentiment analysis

Nguyễn Văn Hoàng-20521346

Table of contents

01 Introduction

02 DATA
PREPROCESSING

03 FEATURE
EXTRACTION

04 SUPPORT VECTOR
MACHINE

05 LOGISTIC
REGRESSION

06 EXPERIMENT

01

Introduction

Insert a subtitle here if you need it

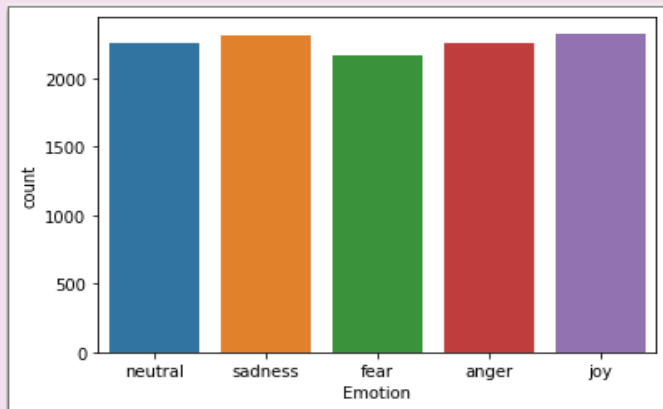


01

Introduction

- The rapid progress of technology has given rise to various online platforms, including blogs, forums, and social networks, enabling users to participate in discussions and share their thoughts.
- These platforms serve as a space for individuals to express grievances, debate current issues, and share political viewpoints.
- User-generated information from these platforms holds value for analyzing human behavior and activities across domains, particularly in e-commerce and government sectors.
- However, sentiment analysis (SA) faces challenges due to the vast amount of data generated online, necessitating efficient storage, access, processing, and reliable results.

Introduction



- This report focuses on exploring Machine Learning techniques, specifically Support Vector Machine and Decision Tree, for sentiment analysis.
- The selected dataset is a combination of dailydialog, isear, and emotion-stimulus datasets, resulting in a balanced dataset with five labels: joy, sadness, anger, fear, and neutral.
- The dataset comprises 11,327 sentences, primarily short messages and dialogue utterances.
- The sentence distribution across classes is as follows: 2,326 joy, 2,317 sadness, 2,259 anger, 2,254 neutral, and 2,171 fear.
- The dataset is divided into a training set with 7,934 sentences and a test set with 3,393 sentences.

02 DATA PREPROCESSING



DATA PREPROCESSING



**Lowercase
conversion**



Remove noise



**Remove
stopwords**



Punctuation



POS tagged

Lowercase conversion:

A → a

a → A

- Converting words to lowercase is important to avoid unintended differentiation caused by capitalization.
- For example, "Artificial" and "artificial" would be perceived as two distinct words by the computer.
- Not converting to lowercase could result in augmented sentence length.

Remove noise

A code editor window with a dark background and three colored window control buttons (red, yellow, green) in the top-left corner. It contains three lines of Python code:

```
import re
re_html = re.compile(r'<[^>]+>')
re_html.sub('', html_text)
```

```
import re
re_html = re.compile(r'<[^>]+>')
re_html.sub('', html_text)
```

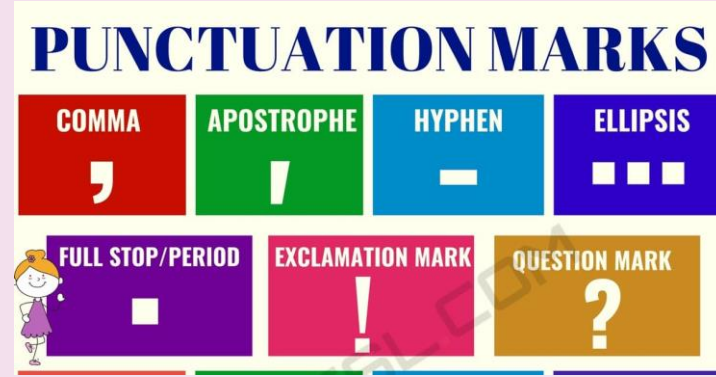
- Removing unnecessary elements such as HTML markup, URLs, hashtags, and @names, as well as punctuation, non-ASCII digits, and whitespace.
- Eliminating noise helps to clean the text data and focus on the relevant content.

Remove noise

Sample text with Stop Words	Without Stop Words
GeeksforGeeks – A Computer Science Portal for Geeks	GeeksforGeeks , Computer Science, Portal ,Geeks
Can listening be exhausting?	Listening, Exhausting
I like reading, so I read	Like, Reading, read

- Stopwords are common words that do not contribute significantly to the meaning of a sentence, such as articles and prepositions.
- However, the presence of the word "not" in a stopword can alter the sentence's meaning.
- Removing stopwords is necessary, but the word "not" should be excluded from the stopwords list.

Punctuation



- Some sentences in the dataset contain words like "I'm," "He's," "She's," or "I didn't," which include punctuation.
- To preserve the intended meaning and context, sentence expansion through punctuation is required.
- This ensures that the word "not" is not lost in cases like "didn't."

POS tagged



- POS tagging is used to convert words from past tense to present tense without treating them as distinct words.
- Nouns, verbs, adjectives, and adverbs are the four categories of words used in this process.
- POS tagging helps in maintaining the correct tense and improving the accuracy of sentiment analysis.

03 FEATURE EXTRACTION



$$TF(t, d) = \frac{\text{number of times } t \text{ appears in } d}{\text{total number of terms in } d}$$

$$IDF(t) = \log \frac{N}{1 + df}$$

$$TF - IDF(t, d) = TF(t, d) * IDF(t)$$

Feature Extraction using TF-IDF:

In NLP, various feature extraction methods are available.

For this problem, we use TF-IDF (term frequency-inverse document frequency).

We utilize the `TfidfVectorizer()` method from the `sklearn` module.

Parameters: `ngram = 1,2` (unigrams and bigrams), `norm = l2` (normalize vector elements), `sublineartf = True` (normalize TF value).

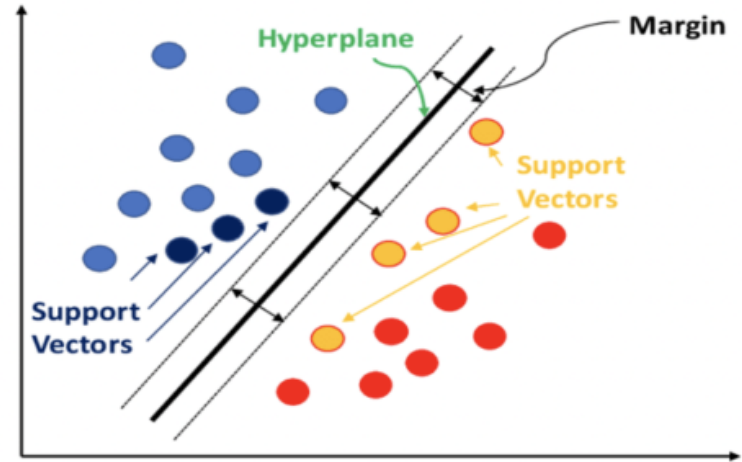
TF-IDF helps in extracting meaningful features and improving sentiment analysis accuracy.

04 SUPPORT VECTOR MACHINE



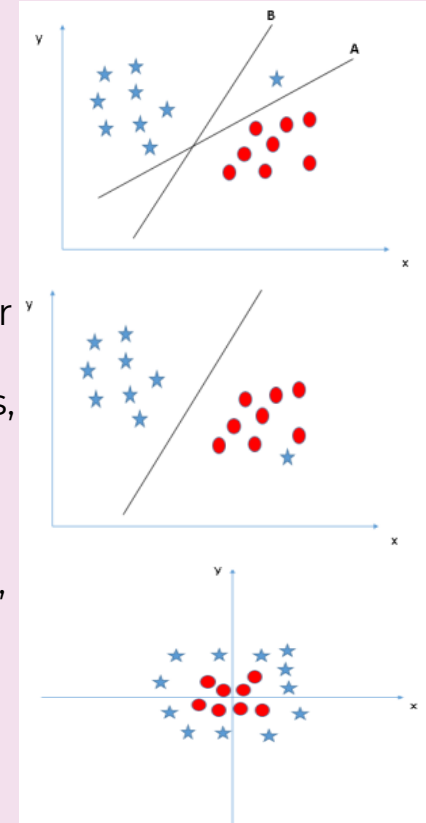
- SVM is a popular supervised learning model used for multi-class classification tasks.
- It aims to find a hyperplane that effectively separates data points based on their categories.

WHAT IS A **SUPPORT VECTOR MACHINE?**



The optimal separating hyperplane is chosen based on criteria:

1. The hyperplane should effectively separate the layers without overlapping.
2. The maximum distance or "margin" between the nearest point of a layer and the hyperplane should be calculated.
3. The primary criterion is to choose a hyperplane that separates the layers, even if it has a smaller margin.
4. SVM can handle exceptional cases and outliers by disregarding them and finding the hyperplane with the widest margin.
5. In cases where a single line cannot partition the data into distinct layers, SVM introduces an additional feature to resolve the problem.



SVM Hyperparameter Tuning using GridSearchCV

GridSearch should be used to optimize SVM with three

hyperparameters: kernel, C and gamma.

C=0.1, 1, 10, 100, 1000

gamma=1, 0.1, 0.01, 0.001, 0.0001

kernel=linear, poly, rbf

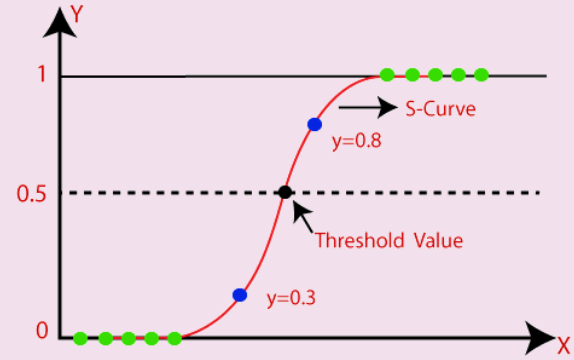
Result when using GridSearchSV:

C = 10, gamma=1, kernel = rbf

05 LOGISTIC REGRESSION

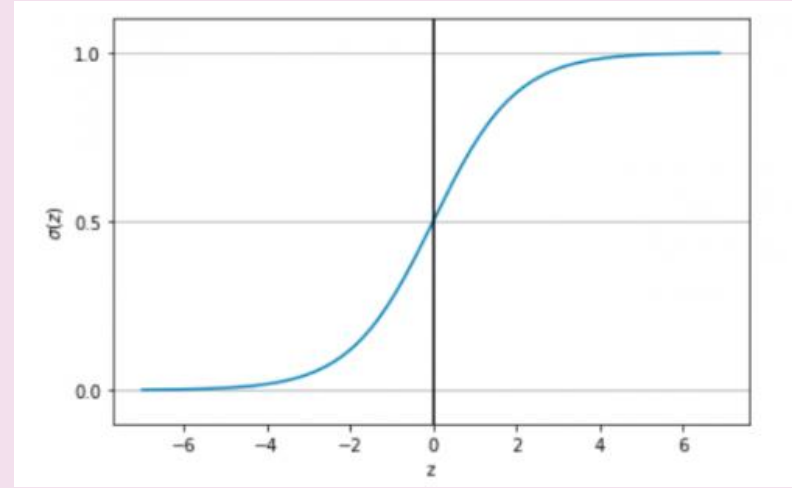


05 LOGISTIC REGRESSION



- Logistic regression is a machine learning algorithm used for classification problems.
- It predicts whether an instance belongs to one class or another.
- It is a supervised learning algorithm that learns the relationship between features and classes.
- It predicts the probability of an event.

Mapping to Probability



- Logistic regression maps input data to a probability between 0 and 1.
- Unlike linear regression, it doesn't give continuous output values.
- It uses the sigmoid function to map the input data to a probability.
- The sigmoid function transforms the weighted sum of input values to a value between 0 and 1.

Decision Boundary and Predictions

- The logistic regression model output represents the probability that the class of the input data is 1.
 - The decision boundary or threshold is used to predict class labels based on the sigmoid output.
 - If the probability is above the threshold, it belongs to one class; otherwise, it belongs to the other class. Logistic regression maps input data to a probability between 0 and 1.
 - Unlike linear regression, it doesn't give continuous output values.
 - It uses the sigmoid function to map the input data to a probability.
 - The sigmoid function transforms the weighted sum of input values to a value between 0 and 1.
-

Model Training

- Logistic regression model parameters are learned using maximum likelihood estimation.
- The cost function is minimized using optimization algorithms like gradient descent.
- Scikit-learn provides different solver algorithms (e.g., newton-cg, lbfgs, liblinear, saga, sag) for training logistic regression models.

Suitable for Binary Classification


- Logistic regression is suitable for binary classification problems.
- It is used when the dependent variable is categorical.
- In contrast, linear regression is used when the dependent variable is continuous.



06

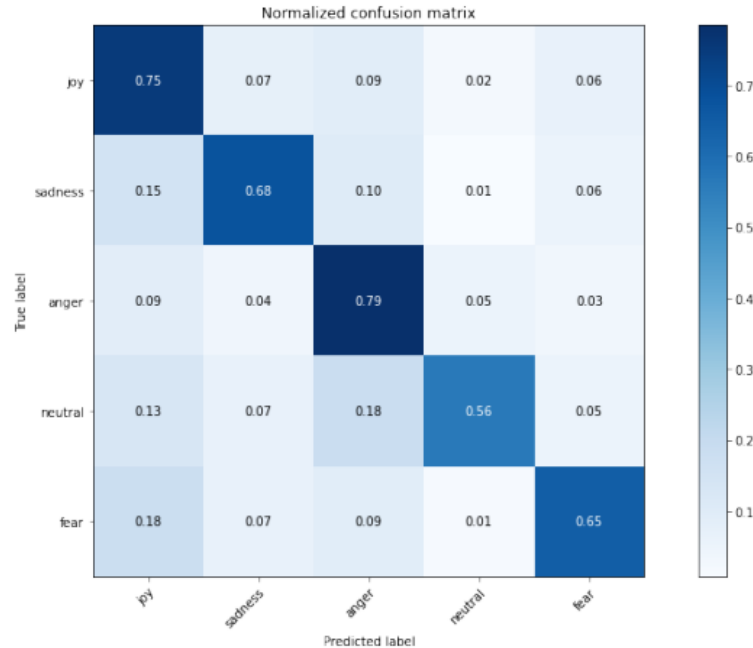
Introduction

Insert a subtitle here if you need it

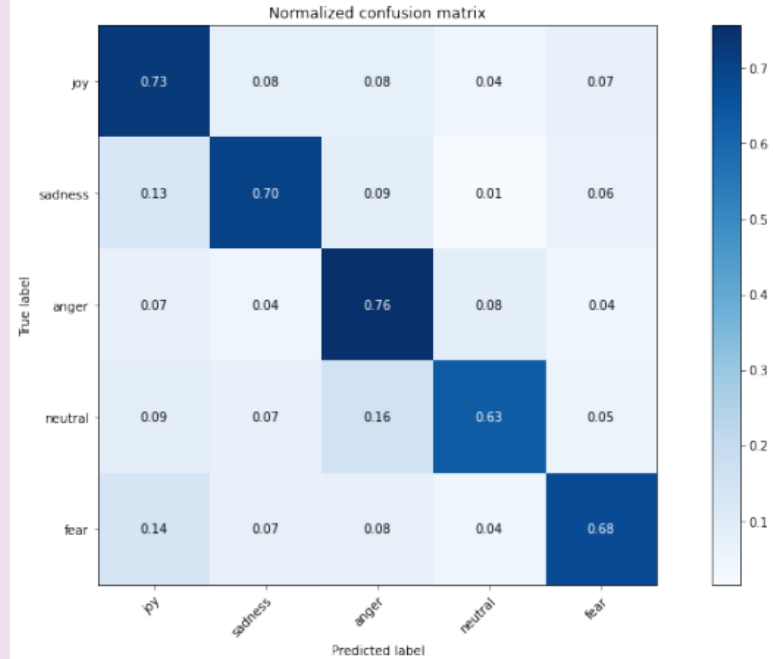


Support Vector Machine

Confusion matrix without Tuning

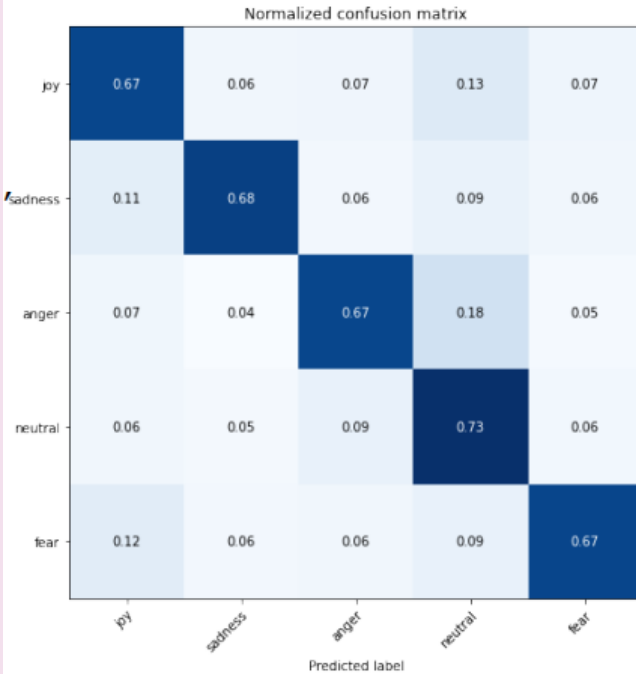


Confusion matrix with Tuning



Logistic Regression

Confusion matrix without Tuning



Confusion matrix with Tuning

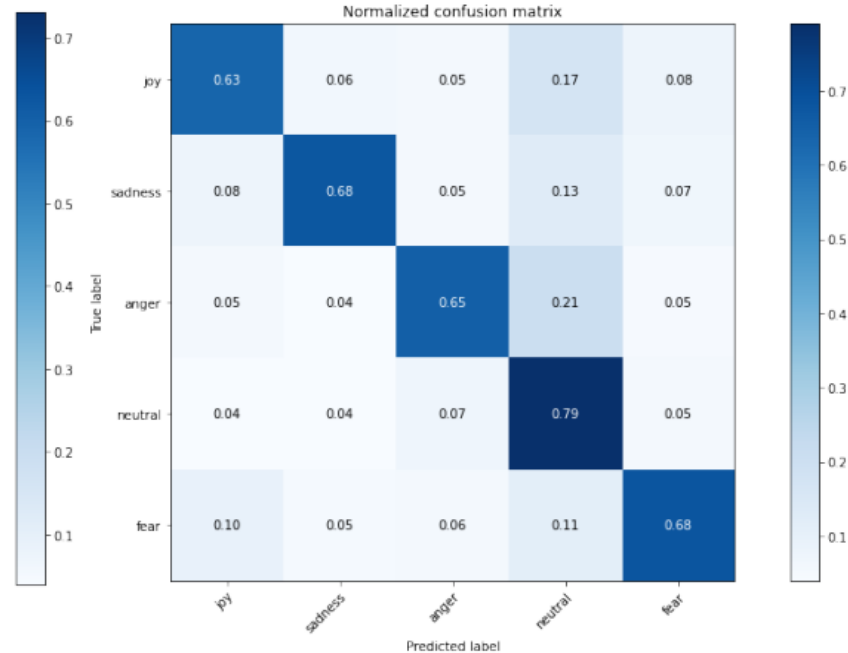


Table result

	<i>SVM</i>	<i>SVM with tuning</i>	<i>Logistic</i>	<i>Logistic with tuning</i>
F1-score	68.9	70	68.5	68.6
accuracy	68.9	70	69.5	68.6

CONCLUSIONS

- The overall results are quite good, approaching 0.7, when using the default function.
- GridSearch method to optimize accuracy did not lead to significant improvements in the results for both machine learning methods.
- It is possible to draw conclusions that the use of GridSearch does not always yield favorable outcomes.
- Reasons could be the group's utilization of parameter values or the insufficient number of parameters

CONCLUSIONS

Support Vector Machine (SVM)

- Using SVM without any parameters resulted in a 68.85 score.
- Employing the GridSearch method increased the score to 70.09 with the following parameters: $C = 10$, $\gamma = 1$, and $\text{kernel} = \text{rbf}$.
- Improved scores: sadness class from 0.68 to 0.7, neutral class from 0.56 to 0.63, and fear class from 0.65 to 0.68.

CONCLUSIONS

Logistic Regression

- Slight increase in metrics (accuracy and f1-score) compared to default settings.
- Metrics are approximately 0.9 to 1 percent lower compared to hyperparameter tuning.
- Tuned hyperparameters are not significantly more efficient.
- Plausible reason is that the customized parameters chosen for tuning are not substantially better than the default parameters.