A MAHAMUDHA BEGAM

Final Project

PROJECT TITLE

Sentimental analysis of movie reviews using CNN

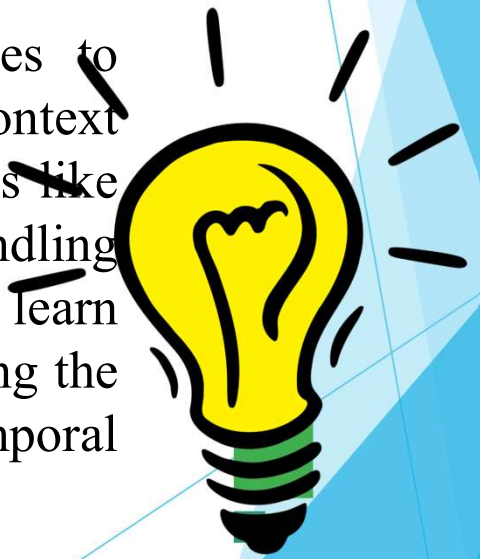
AGENDA

The goal of a sentiment analysis project using Convolutional Neural Networks (CNNs) is to enhance the accuracy of sentiment classification by analyzing large volumes of text data and identifying complex patterns in language usage. The project aims to leverage CNNs to recognize spatial and temporal relationships in movie review data, thereby improving the model's ability to classify sentiment accurately for various types of reviews. Ultimately, the objective is to provide more reliable and informative sentiment analysis results, which can be used for decision-making in various domains such as product reviews, social media sentiment analysis, and market research.

PROBLEM STATEMENT

The problem statement for a sentiment analysis project using Convolutional Neural Networks (CNNs) involves developing a model capable of accurately classifying the sentiment expressed in movie reviews, such as positive or negative sentiment, based on the text data.

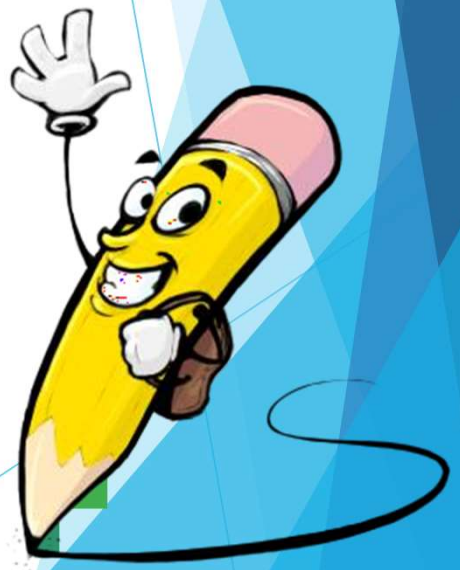
The challenge lies in leveraging CNN architectures to effectively capture the nuanced language patterns and context within the movie review data. This includes addressing issues like understanding sarcasm, detecting sentiment modifiers, and handling variations in writing styles. Additionally, the model needs to learn to extract relevant features from the text data while considering the spatial arrangement of words within sentences and the temporal ordering of sentences within reviews



PROJECT OVERVIEW

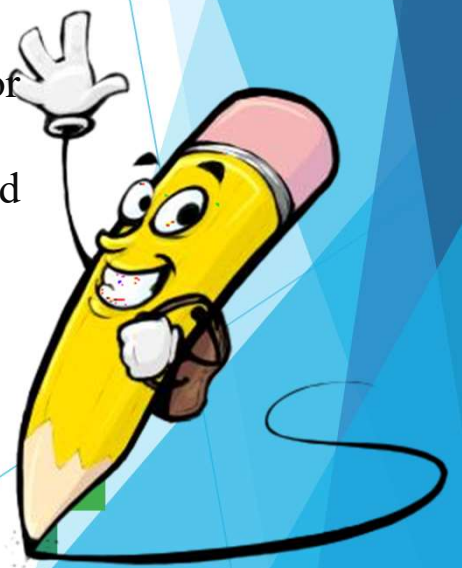
This sentiment analysis project employing Convolutional Neural Networks (CNNs) aims to enhance sentiment classification accuracy by leveraging deep learning techniques. Utilizing extensive datasets of movie reviews, both historical and real-time, the project endeavors to build a robust CNN model capable of accurately discerning sentiment expressed in the text.

Visual representation:
The sentiment analysis project utilizing Convolutional Neural Networks (CNNs) follows several key phases. Initially, large datasets of movie reviews are collected from various sources, including online databases and social media platforms. Subsequently, the text data undergoes meticulous preprocessing to handle noise, remove irrelevant information, and tokenize the reviews into manageable units. Following preprocessing, the data is partitioned into training, validation, and testing sets to facilitate model development and evaluation.



OBJECTIVE:

Develop models capable of accurately classifying movie reviews as positive, negative, or neutral based on the sentiment expressed in the text.
Gain insights into audience opinions and preferences towards movies by analyzing the sentiment conveyed in their reviews.
Enhance movie recommendation systems by incorporating sentiment analysis to tailor recommendations based on individual preferences and sentiments.
Use sentiment analysis to enhance user experience on movie review websites and platforms by providing personalized recommendations and relevant content.



YOUR SOLUTION AND ITS VALUE PROPOSITION

1.Accuracy: Our solution utilizes advanced deep learning techniques to achieve high accuracy in sentiment classification, ensuring reliable insights into audience sentiments towards movies.

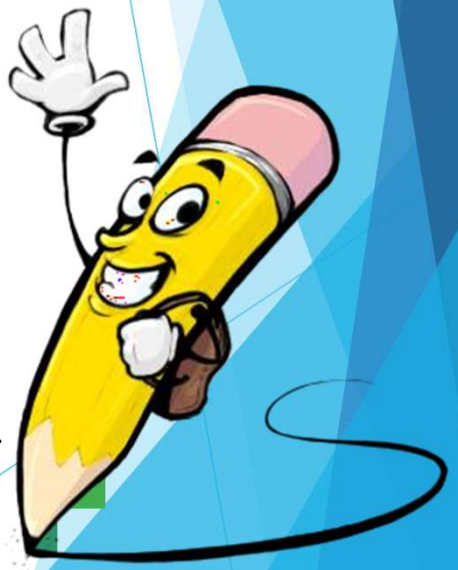
2.Efficiency: The use of CNNs allows for efficient processing of large volumes of text data, enabling quick analysis and decision-making for filmmakers, producers, and distributors.

3.Insightful Analytics: By analyzing sentiment trends across different movies, genres, and time periods, our solution provides insightful analytics to support market research, trend analysis, and strategic decision-making in the entertainment industry.

4.Improved User Experience: By incorporating sentiment analysis into movie recommendation systems and review platforms, we enhance the user experience by delivering personalized recommendations and relevant content tailored to individual preferences and sentiments.

5.Competitive Advantage: Our solution provides a competitive advantage by enabling filmmakers and producers to understand audience reception, identify areas for improvement, and tailor their strategies to meet audience expectations effectively.

.



The Wow Factor in Your Solution

•**Cutting-edge Deep Learning:** Our solution utilizes state-of-the-art Convolutional Neural Networks (CNNs) to analyze movie reviews, allowing for the extraction of intricate patterns and relationships within the text data with remarkable accuracy.

•**Actionable Insights**: Beyond simple sentiment classification, our solution offers advanced analytics and visualization techniques to uncover sentiment trends across movies, genres, and time periods, enabling stakeholders to make data-driven decisions and stay ahead of industry trends.

•**Personalized User Experience:** By seamlessly integrating sentiment analysis into movie recommendation systems and review platforms, our solution delivers personalized recommendations and relevant content tailored to individual preferences and sentiments, enhancing user engagement and satisfaction

RESULTS

SENTIMENT ANALYSIS

Finally, we train our model:

```
N_EPOCHS = 5

for epoch in range(N_EPOCHS):

    train_loss, train_acc = train(model, train_iterator, optimizer, criterion)
    valid_loss, valid_acc = evaluate(model, valid_iterator, criterion)

    print(f'| Epoch: {epoch+1:02} | Train Loss: {train_loss:.3f} | Train Acc: {train_acc*100:.2f}% | Val. Loss: {valid_loss:.3f} | Val. Acc: {valid_acc*100:.2f}% |')
```

```
| Epoch: 01 | Train Loss: 0.495 | Train Acc: 74.73% | Val. Loss: 0.340 | Val. Acc: 85.51% |
| Epoch: 02 | Train Loss: 0.305 | Train Acc: 86.89% | Val. Loss: 0.287 | Val. Acc: 88.24% |
| Epoch: 03 | Train Loss: 0.223 | Train Acc: 91.11% | Val. Loss: 0.279 | Val. Acc: 88.53% |
| Epoch: 04 | Train Loss: 0.151 | Train Acc: 94.40% | Val. Loss: 0.277 | Val. Acc: 89.07% |
| Epoch: 05 | Train Loss: 0.090 | Train Acc: 97.21% | Val. Loss: 0.296 | Val. Acc: 88.99% |
```

DRIVE LINK:
https://colab.research.google.com/drive/1AeZd4CVknobDfWk75PvoHTFi3zdNqRg2?usp=drive_link

Accuracy of the model

```
| Epoch: 05 | Train Loss: 0.090 | Train Acc: 97.21% | Val. Loss: 0.296 | Val. Acc: 88.9
```

We got a good accuracy: 88.99%

```
test_loss, test_acc = evaluate(model, test_iterator, criterion)
print(f'| Test Loss: {test_loss:.3f} | Test Acc: {test_acc*100:.2f}% |')
```

Loading...

```
[ ]  import spacy
     nlp = spacy.load('en')

     def predict_sentiment(sentence, min_len=5):
         tokenized = [tok.text for tok in nlp.tokenizer(sentence)]
         if len(tokenized) < min_len:
             tokenized += ['<pad>'] * (min_len - len(tokenized))
         indexed = [TEXT.vocab.stoi[t] for t in tokenized]
         tensor = torch.LongTensor(indexed).to(device)
         tensor = tensor.unsqueeze(1)
         prediction = torch.sigmoid(model(tensor))
         return prediction.item()
```

An example negative review:

```
predict_sentiment("This film is boring")
```

```
0.050386134535074234
```

An example positive review:

```
[ ]  predict_sentiment("This film is great")
```

```
0.9362199902534485
```