# **SIMATS SCHOOL OF ENGINEERING**

**SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES**

**CHENNAI-602105**

**Inappropriate Comments Scanner**

**A CAPSTONE PROJECT REPORT**

*Submitted in the partial fulfillment for the award of the degree of*

**BACHELOR OF ENGINEERING**

**IN**

**COMPUTER SCIENCE ENGINEERING**

**Submitted by**

**B.Girish kumar (192210500)**

**Under the Supervision of**

**Dr. K.V.KANIMOZHI**

**CSA1383 - Theory of Computation for Natural Language Processing**

DECEMBER (2024)

**Abstract**

The **Inappropriate Comments Scanner** is a Natural Language Processing (NLP) project aimed at detecting and classifying inappropriate comments in text data. Leveraging advancements in NLP techniques, the project focuses on identifying offensive, profane, or harmful language in online communications, fostering a safer and more inclusive digital environment. The scanner involves key processes such as dataset preprocessing, labeling, feature extraction, and machine learning-based classification. This system is designed to enhance moderation tools, empowering platforms to mitigate abusive behavior effectively while promoting constructive interactions.

**Introduction**

With the exponential growth of user-generated content on social media platforms, forums, and other online spaces, managing inappropriate or harmful comments has become a critical challenge. Offensive language, cyberbullying, and hate speech significantly affect the mental well-being of individuals and the integrity of online communities. Automated systems to detect and classify such comments are necessary to address these issues efficiently.

This project focuses on building an **Inappropriate Comments Scanner** using NLP techniques. The goal is to preprocess raw text data, extract relevant features, and classify comments into appropriate or inappropriate categories using machine learning models. By utilizing labeled datasets, this scanner learns to identify patterns and similarities in offensive language, contributing to an improved moderation process.

Key components of the project include:

* **Text Classification**: Categorizing text into predefined labels such as "appropriate" or "inappropriate."
* **Dataset Preprocessing**: Cleaning and preparing textual data for analysis, including tokenization, removing stop words, and stemming/lemmatization.
* **Labeling and Annotation**: Manually tagging text data to create a labeled dataset for supervised learning.
* **Feature Extraction**: Converting textual data into numerical representations using techniques like Term Frequency-Inverse Document Frequency (TF-IDF) or word embeddings.

The system's applications span across various domains, including social media moderation, e-learning platforms, and workplace communication tools, demonstrating its relevance and scalability. This project report outlines the development process, methodologies, and evaluation metrics to build a robust and efficient inappropriate comment detection system.

**Methods and Materials**

**Methods**

1. **Data Collection**
   * A large dataset of user comments or text from online platforms is collected. Publicly available datasets such as the Jigsaw Toxic Comment Classification dataset or manually curated datasets are used.
   * Comments are labeled as "appropriate" or "inappropriate" based on predefined guidelines.
2. **Dataset Preprocessing**
   * Text data is cleaned to remove unwanted characters, HTML tags, special symbols, and numbers.
   * Case normalization is performed by converting all text to lowercase.
   * Tokenization is applied to split text into individual words or tokens.
   * Stop words (e.g., "and," "the") are removed to focus on meaningful content.
   * Lemmatization or stemming reduces words to their root forms to unify similar terms.
3. **Labeling and Annotation**
   * Annotators manually label comments based on guidelines to categorize them as "appropriate" or "inappropriate."
   * Labeling tools like Doccano or Prodigy can be used to streamline this process.
4. **Feature Extraction**
   * Convert text into numerical representations for machine learning models using techniques such as:
     + **TF-IDF**: Assigns weights to words based on their importance in a document.
     + **Word Embeddings**: Techniques like Word2Vec, GloVe, or BERT embeddings to capture semantic meaning.
     + **Bag of Words (BoW)**: Represents text as a vector of word counts or frequencies.
5. **Model Training and Evaluation**
   * Train classification models using supervised learning algorithms like:
     + Logistic Regression
     + Support Vector Machines (SVM)
     + Random Forests
     + Neural Networks (e.g., LSTM, BERT)
   * Split data into training and testing sets for performance evaluation.
   * Evaluate model performance using metrics such as accuracy, precision, recall, and F1-score.
6. **Deployment**
   * Implement the trained model in a real-time application, such as a web or mobile app, to scan comments for inappropriateness.
   * Integrate APIs or frameworks like Flask or FastAPI for seamless user interaction.

**Materials**

1. **Datasets**
   * Jigsaw Toxic Comment Classification dataset (available on Kaggle).
   * Custom datasets curated from online platforms like Reddit or Twitter (ensure compliance with ethical guidelines and privacy policies).
2. **Software and Tools**
   * **Programming Language**: Python for data preprocessing and model development.
   * **Libraries**:
     + NLP: NLTK, spaCy, TextBlob
     + Feature Extraction: scikit-learn, gensim
     + Deep Learning: TensorFlow, PyTorch
   * **Annotation Tools**: Doccano, Prodigy, Label Studio
   * **Deployment Frameworks**: Flask, FastAPI
3. **Hardware Requirements**
   * A computer with sufficient processing power (preferably with GPU support for deep learning models).
4. **Ethical Considerations**
   * Ensure data privacy and anonymity while collecting user comments.
   * Avoid introducing bias during annotation by creating balanced datasets representing diverse perspectives.

This structured approach ensures a methodical and efficient development process for building an **Inappropriate Comments Scanner**, paving the way for a robust and scalable solution.

**Code:**

# Importing required libraries

import pandas as pd

import numpy as np

from sklearn.model\_selection import train\_test\_split

from sklearn.feature\_extraction.text import TfidfVectorizer

from sklearn.linear\_model import LogisticRegression

from sklearn.metrics import accuracy\_score, classification\_report

# Step 1: Load Dataset

# Replace 'path\_to\_dataset.csv' with your dataset path

data = pd.read\_csv("path\_to\_dataset.csv")

# Assuming the dataset has 'comment\_text' and 'toxic' columns

comments = data['comment\_text']

labels = data['toxic'] # 1 for inappropriate, 0 for appropriate

# Step 2: Preprocessing Function

import re

from nltk.corpus import stopwords

from nltk.stem import WordNetLemmatizer

from nltk.tokenize import word\_tokenize

stop\_words = set(stopwords.words('english'))

lemmatizer = WordNetLemmatizer()

def preprocess\_text(text):

# Remove special characters and numbers

text = re.sub(r"[^a-zA-Z\s]", "", text)

# Tokenization

tokens = word\_tokenize(text.lower())

# Remove stopwords and lemmatize

tokens = [lemmatizer.lemmatize(word) for word in tokens if word not in stop\_words]

return " ".join(tokens)

# Apply preprocessing

comments\_cleaned = comments.apply(preprocess\_text)

# Step 3: Feature Extraction using TF-IDF

tfidf = TfidfVectorizer(max\_features=5000)

X = tfidf.fit\_transform(comments\_cleaned).toarray()

# Step 4: Split Data into Training and Testing Sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, labels, test\_size=0.2, random\_state=42)

# Step 5: Train Logistic Regression Model

model = LogisticRegression()

model.fit(X\_train, y\_train)

# Step 6: Evaluate the Model

y\_pred = model.predict(X\_test)

print("Accuracy:", accuracy\_score(y\_test, y\_pred))

print("Classification Report:\n", classification\_report(y\_test, y\_pred))

# Step 7: Real-Time Inference

def predict\_inappropriateness(comment):

processed\_comment = preprocess\_text(comment)

vectorized\_comment = tfidf.transform([processed\_comment])

prediction = model.predict(vectorized\_comment)

return "Inappropriate" if prediction == 1 else "Appropriate"

# Example Usage

new\_comment = "You are a horrible person!"

result = predict\_inappropriateness(new\_comment)

printf ("The comment '{new\_comment}' is classified as: {result}")

**Advantages of the Inappropriate Comment Scanner**

1. **Improved Online Safety**
   * Helps create a safer digital environment by detecting and mitigating offensive, abusive, or harmful language in real-time.
2. **Scalability and Automation**
   * Automates the moderation process, reducing the dependency on manual reviews for large volumes of comments.
3. **Cost-Effective**
   * Reduces the need for extensive human moderation teams, cutting operational costs for online platforms.
4. **Real-Time Detection**
   * Capable of scanning and classifying comments instantaneously, ensuring inappropriate content is addressed promptly.
5. **Customizable and Domain-Specific**
   * Can be tailored to meet the specific needs of various platforms, industries, or languages. For example, it can be adapted to focus on hate speech, workplace harassment, or other specific use cases.
6. **Promotes Inclusivity**
   * Encourages positive interactions and fosters inclusivity by discouraging toxic behavior.
7. **High Accuracy with Advanced Models**
   * With NLP advancements (e.g., BERT, GPT-based models), the scanner can achieve high accuracy, reducing false positives and negatives.
8. **Ease of Integration**
   * Can be integrated into existing platforms through APIs or frameworks, making it a seamless addition to moderation tools.
9. **Support for Multilingual Content**
   * Advanced NLP models enable support for detecting inappropriate content in multiple languages, catering to global audiences.
10. **Data-Driven Insights**
    * Provides insights into user behavior and patterns of inappropriate content, helping platform owners to develop better content policies.
11. **Encourages Responsible Behavior**
    * The presence of a scanner acts as a deterrent, encouraging users to be more mindful of their language.
12. **Flexible and Upgradable**
    * The system can be continuously improved with updated datasets, new features, and evolving NLP models to keep up with trends in online communication.

This tool not only enhances user experiences but also plays a crucial role in maintaining the integrity and reputation of online platforms.

**Future Predictions for the Inappropriate Comment Scanner Project**

1. **Advanced Context Understanding**
   * **Development of Context-Aware Models: Future systems will better understand the nuances of context, sarcasm, and idiomatic expressions, reducing false positives and negatives.**
   * **Integration of transformer-based models (e.g., GPT, BERT, RoBERTa) will allow more accurate interpretations of conversational intent.**
2. **Multilingual and Multimodal Capabilities**
   * **The scanner will expand to support multiple languages and dialects, catering to a global audience.**
   * **Multimodal processing will enable the detection of inappropriate content in text, images, videos, and audio combined.**
3. **Integration with Social Media Platforms**
   * **Seamless integration with major social media platforms and comment sections of blogs, forums, and news websites will become common.**
   * **Real-time detection and immediate content moderation will enhance the user experience.**
4. **Personalized Moderation**
   * **Users may be able to set personalized filters to define what they consider inappropriate, tailoring the experience to individual preferences.**
5. **Ethical AI and Bias Reduction**
   * **The incorporation of fairness-focused algorithms will ensure the scanner is unbiased across different demographics, promoting inclusivity.**
   * **Ethical considerations will play a significant role in designing models to avoid censorship or over-moderation.**
6. **Integration with Regulatory Compliance**
   * **The system will help organizations comply with stricter regulatory frameworks concerning online safety, such as GDPR, CCPA, or other regional content moderation laws.**
7. **Proactive Behavior Modeling**
   * **Predictive analytics will allow the scanner to identify trends in inappropriate behavior, helping platforms address issues before they escalate.**
8. **Gamification of Moderation**
   * **Community moderation tools integrated with the scanner could gamify the process, rewarding users for flagging inappropriate content.**
9. **Enhanced Real-Time Scalability**
   * **With advancements in cloud computing and distributed systems, future scanners will handle exponentially higher comment volumes in real time.**
10. **AI Collaboration with Human Moderators**
    * **The scanner will act as an assistant to human moderators, flagging potential issues while leaving nuanced decisions to humans.**
    * **This hybrid approach will balance efficiency and ethical oversight.**
11. **Sentiment Analysis and Community Insights**
    * **By incorporating sentiment analysis, the scanner will help monitor the overall mood of online discussions, giving community managers actionable insights.**
12. **Integration with Virtual and Augmented Reality**
    * **As AR/VR platforms grow, inappropriate behavior detection in virtual spaces (e.g., metaverse interactions) will become a critical use case.**
13. **Focus on Mental Health and Well-being**
    * **Enhanced systems will not only block inappropriate comments but also identify signs of distress or mental health concerns, offering supportive interventions where appropriate.**
14. **Global Standardization**
    * **Industry-wide standards for detecting and categorizing inappropriate comments will emerge, enabling interoperability and shared learning among platforms.**
15. **Custom AI Models for Organizations**
    * **Companies will have AI systems trained specifically on their data, ensuring context-relevant moderation aligned with their policies and branding.**

**Long-Term Vision**

**In the long run, the Inappropriate Comment Scanner will be a vital tool in fostering healthier digital communities. Its capabilities will evolve to include real-time, nuanced, and context-aware moderation across various content types and platforms, ensuring that the internet remains a safe and productive space for everyone.**

**Conclusion**

**The Inappropriate Comment Scanner is a powerful application of Natural Language Processing (NLP) that addresses the growing need for automated moderation of online content. By leveraging advanced techniques such as text preprocessing, feature extraction, and machine learning-based classification, this project demonstrates the potential to identify and mitigate harmful language effectively.**

**The system not only automates the detection of inappropriate comments but also enhances online safety, promotes inclusivity, and fosters positive interactions in digital spaces. Its scalability, multilingual support, and ability to integrate with various platforms make it a versatile tool for moderating large-scale user-generated content.**

**As the project evolves, advancements in AI, context understanding, and ethical design will further improve its accuracy and fairness. The incorporation of real-time detection, multimodal capabilities, and predictive analytics will ensure its relevance in diverse applications, from social media moderation to workplace communication tools.**

**Ultimately, the Inappropriate Comment Scanner is a testament to how technology can be harnessed to create safer, more respectful, and constructive online communities, paving the way for a more inclusive digital future.**