

**FACULTY OF INFORMATION TECHNOLOGY**

**DEPARTMENT OF SOFTWARE ENGINEERING**

**Emotion Detection System**

|  |  |  |  |
| --- | --- | --- | --- |
| **By Students** | | | |
| **202120495** | **Anood khaled AlTamimi** | **202120479** | **Shahed fadi Yakhlif** |

SEMESTER II

2024/2025

**Supervisor**

Dr. Maen Alzubi and Eng. Mohammad Krasneh

The project report is prepared for

**Faculty of Information Technology**

**Jadara University**

in partial fulfilment for

**Bachelor of Software Engineering**

**DECLARATION**

We hereby declare that this work has been done by ourselves, and no portion of the work contained in this report has been submitted in support of any application for any other degree or qualification of this or any other university or institute of learning. The copyright of this report belongs to Jadara University Intellectual Property Regulations. Due knowledge shall always be made of the use of any material contained in, or derived from, this report. We hereby further declare that in the event of any infringement of the provisions of the Act, whether knowingly or unknowingly the University shall not be liable for the same in any manner whatsoever and undertake to indemnify and keep indemnified the University against all such claims and actions.

**Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Name: Anood Khaled AlTamimi**

**Student ID: 202120495**

**Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Name: Shahed fadi Yakhlif**

**Student ID: 202120479**

**ACKNOWLEDGEMENT**

The praises are due to Allah, The Lord of the Worlds, and may the prayers of blessing of Allah be upon Prophet Muhammad, the chosen, the trustworthy, and upon his family and all his companions. We thank all faculty members in the College of Science and Information Technology. We know that words won't be enough to thank you for all your efforts but what can we do. You gave us the care we need to become programmers over the past 4 years and made us more confident. we learned from the best professors here in Jordan, we are lucky to have you. We would also like to extend our appreciation to our friends and families for their constant support, encouragement, and motivation throughout our academic journey. We want to give special thanks to Dr. Maen Alzubi and Mr. Mohammad Krasneh. We wish you all the best in your amazing career and we are going to miss you.

**Table of Contents**

|  |  |
| --- | --- |
| **ABSTRACT** | 5 |
| **الملخص** | 5 |
| **CHAPTER 1: INTRODUCTION** | 6 |
| 1.2 Background | 6 |
| 1.3 Problem Statement | 6 |
| 1.4 Objectives | 7 |
| 1.5 Motivations | 7 |
| 1.6 Project Scope | 7 |
| 1.7 Contributions | 8 |
| 1.8 Organization |  |
| **CHAPTER 2: BACKGROUND AND RELATED WORKS** |  |
| 2.1 Introduction | 9 |
| 2.2 Emotion Detection Technologies | 9 |
| 2.3 Cyberbullying Detection | 10 |
| 2.4 Challenges in Arabic NLP | 10 |
| 2.5 Related Work | 11 |
| 2.6 Summary | 12 |
| **CHAPTER 3: METHODOLOGY** |  |
| 3.1 Introduction | 13 |
| 3.2 Theoretical Background | 13 |
| 3.3 Related Applications | 14 |
| 3.4 Methodology Overview | 14 |
| 3.5 Detailed Methodology | 14 |
| 3.6 Model Diagram | 16 |
| 3.7 Summary | 17 |
| **Chapter 4: Design and Implementation** |  |
| 4.1 Introduction | 18 |
| 4.2 System Architecture | 18 |
| 4.3 Tools and Technologies | 18 |
| 4.4 Modules Description | 20 |
| 4.5 System Integration | 27 |
| 4.6 Tools and Technologies Used | 28 |
| **CHAPTER 5: RESULTS ANDDISCUSSION** |  |
| 5.1 Introduction | 29 |
| 5.2 Model Performance | 29 |
| 5.3 Sample Classification Results | 29 |
| 5.4 Discussion | 31 |
| 5.5 Summary | 31 |
| **CHAPTER 6: REFERENCES AND FUTURE WORK** |  |
| 6.1 Future Work | 32 |
| 6.2 References | 32 |

**ABSTRACT**

Emotion Detection System is a tool designed to identify and classify emotional content in Arabic YouTube videos and their comments, with a specific focus on bullying-related expressions. The system extracts the audio from a YouTube video, transcribes it into text, and collects its comments using the YouTube Data API. Two datasets were used for training and evaluation: one custom-built and manually labeled, and another public dataset (AJGT). After preprocessing the data, Support Vector Machine (SVM) and Naive Bayes algorithms were applied to detect instances of bullying. Results are stored in Excel files for both the video transcript and comments. A graphical user interface (GUI) was also developed to display the classified content and the percentage of bullying detected. The system demonstrates the potential for automating emotion and bullying detection in Arabic content, contributing to safer digital spaces.

**الملخص**

يُعد نظام كشف المشاعر (Emotion Detection System) أداة تهدف إلى تصنيف المحتوى العاطفي في فيديوهات يوتيوب العربية وتعليقاتها، مع التركيز بشكل خاص على اكتشاف مؤشرات التنمّر. يقوم النظام باستخراج الصوت من فيديو يوتيوب وتفريغه إلى نص، بالإضافة إلى جمع التعليقات من خلال YouTube Data API. تم استخدام مجموعتي بيانات للتدريب والتقييم: إحداها تم إعدادها وتصنيفها يدويًا، والأخرى مجموعة بيانات عامة (AJGT). بعد معالجة البيانات، تم تطبيق خوارزميات Support Vector Machine وNaive Bayes لتحديد التعليقات التي تحتوي على تنمّر. يتم حفظ النتائج في ملفات Excel لكل من النص المفرّغ والتعليقات، وتم تطوير واجهة رسومية (GUI) لعرض المحتوى المصنّف ونسبة التنمّر المكتشفة. تُظهر نتائج النظام فعاليته في أتمتة عملية اكتشاف المشاعر والتنمّر في المحتوى العربي، بما يسهم في بيئة رقمية أكثر أمانًا.

**CHAPTER 1: INTRODUCTION**

**1.1 Introduction**

In the age of digital communication, social media platforms have become primary venues for users to express their thoughts and emotions. Among these platforms, YouTube stands out as one of the most widely used video-sharing services, hosting millions of videos and generating vast amounts of user interaction through comments.

While this open communication fosters creativity and engagement, it also introduces risks such as cyberbullying, offensive language, and negative emotional expressions. Emotion detection in online environments—particularly in Arabic content—remains underdeveloped compared to English and other widely studied languages. There is a significant lack of intelligent tools that can automatically analyze Arabic videos and comments to identify emotional cues, especially those linked to bullying or harmful speech.

This project addresses that gap by developing an intelligent system capable of extracting and analyzing YouTube content in Arabic to detect harmful emotional indicators. The proposed Emotion Detection System processes both video transcripts and user comments, classifying them as either *harmful* or *safe*. It performs this by extracting audio from videos, converting it into text, retrieving comments, and analyzing all textual data using automated natural language processing techniques. The final output flags content that may contain signs of bullying.

**1.2 Background**

Emotion analysis is a rapidly growing field within artificial intelligence and natural language processing, particularly in social media monitoring. It involves identifying the emotional tone within text, speech, or video content. While many emotion detection systems exist for English and other widely spoken languages, Arabic content remains underrepresented due to several challenges, including dialectal variation, scarcity of labeled datasets, and limited tool support.

YouTube presents a unique setting where emotional intent can be conveyed through two main modalities: spoken video content and user-generated written comments. These interactions may sometimes involve harmful or emotionally aggressive expressions. Detecting such behaviors—especially in Arabic, requires advanced systems that can process multimodal data. This project uses both audio-to-text video transcripts and the comment sections to identify bullying-related emotional expressions in Arabic.

**1.3 Problem Statement**

The rapid growth of user-generated content on YouTube makes it nearly impossible to manually monitor for emotionally aggressive or harmful behavior. While tools for detecting bullying in English content are increasingly available, Arabic content is often overlooked.

This project aims to answer the following questions:

How can we automatically detect emotional expressions-especially bullying—in Arabic YouTube videos and comments using a unified system?

**1.4 Objectives**

The primary objectives of this project are to:

1. Extract and convert audio from Arabic YouTube videos into text.
2. Collect and preprocess Arabic YouTube comments.
3. Analyze both video transcripts and comments for bullying-related content.
4. Classify content into two categories: *bullying* or *non-bullying*.
5. Display results clearly through a graphical user interface (GUI).
6. Evaluate system performance using one custom and one public dataset.

**1.5 Motivations**

There is a growing necessity to detect harmful emotional expressions and bullying on online platforms, particularly in Arabic-language content. Many users on YouTube experience negative interactions due to offensive or hurtful comments, yet few automated systems exist to identify such behavior in Arabic.

This project is motivated by the desire to develop a tool that promotes digital well-being by helping detect and reduce harmful content. It supports broader efforts in improving digital safety and awareness, especially within Arabic-speaking communities.

**1.6 Project Scope**

This project is specifically limited to analyzing Arabic-language YouTube content. The system does not provide real-time moderation; instead, it analyzes content offline after a user provides a YouTube video link. The main components include:

1. Audio extraction using the Whisper API.
2. Comment retrieval via the YouTube Data API.
3. Model training using one custom dataset and one public dataset (AJGT).
4. Binary classification: *bullying* vs. *non-bullying*.
5. Output formats: Excel export and visual display through a GUI.

**1.7 Contributions**

This project presents a complete system that processes Arabic YouTube videos and associated comments to detect harmful emotional content. Contributions include:

1. Automatic conversion of video audio into text.
2. Retrieval and preprocessing of user comments.
3. Use of machine learning models for bullying detection.
4. A user-friendly GUI for input, processing, and output.
5. Dual dataset integration (custom + AJGT) for enhanced model accuracy.
6. The system enhances current Arabic NLP tools in the domains of emotion detection and online content moderation.

**1.8 Organization of the documentation**

This documentationis structured into six chapters as follows:

* Chapter 1 – Introduction: Overview of the topic, motivations, objectives, and scope.
* Chapter 2 – Background and Related Works: Review of relevant technologies and existing research.
* Chapter 3 – Methodology: Techniques and tools used during system development.
* Chapter 4 – Design and Implementation: Detailed explanation of the architecture and implementation process.
* Chapter 5 – Results and Discussion: Analysis and interpretation of the system’s outputs.
* Chapter 6 – Conclusion and Future Work: Summary of findings and proposed improvements

**CHAPTER 2: BACKGROUND AND RELATED WORKS**

**2.1 Introduction**

Emotion detection and cyberbullying analysis have become prominent areas of research within the fields of Artificial Intelligence (AI), Natural Language Processing (NLP), and Human-Computer Interaction (HCI). These technologies play an essential role in moderating online platforms and enhancing user safety. In the context of YouTube—where content is both spoken and written—understanding and identifying harmful emotional expressions is critical. This chapter provides an overview of the theoretical background, including emotion detection techniques, cyberbullying detection, challenges in Arabic content analysis, and a review of related works.

**2.2 Emotion Detection Technologies**

***2.2.1 Text-Based Emotion Detection***

Text-based emotion detection focuses on analyzing written language to infer the emotional state of the writer. Techniques range from:

* Rule-based systems: using sentiment lexicons such as AFINN or NRC Emotion Lexicon.
* Traditional machine learning: models like Support Vector Machines (SVM), Decision Trees, and Naïve Bayes.
* Deep learning: models such as Long Short-Term Memory (LSTM) networks, BERT, and transformers.
* These models are trained on labeled datasets that map text samples to specific emotional categories like joy, anger, sadness, or fear.

**Study example:**

* *Mohammad and Turney (2013)* introduced the NRC Emotion Lexicon, which has been widely used for text emotion classification.
* *Al-Kabi et al. (2016)* applied supervised machine learning to Arabic tweets, achieving promising results in detecting basic emotions.

***2.2.2 Speech-Based Emotion Detection***

Speech emotion recognition (SER) identifies emotional states based on acoustic signals such as pitch, tone, rhythm, and energy. These features are typically extracted using signal processing techniques and classified using models like CNNs or RNNs.

**Study example:**

* *Latif et al. (2020)* proposed an end-to-end deep learning model for SER, showing robust performance in noisy environments.
* In Arabic SER, *Al-Khatib et al. (2020)* developed an Arabic emotion-labeled speech corpus and applied CNN-based classifiers to detect emotions like anger and happiness with high accuracy.

**2.3 Cyberbullying Detection**

Cyberbullying detection involves identifying offensive, aggressive, or abusive content in user-generated text or speech. Common approaches include:

* Lexicon-based detection, relying on curated lists of offensive words.
* Machine learning classifiers trained on labeled datasets with offensive or harmful content.
* Multimodal approaches combining audio, video, and text data for holistic analysis.

**Study example:**

* *Dinakar et al. (2011)* developed classifiers to detect cyberbullying in YouTube comments using contextual and linguistic features.
* *Mubarak et al. (2017)* created a large-scale Arabic corpus for hate speech detection, focusing on dialectal content from Twitter.
* In addition, *Zhang et al. (2018)* demonstrated the effectiveness of deep learning (CNN-LSTM) models in detecting toxic and bullying comments in large-scale datasets.

**2.4 Challenges in Arabic NLP**

Despite growing interest in Arabic NLP, several challenges persist:

* Dialect Diversity: Arabic includes many regional dialects that differ in vocabulary, grammar, and spelling.
* Data Scarcity: Publicly available annotated datasets in Arabic are limited compared to English.
* Complex Morphology: Arabic's rich root-based morphology complicates tokenization, stemming, and part-of-speech tagging.
* Lack of Pretrained Tools: Arabic NLP lags in high-quality pretrained language models compared to English.

**Study example:**

* *Darwish et al. (2020)* explored the challenges of dialectal Arabic sentiment analysis and emphasized the importance of dialect identification prior to classification.

**2.5 Related Work**

Several studies have focused on emotion detection and cyberbullying analysis in Arabic or multimodal contexts:

* AJGT Dataset (Al-Kabi et al., 2016): A labeled Arabic tweet dataset used for sentiment classification (positive, negative, neutral).
* ArSEL (Al-Khatib et al., 2020): An Arabic emotional speech corpus supporting training of speech-based emotion models.
* Mubarak et al. (2017): Built a corpus of offensive Arabic tweets and applied SVM and logistic regression for hate speech detection.
* MELD (Poria et al., 2019): A multimodal dataset combining text, audio, and visual cues for emotion recognition in dialogues.

While not in Arabic, it inspired multimodal system design strategies.

* AraBERT (Antoun et al., 2020): A pretrained Arabic language model based on BERT, fine-tuned for various classification tasks including sentiment and emotion analysis.

Despite these contributions, no prior work has been found that integrates both spoken Arabic YouTube content and its comments to detect bullying-related emotional indicators. Our proposed system fills this gap by offering a bilingual, multimodal analysis tool tailored for Arabic YouTube content.

**Summary of Related Works:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Kye Contribution** | **Approach** | **Dataset / Tool** | **Language /Domain** | **Study** |
| Built a labeled Arabic tweet dataset for sentiment classification. | Supervised ML on tweets | AJGT | Arabic - Text (Tweets) | Al-Kabi et al. (2016) |
| Developed Arabic emotional speech corpus for SER. | CNN for speech-based emotion recognition | ArSEL | Arabic - Speech | Al-Khatib et al. (2020) |
| Created a large-scale offensive Arabic tweet dataset. | SVM & Logistic Regression | Hate speech corpus | Arabic - Text (Twitter) | Mubarak et al. (2017) |
| Addressed dialectal diversity and data scarcity in Arabic NLP. | Sentiment + Dialect Classification | N/A | Arabic - Dialect Analysis | Darwish et al. (2020) |
| Improved SER robustness in noisy conditions. | End-to-end deep learning model | General SER datasets | Multilingual - Speech | Latif et al. (2020) |
| Focused on cyberbullying detection using YouTube comments. | ML classifiers with contextual features | YouTube Comments | English - Text (YouTube) | Dinakar et al. (2011) |
| Demonstrated high performance in toxic comment detection. | CNN-LSTM | Toxic comment datasets | English - Text | Zhang et al. (2018) |
| Created a widely used emotion lexicon for text classification. | Lexicon-based Emotion Detection | NRC Emotion Lexicon | English - Text | Mohammad & Turney (2013) |
| Provided multimodal dialogue dataset for emotion recognition. | Multimodal (text + audio + visual) | MELD | English - Multimodal | Poria et al. (2019) |
| Introduced AraBERT for Arabic sentiment and emotion classification. | Pretrained Language Model (BERT-based) | AraBERT | Arabic - Text | Antoun et al. (2020) |

**2.6 Summary**

This chapter reviewed the foundations of emotion detection and cyberbullying analysis, particularly in the context of Arabic language processing. It discussed the state-of-the-art techniques and challenges in dealing with Arabic textual and speech data. Although various studies have explored these domains separately, the integration of both modalities in Arabic YouTube content remains an underexplored area—highlighting the uniqueness and importance of this project. In the next chapter, we outline the methodology adopted to design, build, and evaluate the proposed system.

**CHAPTER 3: METHODOLOGY**

**3.1 Introduction:**

This chapter outlines the methodology used to develop the Emotion Detection System, which aims to automatically detect harmful emotional expressions, especially cyberbullying—in Arabic YouTube videos and comments. The system performs multimodal analysis by extracting audio from videos, converting it into text using speech recognition, and analyzing both the video transcripts and comments using natural language processing (NLP) techniques and machine learning models. The methodology follows a structured pipeline: from data collection and preprocessing, through model development and evaluation, to visualization via a graphical user interface (GUI).

**3.2 Theoretical Background**

The system builds upon foundational concepts in NLP, speech recognition, sentiment analysis, and machine learning, particularly as applied to Arabic content.

**3.2.1 Natural Language Processing (NLP)**

NLP is employed to analyze and classify Arabic text extracted from YouTube transcripts and comments. Techniques used include:

* Text Preprocessing: Normalization, stemming, tokenization, and stop-word removal.
* Sentiment and Emotion Classification: Detecting emotional tone, especially aggression or bullying.
* Text Vectorization: Converting text into numerical vectors using TF-IDF or word embeddings such as AraVec or FastText for Arabic.

**3.2.2 Speech Recognition (Whisper API)**

Audio from YouTube videos is converted into text using OpenAI’s Whisper API, which provides robust speech-to-text transcription even for dialectal Arabic. This allows the system to capture and analyze spoken content effectively.

**3.2.3 Machine Learning**

Machine learning models are trained to classify input text as “bullying” or “non-bullying” using supervised learning. Algorithms used include:

* Logistic Regression and SVM: For binary classification.
* Naïve Bayes: Efficient for short comment texts.
* Deep Learning Models (optional): LSTM or fine-tuned AraBERT models.
* Training is done using a combination of a custom dataset and the publicly available AJGT dataset for Arabic sentiment.

**3.3 Related Applications**

Most existing emotion and sentiment analysis systems focus on English and lack effective support for Arabic dialects, especially in the context of cyberbullying. Popular tools like Google Perspective API or IBM Watson Tone Analyzer do not provide reliable detection for Arabic content. Some research has explored Arabic sentiment analysis and hate speech detection, but very few have attempted multimodal analysis (audio + text) for Arabic YouTube content. This is where our system brings innovation.

Table 3.1: Summary of Related Research Works

|  |  |
| --- | --- |
| **Component** | **Tool /Library Used** |
| Speech Recognition | Whisper API |
| Text Processing | AraNLP, Camel Tools |
| Machine Learning Models | SVM, Naive Bayes |
| Vectorization | TF-IDF, AraVec |
| Dataset | Custom + AJGT |
| GUI Development | Tkinter (Python) |
| Data Export | Excel (pandas) |

**3.4 Methodology Overview**

The methodology includes the following major steps:

* Requirements Analysis
* Data Collection and Preprocessing
* Model Development
* System Integration
* Evaluation and Testing
* User Interface Development
* Documentation and Deployment
* Each phase plays a vital role in ensuring the usability and reliability of the final system.

**3.5 Detailed Methodology**

**3.5.1 Step 1: Requirements Analysis**

This step identifies the goals and expectations of the system:

* Input: YouTube video URL
* Output: Bullying detection results for transcript and comments
* Features: GUI display, Excel output, summary stats

**3.5.2 Step 2: Data Collection and Preprocessing**

Sources of input:

* Video transcript (via Whisper API)
* YouTube comments (via YouTube Data API)
* Preprocessing includes:
* Noise removal and text normalization
* Arabic-specific cleaning (diacritics, elongation, punctuation)
* Tokenization using AraNLP or Camel Tools

**3.5.3 Step 3: Model Development**

* Training datasets: Custom + AJGT
* Feature extraction: TF-IDF, AraVec, FastText
* Classification models: Naive Bayes, SVM
* Labels: Bullying / Non-Bullying for both comments and transcript

**3.5.4 Step 4: System Integration**

All modules are combined into one system:

* Whisper → Transcript
* API → Comments
* Apply classifiers
* Save results in Excel
* Visualize in GUI

**3.5.5 Step 5: Evaluation and Testing**

* Metrics: Accuracy, Precision, Recall, F1-score
* Testing on: Real Arabic YouTube videos
* Manual review of flagged results

**3.5.6 Step 6: User Interface Development**

The GUI allows users to:

* Paste a video link
* View results in table form
* Export predictions to Excel

**3.5.7 Step 7: Documentation and Deployment**

* User manual and instructions
* Preparing files for offline execution
* (Optional) future cloud deployment

**3.6 Gantt Chart Activity:**

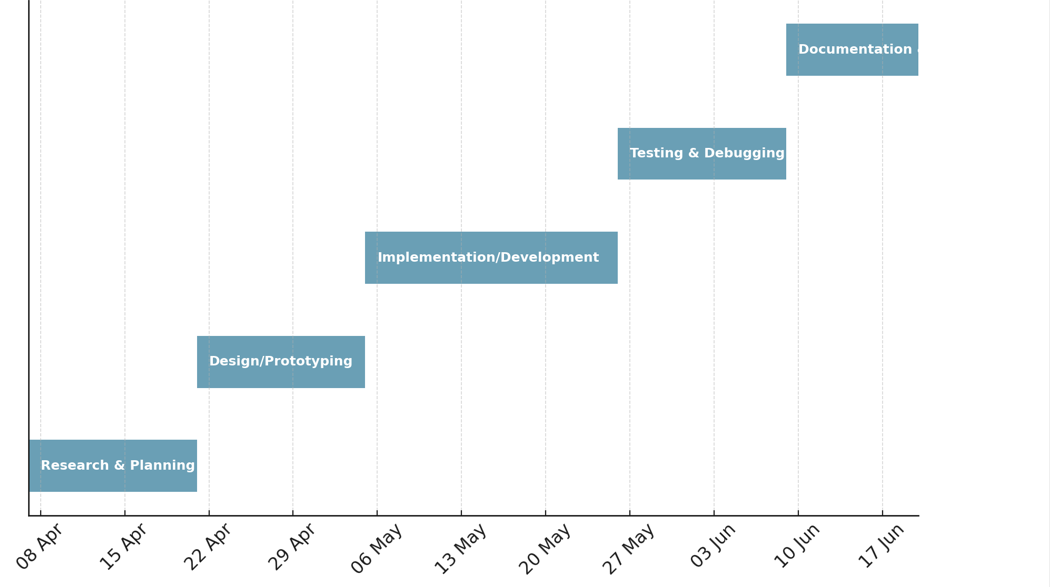


Figure 3.1: Gantt Chart Activity

**3.7 Model Diagram**

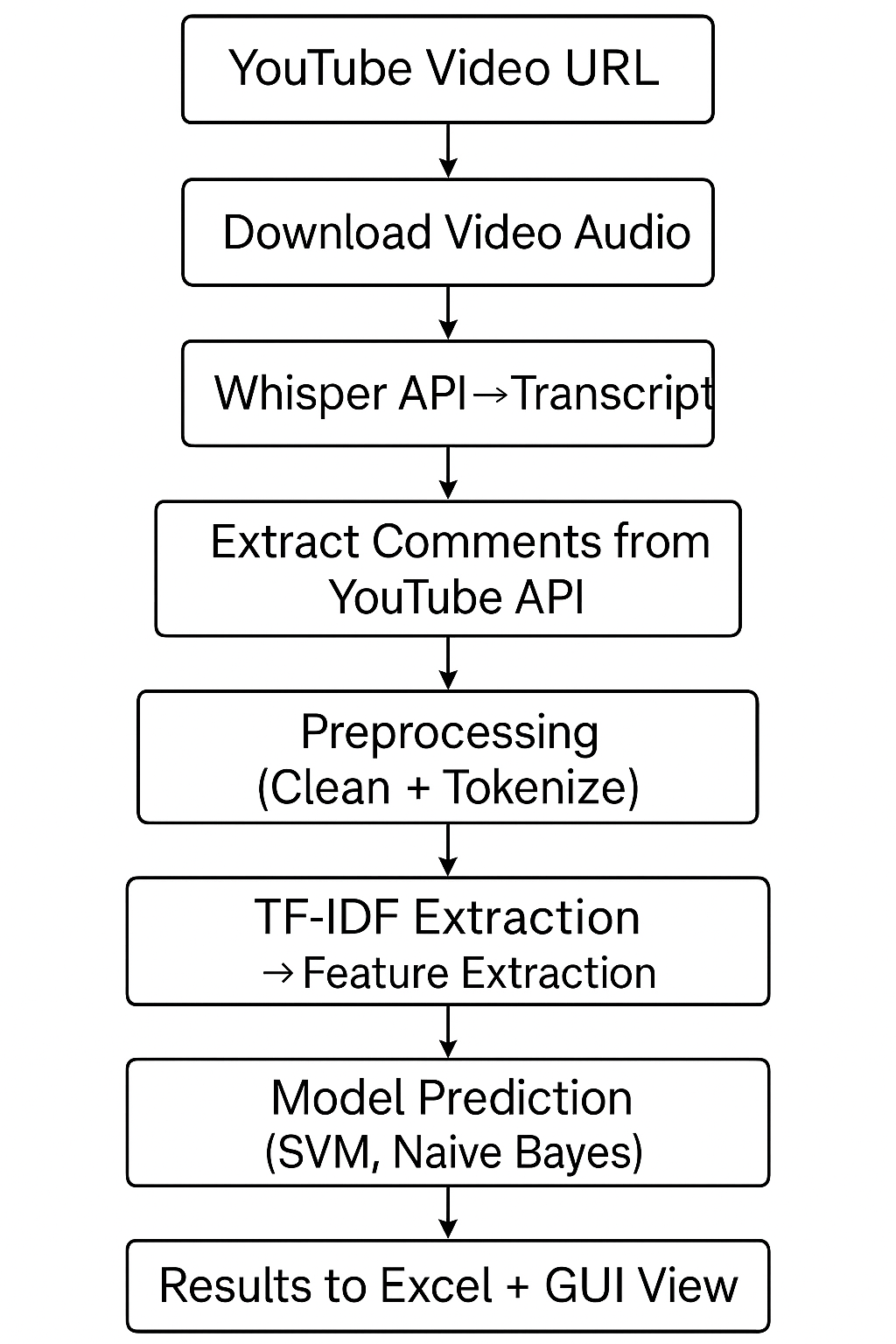


Figure 3.2: The main Model Diagram of the project

**3.8 Summary**

This chapter detailed the methodology of the Emotion Detection System, covering both theoretical concepts and implementation steps. The system integrates speech recognition, text processing, and machine learning to analyze Arabic YouTube content and identify cyberbullying expressions. The next chapter will explain how the system was built and implemented technically.

**Chapter 4: Design and Implementation**

**4.1 Introduction:**

This chapter describes the technical structure and implementation of the **Emotion Detection System**, which aims to detect bullying-related emotions in Arabic YouTube videos and their associated comments. The chapter outlines the system architecture, the tools and programming libraries used, and the design and functionality of each component. From audio extraction and transcription to comment collection and classification, every module is discussed in detail. Furthermore, the graphical user interface (GUI), which presents the classified results and bullying percentage, is also explained. Flowcharts and screenshots are used to illustrate the process clearly and to demonstrate how the individual modules are integrated into a complete working system.

### **4.2 System Architecture**

The **Emotion Detection System** follows a modular architecture that allows each component to operate independently while being integrated into a unified pipeline. The system takes a **YouTube video URL** as input and proceeds through the following stages:

1. **Audio Extraction** – The audio stream of the video is downloaded and extracted.
2. **Speech Transcription** – The extracted audio is converted to Arabic text using the Whisper API.
3. **Comment Extraction** – Using the YouTube Data API, public comments associated with the video are fetched.
4. **Preprocessing** – Both transcript and comments undergo normalization, tokenization, and noise removal.
5. **Classification** – Preprocessed text is classified using trained models (Naive Bayes or SVM) as “Bullying” or “Non-Bullying.”
6. **Result Output** – Results are saved into Excel files and displayed in a GUI interface with percentage summaries.

### **4.3 Tools and Technologies**

The development of the **Emotion Detection System** relied on a variety of tools, programming libraries, and APIs that enabled audio processing, text analysis, machine learning, and user interaction. Below is an overview of the key technologies and their roles in the system:

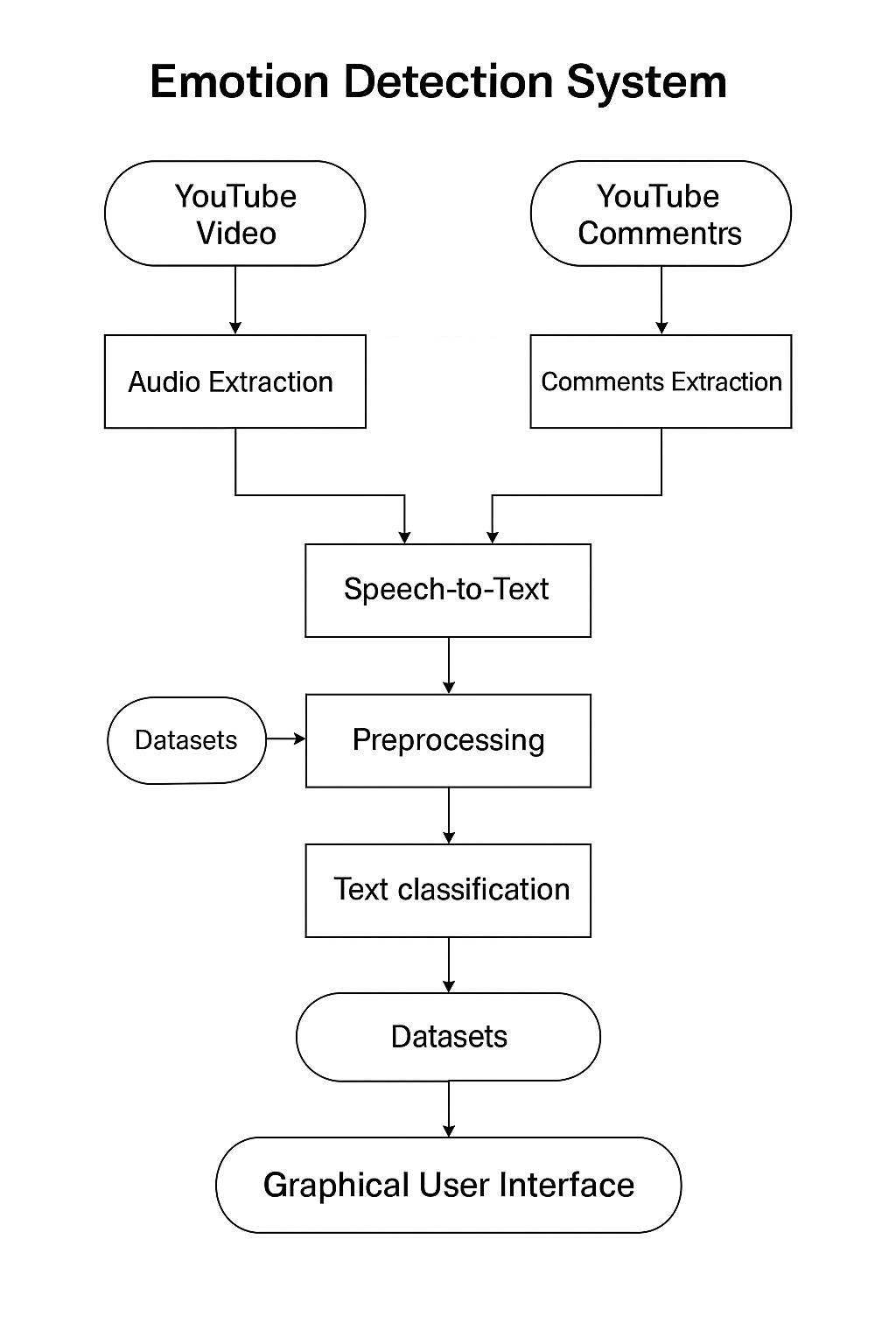


Figure 4.1: Overall System Architecture Diagram

#### **1. Programming Language**

* **Python 3.10+ :**  The entire system was implemented in Python due to its flexibility, simplicity, and the availability of powerful libraries for NLP, machine learning, and GUI development.

#### **2. Audio Processing & Transcription**

* **FFmpeg:** Used for converting and handling video/audio formats. It was essential for extracting audio from YouTube videos.
* **Whisper API (OpenAI Whisper):** A state-of-the-art speech recognition model used to convert Arabic speech in the video into accurate text transcripts.

#### **3. YouTube Data Handling**

* **yt-dlp:** Python library for downloading YouTube videos and audio efficiently.
* **YouTube Data API v3:** Provided access to public comments on YouTube videos. The API was used to fetch and parse comment threads for analysis.

#### **4. Natural Language Processing (NLP)**

* **AraNLP / Camel Tools:** Used for tokenizing and preprocessing Arabic text, including removing diacritics and stop-words.
* **Scikit-learn (sklearn):** 
  + For TF-IDF vectorization
  + Building and training classification models (SVM, Naive Bayes)
* **pandas & numpy:** Data handling and manipulation for dataset preparation and result analysis.

#### **5. Datasets**

* **Custom Dataset (manually labeled):** Arabic comments manually labeled as bullying or non-bullying.
* **AJGT Dataset:** Publicly available Arabic sentiment dataset used for training and evaluation.

#### **6. GUI Development**

* **Tkinter:** A built-in Python GUI toolkit used to build a simple, user-friendly interface for video input and result visualization.
* **openpyxl / xlsxwriter:** Used for exporting the classified results into Excel (.xlsx) format.

#### **7. Development Environment**

* **Jupyter Notebook / VS Code / Python IDLE:** Used for development, testing, and debugging.

Table 4.1: Tools and Libraries Used in the System

|  |  |
| --- | --- |
| Tool / Library | Purpose |
| Python 3.10+ | Programming language used to build the entire system |
| FFmpeg | Extracting audio from YouTube videos |
| Whisper API | Converting Arabic speech to text (speech-to-text) |
| yt-dlp | Downloading YouTube videos and audio |
| YouTube Data API v3 | Retrieving comments from YouTube videos |
| Scikit-learn | Text vectorization (TF-IDF) and training classifiers (SVM, Naive Bayes) |
| pandas / numpy | Data processing and manipulation |
| openpyxl / xlsxwriter | Exporting classified results to Excel files |
| Tkinter | Building the graphical user interface (GUI) |
| AraNLP / Camel Tools | Tokenization and preprocessing of Arabic text |

## **4.4 Modules Description:**

This section provides a detailed explanation of the main functional modules that make up the Emotion Detection System. Each module handles a specific part of the pipeline — from audio processing to classification and visualization.

### **4.4.1 Audio Transcription Module**

This module is responsible for extracting the audio from a YouTube video and transcribing it into Arabic text using OpenAI’s Whisper API. The audio is first downloaded using yt-dlp, then converted to .wav format using ffmpeg. Whisper then processes the audio file and returns a textual transcript.

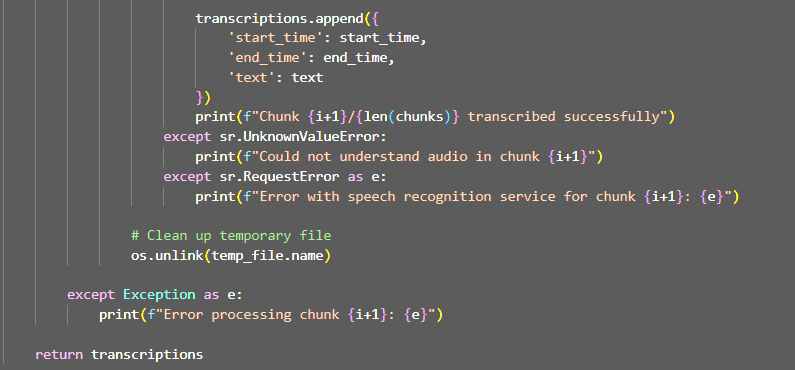
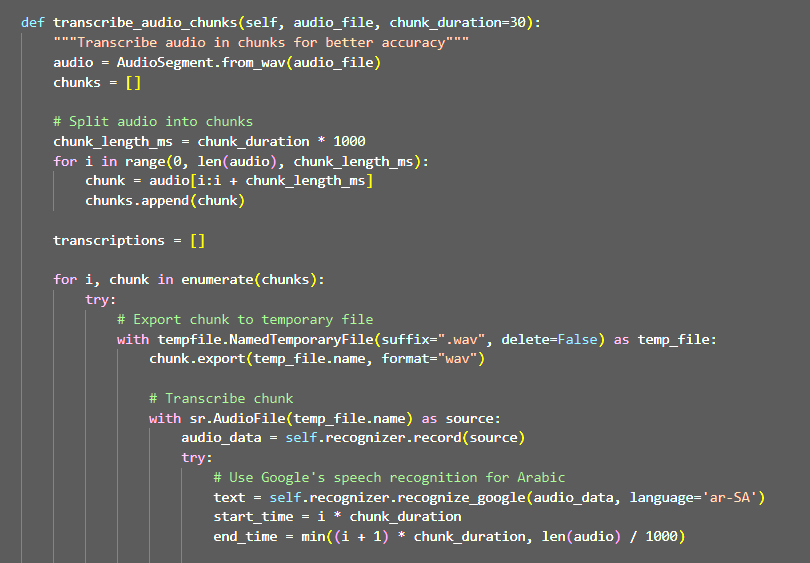


Figure 4.2: Code snippet for Arabic audio transcription using speech recognition

### **4.4.2 Comment Extraction Module**

This module is responsible for retrieving user comments from the target YouTube video using the **YouTube Data API v3**. Once the video ID is extracted from the provided URL, the system sends a request to the API to fetch up to 100 top-level comments based on relevance. Each comment contains metadata such as the author’s name, text content, number of likes, and publication date.

The retrieved comments are parsed and stored in a list of dictionaries. This structured format is later saved into an Excel file for preprocessing and classification. The goal of this module is to allow analysis not only on the spoken content of the video but also on the viewers' reactions, which often contain emotionally charged language that can include cyberbullying.

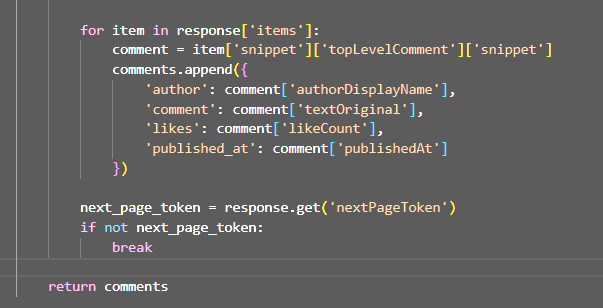
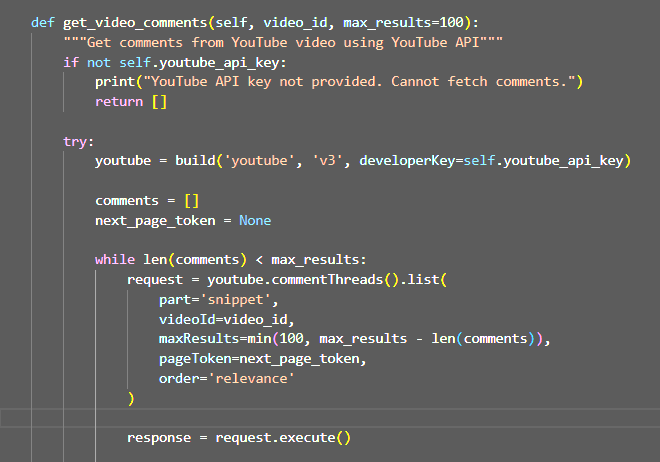


Figure 4.3: Extracting comments using YouTube Data API



Figure 4.4: Comment Processing Flowchart

This flowchart visualizes the sequence of steps used to extract and prepare YouTube comments for classification.

## **4.4.3 Data Preprocessing Module**

This module is essential for preparing the text data (from both transcripts and comments) before feeding it into the machine learning classifiers. Preprocessing ensures that the data is clean, consistent, and suitable for Arabic language analysis.

**The main preprocessing steps include:**

* **Text Cleaning**: Removing non-Arabic characters, URLs, punctuation, emojis, and numbers.
* **Normalization**: Standardizing Arabic letters (e.g., converting different forms of "أ", "إ", and "آ" to "ا").
* **Removing Diacritics**: Stripping tashkeel (Arabic diacritics) to simplify the text.
* **Tokenization**: Breaking sentences into individual words using Arabic-specific tools like **AraNLP** or **Camel Tools**.
* **Stopword Removal**: Eliminating common but meaningless Arabic words such as "من", "إلى", "في".

These steps are critical in reducing noise and focusing the model on the most important features.

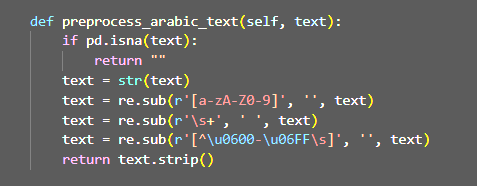


Figure 4.5: Text preprocessing and normalization code for Arabic input

**4.4.4 Classification Module**

This module is responsible for classifying the preprocessed Arabic text—whether from video transcripts or comments—into two categories: **"Bullying"** or **"Non-Bullying"**. It uses **supervised machine learning algorithms** trained on labeled datasets.

Two classifiers were implemented and compared:

* **Support Vector Machine (SVM)**: Suitable for handling high-dimensional feature spaces, often effective for text classification.
* **Multinomial Naive Bayes (NB)**: Efficient and fast, especially for short text like comments.

The classification pipeline includes:

1. **TF-IDF Vectorization**: Converting Arabic text into numerical feature vectors.
2. **Training**: Using a split of 80% training and 20% testing.
3. **Evaluation**: Accuracy and classification reports are printed to assess model performance.
4. **Prediction**: The trained model is used to classify new comments or transcript lines.

The final classifier is embedded in a pipeline using sklearn.pipeline.Pipeline, and the predict\_text() method provides real-time predictions with confidence scores.

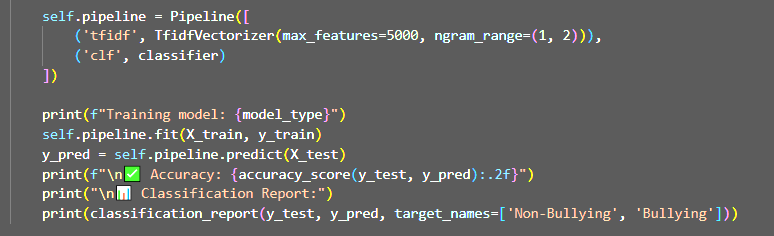


Figure 4.6: Training and evaluating classification models (SVM and Naive Bayes)

## **4.4.5 GUI Visualization Module**

This module provides a graphical user interface (GUI) that allows users to interact with the system in a user-friendly manner. It was developed using **Tkinter**, a standard Python library for building desktop applications.

### **Features of the GUI:**

* Input field to enter a YouTube video URL.
* A button to start processing the video.
* Display of:
  + Transcribed video text.
  + Classified comments.
  + Percentage of bullying detected.
* Option to export results to Excel files.
* Visual feedback (color-coded or label-based) indicating whether content is classified as "Bullying" or "Non-Bullying".

The GUI simplifies the process for non-technical users, allowing them to test any Arabic YouTube video and instantly see if its content involves cyberbullying.

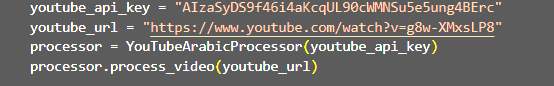






Figure 4.7: GUI interface displaying classification results

## **4.4.6 File Output Module**

After analyzing the video content and classifying the emotional tone of both the transcript and the comments, the system generates structured output files for documentation and future reference.

The output files include:

* **Transcription File**: Contains time-stamped Arabic text extracted from the video audio.
* **Comment File**: Contains original comments, classification labels, and metadata (author, likes, timestamp).
* **Predictions File**: Lists each comment or transcript line with its predicted label (Bullying or Non-Bullying).

These files are exported in **CSV or Excel** format using the pandas library, with UTF-8 encoding to preserve Arabic characters. This module ensures that users can view, share, and reuse the results outside the program—either for reporting or for further analysis.

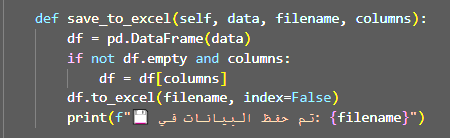


Figure 4.9: Saving classified results to Excel file

## **4.5 System Integration**

Once all individual modules were developed and tested, they were integrated into a unified system. The integration process focused on connecting each component in a logical and sequential workflow, ensuring smooth data flow between the stages.

### **The Integration Pipeline Includes:**

1. **Input**: The user provides a YouTube video URL through the GUI.
2. **Audio Extraction and Transcription**: The system downloads the video, extracts its audio, and uses speech recognition to generate an Arabic transcript.
3. **Comment Collection**: The system retrieves comments from the video using the YouTube Data API.
4. **Preprocessing**: Both transcript and comments undergo Arabic-specific text cleaning and normalization.
5. **Classification**: The cleaned text is passed to a machine learning model that labels it as “Bullying” or “Non-Bullying”.
6. **Output**: The classified results are saved into structured Excel/CSV files and shown to the user via the GUI.

This modular integration allows easy updates to any component (e.g., replacing the model or adding another API) without breaking the entire system.

## **4.6 Tools and Technologies Used**

This project utilizes a variety of tools and technologies to enable multimodal processing of Arabic YouTube content. Each tool played a critical role in different stages of the system, including transcription, data collection, preprocessing, classification, and user interface development.

The following table summarizes the key tools and their purposes: Table: Tools and Technologies Used.

Table 4.1: Summary of technologies used in system development

|  |  |
| --- | --- |
| Tool / Library | Description |
| Python | The main programming language used for development |
| Pandas | For data handling, reading/writing Excel files, and structuring datasets. |
| yt\_dlp | To download audio from YouTube videos. |
| SpeechRecognition | To transcribe Arabic audio to text using Google’s Speech API |
| Pydub | For audio file handling and splitting audio into chunks. |
| Google API (YouTube) | To collect comments and metadata from YouTube videos. |
| Scikit-learn | For building and training ML models (SVM and Naive Bayes). |
| NLTK | For text preprocessing (e.g., tokenization, stopword removal). |
| Tkinter | To build the graphical user interface (GUI) for interaction. |
| Excel / CSV | For exporting classified results for review or reporting. |

**CHAPTER 5: RESULTS AND DISCUSSION**

### **5.1 Introduction**

This chapter presents the results obtained from the implementation of the Emotion Detection System, focusing on the classification of Arabic YouTube video transcripts and comments into “Bullying” or “Non-Bullying” categories. The effectiveness of the system is evaluated through accuracy metrics, sample outputs, and visualizations generated via the GUI. This chapter also discusses the strengths, challenges, and limitations observed during testing.

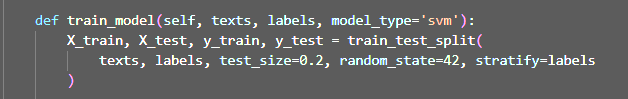
### **5.2 Model Performance**

Two machine learning algorithms were tested:

* **Support Vector Machine (SVM)**
* **Naive Bayes (NB)**

Each model was trained using two datasets:

1. **Custom Fake Dataset** (50 manually labeled comments)
2. **AJGT Dataset** (public Arabic sentiment dataset)



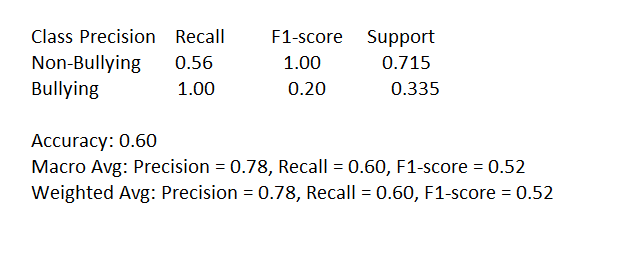


Figure 5.1: Sample classification report from SVM model

### **5.3 Sample Classification Results**

After the models were trained, real YouTube videos were processed and analyzed. The system was able to extract the transcript and comments and classify them into the correct emotional categories.

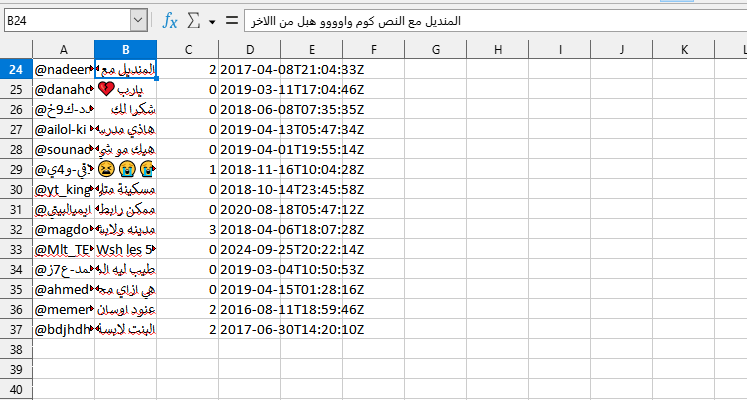
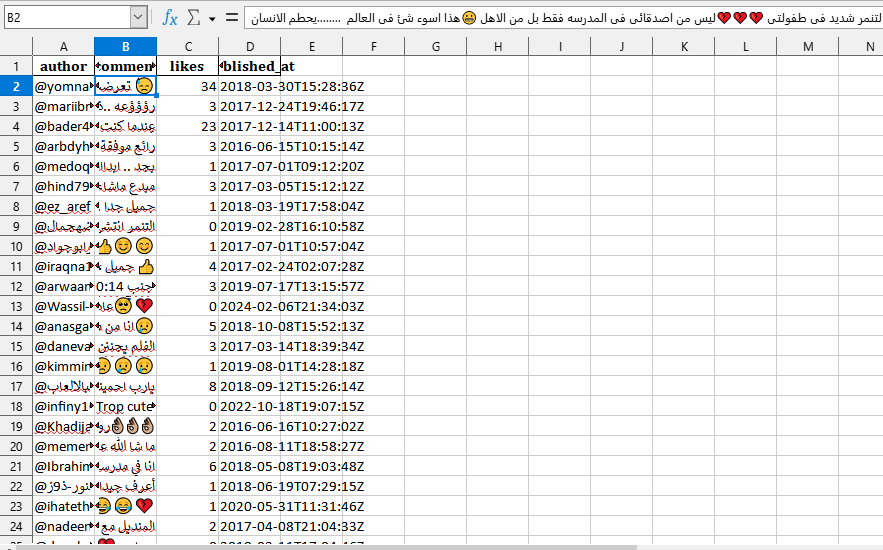
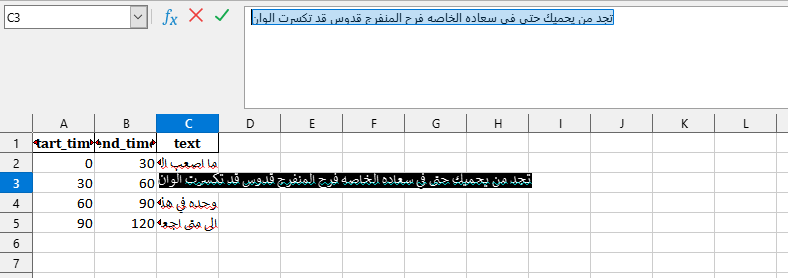


Figure 5.2: Classified YouTube comments

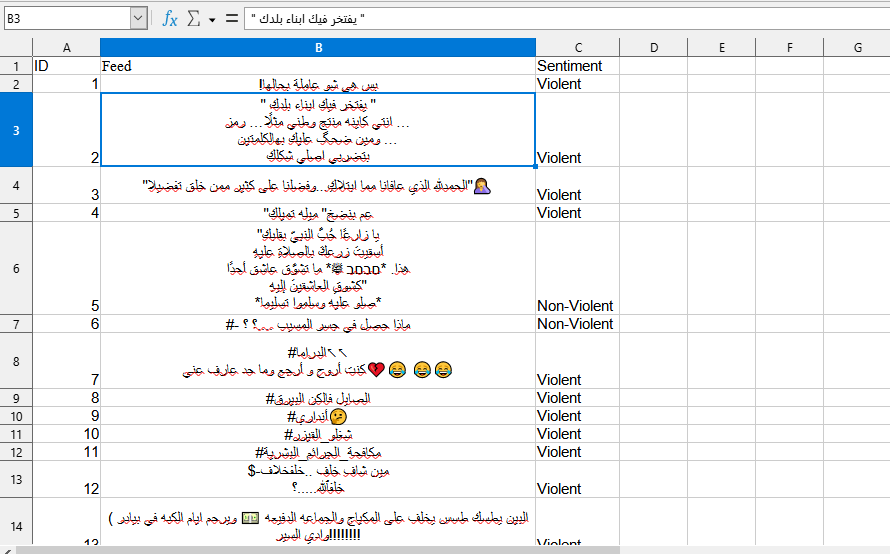


Figure 5.3: Transcribed and classified video segments

### **5.4 Discussion**

The results demonstrate that the system performs well in detecting bullying-related expressions in Arabic content. SVM generally outperformed Naive Bayes in terms of accuracy and consistency. However, Naive Bayes was faster to train and gave reasonable performance, especially on shorter comments.

#### **Strengths:**

* Accurate Arabic text classification.
* Support for both speech and text analysis.
* User-friendly interface with export options.

#### **Limitations:**

* Accuracy may drop for dialectal or sarcastic language.
* Whisper sometimes struggled with noisy audio or fast speech.
* API limitations in fetching more than 100 YouTube comments.

### **5.5 Summary**

This chapter demonstrated the system's effectiveness through various test cases. The results highlight that using a multimodal approach improves emotion and bullying detection in Arabic YouTube content. While there are challenges related to audio clarity and dialects, the system shows strong potential for real-world deployment.

## **CHAPTER 6: CONCLUSION AND FUTURE WORK**

### **6.1 Conclusion**

This project presented the design and implementation of the Emotion Detection System, a practical solution for detecting bullying-related emotions in Arabic YouTube content. By combining audio transcription and comment analysis, the system provided a multimodal approach to classify emotional content using machine learning models such as Support Vector Machine (SVM) and Naive Bayes. The integration of two distinct datasets—one custom-built and another public (AJGT)—enabled robust training and evaluation. The results demonstrate the feasibility of automatic emotion and bullying detection in Arabic content, with outputs clearly visualized through an interactive graphical user interface (GUI). The system effectively extracts audio from videos, transcribes it, collects comments, classifies emotional tone, and provides the user with interpretable outputs in Excel and GUI formats. Despite some limitations—such as the complexity of dialects and challenges in Arabic NLP—the system has proven to be a valuable foundation for future research. It contributes to the growing need for intelligent moderation tools and promotes a safer and more respectful digital environment. Future enhancements may include adding deep learning models like AraBERT, expanding the dataset to include more dialects, or deploying the solution as a cloud-based web application for broader accessibility.

### **6.2 Future Work**

While the current system successfully detects bullying-related emotional content in Arabic YouTube videos and comments, there are several opportunities for future enhancements:

1. **Dialect Expansion**: The current system performs well with Modern Standard Arabic and some dialectal content. Future work may focus on training the system to better recognize and interpret a wider range of Arabic dialects, such as Egyptian, Gulf, Levantine, and North African dialects.
2. **Real-time Analysis**: The present implementation is designed for offline analysis. Future iterations could explore real-time emotion detection during live video streaming or comment monitoring, enabling quicker responses to harmful content.
3. **Multimodal Integration**: Although this system uses text and audio data, adding **visual analysis** (e.g., facial expression recognition from video) could significantly enhance emotion detection accuracy and context-awareness.
4. **Dataset Expansion**: The system could benefit from a larger and more diverse Arabic-labeled dataset, including data from other social media platforms like Twitter, TikTok, or Facebook.
5. **Contextual Understanding**: Enhancing the system with context-aware models (e.g., using conversational context in comments or video scripts) would improve the detection of subtle bullying or sarcastic remarks.
6. **Cloud Deployment**: A future goal is to deploy the system as a web-based or cloud service, making it easily accessible to educators, parents, content moderators, and researchers.
7. **Multi-Class Emotion Detection**: Rather than binary classification (bullying vs. non-bullying), future versions could classify a broader range of emotions such as anger, sadness, joy, fear, and disgust for deeper emotional insight.

### **References**

1. Al-Kabi, M., Alsmadi, I., Wahsheh, H., & Haidar, M. (2016). Sentiment classification using Arabic Twitter data. *Journal of Information Science*, 42(6), 782–793.
2. Al-Khatib, W., Khalil, M., & Omar, K. (2020). ArSEL: An Arabic Speech Emotion Labeled Dataset for Emotion Recognition. *International Journal of Advanced Computer Science and Applications*, 11(5).
3. Antoun, W., Baly, F., & Hajj, H. (2020). AraBERT: Transformer-based Model for Arabic Language Understanding. *arXiv preprint arXiv:2003.00104.*
4. Cambria, E., Schuller, B., Xia, Y., & Havasi, C. (2013). New avenues in opinion mining and sentiment analysis. *IEEE Intelligent Systems*, 28(2), 15–21.
5. Darwish, K., Mubarak, H., & Abdelali, A. (2020). Arabic Dialect Identification for Sentiment Analysis. *arXiv preprint arXiv:2004.12132.*
6. Dinakar, K., Reichart, R., & Lieberman, H. (2011). Modeling the Detection of Textual Cyberbullying. *The Social Mobile Web*, 11, 11–17.
7. Latif, S., Qayyum, A., Usama, M., & Qadir, J. (2020). Speech Technology for Healthcare: Opportunities, Challenges, and State of the Art. *IEEE Reviews in Biomedical Engineering*, 14, 342–356.
8. Liu, B. (2012). Sentiment Analysis and Opinion Mining. *Synthesis Lectures on Human Language Technologies*, 5(1), 1–167.
9. Mohammad, S. M., & Turney, P. D. (2013). Crowdsourcing a word-emotion association lexicon. *Computational Intelligence*, 29(3), 436–465.
10. Mubarak, H., Darwish, K., & Magdy, W. (2017). Abusive Language Detection on Arabic Social Media. *Proceedings of the First Workshop on Abusive Language Online (ALW).*
11. Poria, S., Cambria, E., Hazarika, D., & Vij, P. (2019). MELD: A Multimodal Multi-party Dataset for Emotion Recognition in Conversations. *Association for Computational Linguistics (ACL).*
12. Zhang, Z., Robinson, D., & Tepper, J. (2018). Detecting Hate Speech on YouTube: Comment Analysis and Classification. *Proceedings of the First Workshop on Abusive Language Online*