

# PDF to Audiobook Converter - Project Report : 01

## Overview

The PDF to Audiobook Converter transforms PDF documents into audio files, enabling users to listen to written content while multitasking or addressing accessibility needs. The application features a graphical interface for uploading PDFs, extracting text, and converting it to speech with customizable playback options.

## Tools & Technologies

**Python** (core language), **Tkinter** (GUI), **PyPDF2** (text extraction), **gTTS/pyttsx3** (text-to-speech synthesis), **Pygame** (audio playback), plus **os**, **threading**, and **tempfile** for file management.

## Implementation

**Text Extraction:** PyPDF2 extracts text from multi-page PDFs while handling encoding issues.

**Speech Synthesis:** Integrated gTTS and pyttsx3 engines with adjustable voice selection and speech rates.

**Playback Controls:** Pygame enables play, pause, resume, and stop functionality with MP3 export capability.

**User Interface:** Tkinter-based GUI for file upload, text preview, voice/speed selection, and audio playback/saving.

**Testing Utilities:** Created `create_sample_pdf.py` for generating test documents.

**Dependencies:** Managed via `requirements.txt` for streamlined setup.

## Impact & Future Scope

This tool successfully automates PDF-to-audio conversion, benefiting students, professionals, and users with visual impairments. The modular architecture supports future enhancements including multi-language support, advanced voice customization, and cloud-based TTS integration.

**Repository:** [github.com/Ahanaf-Aziz/Elevate-Labs-Internship/Projects/PDF\\_to\\_Audiobook\\_Converter](https://github.com/Ahanaf-Aziz/Elevate-Labs-Internship/Projects/PDF_to_Audiobook_Converter)

# Fake News Detector - Project Report : 02

## Overview

The Fake News Detector employs machine learning to automatically classify news headlines as real or fake, combating misinformation through an interactive web application. The tool supports both single-headline analysis and batch processing via CSV uploads.

## Tools & Technologies

**Python** (core language), **Streamlit** (web interface), **Pandas/NumPy** (data processing), **scikit-learn** (TF-IDF vectorization and logistic regression), **Pickle** (model persistence), plus **OS** and **sh** for automation.

## Implementation

**Data Preparation:** Preprocessed labeled news datasets with lowercasing and whitespace removal.

**Feature Extraction:** TF-IDF vectorization transforms text headlines into numerical features by calculating term importance across the corpus.

**Model Training:** Logistic regression classifier trained on labeled data with evaluation metrics including accuracy, precision, recall, and F1-score.

**Model Persistence:** Trained model and vectorizer saved via Pickle for deployment reuse.

**Web Interface:** Streamlit application with three tabs—Single Prediction (individual analysis), Batch Upload (CSV processing with downloadable results), and About (documentation).

**Deployment:** `requirements.txt` manages dependencies while `setup.sh` automates environment configuration and model training.

## Impact & Future Scope

This project demonstrates practical NLP applications for misinformation detection at scale. The modular architecture enables users to critically evaluate information while supporting future enhancements like deep learning models, context-aware analysis, and synonym handling to overcome TF-IDF limitations.

**Repository:** [github.com/Ahanaf-Aziz/Elevate-Labs-Internship/Projects/Fake\\_News\\_Detector](https://github.com/Ahanaf-Aziz/Elevate-Labs-Internship/Projects/Fake_News_Detector)