Transcribe RO - Project Summary

Overview

Transcribe RO is a portable command-line tool designed to transcribe audio recordings in any language and translate them to Romanian. Built with OpenAI Whisper and designed for portability, it can run directly from a USB flash drive with minimal setup.

Key Features

Core Functionality

- ✅ **Automatic Language Detection**: Detects source language from 99+ supported languages
- ✅ **High-Quality Transcription**: Uses OpenAI Whisper AI models
- ✅ **Romanian Translation**: Automatically translates to Romanian using Google Translate
- ✅ **Multi-Format Audio Support**: MP3, WAV, M4A, FLAC, OGG, AAC, OPUS, and more
- ✅ **Multiple Output Formats**: TXT, JSON, SRT (subtitles), VTT (web subtitles)
- ✅ **Timestamp Support**: Detailed timestamps for each transcription segment
- ✅ **Portable Design**: Runs from flash drive without installation
- ✅ **User-Friendly CLI**: Simple command-line interface with helpful options

Technical Features

- 🔧 **Multiple Model Sizes**: Choose from tiny, base, small, medium, or large models
- 🔧 **GPU Support**: Optional CUDA acceleration for faster processing
- 🔧 **Offline Capable**: Works offline after initial model download
- 🔧 **Error Handling**: Graceful error handling with clear messages
- 🔧 **Cross-Platform**: Works on Windows, macOS, and Linux

Project Structure

```
transcribe_ro/
├── transcribe_ro.py          # Main application script
├── requirements.txt           # Python dependencies
├── README.md                  # Comprehensive documentation
├── QUICKSTART.md              # Quick start guide for beginners
├── TESTING.md                 # Testing guide and test plans
├── CONTRIBUTING.md            # Contribution guidelines
├── LICENSE                    # MIT License
├── .gitignore                # Git ignore rules
│
├── setup_portable.bat        # Windows portable setup script
├── setup_portable.sh         # Linux/macOS portable setup script
├── run_transcribe.bat        # Windows helper script
├── run_transcribe.sh         # Linux/macOS helper script
└── examples.sh               # Example commands
```

Technical Stack

Core Dependencies

- **OpenAI Whisper**: State-of-the-art speech recognition
- **PyTorch**: Machine learning framework
- **googletrans**: Translation service
- **FFmpeg**: Audio processing

Python Version

- Python 3.8 or higher

Use Cases

1. **Conference Recordings**: Transcribe multilingual meetings
2. **Interviews**: Convert interviews to searchable text
3. **Lectures & Education**: Create lecture transcripts
4. **Podcasts**: Generate accessible transcripts
5. **Voice Notes**: Convert voice memos to text
6. **Video Subtitles**: Generate subtitle files
7. **Legal/Medical**: Professional recording transcription

Getting Started

Quick Installation

```
# Clone the repository
git clone https://github.com/yourusername/transcribe_ro.git
cd transcribe_ro

# Run setup
# Windows: setup_portable.bat
# Linux/macOS: ./setup_portable.sh

# Transcribe an audio file
python transcribe_ro.py audio.mp3
```

Basic Usage Examples

```
# Basic transcription with translation
python transcribe_ro.py audio.mp3

# Fast transcription (tiny model)
python transcribe_ro.py audio.mp3 --model tiny

# High accuracy (medium model)
python transcribe_ro.py audio.mp3 --model medium

# No translation (original language only)
python transcribe_ro.py audio.mp3 --no-translate

# Generate subtitles
python transcribe_ro.py audio.mp3 --format srt

# Export as JSON
python transcribe_ro.py audio.mp3 --format json
```

Model Selection

Model	Size	Speed	Accuracy	Use Case
tiny	~75MB	Fastest	Good	Quick tests
base	~150MB	Fast	Better	**Default**
small	~500MB	Moderate	Great	Quality work
medium	~1.5GB	Slow	Excellent	Professional
large	~3GB	Slowest	Best	Maximum accuracy

Output Formats

TXT (Default)

- Metadata section
- Original transcription
- Romanian translation
- Timestamps for each segment

JSON

- Structured data format
- Includes all metadata
- Segment-by-segment information
- Easy to parse programmatically

SRT/VTT

- Standard subtitle formats
- Timestamp-synced text
- Video subtitle compatible
- Optional translation

Portability Features

Flash Drive Setup

- Run from USB without installation
- Standalone Python environment
- Self-contained dependencies
- Cross-computer compatibility

Offline Capability

- Works offline after model download
- Transcription fully offline
- Translation requires internet
- Use `--no-translate` for complete offline use

Performance Considerations

Speed Factors

- **Model Size**: Smaller = faster
- **Audio Length**: Longer = more time
- **Audio Quality**: Better quality = better accuracy
- **Hardware**: GPU acceleration available

Typical Processing Times

(Base model on CPU)
- 1 minute audio: ~30 seconds
- 5 minute audio: ~2-3 minutes
- 30 minute audio: ~10-15 minutes

Error Handling

The tool includes comprehensive error handling for:
- Missing audio files
- Unsupported formats
- Network issues (translation)
- Insufficient memory
- Corrupted audio files
- Missing dependencies

Future Enhancements

Planned Features

- Batch processing multiple files
- Progress bar for long transcriptions
- Configuration file support
- Better memory management
- Resume interrupted transcriptions

Potential Improvements

- Additional translation services (DeepL)
- GUI version
- Audio preprocessing (noise reduction)
- Speaker diarization
- Real-time transcription

Development

Contributing

Contributions are welcome! See CONTRIBUTING.md (CONTRIBUTING.md) for guidelines.

Testing

Comprehensive testing guide available in TESTING.md (TESTING.md).

Version Control

- Git repository initialized
- Commit history maintained
- Branch-based development encouraged

Documentation

Available Guides

1. **README.md**: Complete documentation
2. **QUICKSTART.md**: Quick start for beginners
3. **TESTING.md**: Testing procedures
4. **CONTRIBUTING.md**: Contribution guidelines
5. **examples.sh**: Command examples

Documentation Quality

- Clear and comprehensive
- Examples for all features
- Troubleshooting guides
- Multi-platform instructions

License

MIT License - Free to use, modify, and distribute.

Acknowledgments

- OpenAI for Whisper model
- FFmpeg project
- Google Translate service
- Open source community

Support

For issues or questions:
- Check documentation
- Review existing issues
- Open new issue with details
- Provide reproduction steps

Statistics

- **Lines of Code**: ~500+ (main script)
- **Supported Languages**: 99+ (Whisper)
- **Audio Formats**: 8+ major formats
- **Output Formats**: 4 (TXT, JSON, SRT, VTT)
- **Model Choices**: 5 size options
- **Platforms**: Windows, macOS, Linux

Target Audience

- **Primary**: Romanian speakers needing transcription/translation
- **Secondary**: Anyone needing multilingual transcription
- **Technical Level**: Beginner to advanced
- **Professional Use**: Suitable for professional work

Competitive Advantages

1. **Portability**: Runs from flash drive
2. **Offline Capable**: Works without internet (after setup)
3. **Free & Open Source**: No subscription fees
4. **Multi-language**: 99+ languages supported
5. **Professional Quality**: State-of-the-art AI models
6. **Flexible Output**: Multiple format options
7. **Easy to Use**: Simple CLI interface
8. **Well Documented**: Comprehensive guides

Project Status

✅ **COMPLETE** - Ready for production use

All core features implemented:
- ✅ Audio transcription
- ✅ Language detection
- ✅ Romanian translation
- ✅ Multiple formats
- ✅ Portable setup
- ✅ Documentation
- ✅ Error handling
- ✅ Testing guides
- ✅ Version control

Contact & Links

- **Repository**: GitHub - transcribe_ro (https://github.com/yourusername/transcribe_ro)
- **Issues**: GitHub Issues
- **Discussions**: GitHub Discussions
- **License**: MIT License

Project Created: January 2026
Version: 1.0.0
Status: Production Ready

Made with ❤️ for the Romanian community