WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

MythSnare

[1] SAJITH J, [2] SANJAY NITHIN S, [3] SUDHAN SANJAY VP.

[1 2 3] UG STUDENTS, SRI SHAKTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY, COIMBATORE.

ABSTRACT:

MythSnare is a multi-modal misinformation detection system that intelligently verifies the authenticity of content submitted as text, audio, video, or live speech. Utilizing real-time transcription through advanced speech recognition and classification via a fine-tuned BERT Sequence Classifier, it distinguishes between factual statements and regional news claims. For factual inputs, it employs Retrieval-Augmented Generation (RAG) in conjunction with OpenRouter's LLaMA-3 to deliver context-aware verifications. News-based inputs are validated against web-scraped sources from credible news outlets. Built with Django and powered by Django Channels for real-time responsiveness, MythSnare offers a seamless, interactive user experience. It not only detects misinformation but also provides reliable corrections, contributing to a more accurate and responsible digital ecosystem.

Keywords: BERT Sequence Classifier, Retrieval-Augmented Generation (RAG), LLaMA-3, Django, Multi-Modal Input, Natural Language Processing (NLP), Misinformation detection.

Introduction:

This In an increasingly interconnected world where digital communication drives social, political, and cultural discourse, the spread of misinformation has emerged as a global concern. From manipulated news articles and doctored videos to out-of-context statements and fabricated statistics, misinformation undermines public trust, disrupts democratic processes, and endangers societal well-being. In response to this growing challenge, *MythSnare* is developed as an advanced, AI-powered misinformation detection system designed to intelligently evaluate and verify the authenticity of user-submitted content across multiple formats—text, audio, recorded video, and live speech.

Unlike conventional fact-checking tools that operate in isolated modalities and often rely on manual intervention, MythSnare embraces a multi-modal approach powered by recent advances in Natural Language Processing (NLP), speech recognition, and transformer-based architectures. The system begins by transcribing non-textual data into structured text using Whisper, an efficient and accurate speech recognition model. Once transcribed, a fine-tuned BERT Sequence Classifier distinguishes between factual assertions and regional news-based claims. This classification allows the system to intelligently route each claim to a suitable verification pipeline.

For factual claims, MythSnare leverages Retrieval-Augmented Generation (RAG), combining the contextual reasoning of large language models with real-time retrieval of corroborating data. Using OpenRouter's implementation of LLaMA-3, the system fetches relevant evidence from trusted web sources, providing accurate and context-aware responses or corrections. On the other hand, regional news claims are validated through targeted web scraping of reliable regional news RSS feeds, ensuring up-to-date comparisons and localized verification. This dual-pipeline architecture ensures that MythSnare adapts dynamically to the nature and origin of each input.

Technically, the platform is built using Django and enhanced with asynchronous capabilities through Django Channels, enabling real-time user interaction regardless of the data format. Users can interact via a modern chatbot-style interface, uploading text, speaking live, or submitting media files, and receive near-instantaneous feedback. This seamless interaction is supported by a modular, scalable backend that can be extended to integrate with APIs, third-party fact-checking engines, or media databases.

In an age where misinformation spreads faster than facts, tools like MythSnare play a critical role in restoring informational integrity. By harnessing AI to automate the detection and correction of false claims across various media types, the system represents a paradigm shift in digital accountability. MythSnare is not merely a detector—it is an educator, a clarifier, and a safeguard for truth in the digital era.

What is the MythSnare?

MythSnare is an AI-powered tool that helps users verify the truthfulness of information shared as text, audio, or video. It uses speech recognition, NLP, and large language models to analyze and classify content, then checks claims against trusted sources. If something is false or misleading, MythSnare explains why and offers corrected information, helping users stay informed and avoid misinformation.

What is the use of MythSnare?

MythSnare helps users verify whether the information they encounter is true or misleading. You can speak, type, or upload content, and it checks the accuracy for you.

Here's how it can help:

1. **Verify Facts:** It tells you if a statement is true, false, or misleading using trusted sources.
2. **Stay Informed:** It helps you avoid falling for fake news or misinformation.
3. **Understand Misinformation**: It explains why something is false and what the truth is.
4. **Smart and Personalized:** The more you use it, the better it understands your input style and provides relevant feedback.

Methodology:

1. Data Collection

- **Objective**: Collect a wide range of factual statements, news articles, and misinformation from reliable sources.
- **Sources**: Use trusted news outlets, fact-checking websites, and user-submitted content to compile the dataset.
- **Data Format**: The dataset consists of text data, including user-submitted claims and factual news articles, labeled as true, false, or misleading.

2. Data Preprocessing

- **Cleaning the Data:** Remove irrelevant, incomplete, or duplicate content to ensure high-quality training data.
- **Text Normalization**: Convert all text to a uniform format (e.g., lowercase, remove special characters) to standardize inputs.
- **Tokenization**: Split text into words or sub-words (tokens) to enable efficient processing by the model.
- **Stop Word Removal**: Eliminate common words (e.g., "and", "the") that do not contribute to the meaning of the statement.

3. Model Selection and Training

- **Model**: Use a pre-trained **BERT Sequence Classifier** for text classification tasks and **T5** or **LLaMA** for fact verification.
- **Training**: Train the model using labeled data to identify whether a statement is factual or belongs to regional news.
- **Fine-Tuning**: Fine-tune the model to improve its accuracy by adjusting hyperparameters and incorporating real-time web data to enhance the system's adaptability.

4. Misinformation Detection and Verification

- **Input**: Users submit text, audio, or video content through the app interface.
- **Processing**: The content is transcribed (if necessary), classified, and verified for factual accuracy using the trained model.
- **Output**: The system generates a verdict on whether the content is factual or misleading, and provides a detailed explanation with references to trusted sources or fact-checking databases.

5. Website Development (Frontend and Backend)

- **Frontend**: Develop a responsive and user-friendly interface using **HTML**, **CSS**, and **JavaScript**. This enables easy interaction where users can submit content, view results, and explore the system's explanations.
- **Backend**: Build the backend using **Django** to handle user input, interact with the models for classification, and return results. Utilize **Django Channels** for asynchronous communication to enable real-time updates and smooth performance. Store and manage data securely in the backend.

6. Evaluation and Testing

- **Accuracy Check**: Continuously test the model on a separate dataset to ensure the accuracy and reliability of the content verification system.
- **User Feedback**: Collect feedback from users on the website to gauge the effectiveness of the system's explanations and the overall user experience.
- **Refinement**: Based on user feedback and performance metrics, refine the models and website interface to improve accuracy, speed, and user engagement.

7. Deployment and Maintenance

- **Website Deployment**: Deploy the website on a hosting platform (e.g., AWS, Heroku), ensuring the system is accessible and functional for users.
- **Continuous Updates**: Regularly update the website with new features, improvements to the model, and bug fixes. This includes refreshing data sources and optimizing the user interface based on user feedback and advancements in AI.

Objective:

- **Misinformation Detection:** Use AI to detect whether a piece of content is factual or misleading.
- **Accurate Verification:** Use advanced AI models like BERT and LLaMA to ensure the content is thoroughly analyzed and verified.
- **User Engagement:** Offer easy-to-understand explanations and corrections for false or misleading claims.
- **Educational Value**: Help users learn about misinformation, its impact, and ways to identify it in digital content

Results

- **Misinformation Detection:** The website will provide users with accurate classifications, identifying whether the content they input is factual or misleading.
- **Real-Time Feedback:** Users will receive immediate results, allowing them to quickly understand the veracity of the content they submit, whether it's text, audio, or video.
- **Personalized Explanations:** The system will offer tailored explanations and corrections based on the specific content, ensuring the feedback is relevant and contextually accurate.
- **Educational Insights:** Users will learn about how misinformation spreads, with practical tips on identifying and evaluating information critically.
- **Enhanced Awareness:** The website will help users become more discerning in the digital space, improving their ability to identify misinformation and make informed decisions.

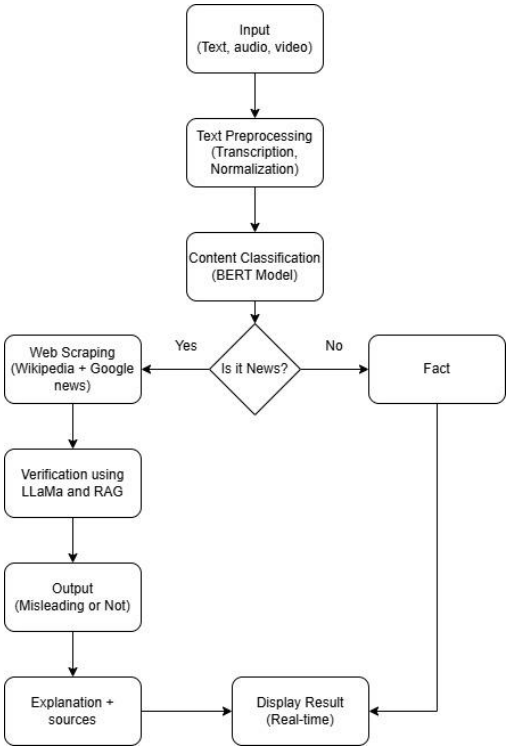

Fig 1 Block Diagram

Conclusion

The MythSnare project effectively demonstrates the power of artificial intelligence in combating misinformation by analyzing and verifying user-submitted content. Utilizing advanced models like BERT for classification and LLaMA for fact-checking, the system provides accurate, real-time results, distinguishing between factual statements and misleading information. The seamless integration of Django for backend processing, combined with a user-friendly frontend, ensures an intuitive experience for users. This project not only addresses the growing challenge of misinformation but also fosters a more informed digital space. Future improvements may include expanding the scope of data sources, enhancing the verification process, and incorporating more dynamic user feedback mechanisms to further refine the system's performance.

REFERENCES:

List all the material used from various sources for making this project proposal

Research Papers:

1. Shu, K., Sliva, A., Wang, S., Tang, J., & Liu, H. (2017). *Fake News Detection on Social Media: A Data Mining Perspective*. ACM SIGKDD Explorations Newsletter, 19(1), 22-36.

2. Zhou, X., & Zafarani, R. (2018). *Fake News: A Survey of Research, Detection Methods, and Opportunities*. ACM Computing Surveys (CSUR), 51(4), 1-40.
3. Horne, B. D., & Adalı, S. (2017). *This Just In: Fake News Packs a Lot in Title, Uses Simpler, Repetitive Content in Text Body, More Similar to Satire than Real News*. arXiv preprint arXiv:1703.09398.
4. Hassan, N., Arslan, F., Li, C., & Tremayne, M. (2017). *Toward Automated Fact-Checking: Detecting Check-Worthy Factual Claims by ClaimBuster*. Proceedings of the 23rd ACM SIGKDD.
5. Zellers, R., Holtzman, A., Rashkin, H., Bisk, Y., Farhadi, A., Roesner, F., & Choi, Y. (2019). *Defending Against Neural Fake News*. NeurIPS.