

Fake News Detection Using Input based News Retrieval And RAG

Joseph Siby¹, Joyal Saji², Sasilekha K S³, Vishnu A⁴, Betsy Joy⁵

¹Bachelor of Technology in CSE, NCERC

²Bachelor of Technology in CSE, NCERC

³Bachelor of Technology in CSE, NCERC

⁴Bachelor of Technology in CSE, NCERC

⁵Assistant Professor, Department of CSE, NCERC

Abstract - The rapid spread of misinformation on social media has become a major challenge, influencing public opinion and decision-making. Traditional fake news detection methods rely on static datasets and extensive training, which often limits their adaptability to emerging news. This project proposes a Fake News Detection system using a Retrieval-Augmented Generation (RAG) approach integrated with a Large Language Model (LLM). The system accepts user input in the form of news content, preprocesses the text, and generates embeddings to retrieve relevant information from trusted sources in real time. The retrieved context is combined with the input and analyzed by the language model to classify the news as real or fake. The system also provides an explanation to improve transparency and user trust. By incorporating real-time retrieval and contextual reasoning, the proposed method improves accuracy, reduces misinformation, and offers an efficient and scalable solution for detecting fake news.

Key Words: Fake News nDetection , RAG , LLM, Social Media, Misinformation, Real-time Rtrieval, Natural Language Processing, Text Embedding , Contextual Reasoning, News Classification

1. INTRODUCTION

The rapid growth of digital communication and social media has transformed how news is produced, shared, and consumed, enabling real-time access to information across the globe. However, this speed has also led to the widespread circulation of misleading content, commonly known as fake news. Such misinformation can influence public opinion, create social divisions, and reduce trust in credible journalism. Traditional detection techniques, including keyword filtering, sentiment analysis, heuristics, and classical machine learning models, struggle to keep pace with evolving misinformation. These approaches often rely on static datasets and superficial textual features, making it difficult to identify deeper contextual and factual inconsistencies in increasingly sophisticated fake content.

To address these limitations, the proposed project introduces a Retrieval-Augmented Generation (RAG) based framework that combines real-time information retrieval with the reasoning capabilities of large language models. By retrieving relevant and verified documents from trusted sources, the system grounds its predictions in factual evidence, improving detection accuracy and reducing unsupported conclusions. The model also emphasizes explainability by providing clear justifications for its decisions, helping build user trust and accountability. With its modular and scalable design, the system can be integrated into various platforms for fact-checking and misinformation monitoring. Overall, this approach aims to create a reliable,

adaptable, and ethical solution for combating fake news and promoting a more informed digital society.

2. LITERATURE REVIEW

A literature review is a structured summary and analysis of previously published research related to a specific topic. It helps in understanding existing knowledge, identifying research gaps, and providing a foundation for new studies. The literature review mainly uses secondary sources such as journals, research papers, and conference publications. It also compares different methodologies, techniques, and findings used by earlier researchers. This process enables researchers to evaluate the strengths and weaknesses of existing approaches and determine the need for further improvements. In this project, the literature review focuses on various fake news detection techniques, including traditional machine learning models, deep learning approaches, transformer-based architectures, and Retrieval-Augmented Generation (RAG) frameworks. It examines how each method contributes to misinformation detection and highlights their limitations, such as lack of real-time verification and dependency on static datasets. By analyzing these studies, the review justifies the need for a more reliable and adaptive system. This supports the development of the proposed RAG-based fake news detection model that combines retrieval and language model reasoning for improved accuracy.

3. Technology used in RAG System

Software used:

Python

Main programming language used to develop the fake news detection system.

FastAPI

Backend framework used to handle API requests and connect frontend with the model.

HTML

Used to design the structure of the web interface.

CSS

Used to style and format the user interface.

JavaScript

Adds interactivity and handles frontend logic.

VS Code

Code editor used for writing and managing project files.

Transformers Library

Provides pre-trained models for language understanding tasks.

SentenceTransformers

Used to generate embeddings for semantic similarity search.

LangChain

Helps integrate LLM with retrieval and external data sources.

FAISS / ChromaDB

Vector database used to store and search embeddings.

Tavily Search API

Retrieves real-time information for fact verification.

Web Browser (Chrome/Edge/Firefox)

Used to run and test the web application.

4. Proposed System

Improvements Over the Existing System:

Higher Accuracy :

The proposed RAG system improves detection accuracy by using real-time retrieved information instead of only trained data.

Real-Time Knowledge Access :

Unlike existing systems, it fetches up-to-date news articles for verification.

Reduced Hallucination :

The model generates responses based on retrieved evidence, reducing false predictions.

Better Fact Verification :

The proposed system validates claims using external trusted sources.

Improved Explainability :

It provides supporting documents, making decisions more transparent.

Dynamic Adaptability :

The system can handle new and trending news topics without retraining.

Lower Retraining Requirement :

No need to frequently retrain the model when new information appears.

Context-Aware Analysis :

Retrieval helps the model understand the context more effectively.

Scalability :

The system can be expanded by adding more data sources.

Enhanced Reliability :

Evidence-based responses increase trust in predictions.

Unlike the existing systems, our system includes the following roles:

1. FastAPI

It acts as the backend server that connects the frontend interface with the AI model and retrieval components. It handles user requests, processes input text, sends it to the embedding model, and returns the classification result

2. Sensor Transformer Framework

The user input news is converted into embeddings using this framework. These embeddings are then compared with stored vectors in the vector database to find similar content.

3. Retrieval Augmented Generation (RAG)

The retrieved context and user query are combined.

Both are passed to the Large Language Model.

This improves accuracy using external knowledge.

4. Tavily Search API

It is used to retrieve real-time information from the web. In this project, it helps fetch recent news articles and facts related to the user input. This improves the credibility of classification. The API acts as an external knowledge source

5. LLM Analysis

The Large Language Model analyzes User input, Retrieved documents, Contextual information, The model evaluates credibility of the news.

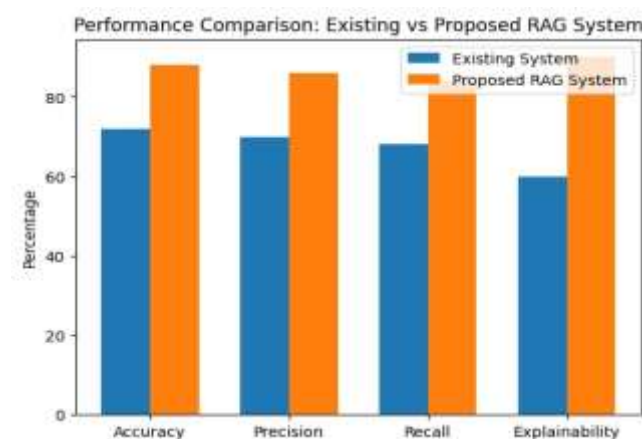


Figure 4.1 : Performance comparison

5.Challenges Faced During Implementation

Retrieval Accuracy

Getting highly relevant documents from search APIs is difficult; irrelevant results can affect final prediction.

API Dependency

The system depends on external search APIs, which may have rate limits, latency, or downtime.

Integration Complexity

Combining embeddings, retrieval, vector database, and LLM into a single pipeline is technically complex.

Response Time

Real-time retrieval plus LLM processing may increase overall system latency.

Handling Ambiguous News

Some news statements lack sufficient evidence, making classification difficult.

Hallucination Risk

Even with RAG, LLMs may sometimes generate unsupported explanations.

Data Quality Issues

Retrieved sources may contain biased or unreliable information.

Embedding Similarity Errors

Poor semantic matching can retrieve unrelated documents.

Hardware Limitations

Running LLMs may require high RAM or GPU resources.

Scalability Challenges

Handling multiple users simultaneously can affect performance.

Prompt Engineering Difficulty

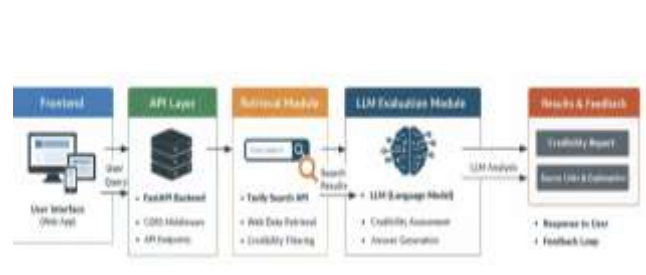
Designing effective prompts for accurate reasoning is challenging.

Evaluation Metrics

Measuring accuracy for real-time fake news detection is complex.

6. System Design

6.1. Architecture of the System



6.2 Module Include

API Input Handling Module

- Access user input through API endpoints.
- Built using FastAPI for efficient request handling.
- Validates and processes incoming requests.
- Sends input data to backend processing pipeline.

DATA Processing & Embedding Module

- Cleans and preprocesses the input news text.
- Convert text into embeddings using NLP models.
- Ensures semantic understanding of input.
- Prepares data for retrieval process.

Retrieval and knowledge base Module

- Searches relevant documents from stored database.
- Uses similarity search to find related news or facts.
- Retrieves supporting evidence for the given input.
- Acts as a knowledge base of the system.

LLM reasoning and classification Module

- LLM analyze input along with retrieved documents.
- Performs reasoning based on contextual evidence.
- Classifies news into real or fake.
- Generates meaningful explanations.

API response and Output Module

- Sends final result through API response.
- Includes prediction and explanation.
- Ensures structured JSON response format.
- Can be integrated with frontend or UI display.

7. Implementation and Results

7.1 System Development

The **RAG system** was developed as a web-based application using modern front-end and back-end technologies. The main aim of the system is to detect fake news using input based news retrieval and RAG.

The system comprises the following modules:

Frontend :

provides a text input area where users can enter news content. A submit button is used to send the input to the backend.

Backend :

This implemented using FastAPI. It handles HTTP requests from the frontend and processes user input. The backend performs preprocessing, embedding generation, retrieval, and model inference.

Embedding Model:

Sentence Transformers are used to convert text into embeddings. The input news content is transformed into numerical vectors.

Vector Database:

When the user submits a query, the system performs similarity search to retrieve relevant documents which are supporting evidence.

Retrieval Augmented Generation :

The retrieved documents and user input are combined. This combined context is passed to the Large Language Model. This approach ensures that the model uses external knowledge for classification.

The Large Language Model :

This analyzes the input and retrieved context. It determines whether the news is real or fake and generates an explanation.

API Integration:

FastAPI endpoints are created to handle User input submission, Model processing, Result retrieval.

7.2 System Testing

To ensure system reliability and performance, the following testing strategies were used :

Unit Testing

Each module such as Input processing module, Embedding generation module, Retrieval module, LLM classification module tested individually.

Integration Testing

Integration testing verifies communication between modules such as Frontend and backend, Backend and embedding model, Retrieval system and vector database, LLM and output module

Functional Test

The system is tested using different news inputs such as True news articles, Fake news headlines, Misleading statements.

Performance Testing

The response time of the system is evaluated. The system is tested with multiple queries to ensure it handles requests efficiently.

User Interface Testing

The frontend is tested for Proper input acceptance, Button functionality, Output display, Error handling.

Accuracy Testing

The system output is compared with verified fact-checking sources to measure accuracy. Different datasets are used to evaluate performance.

Error Handling Testing

The system is tested for Empty input, Invalid text, Network errors, API failures.

7.3 Results:

The system responded effectively by providing output as the news is fake or real after factual verifications and also provided relevant explanations with examples.

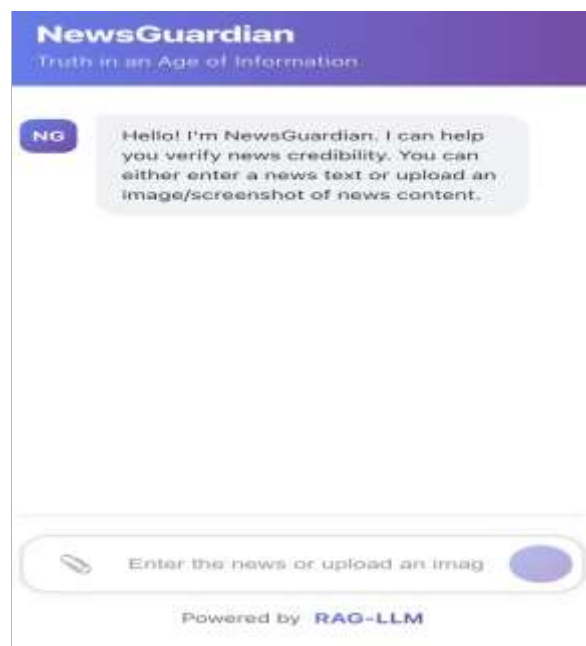


Fig 7.1: Home Page

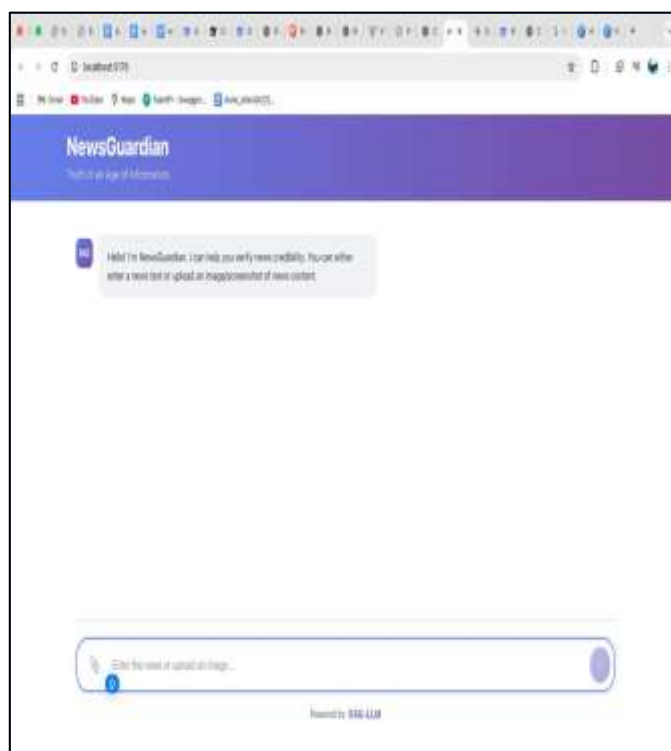


Fig 7.2:User Interface

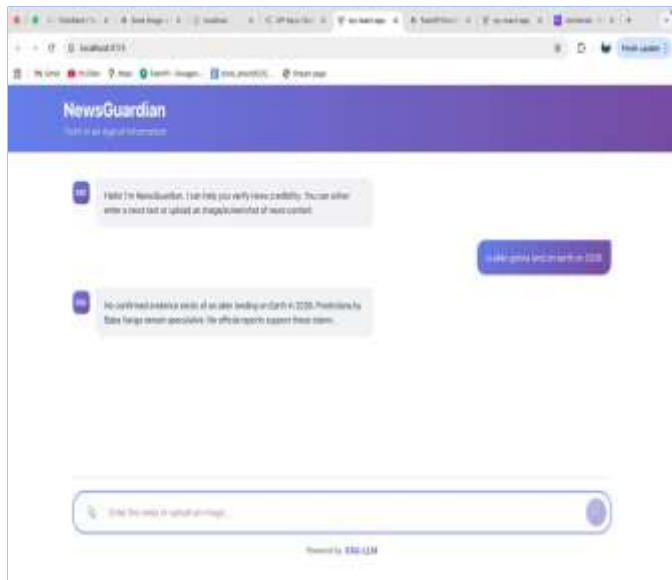


Fig 7.3:User Query and Result

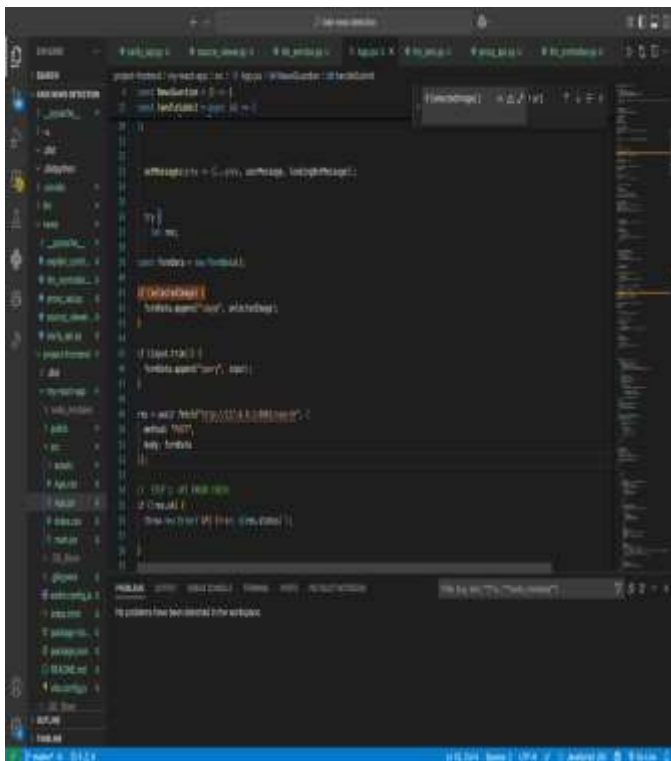


Fig 7.4.Frontend ora code

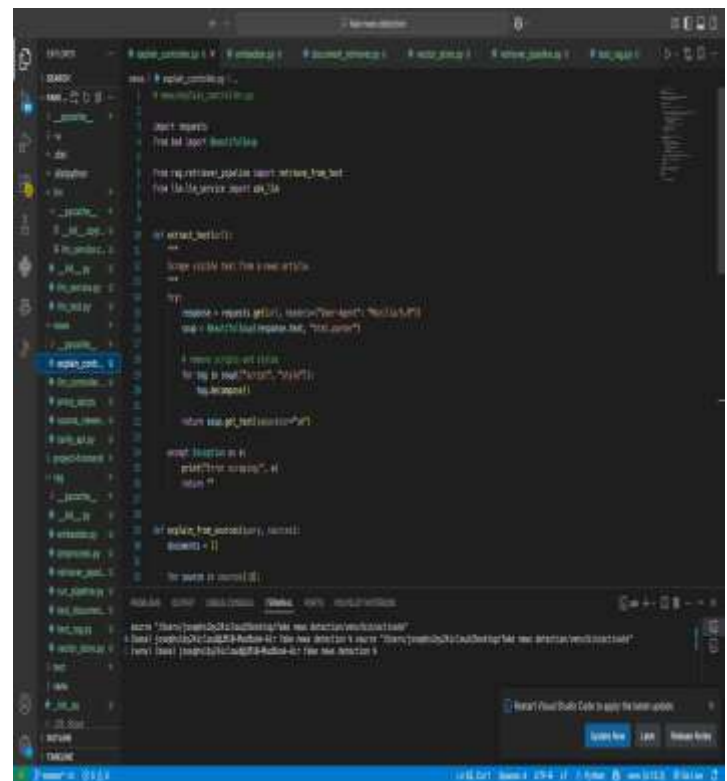


Fig 7.5.Frontend react code

8. DISCUSSION AND ANALYSIS

8.1 Advantages of RAG System

Access to Real-Time Information:

RAG retrieves information from external sources such as search engines or knowledge bases. This allows the system to use up-to-date data, which improves accuracy in tasks like fake news detection.

Reduced Hallucinations in LLMs:

Traditional Large Language Models sometimes generate incorrect or fabricated information. RAG reduces this problem by grounding responses in retrieved factual evidence.

Improved Accuracy and Reliability:

By combining information retrieval with language generation,

RAG produces more reliable results compared to models that rely only on pre-trained knowledge.

No Need for Extensive Model Retraining:

Since RAG retrieves knowledge dynamically, the system does not require frequent retraining whenever new information appears.

Evidence-Based Explanation:

RAG can provide supporting documents or references for its decisions, making the system more transparent and trustworthy.

Scalability and Flexibility:

The architecture can easily integrate with different databases, APIs, or knowledge sources, making it suitable for large-scale applications.

Better Context Understanding:

Retrieved documents provide additional context, allowing the model to analyze complex queries more effectively.

8.2 Future Work

Future work includes extending the system to support multilingual fake news detection for broader usability. The model can be enhanced by integrating more trusted data sources and real-time APIs to improve accuracy. The system may also be expanded to analyze images and videos for detecting multimedia misinformation. Deployment on large-scale platforms for real-time monitoring can further increase its effectiveness. Continuous learning techniques can be added to adapt to evolving misinformation trends.

9. ACKNOWLEDGEMENT

We are immensely grateful for the collective support that enabled the successful completion of this project. First and foremost, we express our sincere gratitude to our project guide Ms. Betsy Joy, Assistant Professor, Department of Computer Science and Engineering, Nehru College of Engineering and Research Centre, for her invaluable guidance, encouragement, and continuous support throughout the development of this work. Her insightful suggestions and expertise greatly helped in improving the quality of the project. We also extend our heartfelt thanks to the faculty members of the Department of Computer Science and Engineering, Nehru College of Engineering and Research Centre, for providing the necessary resources and a supportive academic environment. Finally, we would like to express our sincere appreciation to our family members, friends, and well-wishers for their constant encouragement and motivation throughout the completion of this project.

10. CONCLUSION

The rapid spread of information through social media has increased the prevalence of fake news, which can influence

public opinion and reduce trust in reliable sources. Traditional detection methods rely on static datasets and basic textual analysis, making them less effective against sophisticated and evolving misinformation. To overcome these challenges, this project proposes a Retrieval-Augmented Generation (RAG) based fake news detection system that combines real-time information retrieval with Large Language Models. The system verifies news using trusted sources, improves accuracy, and provides explainable results. Its scalable design and use of evaluation metrics ensure an efficient and reliable solution for combating misinformation.

REFERENCES

1. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. (2023). GPT-4 technical report. arXiv preprint arXiv:2303.08774.
2. Ahmed, H., Traore, I., & Saad, S. (2017). Detection of online fake news using n-gram analysis and machine learning techniques. In Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments: First International Conference, ISDDC 2017, Vancouver, BC, Canada, October 26–28, 2017, Proceedings 1 (pp. 127–138). Springer.
3. Ahmed, H., Traore, I., & Saad, S. (2018). Detecting opinion spams and fake news using text classification. Security and Privacy, 1(1), e9.
4. Allcott, H., & Gentzkow, M. (2017). Social media and fake news in the 2016 election. Journal of Economic Perspectives, 31(2), 211–236.
5. Altheneyan, A., & Alhadlaq, A. (2023). Big data ML-based fake news detection using distributed learning. IEEE Access, 11, 29447–29463.
6. Barbon, R., & Akabane, A. (2022). Towards transfer learning techniques BERT, DistilBERT, BERTimbau, and DistilBERTimbau for automatic text classification from different languages: A case study.
7. Blackledge, C., & Atapour-Abarghouei, A. (2021). Transforming fake news: Robust generalisable news classification using transformers. In 2021 IEEE International Conference on Big Data (Big Data) (pp. 3960–3968). IEEE.
8. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G.,

- Askill, A., et al. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33, 1877–1901.
9. Budi, I., & Suryono, R. R. (2023). Application of named entity recognition method for Indonesian datasets: A review. *Bulletin of Electrical Engineering and Informatics*, 12(2), 969–978.
10. Conroy, N., Rubin, V., & Chen, Y. (2015). Automatic deception detection: Methods for finding fake news. *Proceedings of the Association for Information Science and Technology*, 52(3), 1–4.
11. Corso, G., Ying, Z., Pándy, M., Veličković, P., Leskovec, J., & Liò, P. (2021). Neural distance embeddings for biological sequences. *Advances in Neural Information Processing Systems*, 34, 18539–18551.
12. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In J. Burstein, C. Doran, & T. Solorio (Eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (pp. 4171–4186). Minneapolis, Minnesota: Association for Computational Linguistics.
13. Floridi, L., & Chiriatti, M. (2020). GPT-3: Its nature, scope, limits, and consequences. *Minds and Machines*, 30, 1–1
14. Hu, B., Sheng, Q., Cao, J., Shi, Y., Li, Y., Wang, D., & Qi, P. (2024). Bad actor, good advisor: Exploring the role of large language models in fake news detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 38, 22105–22113.