TextVerify: A Better AI Detection Model

Dev Bhatia
Computer Systems Lab Dr. Yilmaz
Period: 3
Date: 5/21/2025

Problem

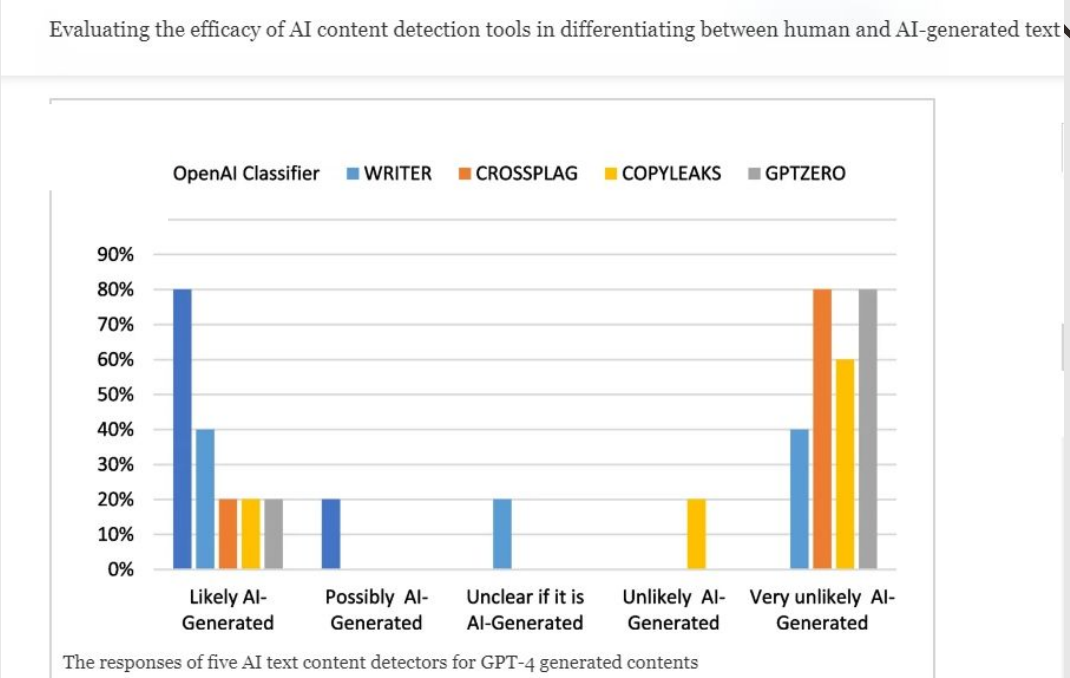
- AI used very prominently
- People tend to paraphrase AI content
- Difficulty telling original work from AI generated text
- Current AI detectors don’t work well
- Leads to lack of knowledge for students
- AI gives untrustworthy or false information
- Need better detection mechanism

Background

- Many AI detectors out there
- Accuracy varies significantly
- Unreliable for teacher use
- Possibility of false positives
- Not close to 100% accuracy

Other Solutions

- Most prominent AI detectors:
  - GPTZero [6]
  - Copyleaks [7]
  - CrossPlag [8]
- Detectors do not perform well against GPT 4 and other AIs
- Many false negatives



Other Solutions Continued

- How these AI detectors work
  - GPTZero:
    - Amount of predictability in the text
    - Looks at the variance in the sentences, AI generated text typically has less variety and is more predictable.
  - Copyleaks
    - Scans the document against different sources
    - Sentence by sentence detection using writing patterns
  - CrossPlag
    - Text analysis using Natural Language Processing
    - Compares to a dataset created by human and AI content
- Pros: Detect well on older GPT generations
- Cons: Fail on GPT 4 and paraphrased AI generations, inconsistencies

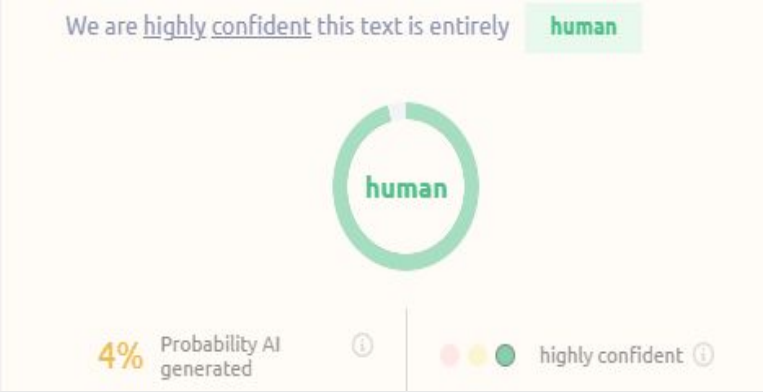
Demo of Current Solutions

- GPTZero:
- CopyLeaks: (doesn't seem to work without subscription)
- CrossPlag:
- Sample AI generated text: “Playing basketball is always exciting. The moment I step onto the court, I feel the energy rise. The sound of the ball bouncing and sneakers squeaking on the floor makes me focus on the game. I enjoy dribbling the ball, passing to my teammates, and shooting for the basket. Each time the ball goes in, it’s a rush of excitement. Working with the team, moving fast, and thinking on my feet keeps me engaged. Basketball is not just a sport to me, it’s a fun way to stay active and connect with others.”

AI Detectors with Paraphrasing continued

"Playing basketball is always exciting. The moment I step onto the court, I feel the energy rise. The sound of the ball bouncing and sneakers squeaking on the floor makes me focus on the game. I enjoy dribbling the ball, passing to my teammates, and shooting for the basket. Each time the ball goes in, it's a rush of excitement. Working with the team, moving fast, and thinking on my feet keeps me engaged. Basketball is not just a sport to me, it's a fun way to stay active and connect with others."

I love basketball all year round. The energy starts to rise as soon as I foot onto the court. The sound of the bouncing ball and the squeaky sneakers on the floor draw my focus to the ga me. I like dribbling the ball, passing it to my teammates, and shooting for the basket. The moment the ball touches down, there is an exhilarating surge. My ability to move quickly, think quickly, and collaborate with others keep me engaged. Basketball is more than just a sport to me; it is an enjoyable way to keep active and make new friends.



Why Is Mine Better?

- TextVerify is more accurate on GPT 4
  - Current detectors struggle on this
- TextVerify is better with paraphrased GPT 4
  - GPT 4 content that people paraphrase but the meaning is the same
- TextVerify works better with DeepSeek, Copilot, Grok content

Novelty

- Using retrieval methods (not used on common AI detectors)
  - Using a database of AI generations to tell whether text is AI generated
    - Use cosine similarity scores
    - Find any matches to AI generations in the database
- Higher accuracy than current AI detecting algorithms have

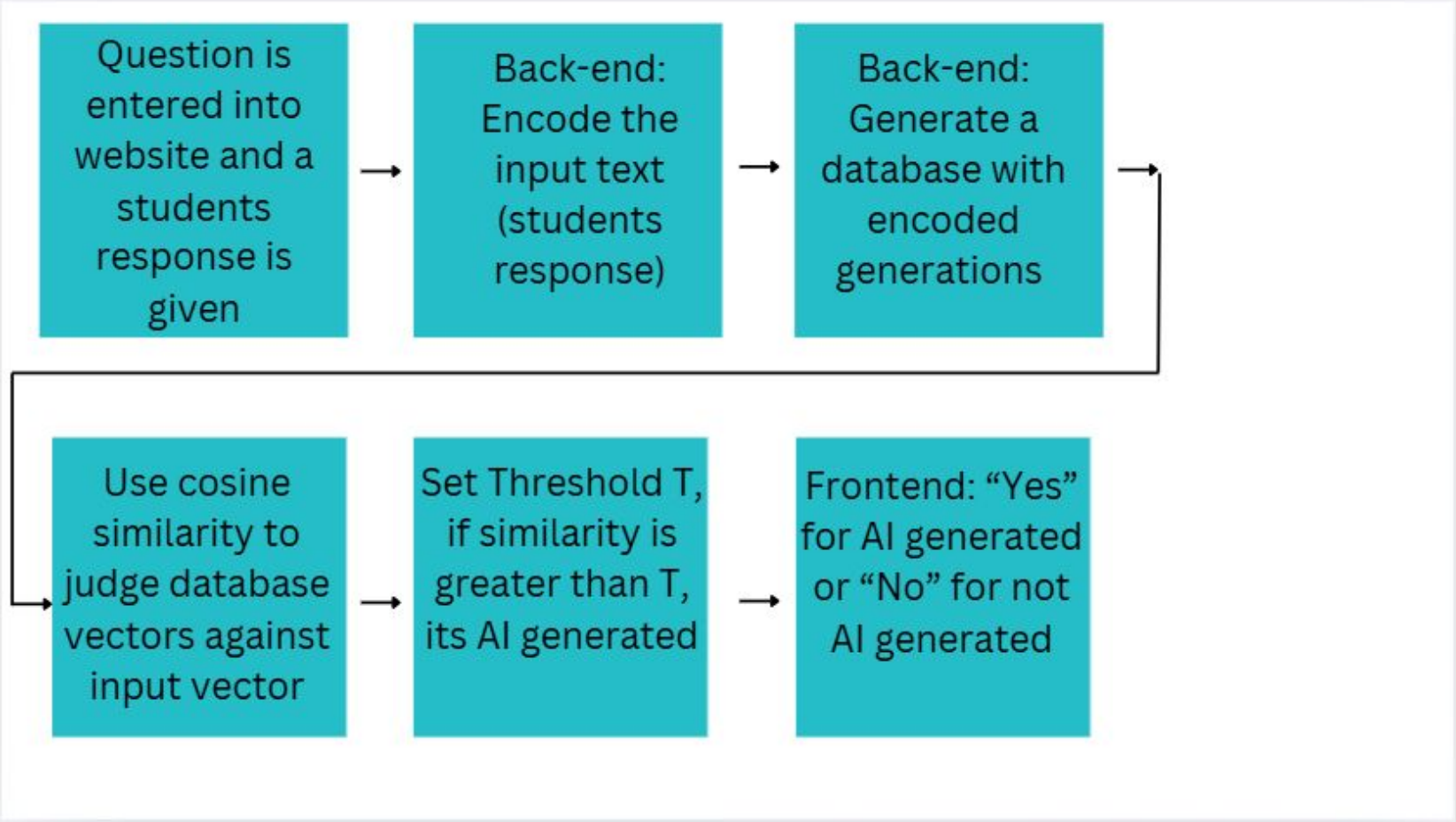
Impact

- Can be used by educators
  - A more reliable way to check AI content
- Trustworthy source of AI detection
- Maintains integrity
- Creates a fair learning environment

Method

1. Given an input text and a question
2. Use a Doc2Vec encoder to encode input text
3. Store Doc2Vec encoded sequences of text that were generated by AI in database in a web server
4. Check to see if input text vector matches database text vector (cosine similarity)
5. Set the threshold (T), if the text score is higher than T, then it is judged as AI Generated
6. Output whether the text is detected as AI generated or not

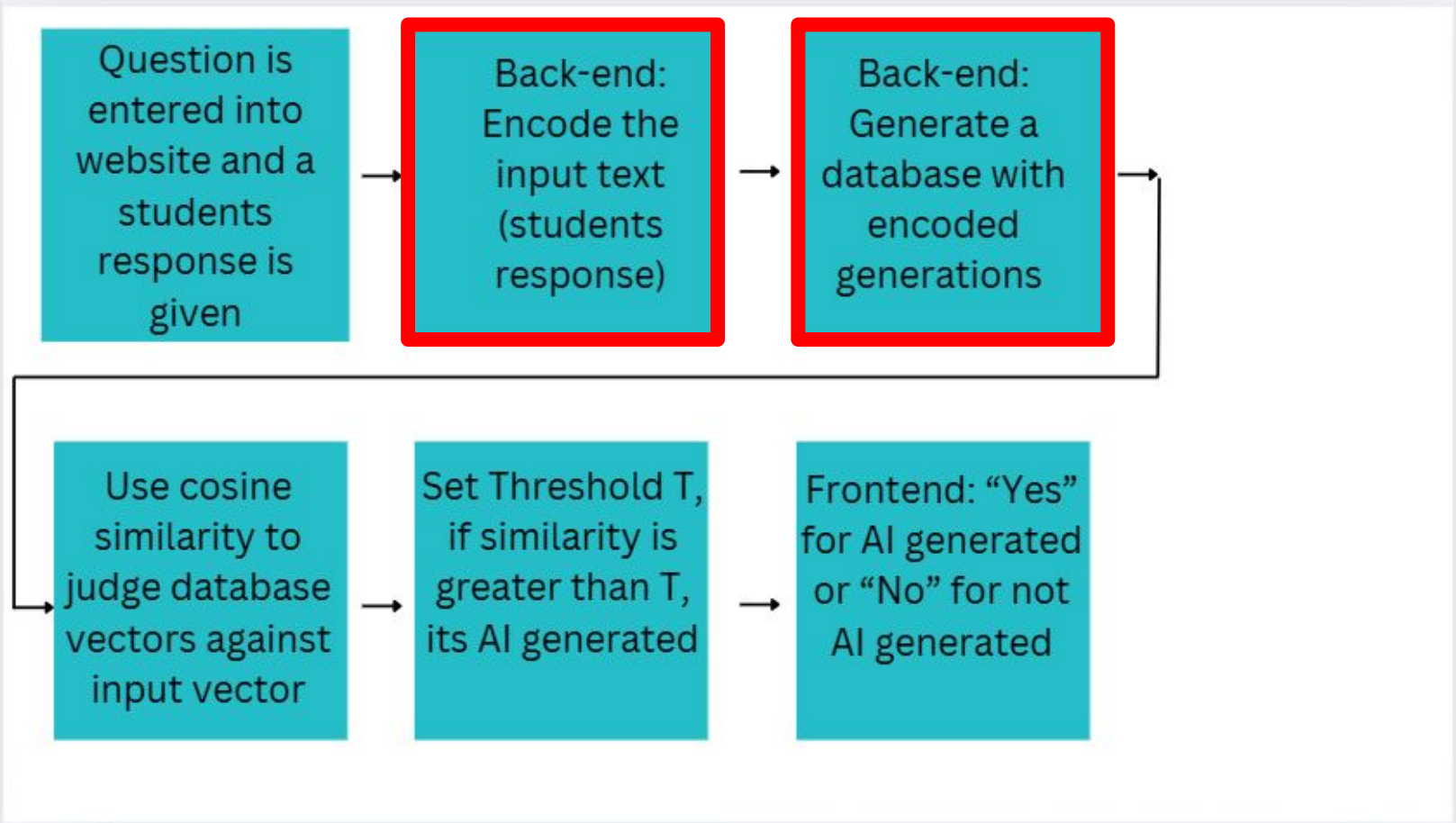
Method: Flow Diagram



Process Overview

- Example:
  - Question (Entered by teacher): Describe the sky
  - Input Text (Sample response): "The sky is clear and blue"
  - Input Text Vector Representation: [0.5,0.3,0.7,0.1,0.4,0.2]
  - Database of generations:
    - “The sky is sunny and bright” Vector: [0.45,0.25,0.65,0.1,0.35,0.3]
    - “The sky has rain” Vector: [0.4,0.2,0.6,0.1,0.3,0.2]
    - “The sky is bright and blue” Vector: [0.51,0.31,0.69,0.12,0.41,0.22]
  - Cosine Similarity "The sky is clear and blue" and "The sky is bright and blue" = 0.80
  - Threshold = 0.60
  - 0.80 > 0.60, therefore AI generated

Process Overview



Encoding - Doc2Vec - Overview

- General Overview:
  - Learn a fixed-length vector representation for document texts.
  - Every document is assigned a unique vector (document vector). This vector is used along with local context words to help predict a target word in the document. Over time this forces the document vector to capture the semantics of the document.
  - The neural network is given a fixed window of words (ie: the correct answer) and has to learn a document vector that fits that window of words. It does this for the whole document and all the documents.
  - Uses a 2-layer neural network

Doc2Vec Continued

- Purpose
  - Training the neural network.
    - Data is coming in the form of documents, these documents are given document vectors that are initially set to random and are updated while training, eventually representing the semantics of the document.

Doc2Vec Model Overview

- Distributed Memory version of Paragraph Vector (PV-DM)
  - Input Layer:
    - Context words and a Document Vector
    - Originally the words and Document Vector are unique vectors (initialized randomly)
  - Hidden Layer:
    - Take the input vectors and average them.
    - Activation Function (ReLU)
  - Use the hidden layer representation to predict the target word
    - Output is a probability distribution over the vocabulary, which is optimized with softmax.
  - Train the network to maximize the probability of the correct (target) word given the combination of the document vector and surrounding context words.
  - Sliding window of words moves around the document

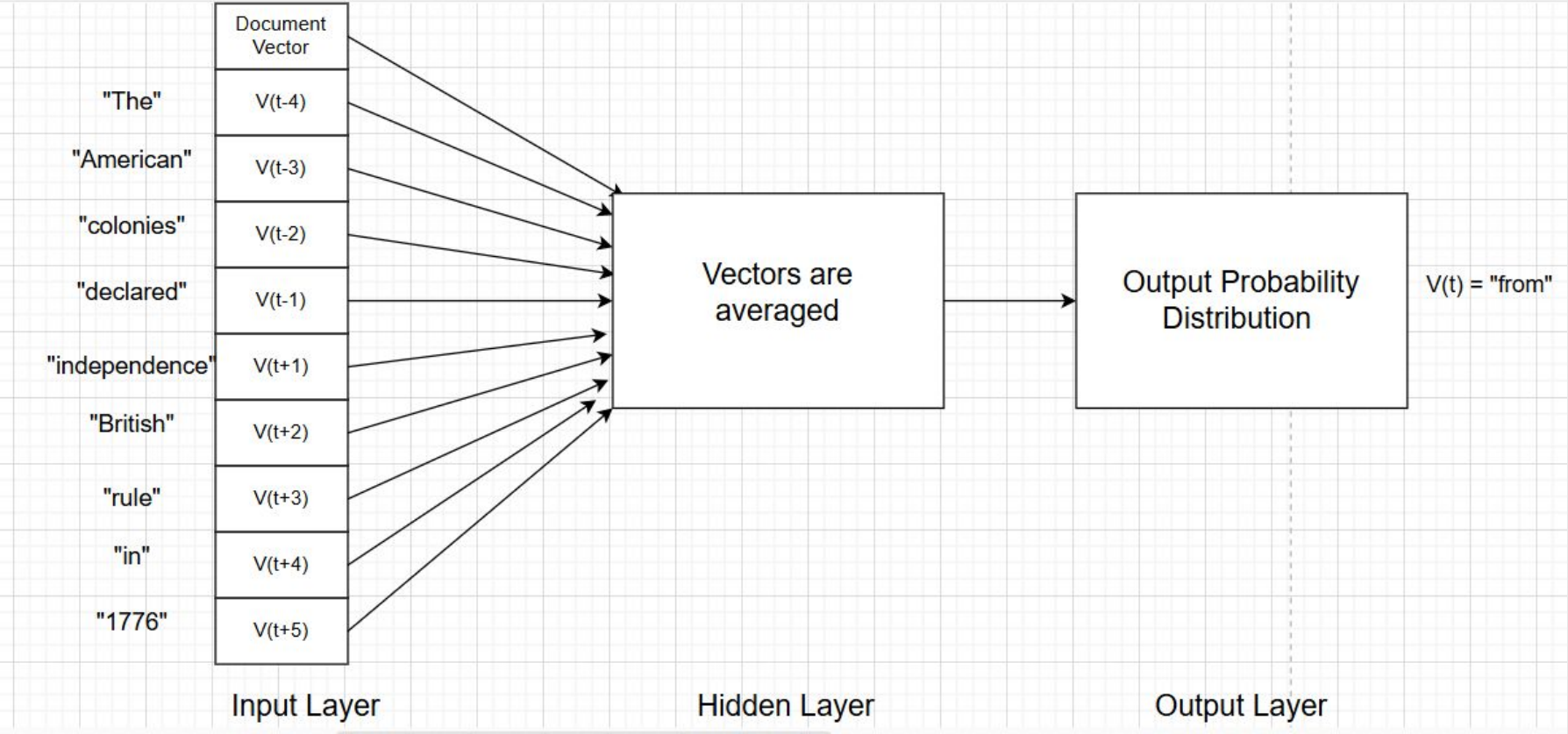
Doc2Vec for a given document

- Sliding window of text goes through the document (ex: 10 word length text). Go window by window into the text and train the NN using that text window).
- Example of a Document:
  [The American colonies declared independence **from** British rule in 1776.] [George Washington bravely led the **Continental** Army through fierce battles.] [Treaties and compromises eventually united **the** young nation after struggles.] [Rapid industrial growth later transformed **America** into modern global power.]
- Ex: It goes 10 words at a time, the NN is training to guess the word in bold. The rest of the words in a text window and the document vector are used as inputs for the NN.

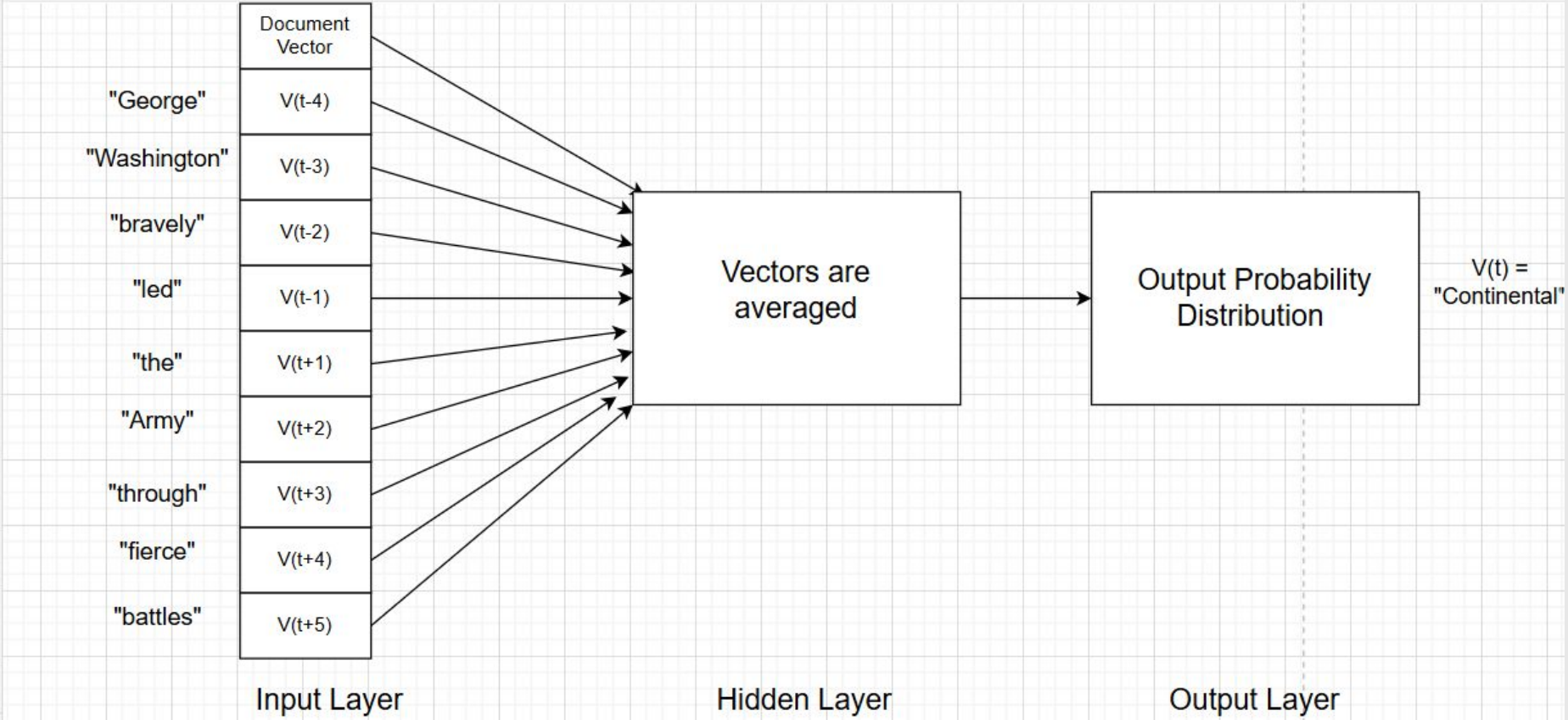
Explanation of the Model

- Go over each document in the set of documents
  - Initialize the Document Vector to random
  - Go over each window of words (length = 10 words) in the document
    - Example window of words that goes into the Neural Network "The American colonies declared independence from British rule in 1776."
    - Input
      - Document Vector
      - Rest of the context words
        - "The American colonies declared independence British rule in 1776" not including middle word "from"
        - Each word is translated into random word vectors
    - Concatenation/average
      - Average the word vectors and Document Vector (then apply ReLU)
      - Output is a probability distribution over the vocabulary, the word with the highest probability is chosen as the neural network's predicted word.
      - This is then compared with the middle word.
    - Do backpropagation to minimize the error to the actual word

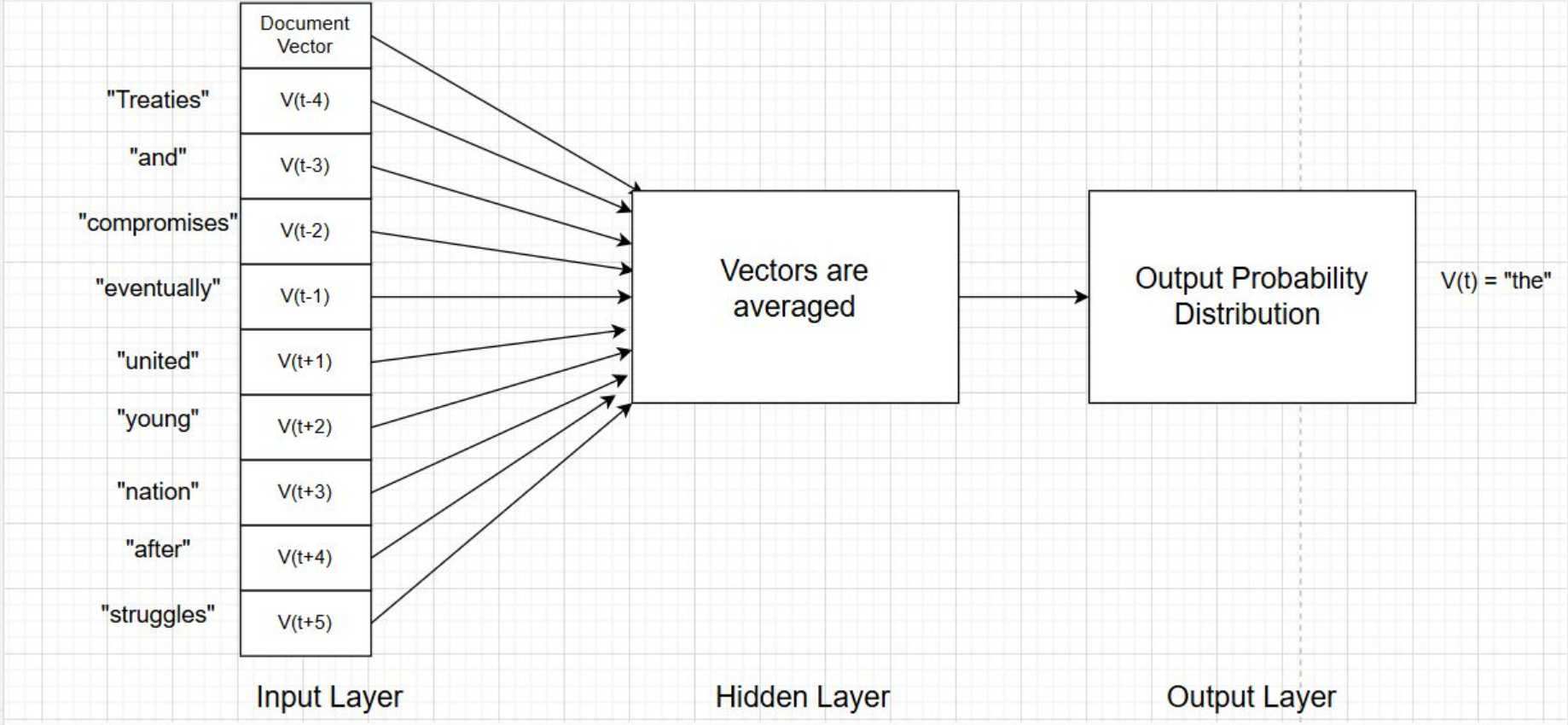
Model Diagram 1



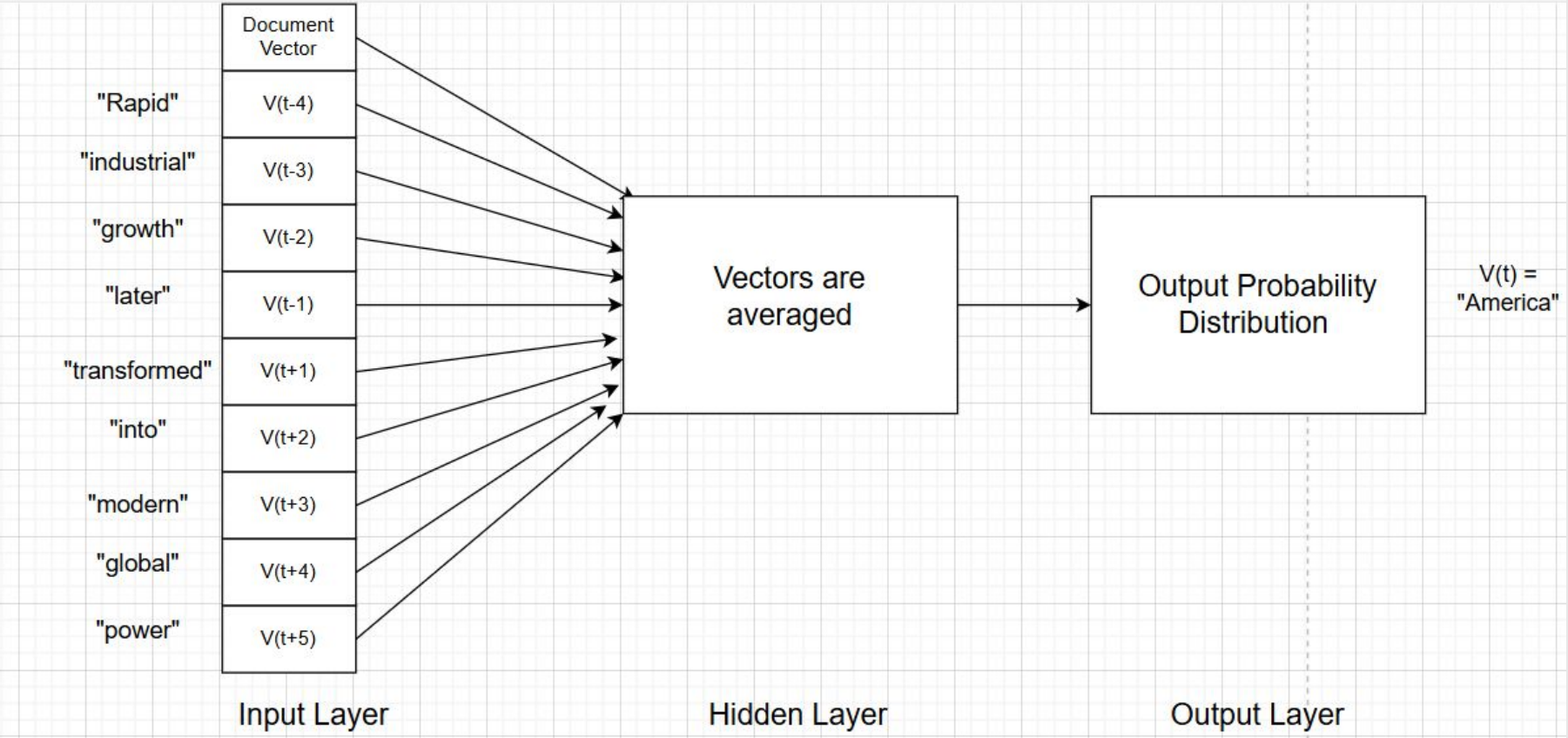
Model Diagram 2



Model Diagram 3



Model Diagram 4



Model in the Long Term

- Over multiple training iterations, the model adjusts the word vectors and document vector to minimize the output prediction error (how far off the output is from the predicted middle word)
- Using backpropagation and gradient descent
- This makes the document vector align with the documents' overall semantics
- Documents that have similar ideas and topics will have overlapping sets of context words. During training, their document vectors will change to reflect the similarities in the documents.
    - The vectors for similar documents will converge toward similar areas in the vector space

Final Vector for a Document

[-0.449723541736602 8, -1.1115018129348755, -0.4288167059421539 3, 0.1373791694641113 3, -0.366965144872665 4, 0.3460772335529327 4, 0.3073736131191253 7, 0.791716396808624 3, -1.177811980247497 6, -0.3252505660057068, 0.7291356921195984, 0.590885996818542 5, 0.1598273068666458, -0.0948212221264839 2, 0.1561938673257827 8, -0.0816936790943145 8, 0.0860782414674758 9, 0.707521438598632 8, -1.770843744277954, -0.384193480014801, -0.1766903549432754 5, 0.811805009841919, 0.0343526452779769 9, 1.234577059745788 6, 0.661862194538116 5]

Database

- Created using AI generated text that answers the teachers question
- The text is generated using Chat GPT 4o mini’s API
- That text is then encoded with Doc2Vec
- Then that vector is stored in the database

Cosine Similarity

- Used to compare how similar two Doc2Vec encoded vectors are (comparing students input vector to the closest database vector)
- Gives a value in the range 0 to 1
- Close to 1 means the generations are similar
- Close to 0 means the generations are not similar
- Example:

$$\cos\theta = \frac{\vec{a}\cdot\vec{b}}{\|\vec{a}\|\|\vec{b}\|}$$

$$\|\vec{a}\| = \sqrt{a_1^2 + a_2^2 + a_3^2 + \cdots + a_n^2}$$

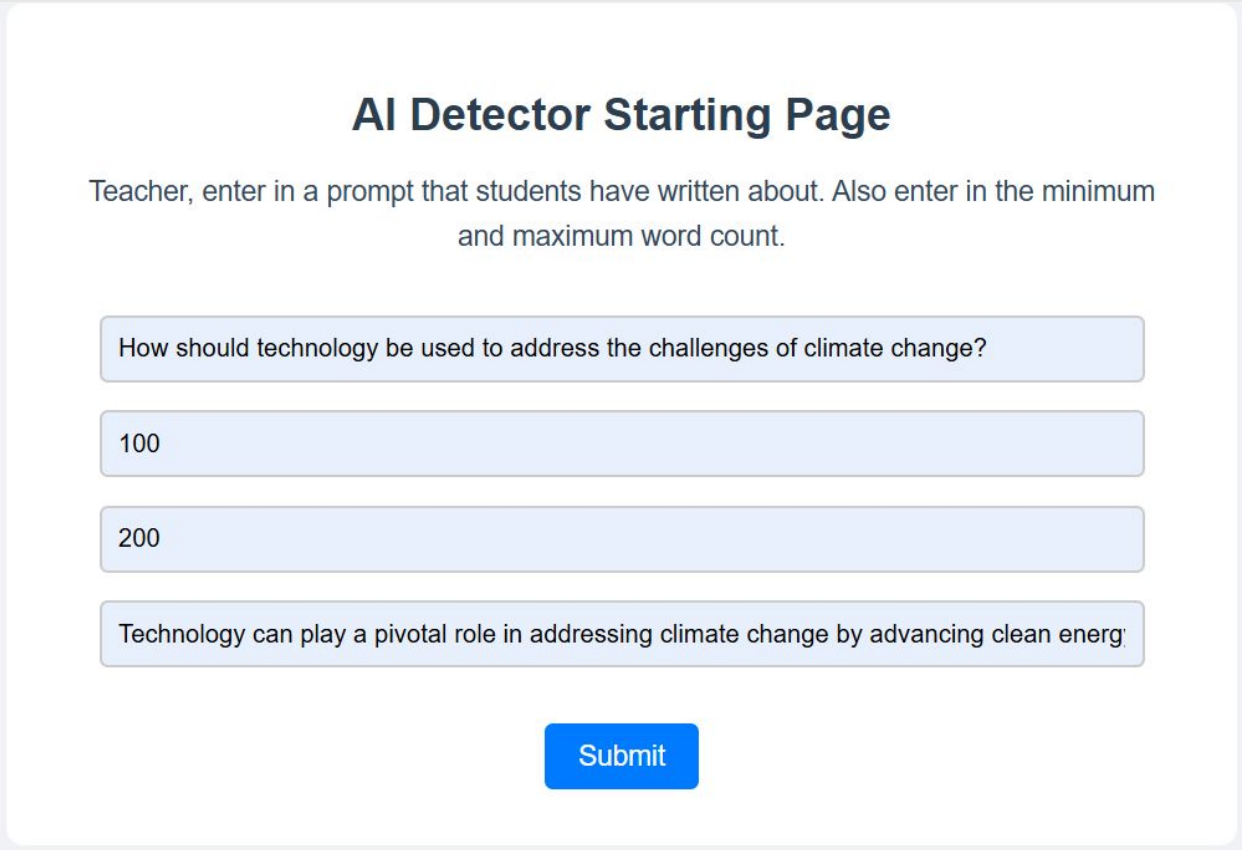
$$\|\vec{b}\| = \sqrt{b_1^2 + b_2^2 + b_3^2 + \cdots + b_n^2}$$

Document 1: [1, 1, 1, 1, 1, 0]
Let's refer to this as A

Document 2: [1, 1, 1, 1, 0, 1]
Let's refer to this as B

Cosine Similarity (A, B) = (4) / (2.2360679775*2.2360679775) = 0.80
(80 percent similarity between the two documents)

Demo


AI Detector Starting Page
Teacher, enter in a prompt that students have written about. Also enter in the minimum and maximum word count.
How should technology be used to address the challenges of climate change?
100
200
Technology can play a pivotal role in addressing climate change by advancing clean energ
Submit

Demo Continued

Generated Responses

The question you have asked to detect on:

How should technology be used to address the challenges of climate change?

minimum word count: **100**

maximum word count: **200**

Go Back | Check if text was AI gen or not

Demo Continued

AI Detection Results:

This is the input text:
Technology can play a pivotal role in addressing climate change by advancing clean energy solutions, improving energy efficiency, and reducing emissions. Renewable energy technologies, such as solar, wind, and hydropower, can replace fossil fuels, while innovations in energy storage and grid management enhance reliability and reduce waste. Additionally, carbon capture and storage (CCS) can help mitigate emissions from industrial sources, and smart agriculture technologies can optimize resource use and reduce environmental impact. Furthermore, data analytics, AI, and IoT can monitor environmental changes in real-time, enabling informed decision-making and fostering more sustainable practices across industries. By leveraging these technologies, we can create a more resilient, low-carbon future.

This is the piece of text the input was closest matched to:
Technology plays a crucial role in addressing climate change by enabling innovative solutions that reduce greenhouse gas emissions and enhance sustainability. Renewable energy technologies, such as solar, wind, and hydroelectric systems, can significantly decrease reliance on fossil fuels. Smart grid technologies optimize energy distribution and consumption, improving efficiency and reducing waste. Additionally, advancements in carbon capture and storage (CCS) can help mitigate emissions from industrial processes by capturing CO2 before it enters the atmosphere. Electric vehicles and energy-efficient transportation technologies can further reduce carbon footprints in urban areas. Moreover, agricultural technologies, such as precision farming and vertical agriculture, can increase food production while minimizing environmental impact. Data analytics and AI can help monitor and predict climate patterns, allowing for better resource management and disaster response. Finally, technological innovations in recycling and waste management can minimize landfill contributions to greenhouse gas emissions. By integrating these technologies into policies and practices, society can create a more sustainable future while actively combating climate change.

The input text is AI Generated

Using a 0.6 value for threshold and a cosine similarity score of 0.7541999727397481

Go Back

Results

- 91.9% accuracy on detecting text generated by AI
- TextVerify takes input text
  - Could be paraphrased
  - Could be from any AI
  - Tells whether it was AI generated or not
- Got better performance than current AI detectors

Model/Category	Correctly Classified	Total Tested	Accuracy (%)
GPT-4 Output	10	10	100
Human Written	9	10	90
DeepSeek Output	2	3	66.7
Copilot Output	2	3	66.7
Grok Output	3	3	100
GPT Paraphrased	2	2	100
DeepSeek Paraphrased	2	2	100
Copilot Paraphrased	2	2	100
Grok Paraphrased	2	2	100
Total Accuracy	34	37	91.9

Limitations

- Text not generated by students
- Text with shorter number of words
- Not using watermarking and statistical outlier methods
- Text written in other languages

Future Work

- Improvements in text generated for other things besides school
- Using retrieval methods along with other methods (watermarking, statistical outlier) to improve shorter text classification
- Using retrieval methods with AI generations in other languages

Conclusion

- I solved the problem of AI Detection using Retrieval Methods

References

[1]A. M. Elkhatat, K. Elsaid, and S. Al-Meer, “Evaluating the efficacy of AI content detection tools in differentiating between human and AI-generated text,” International journal for educational integrity, vol. 19, 1, Sep. 2023, doi: https://doi.org/10.1007/s40979-023-00140-5

[2]A. Singh, “A Comparison Study on AI Language Detector,” IEEE Xplore, Mar. 2023, doi: https://doi.org/10.1109/ccwc57344.2023.10099219

[3]M. Perkins et al., “Simple techniques to bypass GenAI text detectors: implications for inclusive education,” International Journal of Educational Technology in Higher Education, vol. 21, no. 1, Sep. 2024, doi: https://doi.org/10.1186/s41239-024-00487-w

[4]K. Krishna, Y. Song, M. Karpinska, J. Wieting, and M. Iyyer, “Paraphrasing evades detectors of AI-generated text, but retrieval is an effective defense Mohit Iyyer,” Oct. 2023, doi: https://doi.org/10.48550/arXiv.2303.13408

[5]D. Dukić, D. Keča, and D. Stipić, “Are You Human? Detecting Bots on Twitter Using BERT,” IEEE Xplore, Oct. 01, 2020. doi: https://doi.org/10.1109/DSAA49011.2020.00089. Available: https://ieeexplore.ieee.org/document/9260074

[6]G. Shperber, “A gentle introduction to Doc2Vec,” Medium, Nov. 05, 2019. Available: https://medium.com/wisio/a-gentle-introduction-to-doc2vec-db3e8c0cce5e

[7]A. Prakash, “Understanding Cosine Similarity: A key concept in data science,” Medium, Sep. 21, 2023. Available: https://medium.com/@arjunprakash027/understanding-cosine-similarity-a-key-concept-in-data-science-72a0fcc57599

[8]“OpenAI Platform,” Openai.com, 2025. Available: https://platform.openai.com/docs/quickstart?api-mode=chat

[9]E. Tian, “GPTZero,” gptzero.me, 2022. Available: https://gptzero.me/

[10]“Copyleaks: AI & Machine Learning Powered Plagiarism Checker,” copyleaks.com. Available: https://copyleaks.com/

[11]“Crossplag,” app.crossplag.com. Available: https://app.crossplag.com/individual/detector

[12]V. Chen, “How Do AI Detectors Work? | GPTZero,” AI Detection Resources | GPTZero, Oct. 14, 2024. Available: https://gptzero.me/news/how-ai-detectors-work/

[13]“AI Content Detector FAQs How It Works Understanding the Results Detection Capabilities & Limitations.” Available: https://copyleaks.com/wp-content/uploads/2023/05/ai-content-detector-faqs.pdf

[14]Agnesa Nuha, “Detecting if a text is AI generated - Crossplag,” Crossplag, Dec. 19, 2022. Available: https://crossplag.com/detecting-if-a-text-is-ai-generated/. [Accessed: Apr. 24, 2025]

Q&A

THANKS!