Semantic Plagiarism Detection: Enhancing Jaccard Similarity with Graph-Based Normalization and Fuzzy Sets

Used To Fulfill The "IF1220 – Discrete Mathematic" Paper Assignment

Michael James Liman - 13524106
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail: jamesliman7@gmail.com , 13524106@std.stei.itb.ac.id

***Abstract*—Traditional plagiarism detection tools that rely solely on lexical matching are often ineffective against semantic paraphrasing, where the underlying meaning of a text is preserved but the vocabulary and sentence structure modified. This paper proposes a multi-phase methodology to overcome this limitation. The approach uses a graph-based normalization using WordNet database, which traverses each word's synonym and derivational relationship to get their root word. Then, shingling technique is used to represent the normalized text as sets of overlapping phrases. Similarity is then measured using a Fuzzy Jaccard Index, which provides a more diverse score than classical set-based methods by accounting for the frequency of these phrases. The implemented system demonstrates a superior ability to detect conceptual overlap in heavily paraphrased texts, validating that a combination of graph theory and fuzzy logic provides an effective method for semantic plagiarism detection.**

Keywords—plagiarism detection, graph, fuzzy set, jaccard similarity

I. Introduction

In the digital age, plagiarism has emerged as a challenge in academic and research environments, threatening the integrity of intellectual work and publications. The widespread availability of digital content and easy access to textual materials have significantly increased the demand for robust automated plagiarism detection systems [1].

Plagiarism detection has evolved significantly with the advancement of computational techniques, moving beyond simple lexical matching to more sophisticated semantic analysis. Traditional plagiarism detection systems primarily relied on text representation and similarity approaches that identify matching words appearing in both suspicious and source documents [1]. These conventional methods, while effective for detecting verbatim copying, face significant limitations when confronted with semantic plagiarism, where ideas are appropriated but expressed using different vocabulary and sentence structures.

The limitations of existing plagiarism detection methods highlight the need for integrated approaches that combine multiple techniques to address the complex challenge of semantic plagiarism [1]. The integration of graph-based normalization with fuzzy set theory and enhanced Jaccard similarity represents a promising direction for advancing the field of plagiarism detection. By leveraging the structural representation capabilities of graphs, the flexible membership concept of fuzzy sets, and the established foundation of Jaccard similarity, such an integrated approach can address many of the limitations faced by traditional methods.

II. Theoretical Framework

A. Graph

Graph theory provides a fundamental mathematical framework for representing relationships between discrete objects. In discrete mathematics, a graph is formally defined as an ordered pair

$$G = (V, E), \quad (1)$$

where V represents a nonempty set of vertices (or nodes) and E represents a set of edges that each connects to one or two vertices, which called endpoints [2].

The mathematical definition of a graph encompasses several key properties that make it particularly suitable for representing textual and semantic relationships. Vertices in a graph can represent any discrete entity, such as words, concepts, or documents, while edges capture the relationships or connections between these entities. Various graph types arise by imposing constraints on vertices, edges, or their relationships. Below is an overview of the most common graph types [2]:

TABLE I. Graph Terminology [2]

Type	*Edges*	*Multiple Edges*	*Loops*
Simple graph	Undirected	No	No
Multigraph	Undirected	Yes	No
Pseudograph	Undirected	Yes	Yes
Simple directed graph	Directed	No	No

Type	*Edges*	*Multiple Edges*	*Loops*
Directed multigraph	Directed	Yes	Yes
Mixed graph	Directed & undirected	Yes	Yes

1) Based on loops and parallel edges

a) Simple Graph: An undirected graph with no loops or multiple edges between the same pair of vertices

b) Multigraph: An undirected graph that allows multiple edges (parallel edges) between the same vertices.

c) Pseudograph: A multigraph that permits loops connecting a vertex to itself.

2) Based on direction of edges

a) Undirected Graph: Edges are unordered pairs {u,v}{u,v}, implying a two-way relationship.

b) Directed Graph (Digraph): Edges are ordered pairs (u, v), indicating a one-way connection from *u* (tail) to *v* (head).

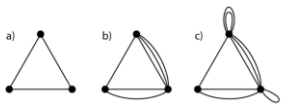

Fig. 1 Simple graph (a), multigraph (b), pseudograph (c) [3]

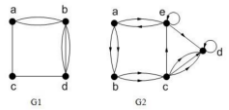

Fig. 2 Undirected Graph (G1) and Digraph (G2) [3]

B. Set

Set is the foundational discrete structure upon which all other discrete structures are built. A set is a group of distinct items, called its elements or members, where the order of these items doesn't matter. When an item *a* is part of a set *A*, we write this as $a \in A$. If *a* is not in set *A*, we write it as $a \notin A$. Two sets, *A* and *B*, are considered equal if and only if they contain precisely the same elements. This means that for any object *x*, *x* is an element of *A* if and only if *x* is also an element of *B*. We denote this equality as $A = B$. The size of a union *A* denoted as $|A|$ [2].

There are four main set operations that are mathematical ways to manipulate sets [2]:

1) Union (U)

Let *A* and *B* be sets. The union of the sets *A* and *B*, denoted by $A \cup B$, is the set that contains those elements that are either in *A* or in *B*, or in both

2) Intersection (cap)

Let *A* and *B* be sets. The intersection of the sets *A* and *B*, denoted by $A \cap B$, is the set containing those elements in both *A* and *B*.

3) Difference (-)

Let *A* and *B* be sets. The difference of *A* and *B*, denoted by $A - B$, is the set containing those elements that are in *A* but not in *B*. The difference of *A* and *B* is also called the complement of *B* with respect to *A*.

4) Complement

Let *U* be the universal set. The complement of the set *A*, denoted by $\bar{A}$, is the complement of *A* with respect to *U*. Therefore, the complement of the set *A* is $U - A$.

C. Jaccard Index

The Jaccard similarity coefficient, also known as the Jaccard index, is a traditional metric used in plagiarism detection that measures the similarity between finite sample sets by calculating the ratio of the size of the intersection to the size of the union of the sets, expressed mathematically as in

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \quad (2)$$

The Jaccard index satisfies several important mathematical properties that make it a robust similarity measure. It ranges from 0 to 1, where 0 indicates no similarity (disjoint sets) and 1 indicates perfect similarity (identical sets). The index is symmetrical, meaning

$$J(A,B) = J(B,A), \quad (3)$$

and it satisfies the triangle inequality when transformed into a distance metric through the relationship: Jaccard Distance = 1 - Jaccard Index. While effective for exact matching, traditional Jaccard similarity has limitations when dealing with semantic variations and paraphrasing [4].

D. Fuzzy Set

Fuzzy set theory, introduced by Lotfi A. Zadeh in 1965, extends classical set theory by allowing elements to have degrees of membership in a set rather than the binary membership of classical sets [5]. The mathematical foundation of fuzzy sets rests on the concept of a membership function $\mu_{A(x)}$, which assigns to each element x in the universe of discourse a membership degree in the interval. When $\mu_A(x) = 1$, the element has full membership; when $\mu_A(x) = 0$, the element has no membership; and when $0 < \mu_A(x) < 1$, the element has partial membership [6].

The algebraic operations on fuzzy sets are defined through their membership functions. The intersection of two fuzzy sets A and B is typically defined using the minimum operation:

$$\mu_{(A \cap B)}(x) = \min(\mu_A(x), \mu_B(x)). \quad (4)$$

The union is defined using the maximum operation:

$$\mu_{(A \cap B)}(x) = \max(\mu_A(x), \mu_B(x)). \quad (5)$$

The complement of a fuzzy set A is defined as:

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x) \text{ [6]}. \quad (6)$$

Fuzzy set theory provides a mathematical framework for handling uncertainty and imprecision in real-world applications. The theory has been extended to include Type-2 fuzzy sets, where the membership functions themselves are fuzzy, providing additional layers of uncertainty modeling [7].

E. *WordNet*

WordNet represents a lexical database that organizes words according to their semantic relationships rather than alphabetical ordering. The fundamental structure of WordNet is built around the concept of synsets (synonym sets), which are collections of words that share the same meaning or sense. Each synset represents a unique concept or word sense, providing the basic building blocks for the semantic network [8], [9].

The mathematical structure of WordNet can be viewed as a directed graph where synsets serve as vertices and semantic relations serve as edges. The primary semantic relations in WordNet include hypernymy (is-a relationships), hyponymy (kind-of relationships), meronymy (part-of relationships), and antonymy (opposite relationships). These relations create a hierarchical structure that captures the conceptual organization of natural language as in Fig. 1 [10].

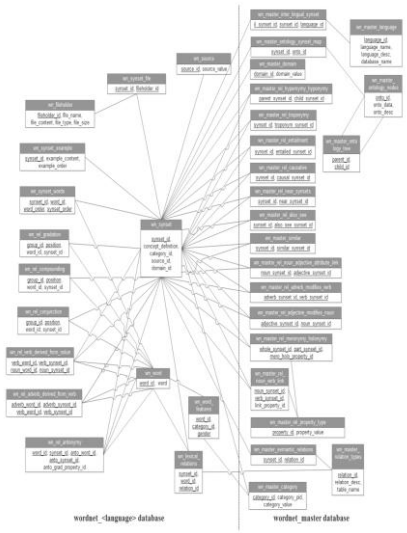

Fig. 3. Database model diagram of WordNet [11]

WordNet's database structure typically consists of multiple interconnected tables that store different types of linguistic information. The synset table contains the core semantic units, while relation tables capture the connections between synsets. Additional tables store lexical information, including word forms, part-of-speech tags, and usage frequencies [9].

III. Methodology

The core of the tool is a multi-phase process that transform raw sentences into discrete mathematical representation, which can then be analyzed.

A. *Graph-Based Synonym Normalization*

The problem with traditional similarity checking rests on the inability to compare two semantically exact sentences with completely different words by the process of changing every word in the sentences with its synonym or even with different parts of speech. To address the issue of semantic equivalence across different parts of speech and synonyms, we use a multi-step normalization process rooted in the graph structure of the WordNet lexical database.

1) Part-of-Speech Tagging: Each word in the input text is first tagged with its grammatical part of speech (e.g., noun, verb, adjective). This step is crucial for accurate lemmatization.
2) Lemmatization: Using the POS tag, we find the dictionary root form, or "lemma," of each word. For example, the verb "destroying" is correctly lemmatized to "destroy".
3) Derivational Graph Traversal: The key innovation lies in using WordNet's graph structure to connect words that share a root but have different parts of speech. For a given lemma, we traverse its "derivationally related forms." This allows the algorithm to discover that the noun "destruction" is derived from the verb "destroy". By prioritizing the verb as the canonical root, we can map both "destruction" and "destroying" to the single token "destroy". This process can be visualized as finding a common root node in a complex graph of related words by Fig. 2.

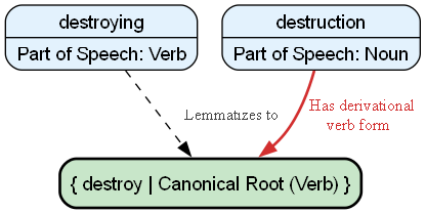

Fig. 4 Graph of "destroying" and "destruction" lemmatization

The normalization process was taken out with python library *nltk* (Natural Language Toolkit) by using its lemmatizer to search for the shortest path from each word in a text to a possible lemma in WordNet. Furthermore, we traverse each derivational related form until we got a synset of verb. If a verb derivational related synset was not found, then we lemmatize the raw word as a verb. Those processes carried out by:

```
def normalize_text(text):
    stop_words = set(['the', 'a', 'an', 'is', 'of', 'for', 'in', 'to', 'and', "'s", 'by', 'was', 'done',
'their'])
    lemmatizer = nltk.WordNetLemmatizer()

    raw_tokens = nltk.word_tokenize(text.lower())
    pos_tagged_tokens = nltk.pos_tag(raw_tokens)

    normalized_tokens = []

    for word, tag in pos_tagged_tokens:
        if not word.isalpha() or word in stop_words:
            continue

        wn_pos = get_wordnet_pos(tag)
        lemma = lemmatizer.lemmatize(word, pos=wn_pos)
        canonical_form = lemma

        # If the word is a noun, try to find a related verb form to use as the root.
        if wn_pos == wordnet.NOUN:
            found_verb_root = False
            # Try to find a verb root from the noun's derivational forms
            synsets = wordnet.synsets(lemma, pos=wn_pos)
            if synsets:
                for related_lemma in synsets[0].lemmas():
                    for related_form in related_lemma.derivationally_related_forms():
                        if related_form.synset().pos() == wordnet.VERB:
                            canonical_form = related_form.name()
                            found_verb_root = True
                            break
                    if found_verb_root:
                        break

            # If no derivational verb root was found (e.g., for "destroying" tagged as a noun),
            # try lemmatizing the original word as a verb. This handles POS tagging errors.
            if not found_verb_root:
                verb_lemma = lemmatizer.lemmatize(word, pos=wordnet.VERB)
                if verb_lemma != word:
                    canonical_form = verb_lemma

        normalized_tokens.append(canonical_form)

    return normalized_tokens
```

Fig. 5 Python code for text normalization

B. *Shingling and Crisp Set Representation*

After synonym normalization, the documents are converted into a format suitable for set-based comparison. This is achieved through **k-shingling**. A k-shingle is a contiguous sequence of k words from the text. This technique preserves more context than single-word analysis. For our tool, we use a default k of 2. Therefore, if a document has N tokens after pre-processing, the process of k-shingling generates sets of token of $N - k + 1$ or $N - 1$ (by default) overlapping subset of pre-processed set of tokens. The processed text is thus converted into a list of shingles, which we then represent as a classical (or "crisp") mathematical set. In a crisp set, an element is either a member or it is not. This process removes all duplicate shingles.

The process carried out by:

```
def create_shingles(tokens, k=2):
    if len(tokens) < k:
        return [" ".join(tokens)] if tokens else []
    shingles = []
    for i in range(len(tokens) - k + 1):
        shingles.append(" ".join(tokens[i:i+k]))
    return shingles
```

Fig. 6 Python code for creating shingles

C. *Fuzzy Set Representation and the Fuzzy Jaccard Index*

To overcome the binary nature of the standard Jaccard Index, we introduce **fuzzy set theory**. While the previous step used combinatorics to generate a set of unique shingles, this phase analyzes the importance of those shingles. The foundation of this analysis is the **universe of discourse**, X, which is the union of all unique shingles from both documents. The creation of this universe is itself a combinatorial and set-theoretic result of the shingling process.

In fuzzy set theory, an element's membership is not absolute but is instead a real number between [0, 1]. This value, μ(x), is the **degree of membership**. We redefine each document as a fuzzy set of shingles, where the degree of membership for each shingle is determined by its **normalized term frequency (TF)**. This allows us to capture the idea that shingles that appear more frequently are more central to the document's theme.

The membership function for a shingle x in document D is:

$$\mu_D(x) = \frac{f_D(x)}{\max(f_D)}, \qquad (7)$$

where $f_D(x)$ is the frequency of shingle x in document D, and $nax(f_D)$ is the frequency of the most common shingle in that document.

With documents now represented as fuzzy sets, we can now define a **Fuzzy Jaccard Index** using fuzzy equivalents of intersection and union, defined at *Section II.D Fuzzy Set*. The size of a fuzzy set is the sum of its membership degrees. Therefore, the Fuzzy Jaccard Index is:

$$J_F(A,B) = \frac{\sum_{x \in X} \min(\mu_A(x), \mu_B(x))}{\sum_{x \in X} \max(\mu_A(x), \mu_B(x))}. \qquad (8)$$

Those processes carried out by:

```
def calculate_fuzzy_jaccard(shinglesA, shinglesB):
    freqA = {shingle: shinglesA.count(shingle) for
shingle in set(shinglesA)}
    freqB = {shingle: shinglesB.count(shingle) for
shingle in set(shinglesB)}
    universe = set(shinglesA) | set(shinglesB)
    if not universe: return 1.0

    max_freq_A = max(freqA.values()) if freqA else 1
    max_freq_B = max(freqB.values()) if freqB else 1

    fuzzy_intersection_sum = 0
    fuzzy_union_sum = 0

    for shingle in universe:
        muA = (freqA.get(shingle, 0)) / max_freq_A
        muB = (freqB.get(shingle, 0)) / max_freq_B
        fuzzy_intersection_sum += min(muA, muB)
        fuzzy_union_sum += max(muA, muB)

    if fuzzy_union_sum == 0: return 1.0
    return fuzzy_intersection_sum / fuzzy_union_sum
```

Fig. 7. Python code for calculating the similarity score

IV. Result

The tool then tested against various plagiarism checker in the internet to compare multiple English test texts with the paraphrased version using Gemini 2.5 Flash with prompt of “Paraphrase this text below (give only one result)”.

A. *Sample Text*

1) *Base text*

a) WordNet is a semantic lexicon for the English language that is used extensively by computational linguists and cognitive scientists. WordNet groups words into sets of synonyms called synsets and describes semantic relationships between them. One such relationship is the is-a relationship, which connects a hyponym (more specific synset) to a hypernym (more general synset). For example, a plant organ is a hypernym to plant root and plant root is a hypernym to carrot [12].

b) Proponents of geoengineering have never regarded the earth-changing engineering projects as a complete solution. Nevertheless, the concept as a whole attracts many criticisms. One is that the problem of climate change is of such huge scale and complexity that there will not be one single solution. All proposals so far have advantages and disadvantages. The biggest problem of all is that many of the projects are untested and any of the proposals may have unforeseen consequences. For example, we could not suddenly stop a geoengineering scheme: keeping temperatures artificially low for a period then taking away the cause of this would cause the temperature to rise again rapidly. Furthermore, global engineering solutions to the problem of climate change would need the agreement of all the world's leaders: having an American solution, a Chinese solution, a Brazilian solution, and so on simply wouldn't be

politically acceptable. But the biggest downfall is that geoengineering projects could reduce the political and popular pressure for reducing carbon emissions, as politicians point to geoengineering for an answer rather than tackling the real cause of climate change: human activity [13].

c) Science produces ideas about how the world works, whereas the ideas in technology result in usable objects. Technology is much older than anything one could regard as science and unaided by any science. Technology gave rise to the crafts of early humans, like agriculture and metalworking. It is technology that carries with it ethical issues, from motorcar production to cloning a human [14].

2) Paraphrased text by Gemini 2.5 Flash

a) WordNet is an English dictionary that organizes words into synonym sets (synsets) and shows how they relate semantically. It's widely used by researchers in computational linguistics and cognitive science. For instance, it defines "is-a" relationships where a more specific term (hyponym) is linked to a more general one (hypernym); for example, "plant root" is a more specific type of "plant organ," and "carrot" is a more specific type of "plant root."

b) While supporters don't see geoengineering as a full answer to climate change, the idea faces significant criticism. Critics argue that climate change is too vast and complex for a single solution, and all current geoengineering proposals have pros and cons, with the biggest concern being the untested nature of many projects and potential unforeseen consequences. For example, abruptly halting a geoengineering effort could lead to a rapid temperature rebound. Furthermore, implementing global geoengineering solutions would require universal agreement among world leaders, as nationally specific approaches would be politically unfeasible. Ultimately, the most significant drawback is the risk that geoengineering could diminish the urgency to reduce carbon emissions, as politicians might favor technological fixes over addressing the root cause of climate change: human activity.

c) Science focuses on understanding how the world operates, while technology translates ideas into practical tools and objects. Technology predates and developed independently of formal science, giving rise to ancient crafts like farming and metalworking. Furthermore, technology, from car manufacturing to human cloning, is inherently linked to ethical considerations.

B. Performance Result

To validate the performance of the proposed methodology, a comparative analysis was conducted against two other tools on the internet. The three pairs of documents were analyzed by those tools which have its similarity score from each tool presented by Table II.

TABLE II. PERFORMANCE RESULT

Text Number	***Similarity Score [0, 1]***		
	Proposed Solution	***Tool A***[a]	***Tool B***[b]
a	0,0655	0,1	0,06
b	0,0769	0,19	0,08
c	0,0494	0,06	0,04

[a] https://gowinston.ai/text-compare/

[b] *https://www.prepostseo.com/plagiarism-comparison-search*

V. CONCLUSION AND LIMITATION

A. Conclusion

The proposed solution uses multi-layered plagiarism checker by synthesizing graph theory, set theory, and fuzzy logic. Therefore, the proposed solution moves beyond naïve lexical matching to more robust semantic analysis. The graph-based normalization allowing the tool to be effective even under paraphrasing. Then, the use of Fuzzy Jaccard Index provided more diverse and accurate similarity score than a traditional binary approach. The comparative analysis presented in Table II proves that the proposed solution generates consistent and sensitive results to plagiarized texts even though the texts have been paraphrased using advanced generative AI model.

B. Limitation

While the proposed solution shows effectiveness and sensitivity across all sample texts, it still has some limitations:

1) **Lack of grammartical awareness:** The shingling method captures local word order and context but does not get the grammartical structure of each sentence. Therefore, it could be tricked by inverting its grammartical structure or changing its tense.

2) **Naive term frequency:** The current implementation of term frequency weight each word equally thus it cannot distinguish between commond words (e.g., “system”, “analysis”) and highly specific terms.

VI. APPENDIX

1) Code at Github :
https://github.com/MichaelJamesL/DiscreteMath-Paper
2) Video Presentation at Youtube :
https://youtu.be/JUSWTwh5XpA

ACKNOWLEDGMENT

First, the author would like to express his gratitude to God Almighty for His blessings, directions, and grace that were important in the success of this paper. Sincere appreciation and gratitude are also directed to Dr. Ir. Rinaldi Munir, M.T. as the author of Discrete Mathematic Material of STEI ITB and Arrival Dwi Sentosa, S.Kom., M.T as author’s lecturer, for their dedication and expertise in teaching. Finally, the author also incredibly grateful for the support from his family and friends,

which have been the source of motivation throughout the writing of this paper.

REFERENCES

[1] A. A. M. Saeed and A. Y. Taqa, "An Intelligent Approach for Semantic Plagiarism Detection in Scientific Papers," in *2022 8th International Conference on Contemporary Information Technology and Mathematics (ICCITM)*, IEEE, Aug. 2022, pp. 107–112. doi: 10.1109/ICCITM56309.2022.10031641.

[2] Kenneth H. Rosen, *Discrete Mathematics and Its Applications*, 7th ed. New York: McGraw-Hill, 2012.

[3] R. Munir, "20-Graf-Bagian1-2024," *Bahan Kuliah IF1220 Matematika Diskrit*, 2024.

[4] D. Ogwok and E. M. Ehlers, "Jaccard Index in Ensemble Image Segmentation: An Approach," in *Proceedings of the 2022 5th International Conference on Computational Intelligence and Intelligent Systems*, New York, NY, USA: ACM, Nov. 2022, pp. 9–14. doi: 10.1145/3581792.3581794.

[5] K. Gupta, D. K. Tayal, and A. Jain, "Evolution of Fuzzy Set Theory under Monotonic Constraints," in *2023 International Conference on Advancement in Computation & Computer Technologies (InCACCT)*, IEEE, May 2023, pp. 41–45. doi: 10.1109/InCACCT57535.2023.10141786.

[6] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, no. 3, pp. 338–353, Jun. 1965, doi: 10.1016/S0019-9958(65)90241-X.

[7] Y. Güven, A. Köklu, and T. Kumbasar, "Zadeh's Type-2 Fuzzy Logic Systems: Precision and High-Quality Prediction Intervals," in *2024 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, IEEE, Jun. 2024, pp. 1–6. doi: 10.1109/FUZZ-IEEE60900.2024.10611797.

[8] P. Gulhayo, "WORDNET – A LEXICAL DATABASE FOR LINGUISTIC ONTOLOGIES," *Current Research Journal of Philological Sciences*, vol. 5, no. 12, pp. 27–32, Dec. 2024, doi: 10.37547/philological-crjps-05-12-06.

[9] H. Redkar, S. Bhingardive, D. Kanojia, and P. Bhattacharyya, "World WordNet Database Structure: An Efficient Schema for Storing Information of WordNets of the World," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, no. 1, Mar. 2015, doi: 10.1609/aaai.v29i1.9276.

[10] A. Cocos, M. Apidianaki, and C. Callison-Burch, "Mapping the Paraphrase Database to WordNet," in *Proceedings of the 6th Joint Conference on Lexical and Computational Semantics (*SEM 2017)*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2017, pp. 84–90. doi: 10.18653/v1/S17-1009.

[11] H. Redkar, S. Bhingardive, D. Kanojia, and P. Bhattacharyya, "World WordNet Database Structure: An Efficient Schema for Storing Information of WordNets of the World," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, no. 1, Mar. 2015, doi: 10.1609/aaai.v29i1.9276.

[12] "WordNet," https://www.cs.princeton.edu/courses/archive/spr07/cos226/assignments/wordnet.html.

[13] "IELTS Reading Academic 34," https://www.ielts-writing.info/EXAM/docs/reading/IELTS_Reading_Academic_54.htm.

[14] "Academic Reading 26 - Passage 1," https://www.ielts-writing.info/EXAM/docs/reading/IELTS_Reading_Academic_26_Passage_1.htm#google_vignette.

STATEMENT

I hereby declare that the paper I wrote is my own writing, not an adaptation or translation of someone else's paper, and is not plagiarized.

Bandung, 20th June 2025

Michael James Liman
13524106