International Journal of Engineering and Management Research
e-ISSN: 2250-0758 | p-ISSN: 2394-6962
https://ijemr.vandanapublications.com
Peer Reviewed & Refereed Journal
Volume-13, Issue-5 (October 2023)
https://doi.org/10.31033/ijemr.13.5.4

Harnessing Knowledge-Based Concepts for Streamlined Mind Mapping in Academic Study

Weerasekara N.N[1], Senaweera T.I.S[2], Wijayasena W.D.N.D.T[3], Colambage K.G[4], Prasanna S. Haddela[5] and Sathira Hettiarachchi[6]
[1]Student, Faculty of Computing, Sri Lanka Institute of Information Technology, Colombo, SRI LANKA
[2]Student, Faculty of Computing, Sri Lanka Institute of Information Technology, Colombo, SRI LANKA
[3]Student, Faculty of Computing, Sri Lanka Institute of Information Technology, Colombo, SRI LANKA
[4]Student, Faculty of Computing, Sri Lanka Institute of Information Technology, Colombo, SRI LANKA
[5]Senior Lecturer, Faculty of Computing, Sri Lanka Institute of Information Technology, Colombo, SRI LANKA
[6]Lecturer, Faculty of Computing, Sri Lanka Institute of Information Technology, Colombo, SRI LANKA

[3]Corresponding Author: dinethwija1806@gmail.com

Received: 03-09-2023 Revised: 19-09-2023 Accepted: 03-10-2023

ABSTRACT

The main goal of this research project is to build and implement a knowledge-based system that will produce more accurate and significant mind maps from textual input. This study suggests the creation of a knowledge-based system devoted to mind map construction, as opposed to current text mining algorithms and machine learning techniques, which might not be able to properly capture the semantic subtleties of input material. The suggested system aims to reduce the constraints of conventional techniques by incorporating domain-specific information and rules into the analytical process. The accuracy and relevance of the produced mind maps are projected to improve with the addition of this knowledge. This project's main objective is to build a knowledge-based framework that overcomes the limitations of the existing text mining and machine learning approaches, resulting in more sophisticated and theoretically rich mind maps from textual data.

Keywords— Knowledge-Based System, Machine Learning, Mind Maps, Text Mining

I. INTRODUCTION

Mind Mapping is a widely recognized method for taking notes, which has proven to be highly effective in facilitating learning and self-study. This technique allows one to create visual representations of concepts and ideas in a non-linear, free-form manner that is easy to understand and remember. Mind mapping can play a crucial role in the self-learning process for students. By allowing the mind to draw correlations and connections between many pieces of knowledge, this method promotes creativity and aids in the development of critical thinking skills. Mind mapping can help students to visualize the relationships between different concepts and ideas. Mind maps provide a graphical representation of the relationships between different topics, making it easier to understand complex information and identify connections between different concepts. The thing is many students don't use mind maps because creating mind maps requires extra mental efforts. Only a few are able enough to draw good mind maps and the rest don't have that ability to draw mind maps. My point is that if every student gets a chance to summarize their work using a mind map, they could improve their memory retention significantly. This paper provides a tool that builds a mind map according to the given study material by the student. This will be helpful for the student who don't know how to create or make a mind map. Overall, mind mapping is a flexible and useful tool that may be used in a variety of contexts, including education, business, and personal growth, by people of all ages and professions. In conclusion, Mind mapping can be a powerful way to help students understand and apply it in their own learning process. This technique can help to simplify complex information and provide a visual representation of the relationships between different concepts, making it easier for students to understand and recall the material. By using this tool, students can improve their self-studying process, which can lead to improved academic performance and success in their academic pursuits.

A. Initial Survey

In the beginning, we carried out a thorough study to determine students' capacity for mind mapping and their levels of satisfaction with the available mind mapping tools. We were able to learn more about students' abilities to create mind maps and their opinions on the efficiency of the available tools in this field by gleaning insightful information from this survey.

This work is licensed under Creative Commons Attribution 4.0 International License.

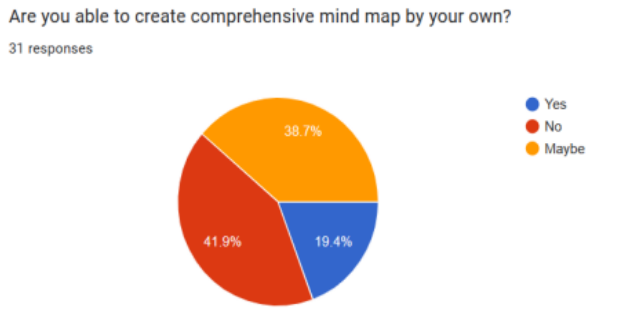

Figure 1: Responses for the ability to create a mind map.

Our study's key result was that most people lack the abilities necessary to efficiently create comprehensive mind maps. Additionally, the evaluation of current mind map creation technologies revealed that customers are often dissatisfied. These results highlight the relevance of our suggested knowledge-based methodology, which strives to solve the shortcomings in current approaches and enable users to produce more accurate and meaningful mind maps.

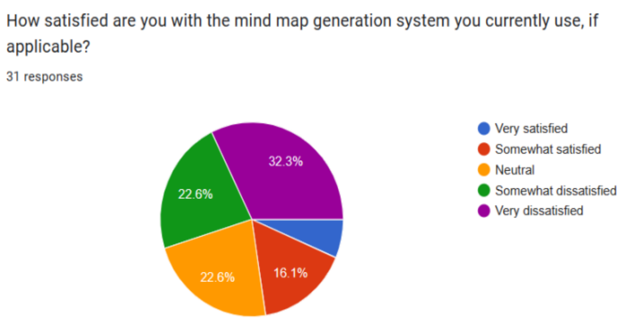

Figure 2: Responses for the current systems

II. LITERATURE REVIEW

In recent years, academics have focused a lot of emphasis on mind map creation systems as they look into different technologies to offer the best solution to consumers. The methods and resources that were employed by the researchers are discussed in the section that follows.

In 2021, Yeoil, Taejin and Namgyu Kim who were Graduates in Kookmin University in faculty of Business IT have done research about Deep Learning-Based Knowledge Graph Generation for COVID-19 [1]. In order to create a knowledge network particular to COVID-19, this study suggests an Open Information Extraction (OpenIE) method based on unsupervised learning. The system uses a COVID-19 entity dictionary and a fine-tuned BERT language model to extract connecting words between entities and outperforms original BERT in terms of accuracy and score. It highlights the two methods of natural language processing - statistical and neural network - and emphasizes the benefits of using neural networks, such as the ability to efficiently expand the model.

Researchers from the University of Mexico, Ivan Lopez-Arevalo, Jose L. Martinez-Rodriguez, and Ana B. Rios-Alvarado conducted research in 2018 on an OpenIE-based method for building knowledge graphs from text. [2] This method involves the use of Natural Language Processing (NLP) and Information Extraction (IE) techniques to convert the input text into a machinereadable format consisting of RDF triples. RDF graphs are used to represent entities and relationships between them, where entities can be anything that has a unique identifier, such as a person, place, concept, or event. Relationships between entities are represented as edges in the graph, with properties that describe the nature of the relationship. By constructing a graph from these triples, Knowledge Graphs can be used to represent knowledge from multiple sources and domains and to support various applications, such as information retrieval and natural language processing.

A research study for creating mind maps from articles using machine learning was completed in 2019 by M.F. Kuroki, L.S. Riza, and Rasim at the department of computer science education at Universitas Pendidikan Indonesia. [3] For the data collection they have used articles. The topic sentence of a paragraph is chosen using the information retrieval approach, pre-processing, core NLP, and feature extraction approaches in model creation. Application development uses a stageby-stage, linear process model. In experiments, experts choose the criteria for articles, and the application and the experts choose the topic phrases. The results are articles with topic sentences in each paragraph. With a title, subtitle, and topic sentence for each paragraph, this system creates teaching materials in the form of a mind map. The system's output is contrasted with the average values produced by two human experts, whose accuracy rate averages 53.55%. This indicates a moderate level of system correctness.

In 2013, Ayu Purwarianthi, Athia Saelan, Irfan Afif, Filman Ferdian, and Alfan Farizki conducted research on developing an autonomous mind map generator in Indonesian language at the school of electrical engineering and informatics institution in Indonesia. [4] Indonesian Mind Map Generator utilizes Indonesian natural language understanding tools such as a POS tagger, syntactic parser, and semantic analyzer to facilitate easy creation of Mind Map objects. The tools' accuracy rates for the POS tagger, syntactic parser, and semantic analyzer are 96.5%, 47.22%, and 62.5%, respectively. They were created with the aim of addressing the dearth of Indonesian language resources. The Mind Map generator also employs radial drawing visualization and an editor for modifications. In evaluation, the Mind Map object was easily understood for simple sentences by 5 respondents.

This work is licensed under Creative Commons Attribution 4.0 International License.

In 2020 Danilo Dessi, Franceso Osborne, Diego Recupero, Davide Buscaldi and Enrico Motta have done research on knowledge graphs creation from NLP. [5] The abundance of scientific literature makes analysis difficult, necessitating technical infrastructures for effective browsing, analysis, and research forecasting. Although knowledge graphs, which are extensive networks of entities and relationships, are useful tools, they do not explicitly represent the knowledge found in research papers. This work introduces a unique architecture that uses NLP and ML methods to extract entities and relationships from research publications and add them to a knowledge graph. A scientific knowledge network was created using the hybrid approach's accurate extraction of 109,105 triples from 26,827 abstracts. The method is universal and adaptable to any domain.

In 2011 Robert, Mirko and Mladen from university of Zagreb, Croatia has implemented a mind map generating software model by using text mining algorithm. [6] This software is compatible with desktop, laptop, PDA, and mobile devices. A web service based on SOA is advised because PDAs and mobile phones may execute slowly. The algorithms will be run by the web service, which will then produce a mind map that will be saved on a database server. All mind maps can be searched for and downloaded by users, and the database can be used to integrate mind maps and conduct additional research. This software has some kind of accuracy related to other systems.

III. METHODOLOGY

Our thorough step-by-step process offers a disciplined framework for producing in-depth mind maps using study materials. This strategy effortlessly incorporates many steps, each of which plays a crucial part in the process. We guarantee that the research material is ready for analysis by starting with thorough data pretreatment and cleaning and organizing it. This stage ensures that the succeeding processes have access to accurate and consistent data by removing duplication and dealing with different document formats.

The entity and relation extraction stage begins after data preparation. throughout this crucial stage. This makes it possible for us to efficiently detect and categorize things and their connections within the research material. Our Mind maps are constructed using these things, and the relationships between them show how intricately connected they are.

Our approach moves on to the extraction of relation triplets after successfully extracting entities and relations. The fundamental framework of our knowledge base is composed of these triplets, which include a subject entity, a predicate (the relation), and an object entity. They help students better understand complicated interdependencies by encapsulating the underlying relationships that underpin the concepts in the study material.

The next step is to build these things methodically now that we have knowledge-based objects at our disposal. This guarantees that the learned information is arranged logically and saved in a structured fashion. This stage not only increases the effectiveness of our mind map creation but also provides students looking for a deeper comprehension of the subject with an invaluable resource.
We use entity filtering and normalization to further improve the coherence and clarity of our mind maps. Issues with synonyms, abbreviations, and variations in entity names are resolved in this stage. We improve the final mind maps' readability and efficacy by standardizing entity representations.

In the end, our process results in the creation of comprehensive mind maps. We create visually perceptible representations of the structured data and knowledge-based items by utilizing graph theory and visualization approaches. The outcome is a dynamic mind map that accurately replicates the study material's underlying hierarchical structure and complex linkages. Entities become nodes; relationships become edges.

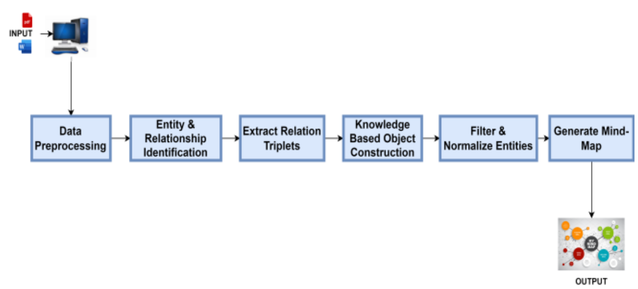

Figure 3: Methodology

A. Text Preprocessing (Step 1)

In this system, text preprocessing is a crucial first step that entails cleaning and converting unstructured text input into a more organized and manageable format. This procedure often entails steps like deleting superfluous letters, punctuation, and special symbols, changing the case of the text, and tokenizing the text to separate it into words or tokens. Text preprocessing delivers more accurate and useful results in many text-based applications. Text preprocessing helps enhance the quality of text data by making it more consistent and acceptable for later analysis.

B. Entity and Relationship Extraction (Step 2)

Identification The process of locating and classifying certain things or objects inside a text is known as entity extraction, often referred to as named entity recognition (NER). These entities might be actual things

This work is licensed under Creative Commons Attribution 4.0 International License.

like people, companies, places, or dates, or they could be more domain-specific terminology like diseases, substances, or financial instruments. Entity extraction's main objective is to discover and correctly classify these entities inside the text, frequently by labeling them with standard terms like "person," "organization," or "location," or with bespoke terms made specifically for the intended application.

Finding and extracting significant links or linkages between entities stated in a text is referred to as relationship extraction or relation extraction. Recognizing how things relate to one another and if they play certain roles or have connections that may be captured by predetermined relationships is necessary to achieve this. For instance, connection extraction in a news item would entail noting that "Apple Inc." (entity) purchased "Tesla" (entity) and designating their relationship as a "acquisition." To properly identify the nature of the relationship, relationship extraction often requires knowledge about the context of the entities and the surrounding language. Information retrieval, knowledge graph construction, and the creation of organized databases from unstructured text all benefit from its use.

In tasks involving natural language processing, named entity recognition (NER) and relationship classification (RC) are often used algorithms for entity and relationship extraction. The conventional pipeline, which employs NER before moving on to RC, might, however, generate mistakes that spread throughout the procedure. This sequential method can have its limitations, particularly when tackling connections in text data that are intricate or subtle. A preset set of relation types is another restriction placed on RC, which may not fully account for all possible links between entities.

Recent developments in natural language processing have proposed creative end-to-end techniques that try to handle both problems concurrently in order to solve the difficulties presented by the sequential application of Named Entity Recognition (NER) followed by Relation Classification (RC). Relation Extraction (RE) is the term used most frequently to describe this integrated process. In the framework of this post, we will go into the use of the amazing REBEL end-to-end model, which was created by BabelScape. Researchers and practitioners can accelerate the entity and connection extraction process, reducing the possibility of error propagation while increasing the ability to capture a greater variety of relation types, by implementing REBEL and other cutting-edge models.

In order to translate a phrase with entities and implicit relations into a sequence of triplets that explicitly refer to those relations, BabelScape trained the text-to-text model REBEL by fine-tuning BART. More than 200 distinct relation kinds were used to train it. Using entities and relations discovered in Wikipedia abstracts and Wikidata, the authors constructed a bespoke dataset for REBEL pre-training and filtered it using a RoBERTa Natural Language Inference model (similar to this model). To learn more about how the dataset was created, check the paper. On a variety of benchmarks for Relation Extraction and Relation Classification, the model performs well. And for this step I'm using this pre-trained model to extract entities and relations in an accurate way.

C. Extract Relation Triplets from the Processed Text (Step 3)

The creation of a custom function that can parse the structured strings produced by the REBEL model and convert them into relation triplets is a vital next step in the procedure. The subject, relation type, and object of each extracted piece of knowledge are all contained inside these triplets, which operate as the basic units of extracted knowledge. The function, which facilitates this translation, accounts for the addition of additional tokens during the model training phase, such as placeholders like "<triplet>," "<subj>," and "<obj> ". The borders and functions of entities and relationships inside the produced strings are clearly defined by these tokens.

Since each connection is represented as a dictionary, the function analyzes these strings to create a list of relations. These dictionaries have three fundamental words: "head" to indicate the subject (for example, "Fabio"), "type" to indicate the sort of relation (for example, "lives in"), and "tail" to indicate the object (for example, "Italy"). By using this function, we fill the gap between the model's output and a structured, useable representation of the knowledge that was extracted, making it more available and usable for later applications like knowledge graph generation and information retrieval.

D. Passed Triplets Into Knowledge-Based Object (Step 4)

The KB class is structured around a list of relations and features methods to add and print relations, as well as to determine if two relations are equal based on their attributes. The __init__ method initializes the KB object with an empty list of relations. The are_relations_equal method checks if two relations are equal by comparing their 'head,' 'type,' and 'tail' attributes, facilitating the identification of duplicate relations. The exists_relation method determines whether a given relation already exists within the KB, employing the are_relations_equal method. The add_relation method adds a new relation to the KB only if it doesn't already exist, ensuring relation uniqueness.

The KB class is structured around a list of relations and features methods to add and print relations, as well as to determine if two relations are equal based on their attributes. The __init__ method initializes the KB object with an empty list of relations. The are_relations_equal method checks if two relations are

This work is licensed under Creative Commons Attribution 4.0 International License.

equal by comparing their 'head,' 'type,' and 'tail' attributes, facilitating the identification of duplicate relations. The exists_relation method determines whether a given relation already exists within the KB, employing the are_relations_equal method. The add_relation method adds a new relation to the KB only if it doesn't already exist, ensuring relation uniqueness.

E. Filter and Normalize Entities (Step 5)

Our knowledge base creation approach includes a filtering stage as well, to improve accuracy and coherence. Entity linking, which includes comparing the retrieved entities with Wikipedia pages, is a successful filtering strategy. This stage involves checking to see if terms like "Cristiano Ronaldo" and "Cristiano" have a similar Wikipedia page. In the event that such a relationship is found, the entities are normalized to the page's title, combining them into a single representation. It's vital to note that this approach is predicated on the notion that Wikipedia consistently has accurate information on these entities as a result of user contributions. A more accurate and focused depiction of entities is made possible by temporarily excluding from the knowledge base any entities without associated Wikipedia entries.

A significant element of our knowledge base design is the method "are_relations_equal," which provides a way to determine whether two relations are equivalent based on the terms "head," "type," and "tail." This technique is crucial for knowledge base management because it enables us to establish whether two relations belong to the same information or to different facets of the same notion. We construct a related equality criteria by comparing these qualities, improving the accuracy and effectiveness of operations inside the knowledge base. When working with huge datasets or updating the knowledge base, this functionality is especially useful for ensuring that redundant or repeated relations are properly detected and maintained. Overall, "are_relations_equal" is a crucial technique that improves precision and coherence.

F. Visualize the Mind-Map

A crucial step in turning abstract data into a clear and visual representation is the creation of a graph visualization using the relations held in a knowledge base (KB). The Pyvis package, a potent tool for creating interactive network graphs, is used in this visualization stage. Using Pyvis, we can convert the intricate network of relations into an understandable and aesthetically pleasing graph structure. A visual representation of the underlying knowledge structure is created by turning each relation into a node and representing the connections between them as edges.

Once the graph has been created, it allows users to explore the links and interconnections within the knowledge base dynamically while also being visually instructive and engaging. The generated graph is stored as an HTML file for easy distribution and additional analysis. This ensures accessibility and shareability. This visual representation not only makes the knowledge base easier to grasp, but it also provides researchers, analysts, and learners with a thorough understanding of the links between the many entities in the data.

IV. RESULTS AND DISCUSSSION

The main goal of this painstakingly created test plan is to carefully assess and certify the quality and authenticity of the mind map produced by using the knowledge graph. In essence, it aims to make sure that the finished mind map serves as an accurate and thorough representation of the information included in the given text. This multidimensional review procedure is intended to make sure that the mind map properly captures every aspect, subtlety, and aspect of the document's content. By doing this, it not only demonstrates the durability of the knowledge graph extraction process but also highlights how well it can organize and present complicated textual information. In the end, this test strategy acts as a crucial quality control mechanism, ensuring that the mind map accurately replicates the key ideas and relationships in the text and equipping users with a potent tool for data exploration and understanding.

EXPERIMENT 1: Input a simple document with a single concept and verify that the generated mind map accurately reflects the concept.

1. Preparation: Choose a straightforward paper with a single, distinct topic, such as "The Earth's Rotation."

2. Execution: Input the document into the system for mind map generation.

3. Validation Steps:

- Check the mind map that was developed to make sure the core node is "The Earth's Rotation," a single notion.

- Make sure the mind map doesn't contain any more unconnected connections or thoughts.

4. Verification: The generated mind map precisely and completely captures the single notion from the paper.

EXPERIMENT 2: Input a document with multiple concepts and verify that the generated mind map accurately reflects all the concepts.

1. Preparation: Pick a paper containing a variety of clear themes, such "Renewable Energy Sources."

2. Execution: Input the document into the system for mind map generation.

3. Validation Steps:

- Check the mind map that was created to make sure it has all the different concepts, the right nodes, and the right connections.

- Verify that the document's important ideas are not omitted from the mind map.

This work is licensed under Creative Commons Attribution 4.0 International License.

4. Verification: All of the concepts from the input paper are represented precisely in the generated mind map.

EXPERIMENT 3: Input a complex document with multiple sub-concepts and verify that the generated mind map accurately reflects all the sub-concepts.

1. Preparation: Create a complicated text with several levels of supporting ideas, such as "Artificial Intelligence and Its Applications."

2. Execution: Send the complicated document to the mind map generating system.

3. Validation Steps:

- Make sure the created mind map properly captures all the sub-concepts and establishes a hierarchical structure by carefully going through it.

4. Verification: The complicated document's convoluted structure of concepts is successfully represented by the mind map that was generated.

EXPERIMENT 4: Input a document with no clear concepts and verify that the generated mind map does not produce any misleading information.

1. Preparation: Choose a piece of writing that lacks clear concepts or a clear framework, such a collection of phrases at random.

2. Execution: Provide the document to the system for mind map generation.

3. Validation Steps:

- Check the created mind map carefully to make sure it doesn't include any fictitious concepts or links in the absence of a defined content structure.

- Make sure there are no new details added to the mind map that are not necessary.

4. Verification: The created mind map suitably abstains from supplying misleading or unnecessary information.

V. CONCLUSION

In conclusion, this research project represents a sizable step forward in knowledge structuring and extraction from unstructured textual material. We have exhibited a strategic way for converting unstructured material into organized and aesthetically pleasing representations by methodically detailing a thorough process. We are able to efficiently extract significant connections and entities from textual content by combining state-of-the-art natural language processing techniques, such as named entity identification and relation extraction, with end-to-end models like REBEL.

To sum up, this research endeavor marks a significant advancement in the structuring and extraction of knowledge from unstructured textual data. By carefully outlining a detailed approach, we have shown how to use strategy to turn unstructured content into arranged and visually acceptable representations. By integrating cutting-edge NLP methods, such as named entity identification and relation extraction, with end-to-end models like REBEL, we are able to effectively extract meaningful relationships and entities from textual information.

The implementation of span-based text splitting also recognizes the difficulties presented by lengthy text documents and improves computational effectiveness, guaranteeing that no important information is missed. Collectively, these methodological developments provide a complete framework for developing extensive knowledge bases, laying the groundwork for knowledge graph development, information retrieval, and data-driven decision-making.

This research gives us the skills we need to harness the power of data in an era marked by the flood of textual information, allowing us to gain new insights, forge connections, and improve our comprehension of the world. These techniques will continue to be crucial in bridging the unstructured text and structured information divide as we move forward, encouraging innovation and insight across several areas.

REFERENCES

[1] T.Kim & N.Kim. (2021). Deep learning-based knowledge graph generation for covid-19. *Graduate School of Business IT, Kookmin University, Seoul, Korea*.

[2] Jose L. Martinez-Rodriguez, Ivan Lipez-Arevalo & Ana B. RiosAlvarado. (2018). OpenIE based approach for Knowledge Graph construction from text. *Expert Systems with Application*. DOI: 10.1016/j.eswa.2019.07.017.

[3] M. F. Nurrokhim, L. S. Riza & Rasim. (2019). Generating mind map from an article using machine learning. *J. Phys.: Conf. Ser., 1280*, 032023. DOI: 10.1088/1742-6596/1280/3/032023.

[4] A. Saelan & A. Purwarianti. (2013). Generating mind map from indonesian text using natural language processing tools. *Procedia Technology, 11*, 1163-1169.

[5] D. Dessì, F. Osborne, D. Reforgiato Recupero, D. Buscaldi & E. Motta. (2018). Generating knowledge graphs by employing natural language processing and machine learning techniques within the scholarly domain. *Knowledge-Based Systems, 154*, 1-14.

[6] R. Kudelić, M. Konecki & M. Maleković. (2011). Mind map generator software model with text mining algorithm. In: *Proceedings of the ITI 2011 33rd Int. Conf. on Information Technology Interfaces, Cavtat, Croatia.*

This work is licensed under Creative Commons Attribution 4.0 International License.