Masters Program in Geospatial Technologies

EO4GEO BOK ANNOTATION OF GI RESOURCES

Upeksha Indeewari Edirisooriya Kirihami Vidanelage

Dissertation submitted in partial fulfilment of the requirements for the Degree of *Master of Science in Geospatial Technologies*

NOVA IMS Information Management School
Universitat Jaume I
Westfälische Wilhelms-Universität Münster

EO4GEO BOK ANNOTATION OF GI RESOURCES

Dissertation Supervised by:
Sven Casteleyn, PhD
Associate Professor, Institute of New Imaging Technologies (INIT),
Universitat Jaume I (UJI),
Castellon de la Plana, Spain

Dissertation Co-supervised by:
Carlos Granell, PhD
Associate Professor, Institute of New Imaging Technologies (INIT),
Universitat Jaume I (UJI),
Castellon de la Plana, Spain

Dissertation Co-supervised by:
Marco Painho, PhD
NOVA IMS, Universidade Nova de Lisboa
Lisbon, Portugal

Dissertation Co-supervised by (External):
Rob Lemmens, PhD
Assistant Professor, Faculty of Geo-Information Science and Earth Observation,
University of Twente,
Enschede, Netherlands

February 20, 2024

DECLARATION OF ORIGINALITY

It is my declaration that I, Upeksha Indeewari Edirisooriya Kirihami Vidanalage, a master's student in the Geospatial Technology Erasmus Mundus program, have written thesis titled "EO4GEO BOK ANNOTATION OF GI RESOURCES" entirely on my own. As an author, I take full responsibility for its content and authenticity.

In my study, I properly credited all sources that I used, including papers, books, journals, and online resources. In order to avoid plagiarism, I understand the need of citing direct quotations and properly attributing paraphrased or condensed content.

I recognize that presenting someone else's work as my own or presenting their ideas without acknowledgment makes academic dishonesty. Having a thorough awareness of plagiarism, I am aware that any instances of it in my thesis will result in rejection.

Signed:

……..

Upeksha Indeewari Edirisooriya Kirihami Vidanalage
Castellon de La Plana,
February 20, 2024

ACKNOWLEDGEMENTS

First, I would like to express my gratitude to the Erasmus Mundus Program for granting me a scholarship to pursue a master's in Geospatial Technology. I am sincerely thankful ad grateful to my principal supervisor, Dr. Sven Casteleyn in University of Jaume I, Spain, whose encouragement, support, and feedback motivated me to work hard and achieve excellence. His strong guidance and support incredibly inspire me.

I want to express my deep appreciation to my co-supervisors, Dr. Rob Lemmens and Dr. Carlos Granell, and Dr. Marco Painho for their guidance, support, and suggestions to develop my research. I truly appreciate the expertise they have offered throughout my research journey.

I am equally thankful to the professors from UJI, Ifgi, and Nova IMS for sharing their diverse knowledge and skills, guiding my professional development, and promoting a multidisciplinary approach to learning.

I would also like to thank the administrative staff at UJI and Ifgi for their continuous assistance, guidance, and support during my master's program.

Lastly, I'm thankful to my loving parents, husband, and friends for their encouragement, endless love, and constant support to succeed in my academic journey.

Additionally, I would like to acknowledge to Sri Lankan free education system and the European Union for their contributions to my education and growth.

EO4GEO BOK ANNOTATION OF GI RESOURCES

ABSTRACT

The Earth Observation for Geospatial Information (EO4GEO) Body of Knowledge (BoK) serves as a foundational framework encompassing essential geospatial concepts necessary for leveraging Geographic Information and Earth Observation data effectively. Based on the BoK a set of tools was developed at the University Jaume I, including the BoK Annotation Tool (BAT), which facilitates the annotation of any PDF document with EO4GEO BoK concepts, streamlining the process of knowledge association. These annotations are added manually through an easy-to-use "what you see is what you get" (WYSIWYG), which presents significant challenges, including time consumption and the need for domain expertise. To address this challenge, this master thesis studies the use of Natural Language Processing (NLP) techniques to automate the BoK annotation process. Concretely twelve NLP-based tools were applied, utilizing three key phrase extraction algorithms (YAKE, PatternRank, KeyBert) and four semantic similarity measures (Cosine, Jaro-Wrinkler, Latent semantic similarity, Word2Vec), in order to (semi)-automatically generate BoK annotation. To assess the performance, a comparative evaluation was carried out using various annotation approaches (i.e. using top-level concepts, leaf concepts and all concepts) and evaluation methods (i.e. direct matching, parent-child matching, ranking). Results revealed that YAKE_JaroW emerges as a standout performer (F1-score 28.28%), particularly in the parent-child evaluation method. This research helps to annotate existing resources with EO4GEO BoK concepts easier and more efficiently, helping to share knowledge more effectively in geospatial fields. It also emphasizes the importance of having annotation tools designed specifically for the EO4GEO BoK, which fills a crucial gap in geospatial knowledge management.

Keywords

EO4GEO, Body of Knowledge, Annotation, NLP, Similarity measure, Key phrase extraction

INDEX OF TEXT

DECLARATION OF ORIGINALITY.....i
ACKNOWLEDGMENT.....ii
ABSTRACT.....ii
INDEX OF TEXT.....iv
INDEX OF FIGURES.....vii
INDEX OF TABLES.....viii
ACRONYMS.....ix
1 INTRODUCTION.....1
1.1 Background and Motivation.....1
1.2 Research Objectives and Question.....2
1.3 Research Methodology and Methods.....4
1.4 Thesis Structure.....4
2 BACKGROUND AND RELATED WORKS.....5
2.1 Research in the GIS-related Body of Knowledge (BoK).....5
2.2 EO4GEO Body of Knowledge (BOK).....6
2.2.1 BoK Visualization and Search.....6
2.2.2 Living Textbook (LTB).....8
2.3 BoK Annotation Tool (BAT).....9
2.4 Algorithms and Technologies.....11
2.4.1 Annotation in NLP.....11
2.4.2 Key Phrase Extraction in NLP.....11
2.4.3 Text Similarity in NLP.....13
2.5 Research Related to Key phrase extraction and Similarity Measures in NLP.....14
3 METHODOLOGY AND EXPERIMENTAL DESIGN.....17
3.1 Exploratory Literature Review.....17
3.2 Data Acquisition and Pre-processing.....18
3.2.1 Data Collection.....18
3.2.2 Data Preprocessing.....18
3.3 Design and Development of NLP-based Tools.....22
3.3.1 Architecture of the Proposed NLP-Based Tool.....22
3.3.2 Development of Web Application.....26

4 EXPERIMENTAL EVALUATION 31
4.1 Annotation 31
4.2 Evaluation 32
4.2.1 Evaluation Parameters 33
4.2.2 Evaluation Approaches 34
5 RESULTS AND DISCUSSION 37
5.1 Results for Evaluation Parameters 37
5.1.1 Direct Matching Evaluation Method 38
5.1.2 Parent-child Matching Evaluation Method 40
5.1.3 Ranking-based Evaluation Method 41
5.1.4 Overall Performance 44
5.2 Results for Matching Percentages 45
6 CONCLUSION 51
6.1 Conclusion 51
6.2 Limitations 53
6.3 Future Works 53
7 Bibliography 55

INDEX OF FIGURES

Figure 1: EO4GEO BoK in the BoK Visualization and Search (EO4GEO Alliance, 2022) 7
Figure 2:The LTB tool, showing GI concepts imported from the PP1-6 BoK (University of Twente. 2021.).....8
Figure 3:BAT in EO4GEO (EO4GEO Alliance, 2022) 10
Figure 4: Browsing BoK in BAT in EO4GEO (EO4GEO Alliance, 2022) 10
Figure 5: Annotated PDFs in BAT in EO4GEO (EO4GEO Alliance, 2022).....10
Figure 5: Annotated PDF (EO4GEO Alliance, 2022).....10
Figure 6: Functional details of the proposed methodology 17
Figure 7: Workflow for the Data Pre-Processing.....19
Figure 8: Word Count Comparison Before and After Text Cleaning for (Mocnik, 2023) .22
Figure 9: The pipeline of the proposed NLP tool.....23
Figure 10: The output of the proposed NLP-based tool as a. JSON format.....26
Figure 11: The workflow of development of web application using Flask framework. ...28
Figure 12: 'Home' page in proposed web application 29
Figure 13: 'Instructions' page in proposed web application 29
Figure 14: Output of the proposed web application.....30
Figure 15: Output of the proposed web application.....30
Figure 16: Entire evaluation workflow.....31
Figure 17: P (%), R (%) and F (%) values for NLP based tool employed for direct matching evaluation method for (a) FULL (b) LEAF and (c) TOP approaches 39
Figure 18: The overall performance for NLP based tool employed for direct evaluation method.....39
Figure 19: P (%), R (%) and F (%) values for NLP based tool employed for parent- child matching evaluation method for (a) FULL (b) LEAF and (c) TOP approaches. 40
Figure 20: The overall performance for NLP based tool employed for parent-child matching evaluation method 41
Figure 21: P (%), R (%) and F (%) values for NLP based tool employed for ranking-based evaluation method for (a) FULL (b) LEAF and (c) TOP approaches. 42
Figure 22: The overall performance for NLP based tool employed for ranking evaluation method.....43
Figure 23: The overall F1-score variation for proposed NLP based tool employed for each evaluation method.....44

Figure 24: Number of NLP-based tools for each PDF document which having matching percentage more than 50% (a)Direct Matching (b) Ranking-based (c) Parent-child matching 49

INDEX OF TABLES

Table 1: Proposed NLP-based annotation tools.....26
Table 2: Performance of the NLP based tool for annotation with EO4GEO BoK concepts in terms of annotation approaches and evaluation approaches 38
Table 3: Calculated matching percentages (%) for each PDF documents for each evaluation and annotation stages.....47
Table 4: Annotation results given for P01 by YAKE_Word2Vec tool and BAT tool for all annotation approaches.....48
Table 5: Number of NLP-based tools which have more than 50% matching for each PDF document.....49

ACRONYMS

Body of Knowledge (BoK)

Geographic Information Systems (GIS)

Earth Observation for Geospatial Information (EO4GEO)

Earth Observation (EO)

Geo- Information (GI)

BoK Annotation tool (BAT)

BoK matching tool (BMT).

Natural Language Processing (NLP)

University Consortium of Geographic Information Science (UCGIS)

Geographic Information Science and Technologies (GIS&T)

Living Textbook (LTB)

YAKE (Yet Another Keyword Extractor)

Latent Semantic Analysis (LSA),

Natural Language Toolkit (NLTK)

CHAPTER 01

1 INTRODUCTION

1.1 Background and Motivation

A Body of Knowledge (BoK) simply represents an organized map of information in a particular topic. It includes important concepts, explanations, and related activities or skills. People in that field could utilize it as a reference for learning and applying knowledge in their studies and careers (Dibiase et al., 2006). It is shown as a mind map or node diagram. In this map, concepts are linked together by relationships, resulting in a structured network known as an ontology. Each concept, be it a theory, method, or technology, is systematically linked to a complete knowledge description. This formal structure contributes to creating an organized comprehension of the topic (Dubois et al., 2021).

BoK in Geographic Information Systems (GIS) are considered as a reference guide in the field. It provides a formal and regularly updated compilation of fundamental knowledge, skills, and competences required in a particular field. It may serve as a reference vocabulary, and as such, can serve various purposes. For example, it may be used as a foundational guide for developing curricula in various professional programs and vocational training. This includes designing accreditations at different qualification levels. The BoK may also support the recruitment process and aid in assessing job requirements by defining the essential knowledge and competencies needed. Particularly valuable for career planning, the BoK is also beneficial for individuals looking to enter, transition, or enhance qualifications within a specific field (Stelmaszczuk-Górska et al., 2020).

The Earth Observation for Geospatial Information (EO4GEO) project is an Erasmus+ Sector Skills Alliance initiative to address the gap between the availability and demand of education and training in the Earth Observation (EO) and Geo- Information (GI) sectors. The project achieves this goal by developing a BoK along with various tools that facilitate the creation and discovery of online resources, enabling stakeholders to develop and find curricula, occupational profiles, job opportunities, and learning materials (Lemmens et al., 2022). The EO4GEO BoK encompasses concepts related to EO, thematic and application domains including the BoK search and visualization tool,

the BoK Annotation tool (BAT) (EO4GEO Alliance, 2022), and the BoK matching tool (BMT). The BAT enables the annotation or association of semantic knowledge, specifically about the concepts within the BoK, to any resource in the form of a PDF file (Monfort et al., 2020). Once a resource is annotated, it becomes possible to compare it with other BoK-annotated resources using the BMT (EO4GEO Alliance, 2018).

Annotation with the BOK ensures that annotated resources are appropriately classified and matched with the domain's key principles. Since BoK concepts provide a standardized vocabulary and framework for domain annotation, by adhering to a shared set of concepts, annotations become more consistent and interoperable across diverse resources, platforms, and applications. Annotation with BoK concepts not only categorizes resources, but it also aids in skill development by tying annotated information to specific competencies and learning objectives stated in the BoK. This alignment helps to establish targeted training programs, curriculum creation, and competency assessments within the domain.

However, there is a lack of automated or semi-automated methods for annotating resources with EO4GEO BoK concepts. The manual annotation process offered by the BAT is time- consuming, requires domain expertise, and limits the scalability and usability of the BAT (Neves & Ševa, 2021). Furthermore, resource-intensive, and subject to human error. Therefore, suitable tools are needed to automatically extract EO/GI knowledge from textual resources and annotate them with relevant BoK concepts to enhance the efficiency and accuracy of annotating resources.

1.2 Research Objectives and Question

The primary focus of this project is to address the existing knowledge gap in automated methods for annotating resources with concepts derived from the EO4GEO BoK, which hinders the exploration and utilization of a vast amount of knowledge contained within unannotated resources. To this aim, the overall goal of this is to (semi)-automate the process of annotating existing text-based resources with BoK concepts through the use of Natural Language Processing (NLP) techniques. By automating the manual annotation process, the project aims to improve efficiency while enabling the discovery of valuable knowledge that may have remained hidden or uncovered within the text (Rehbein, 2012). It furthermore aims to enhance accessibility by empowering non-

experts to participate in the annotation process and bridges the gap between research knowledge and practical applications.

Concretely, this research addresses the following research question:

- Which NLP-based tools are suitable for extracting key EO/GI knowledge from text?
- How can the identified NLP-based tools be used to associate EO/GI knowledge, in terms of EO4GEO BoK concepts, with text documents?
- How do the identified NLP-based tools perform in extracting and associating EO/GI knowledge with text documents?
- How can the identified NLP-based tools be made available to the community of EO/GI researchers and practitioners?

This project has the potential to revolutionize the utilization of the EO4GEO BoK, facilitating its integration into software applications and driving advancements in the EO/GI field. By developing an NLP-based tool, significantly reducing the manual effort required to annotate resources with EO4GEO BoK concept may be significantly reduced, reducing alleviating the burden on experts and enabling a wider range of users to participate in the annotation process.

This study's target audience includes individuals and organizations working within the EO and GI field who want to exploit the utilize the concept of EO4GEO BoK and the associated software tools. This includes educators, researchers, practitioners, industry professionals, and policymakers who rely on the BoK for accessing and applying knowledge in their respective domains (Dibiase et al., 2006). This will mainly benefit those who involved in resource annotation processes, including experts responsible for associating BoK concepts with resources, as well as non-experts who can now participate in the annotation process with the assistance of the proposed NLP tools. Additionally, software developers and application designers who integrate the BoK into their software applications will also benefit from the improved automation and accessibility of resource annotation.

1.3 Research Methodology and Methods

To achieve this research goal and answer the research questions, this thesis will explore investigate cutting-edge NLP techniques to text analysis, extraction of t key knowledge and automatically associate EO/GI-related knowledge with relevant EO4GEO BoK concepts. Concretely, a multi-phase research methodology is used, combining scientific methods, software development and experimental validation. The methodology constitutes the following steps, in which mention the applied research method(s) in each step:

1. Conducting an exploratory literature review to gain insights into NLP-based methods for annotating documents.
2. Identifying the required functionalities of proposed NLP-based tools to annotation with EO4GEO BoK concepts.
3. Selecting appropriate NLP-based methods that match with the functionalities of the proposed NLP-based tools.
4. Design and develop NLP-based tools to automate the annotation process with EO4GEO BoK concepts.
5. Introducing proposed NLP-based tools as a web application for user-friendly accessibility.
6. Evaluating the performance of proposed NLP-based tools and assessing the results to ensure the effectiveness of the annotation process.

1.4 Thesis Structure

The structure of this thesis is the following:

- Chapter One comprises of contextual background and motivation, research objectives and questions, and a brief methodology and methods.
- Chapter Two focuses on the related work in BOK in the GIS field, the EO4GEO project, a short description of the NLP-based tools and technologies used in the project, as well as related studies for annotation using NLP.
- Chapter Three discusses detailed experimental design and development.
- Chapter Four provides a description of the experiment evaluation procedure.
- Chapter Five provides results and discussion from the evaluation.
- Chapter Six includes conclusion of work, limitations, and future works.

CHAPTER 02

2 BACKGROUND AND RELATED WORKS

2.1 Research in the GIS-related Body of Knowledge (BoK)

Research in GIS related BoK encompasses ongoing and finished attempts to define and organize the important knowledge and skills within the geospatial domain.

The University Consortium of Geographic Information Science (UCGIS) launched the first version of Geographic Information Science and Technologies (GIS&T) BoK in 2006, with contribution from GIS researchers and support from Esri. Its aim was to outline what both current and future geospatial professionals should understand and apply in terms of geospatial methods, theories, and tools. The initiative aimed to set a standard for the essential knowledge and skills in the field at that time (DiBiase et al., 2007). Starting in 2013 discussions at the 2014 UCGIS Symposium led to the initiation of a new project: the second edition of GIS&T BoK. This time, the focus shifted to creating a web-based version for easy access and use and the project emerged from various activities, reflecting a commitment to a more modern and user-friendly environment (Waters, 2013; Wilson, 2016).

The United States Geospatial Intelligence Foundation looks at different jobs across industries to figure out what's important for the Geospatial Intelligence (GEOINT) workforce. They summarize this knowledge, skills, and abilities through the Geospatial Intelligence Essential Body of Knowledge (EBK) (DiBiase et al., 2007). In South Africa, a university course framework was created by tailoring it to align with both local and global standards, drawing inspiration from the GIS&T BoK. This customization ensures that the course meets the specific needs and standards, both within the country and internationally (du Plessis & van Niekerk, 2013). In Europe, EO4GEO BoK is being developed (Hofer et al., 2020). This BoK is redesigned and expanded based on the framework of GIS&T BoK, with a specific emphasis on enhancing skills. Its primary goal is to address the gap between the educational and training offerings in earth observation and geospatial sciences, effectively bridging the gap between what is available and what is required. These projects demonstrate a strong commitment and enthusiasm among different organizations, countries, organizations, and experts for the development of a GIS-related BoK.

2.2 EO4GEO Body of Knowledge (BOK)

The Earth Observation and Geographic Information/Geoinformation sector (EO*GI) is experiencing rapid growth, driven by technological advancements in collecting and analyzing large EO data. This shift in technology builds for a new approach to learning and knowledge transfer, especially in designing future EO*GI curricula and training programs. The project *'Towards an innovative strategy for skills development and capacity building in the space geoinformation sector supporting Copernicus user uptake'* known as EO4GEO, is an Erasmus+ Sector Skills Alliance that aims to address the existing and future challenges in (EO*GI) education (Dubois et al., 2021).

The EO4GEO BoK, developed as part of the EO4GEO project[1], is built upon a revision of the GIS&T BoK (Vandenbroucke & Vancauwenberghe, 2016). The EO4GEO BoK was developed with participation from a network of over 150 domain specialists. Currently, the European GIS&T BoK is undergoing expansion by incorporating EO concepts and their relationships, along with skills indicating the practical application of these concepts. This revision process utilizes conventional techniques such as expert interviews, discussion sessions, and workshops, organized into working groups that focus on specific knowledge areas within the field (Hofer et al., 2020).

The EO4GEO BoK offers two distinct exploration modes.

1. BoK Visualization and Search (hierarchical view)
2. Living Textbook[2] (LTB) environment (concept map view)

2.2.1 BoK Visualization and Search

In BoK Visualization and Search mode (Figure 1), BoK concepts serve as the foundational building blocks for representing the EO*GI knowledge domain. These concepts are created and maintained across seven working groups, covering 14 top-level domains. Each concept in the BoK is enriched with a set of elements, which include (Lemmens et al., 2022):

- **Code:** unique identifier assigned by the leader of the respective working group such as [AM6] or [PP2-3-1].
- **Name:** A concise identification of the concept in 1-5 words.

[1] http://www.eo4geo.eu/

[2] http://www.eo4geo.eu/tools/living-textbook/

- **Description:** This element consists of plain narrative text, serving as an abstract about the concept. The description is crafted to be self-contained and accessible to a broader audience, with a maximum limit of 500 words (excluding pictures or tables). This provides a comprehensive understanding of the concept's significance within the EO*GI knowledge domain.
- **Skills:** Each concept is associated with practical skills, offering detailed insights into the real-world application of EO*GI knowledge.
- **References:** Authoritative sources used for concept descriptions within the EO4GEO BoK.
- **Relationships:** The relationships between concepts in the EO4GEO BoK are connected through outgoing and incoming relations, following a Resource Description Framework (RDF)-triple style (Ronzhin et al., 2018). This style involves the current concept (subject) connected to another concept (object) through a defined relationship (predicate).
    - ***Sub-concept***: This relationship signifies that a concept operates on a lower granularity level. A single concept can serve as a sub-concept to more than one other concept.
    - ***Prerequisite:*** This relationship denotes that a concept must be understood or known to comprehend another concept.
    - ***Similarity:*** This relationship highlights a similarity between concepts.
- **Concept status:** Concept status serves as an indicator of its position within the ongoing revision or definition process such as "deprecated," "new," "in progress," "completed," and others.

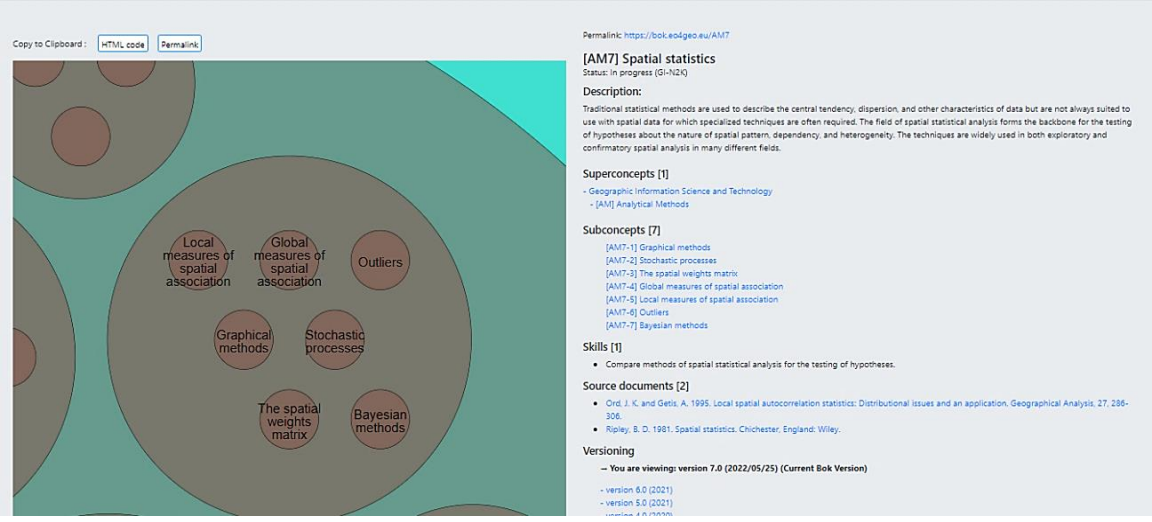


Figure 1: EO4GEO BoK in the BoK Visualization and Search (EO4GEO Alliance, 2022)

2.2.2 Living Textbook (LTB)

The development of the concepts, relationships, and skills forming the ontology-based EO4GEO BoK relies on the use of the Living Textbook environment (Figure 2) (Ronzhin et al., 2018). LTB is a web tool crafted by the University of Twente, designed to model, and visualize domain knowledge for educational and knowledge-sharing purposes. Its interface integrates a wiki-style text window with a concept map, offering a collaborative space for domain experts and teachers to act as content developers. Within this tool, they can collectively generate detailed descriptions of concepts and establish connections through self-defined relationships. The LTB provides a dynamic and interactive environment, fostering collaborative knowledge creation and exchange within the EO4GEO project (Hofer et al., 2020).

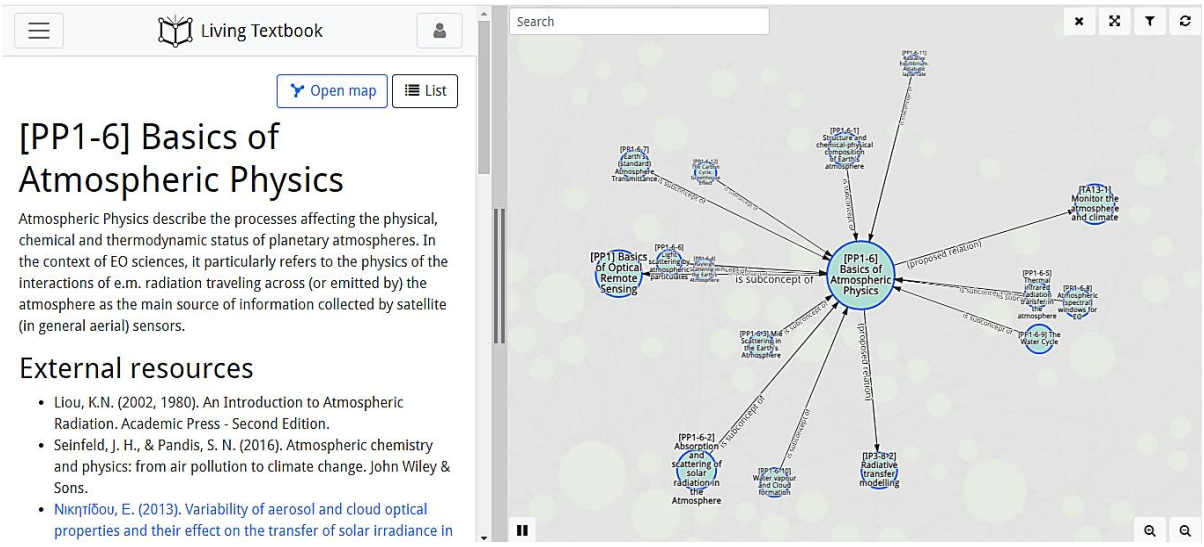


Figure 2:The LTB tool, showing GI concepts imported from the PP1-6 BoK (University of Twente. 2021.)

This platform enables users to easily edit, create, and explore concepts within the BoK. Using a concept map as a foundational element, the LTB provides a visual representation of concepts and their interconnections, providing an efficient means to identify linked concepts along with their detailed descriptions. The concept map serves as a user-friendly interface, that enables contributors to develop and maintain a comprehensive overview of the BoK content. In essence, the LTB not only facilitates the creation and editing of the BoK but also offers a visual representation that aids in understanding the relationships among various concepts within the EO4GEO BoK (EO4GEO Alliance, 2022).

2.3 BoK Annotation Tool (BAT)

The BoK Annotation Tool (BAT) is a product resulting from the EO4GEO project, primarily using the concepts from the EO4GEO BoK. BAT facilitates the annotation (or association) of any PDF document with relevant EO4GEO BoK concepts. These annotations are intended for later use within the BoK Matching Tool (BMT) to identify the best matches. BAT operates by automatically editing the metadata of PDF files. This automated process makes easy the task of associating PDF documents with BoK concepts, enhancing the efficiency and effectiveness of utilizing EO4GEO resources. The following describes the process of annotation using BAT.

1. **Login & registration:** There are two types of users known as anonymous and registered, who can utilize BAT. Anonymous users can directly annotate PDFs without logging in, but to save annotated documents for future access, registration is required.
2. **Annotate PDF:** BAT offers two distinct workflows for users. If the user only needs to annotate a PDF and download it into personal folders, logging in is unnecessary. The system allows for direct PDF uploading by clicking the 'Browse' button (1) and selecting a PDF file from the user's computer. To save an annotated PDF in BAT for later use, log in and provide basic information about the document (2,3) is needed. After uploading the PDF (1) (Figure 3), then annotations should be added using the BoK Visualization and Search component (4). Users can browse the BoK graphically or textually, and search for specific concepts (5,6,7) (Figure 4) and can add desired concepts to the 'Knowledge annotated' list (7) (Figure 4). Annotations are stored in the metadata for interpretation by BMT.
3. **Search previously annotated documents:** A user can check her/his previously annotated PDFs saved in 'My annotated PDFs' accessible by the top menu (Figure 5). After annotation PDF can be saved in BAT and if user needs it can download or save it in the computer.

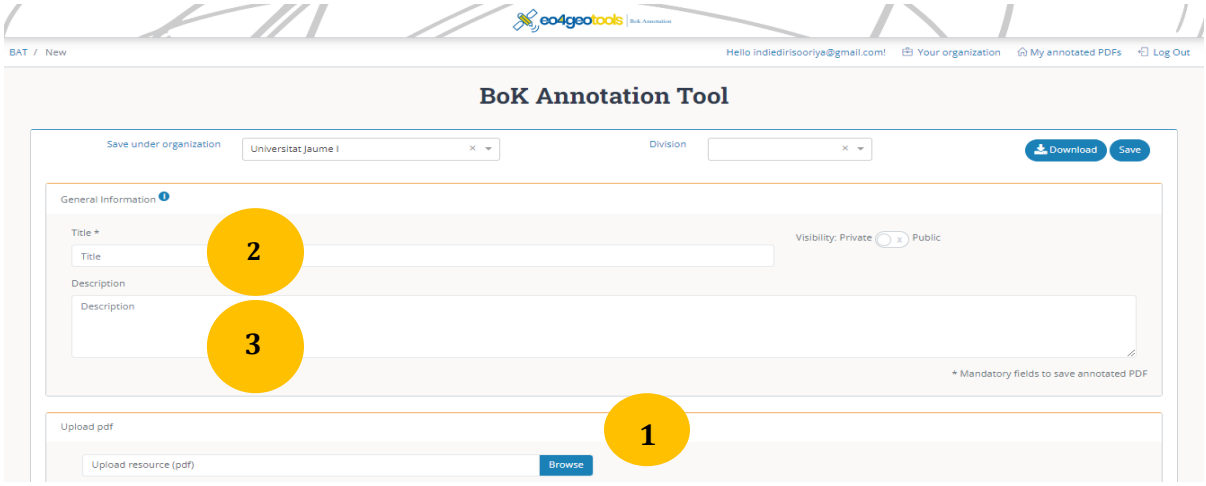
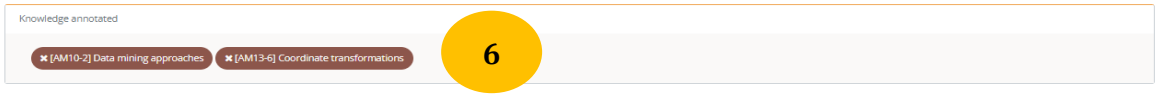


Figure 3:BAT in EO4GEO (EO4GEO Alliance, 2022)



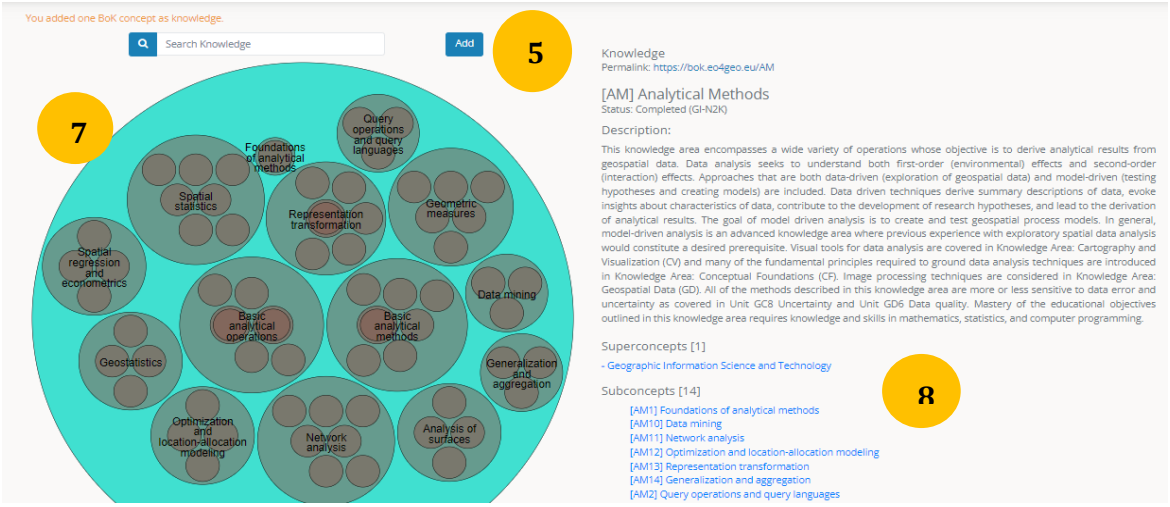


Figure 4: Browsing BoK in BAT in EO4GEO (EO4GEO Alliance, 2022)

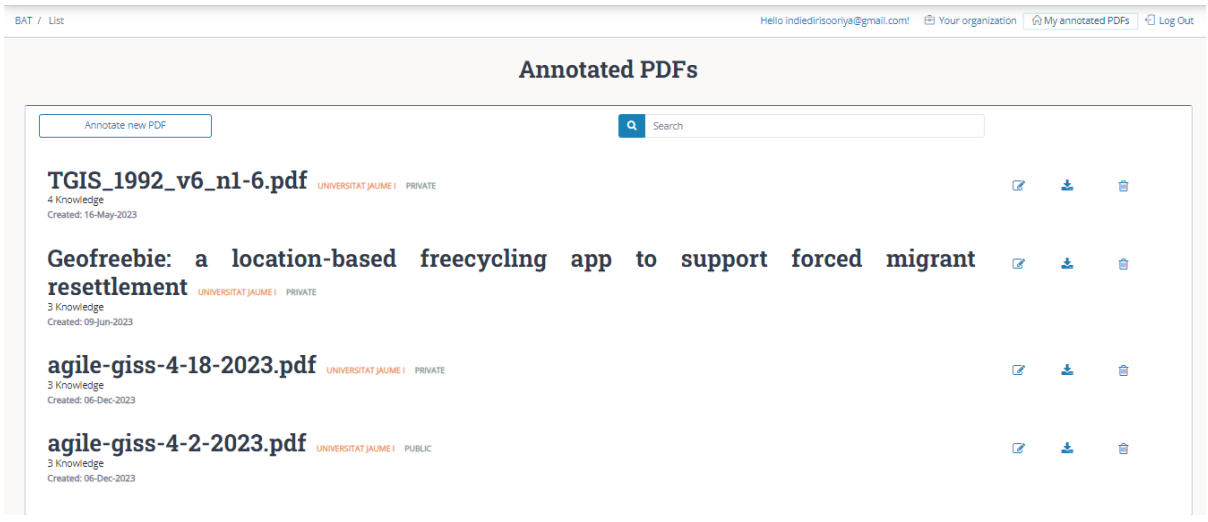


Figure 5: Annotated PDFs in BAT in EO4GEO (EO4GEO Alliance, 2022)

2.4 Algorithms and Technologies

This chapter provides details about the algorithms and technologies used to design and develop proposed NLP-based tools.

2.4.1 Annotation in NLP

Annotation involves the process of reading a specific preselected document and providing it with additional information. These annotations can be applied at various linguistic levels such as words, sentences, paragraphs, phrases, or even individual characters. Document annotations are particularly valuable for tasks related to document classification, providing additional context that contributes to the understanding and categorization of the document (Neves, 2011). In NLP, annotations can be broadly classified into key categories. Sentiment annotation involves labeling text data with sentiments, such as positive, negative, or neutral expressions (Panchal et al., 2022). Entity Annotation focuses on identifying and labeling entities such as people, locations, and organizations within text (Ikhwan Syafiq et al., 2019). Text Classification is concerned with assigning predefined categories to text documents based on their content (Li et al., 2022). Entity Linking is connecting identified entities in the text to larger knowledge repositories or databases, providing additional information about these entities (Tedeschi et al., 2021). Lastly, Linguistic Annotation is a process that involves tagging language data within text or audio recordings (Ide, 2017).

Text classification, also known as text tagging or categorization, involves assigning pre-defined labels to text. It organizes textual data into specific groups by assigning labels or classes, contributing to the automation of numerous processes such as survey analysis, and document summarization etc. One of the key advantages of employing text classification through NLP is its scalability and accuracy in extracting specific information from large volumes of textual data. The classification can be achieved through either manual annotation or automatic labeling (Li et al., 2022; Minaee et al., 2020).

2.4.2 Key Phrase Extraction in NLP

The main task of key phrase extraction is to find a single word or phrase that summarizes the main content of the text paragraph. Instead of using the entire text, a document can be represented by its key phrases, providing a more concise input to the

document classification tasks (Sun et al., 2020). Key phrase extraction in NLP is categorized into supervised methods and unsupervised methods. In unsupervised approaches, the task of key phrase extraction is treated as a ranking problem, and it is performed without the need for prior knowledge or labeled data. Unsupervised key phrase extraction methods fall into statistical-based, graph-based and embedding-based approaches. One significant advantage of unsupervised methods, as opposed to supervised approaches, is that they do not rely on manually labeled datasets (Alami Merrouni et al., 2020; Hu et al., 2018).

2.4.2.1 Unsupervised Key Phrase Extraction

Statistical-based Approach: YAKE

YAKE is a tool designed for extracting significant keywords from unstructured documents across a range of document lengths. It focuses on local text features and statistical information like term frequencies and co-occurrences. Through text preprocessing and the calculation of features such as casing, word positional importance, word frequency, word relatedness to context, and word differential sentence occurrence, YAKE assigns a score to each term. This scoring system efficiently extracts keywords, with indicating greater importance and relevance to the document's content and key topics (Amur et al., 2023; Papagiannopoulou & Tsoumakas, 2020; R. Wang et al., 2014).

Graph-based Approach: PatternRank

PatternRank is one of the graph-based approaches, that relies on Pretrained Language Models (PLMs) and Part of Speech (PoS) tagging. The process begins with a single text document as input, which is then tokenized into individual words. Part-of-speech tags are assigned to each word token, and a predefined pattern is applied to select specific tokens as candidate key phrases. These candidates undergo semantic embeddings using a pre-trained language model to capture their meaning and context. Cosine similarities between the vector representations of the document and candidates are calculated to measure semantic similarity. Then candidate key phrases are ranked based on these scores (Tsvetkov & Kipnis, 2023; Schopf et al., 2022).

Embedding- based Approach: KeyBERT

KeyBERT uses Bidirectional Encoder Representations from Transformers (BERT) embeddings to extract keywords from text documents. BERT, a transformer-based model, provides contextualized word embeddings, capturing semantic relationships based on the entire left and right context of each word. It is pre-trained in extensive text corpora, including Wikipedia, and it has a broad understanding of language. The KeyBERT Python library utilizes pre-trained BERT models to extract contextual attributes of words in sentences, allowing for tasks like text classification and clustering. It calculates embeddings for each word and the entire document, sorts them in descending order, and selects top keywords based on their similarity to the document embeddings (Khan et al., 2022).

2.4.3 Text Similarity in NLP

Text similarity measures in NLP are used to assess the similarity between words, sentences, paragraphs, or documents. Text similarity can be categorized into two main types: lexical similarity and semantic similarity. Lexical similarity measures focus on assessing the similarity between words based on their character sequences. Semantic similarity measures, measures the likeness among words or documents based on their contextual meaning and relationships. Understanding the contextual meaning involves grasping the significance of individual words or phrases within the broader context of the document. Analyzing these relationships involves identifying connections such as synonyms (words with similar meanings), antonyms (words with opposite meanings), and associations (words that frequently occur together). This combined approach allows for a comprehensive understanding of the semantic structure of the document, facilitating the interpretation of its underlying meaning and message. (H.Gomaa & A. Fahmy, 2013 ; Hameed et al., 2022).

2.4.3.1 Lexical Similarity

Term-Based Similarity Measures: Cosine similarity

Cosine similarity is a widely used method for measuring the similarity between text documents which are represented as vectors. In order to determine their similarity, it calculates the cosine value between the term vectors of two documents. This approach is applicable to various text units, such as sentences, paragraphs, or entire documents.

The cosine similarity primarily evaluates the orientation or angle between vectors, making it well-suited for comparing text documents. The similarity values range from -1 to 1, with values close to 1 indicating similarity (parallel vectors) and values near 0 signifying dissimilarity (perpendicular vectors) (Rahutomo et al., 2012; Lehal, 2017).

Character-Based Similarity Measures: Jaro-Winkler Similarity

The Jaro-Winkler similarity is a semantic similarity measure for comparing two strings. It is an extension of the Jaro similarity but incorporates additional modifications to give higher weights to strings that have a common prefix. This modification makes it particularly suitable for cases where strings are expected to have a common prefix, such as names (H.Gomaa & A. Fahmy, 2013).Semantic Similarity

Corpus Based Similarity: Latent Semantic Analysis (LSA)

Latent Semantic Analysis (LSA) is a popular, widely used technique in corpus-based similarity analysis. It operates on the assumption that words with similar meanings tend to appear in similar contexts within a text. The process involves constructing a matrix that represents word occurrences per paragraph, where rows represent unique words, and columns represent individual paragraphs or documents. In LSA, word similarity is measured by calculating the cosine of the angle between the vectors formed by any two rows in the matrix. This cosine similarity reflects the degree of similarity between the meanings of the corresponding words (H.Gomaa & A. Fahmy, 2013)

Knowledge Based Similarity: Word2Vec

Word2Vec, is a NLP-based technique that represents words as N-dimensional vectors, or word embeddings. The technique measures the similarity between words and creates a vector space where similar words are located close to each other. These vectors capture the semantic meaning of words and are learned from a large corpus of text. (Sitikhu et al., 2019 ; Wang & Dong, 2020).

2.5 Research Related to Key phrase extraction and Similarity Measures in NLP

In the realm of automatic unsupervised key phrase extraction methods and similarity measures, various studies have contributed to advancing the field and addressing key challenges. Several methods have been developed, in the domain of automatic

unsupervised key phrase extraction, each offering unique insights and approaches. YAKE[3] (Yet Another Keyword Extractor) is one such method that focuses on extracting key phrases by considering their statistical significance within the text. YAKE leverages both statistical and linguistic features to identify key terms that are highly relevant to the document's content (Campos et al., 2020). Study shows that YAKE stands out for its ability to achieve the highest F1-score values in key phrase extraction (Piskorski et al., 2021). PatternRank[4], utilizing pretrained language models and part-of-speech information, excels in unsupervised key phrase extraction from single documents. Experimental results indicate that Pattern Rank outperforms previous state-of-the-art approaches in terms of precision, recall, and F1-scores (Schopf et al., 2022). Another notable method is KeyBERT[5], proposed by Grootendorst (2020), which utilizes pretrained BERT-based embeddings for keyword extraction. Studies found that the KeyBERT model outperformed traditional approaches in producing similar keywords to the authors' provided keywords (Khan et al., 2022; Koloski et al., 2022).

In terms of similarity measures, Cosine similarity[6] is another popular measure for assessing the similarity between documents. It has become increasingly popular among researchers for various studies (de Vos et al., 2021; Sitikhu et al., 2019). Jaro-Winkler distance[7], introduced by Jaro (1989) and later extended by Winkler (1990), is a string similarity measure commonly used in text processing tasks. Jaro-Winkler distance computes the similarity between two strings based on the number of matching characters and their positional proximity, providing a robust measure of similarity for text data. Studies across different domains and applications have consistently highlighted the effectiveness of Jaro-Winkler similarity in measuring the similarity between strings (Ahamed et al., 2021; Alenazi et al., 2017; Efriyanto & Hayaty, 2022; Ouarda et al., 2023).

Latent Semantic Analysis (LSA)[8] is considered as an another similarity measure in NLP which has significant attention across various disciplines for its ability to quantify document similarity based on underlying semantic structures (Deerwester et al., 1990). Research expanding multiple disciplines, including linguistics, information retrieval, and NLP, has extensively explored the applications and implications of LSA. The widespread adoption and adaptation of LSA across different disciplines underscore its

[3] https://pypi.org/project/yake/#:~:text=Yet%20Another%20Keyword%20Extractor%20(Yake)&text=YAKE!%20is%20a%20light%2Dweight,important%20keywords%20of%20a%20text.

[4] https://pypi.org/project/keyphrase-vectorizers/

[5] https://maartengr.github.io/KeyBERT/

[6] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.pairwise.cosine_similarity.html

[7] https://en.wikipedia.org/wiki/Jaro%E2%80%93Winkler_distance

[8] https://en.wikipedia.org/wiki/Latent_semantic_analysis

versatility and effectiveness in capturing semantic relationships between textual documents (Aseervatham, 2008; Foltz, 1996; Vetriselvi et al., 2022) .

Word2Vec[9], another similarity measure in NLP introduced by Mikolov et al. (2013), has covered numerous studies and research endeavors due to its innovative approach to word embedding. This technique represents words as dense vectors in a continuous vector space, enabling the capture of semantic relationships between words. Consequently, Word2Vec facilitates the computation of semantic similarity between documents based on the similarity of their constituent words. Research efforts have touched the applications of Word2Vec across various domains, including NLP, information retrieval, sentiment analysis, and machine translation. Studies have investigated its effectiveness in tasks such as document classification, text summarization, question answering, and recommendation systems (Asudani et al., 2023; Imaduddin et al., 2019; Mahata et al., 2018; Yildiz, 2019).

[9] https://www.tensorflow.org/text/tutorials/word2vec

CHAPTER 03

3 METHODOLOGY AND EXPERIMENTAL DESIGN

The entire methodology of this study is divided into six distinct stages to ensure a systematic approach: (i) exploratory literature review about NLP techniques for annotation with BoK concepts (ii) data acquisition and pre-processing, (iii) design and development of NLP-based tools incorporating key phrase extraction algorithms and similarity computation techniques in NLP, (iv) development of web application using the proposed NLP tools, (v) manual annotation using the BAT by users, (vi) experimental evaluation. The entire methodology, illustrated in Figure 6, encompasses a structured workflow. This chapter discusses detailed description for the first four stages in the methodology.

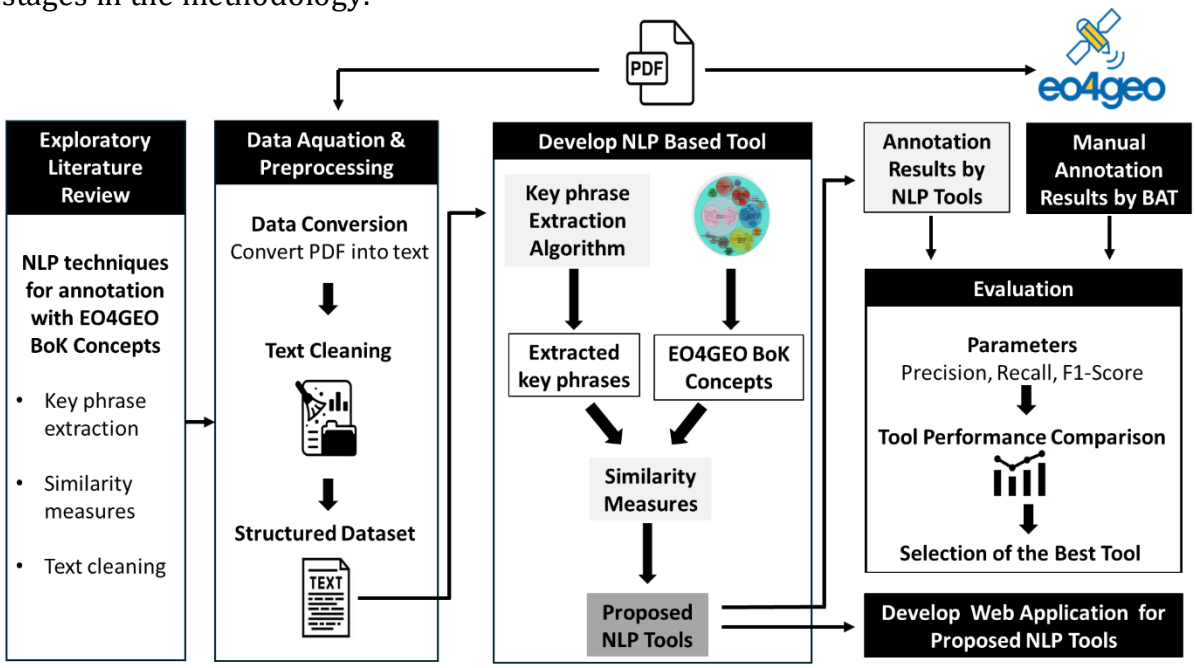


Figure 6: Functional details of the proposed methodology

3.1 Exploratory Literature Review

In the initial phase of designing and developing NLP-based tools for annotating PDFs with EO4GEO BoK techniques, exploratory literature review is done using wide range of resources such as papers, articles, websites, and videos. These resources serve as valuable sources of information on NLP methodologies and techniques. Through review and analysis, suitable methods, and techniques for key phrase extraction, similarity measurement, and text cleaning are identified. This process involves evaluating the

strengths and limitations of various approaches and selecting those that align best with the project's objectives and requirements.

3.2 Data Acquisition and Pre-processing

3.2.1 Data Collection

This study utilizes randomly selected full paper publications in PDF format obtained from the " AGILE: GIScience Series[10]" website as the primary input data for the proposed annotation tools. The AGILE: GIScience Series (AGILE-GISS) serves as a repository for a collection of papers presented at the annual AGILE conferences, providing a valuable platform for the exchange of scientific ideas, methodologies, and experiences within the field of geographical information science (GIScience). The papers encompass a wide range of topics, including basics and computational issues of geographic information, as well as the design, implementation, and utilization of geographical information for diverse applications. Encompassing developments in computer science, geography, cartography, engineering, data science, and artificial intelligence, these publications offer a comprehensive exploration of GIScience. For this study, PDFs were specifically obtained from the 26th AGILE Conference on Geographic Information Science, themed "Spatial data for design," held in Delft, the Netherlands, from June 13 to 16, 2023. The reason to select PDF documents from the AGILE: GIScience Series is motivated by the alignment of the conference themes and topics with the Geographic Information fields (and thus, the EO4GEO BoK).

3.2.2 Data Preprocessing

The data processing procedure involved two primary tasks: (i) content identification of the PDF and extract required portion (ii) text cleaning for the selected portions shown in Figure 7.

Content Identification and Extraction of the Relevant Text Portions

For content identification, it's important to note that AGILE papers follow a consistent format, encompassing headers, footers, titles, abstracts & keywords, main text, and references. This study focuses on annotation text within the abstracts & keywords and main text of the publication. The selection of the abstract & key words and main text

[10] https://www.agile-giscience-series.net/

portion of the paper for annotation is motivated by a combination of factors that contribute to the effectiveness and efficiency of the NLP tools.

- The abstract & key words and main content of the papers contains the key information, insights, and findings related to the research topics.
- Removing unnecessary parts can reduce noise and irrelevant information, allowing NLP tools to focus on the essential components for annotation.
- Annotation precision is crucial for the accuracy of NLP tools. By narrowing the focus to the main text, increase the precision of tools in identifying and extracting key information.

First, PDF is read using python language. It is achieved using 'PdfReader[11]' which a Pythonic API designed for extracting various types of data, including text and images, from PDF documents (Python Software Foundation, 2016) imported from PyPDF2[12] library (version-3.0.1). Consequently, components such as headers, footers, titles, references were removed. To remove headers and footers is PDF document PyMuPDF[13](version- 1.23.6) library is utilized. Moreover, the main title along with author details and the DOI were extracted from the papers for further reference and potential needs in subsequent stages of the study.

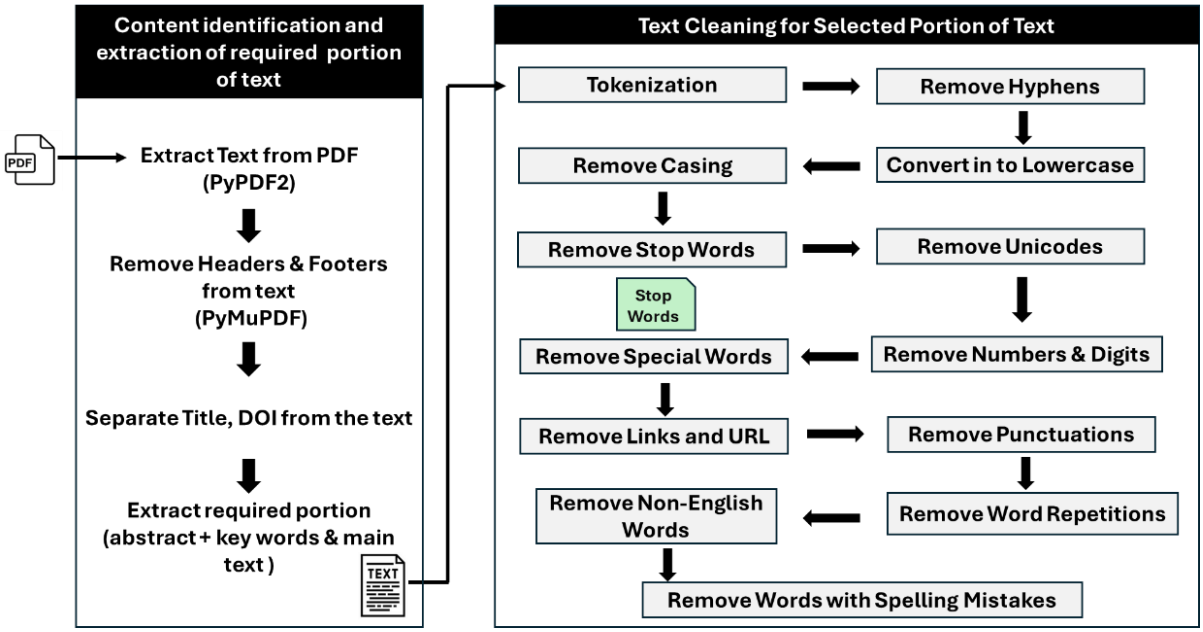


Figure 7: Workflow for the Data Pre-Processing

[11] https://pypi.org/project/pdfreader/
[12] https://pypi.org/project/PyPDF2/
[13] https://pypi.org/project/PyMuPDF/

Text Cleaning for the Selected Portion

Subsequently, the extracted abstract, key words and main text portion undergoes a text cleaning process, involving a series of steps to enhance the quality and consistency of the data.

1. **Tokenization:** Tokenization involves breaking the text into individual tokens or words. This was done by using NLTK[14] library (version- 3.5) which is a powerful library for working with human language data using the Python programming language (NLTK, 2009).
2. **Remove Hyphens within Words:** The removal of hyphens (e.g. 'state-of-art' → state of art) within words ensures that hyphenated words are treated as distinct entities during analysis, preventing potential misinterpretations. This task is accomplished by developing a Python script tailored to this specific purpose.
3. **Convert Main Text to Lower Case:** Converting the entire main text to lowercase standardizes the text, eliminating variations in case and ensuring uniformity in subsequent processing. This is done by writing python script. This task is accomplished by developing a Python script tailored to this specific purpose.
4. **Remove Casing:** This step involves the removal of casing information. This task is accomplished by developing a Python script tailored to this specific purpose.
5. **Remove Stop words:** Stop words are typically defined as the most common words in a language (e.g. prepositions, conjunctions, and common pronouns). With NLP, stop words are generally removed because those aren't significant, and heavily distort any word frequency analysis. The removing of stop words in pre-processing is formulated using 'stopwords' module in NLTK which contains various lists of stop words for different languages. After downloading the stop words (nltk.download('stopwords')) corpus, access the stop words lists in Python script using NLTK's stopwords module, allowing to easily filter out stop words from text data.
6. **Remove Unicode Symbols:** Unicode symbols may include special characters, emojis, or non-standard characters, are removed to address any potential encoding issues and ensure compatibility. This task is accomplished by developing a Python script with regular expression tailored to this specific purpose.

[14] https://www.nltk.org/

7. **Remove Numbers and Digits:** The removal of numerical digits contributes to the focus on textual content and eliminates potential interference from numerical information. This task is accomplished by developing a Python script with regular expression tailored to this specific purpose.
8. **Remove Special Words Mixed with String, Numbers, Punctuation, and Other Symbols:** Special words that may be a mix of alphanumeric characters and symbols are removed to maintain clarity and prevent misinterpretations. This task is accomplished by developing a Python script with regular expression tailored to this specific purpose.
9. **Remove Links and URLs**: Links and URLs are eliminated from the text to avoid interference with subsequent analysis and maintain the focus on the textual content. This task is accomplished by developing a Python script with regular expression tailored to this specific purpose.
10. **Remove Punctuation:** The removal of punctuation (e.g., @, #, $, etc.). This task is accomplished by developing a Python script with regular expression tailored to this specific purpose. This task is accomplished by developing a Python script with regular expression tailored to this specific purpose.
11. **Remove Word Repetitions:** Repetitive words are removd to reduce redundancy and enhance the clarity and conciseness of the text. This task is accomplished by developing a Python script with regular expression tailored to this specific purpose.
12. **Remove Single Characters and Non-English Words:** Single characters and non-English words are removed to enhance the quality of the text. This was achieved by importing the words corpus from the NLTK library. This words corpus contains a collection of words from various languages, often used for tasks such as spell checking, text generation, and linguistic analysis.
13. **Remove Words with Spelling Mistakes:** Words with spelling mistakes are excluded to ensure the accuracy and reliability of the processed text. Spelling errors can also be corrected during the analysis. Python's pyenchant[15] (version – 3.2.2) library proves invaluable in addressing this concern.

[15] https://pypi.org/project/pyenchant/

[16] https://github.com/UpekshaIndeewari/geotec_thesis_EO4GEO/tree/main/Data_processing

Figure 8 displays the word count comparison before and after the text cleaning process[16]. The graph was plotted using visualization libraries in python pandas (version 2.0.3) and matplotlib (version- 3.8.0). This visual representation illustrates the impact of text cleaning on the number of words in the main text. This comparative analysis serves to highlight the significance of text cleaning and enhancing the quality of the textual data for subsequent processing and analysis.

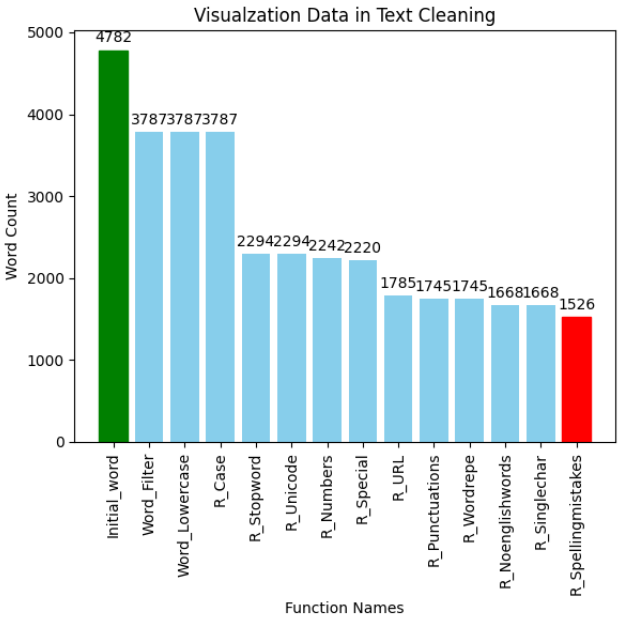


Figure 8: Word Count Comparison Before and After Text Cleaning for (Mocnik, 2023)

3.3 Design and Development of NLP-based Tools

3.3.1 Architecture of the Proposed NLP-Based Tool

The primary objective of developing the NLP-based tools is to automate the annotation process of PDF documents using EO4GEO BoK concepts. The proposed tools are developed using Python language and tools comprise main five sections to achieve the annotation procedure.

(i) In the first section, once PDF resource is browsed, text processing techniques are applied to extract the relevant portions of text, followed by a text cleaning process to enhance data quality.

(ii) Second section, key phrases, which represent important terms in the text, are extracted using key phrase extraction techniques in NLP.

(iii) In the third section EO4GEO BoK concepts are extracted from the EO4GEO API.

(iv) Then, in the fourth section, the extracted key phrases and BoK concepts undergo matching using similarity measures in NLP.

(v) In the fifth section, BoK concepts that exceed a predefined threshold value are presented as output in JSON format.

Based on the outlined architecture, a total of 12 NLP-based tools[17] are designed and developed using the combination of 03 key phrase extraction and 04 similarity measures in NLP (Table1)

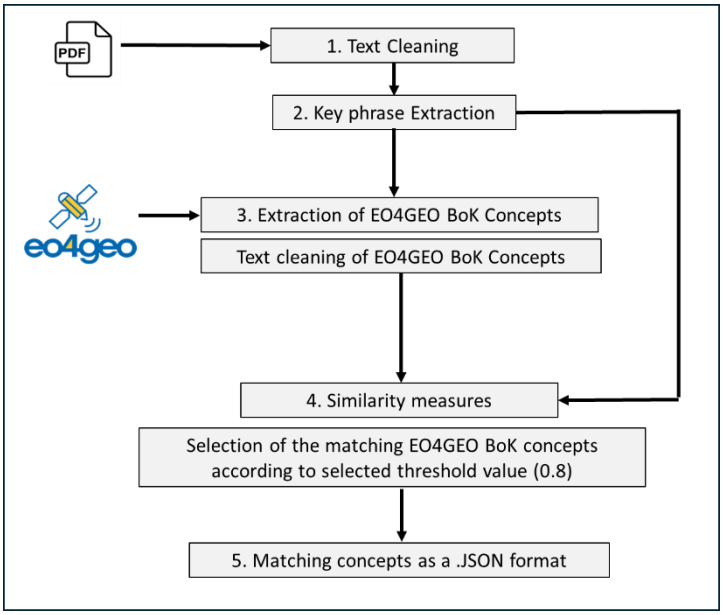


Figure 9: The pipeline of the proposed NLP tool

The detail functionality of key sections include:

Section 01: Text cleaning

This section (Text_Cleaner) is responsible for preprocessing the text extracted from PDF documents. It involves the tasks mentioned in the data preprocessing section (3.2.2.2). After completing data preprocessing, the list of cleaned text is converted into string and then entire string is used as an input to key phrase extraction function.

Section 02: Key phrase extraction for cleaned text

In this section, (YAKE_Extractor, PATTERNRANK_Extractor, KEYBERT_Extractor) focus on the extraction of key phrases from the preprocessed text. In this study, three distinct

[17] https://github.com/UpekshaIndeewari/geotec_thesis_EO4GEO/tree/main/NLP_Tools

algorithms namely, YAKE, PatternRank, and KeyBert have been selected from the unsupervised key phrase extraction methods in NLP. Each algorithm contributes its unique approach to identifying and isolating essential phrases in the text. in this stage for YAKE_Extractor, YAKE library (version - 0.4.8), for PATTERNRANK_Extractor, Python package named keyphrase_vectorizers (version -0.0.11) and for KEYBERT_Extractor using KeyBert (version – 0.8.1) library in Python.

For each algorithm, users can adjust the number of extracted key phrases, providing a customizable experience. This facilitates users in obtaining the key phrases with the highest scores by allowing them to specify the desired number. For instance, if a user specifies "50," the tool will provide the top 50 extracted key phrases based on their scores. This functionality empowers users to customize the output based on their specific requirements.

Section 03: Extraction of the list of EO4GEO BoK concepts and pre-processing the extracted list

This section (EO4GEO) involves automating the extraction of EO4GEO BoK concepts from an EO4GEO BoK visualization and search tool[18] and EO4GEO BoK RESTfull API v2[19]. The extracted BoK list is then pre-processed, potentially involving tasks mentioned in the data preprocessing section (section 3.2.2.2).

Section 04: Measure similarity between extracted key phrases and EO4GEO concepts

The role of these similarity measurement section (Cosine_Similarity, Jarowinkler_Similarity, LSA_Similarity, Word2Vec_Similarity) is to compute the similarity between the list of extracted key phrases and the pre-processed EO4GEO BoK concepts. In this study, four distinct similarity technologies within the lexical and semantic similarity have been employed. These technologies encompass Cosine Similarity, Jarow-Winkler Similarity, LSA Similarity, and Word2Vec Similarity measures.

In order to develop cosine similarity, 'cosine_similarity' function which computes the cosine similarity between pairs of samples in two sets of vectors was used from the sklearn. metrics.pairwise[20] module and 'CountVectorizer' class which used for converting a collection of text documents into a matrix of token counts were imported from the sklearn.feature_extraction.text[21] module. Jarow-Wrikler similarity was

[18] https://bok.eo4geo.eu/GIST
[19] https://eo4geo-uji.web.app/documentation/API.pdf
[20] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.pairwise.cosine_similarity.html
[21] https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html
[22] https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.TruncatedSVD.html

developed using a Python script tailored to this specific purpose. For develop Word2Vec similarity 'CountVectorizer' class from sklearn.feature_extraction.text module, 'TruncatedSVD' class from sklearn.decomposition[22], and 'Normalizer' class from the sklearn.preprocessing[23] module were utilized. TruncatedSVD was used to reduce the dimensionality of the input data by projecting it onto a lower-dimensional space and Normalizer is used for feature-wise normalization of data.

Each of these methods contributes a unique perspective to measure the likeness between the extracted key phrases and the predefined EO4GEO BoK concepts, providing a comprehensive evaluation of semantic relationships. This offers a range of outputs and features designed to enhance the user's interaction with the annotation results:

- All instances of similarity occurrences between the extracted key phrases and EO4GEO BoK concepts are presented in descending order and organized based on the threshold similarity scores.
- Users can obtain a count of all matching BoK concepts which belong within the specified threshold.
- Users benefit from the flexibility to display matching BoK concepts according to their specific needs by adjusting the threshold values. This interactive adjustment enhances users to focus on similarities that meet their desired level of relevance.
- EO4GEO BoK concepts with the highest similarity scores gives as the most closely aligned concepts related to the content of the PDF.
- Provides a comprehensive overview by presenting the total number of EO4GEO BoK concepts matched to the given PDF and their respective frequencies. Further, this summary provides users in identifying and prioritizing the most relevant BoK concepts associated with the content of the PDF based on the frequencies.

Section 05: Download Output Results as a JSON Format

This section (Create_Json) allows users to download the output results in a structured format, specifically JSON. To accomplish this task 'json' library (version – 0.9.14) in Python was used. This gives the DOI, title of the relevant paper, and the matching EO4GEO BoK concepts (with original names) for given PDF document shown in figure 10.

[23] https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.Normalizer.html

```
{
    "DOI": "https://doi.org/10.5194/agile-giss-4-41-2023",
    "Title": "Individualization in Spatial Behaviour and Map Reading Angela Schwering1 , Jakub Krukar1 , Jana Seep1
    and Yousef Qamaz1 1 Institute for Geoinformatics , University of Muenster , Muenster , Germany Correspondence :
    Angela Schwering ( schwering @ uni muenster.de )",
    "Concepts": [
        "Local variance",
        "Map production",
        "Map projections",
        "Semantic discovery",
        "Spatial thinking",
        "Semantic categorisation",
        "Tablet digitizing",
        "Machine learning"
    ]
}
```


Figure 10: The output of the proposed NLP-based tool as a. JSON format

Table 1: Proposed NLP-based annotation tools

Key phrase extraction method	Similarity measure	Proposed NLP tool
YAKE	Cosine	YAKE_Cosine
	Jarow-Wrinkler	YAKE_JaroW
	Latent Semantic Analysis	YAKE_LSA
	Word2Vec	YAKE_Word2
PatternRank	Cosine	PATTERN_Cosine
	Jarow-Wrinkler	PATTERN__JaroW
	Latent Semantic Analysis	PATTERN__LSA
	Word2Vec	PATTERN__Word2
KeyBert	Cosine	KYBERT_Cosine
	Jarow-Wrinkler	KYBERT __JaroW
	Latent Semantic Analysis	KYBERT __LSA
	Word2Vec	KYBERT __Word2

3.3.2 Development of Web Application

To present the proposed NLP-based tools to users in a user-friendly and interactive manner, a web application[24] was developed. Given that the entire codebase is written in the Python language, a suitable framework was selected to integrate with Python's strengths and facilitate rapid web development. After careful consideration, Flask (version – 3.0.0) (Pallets, 2010) was chosen as the preferred framework for building the web application. The reasons to select Flask framework are as follows.

[24] https://github.com/UpekshaIndeewari/geotec_thesis_EO4GEO/blob/main/Website_demo.mp4

- Flask is a small and lightweight Python web framework, offering a set of valuable tools and features that simplify the process of creating web applications in Python.
- Its lightweight nature provides developers with a high degree of flexibility, making it an accessible framework, especially for those new to web development.
- One special advantage of Flask is its ability to facilitate rapid web application development using just a single Python file. This approach not only accelerates the development process but also makes it more approachable for developers looking to quickly prototype or build small to medium-sized applications.
- is highly extensible, allowing developers to tailor their applications according to specific needs.
- Unlike some frameworks, Flask doesn't impose a rigid directory structure or mandate boilerplate code before diving into development. This provides developers with the freedom to organize their projects in a way that best suits their preferences and requirements.
- Flask's ease of use, extensive documentation, and vibrant community support were key factors in the decision-making process, ensuring a smooth development experience and robust functionality.

In addition to Flask, the Bootstrap toolkit (Otto, 2022) has been employed to enhance the visual appeal and styling of the web application. Bootstrap serves as a valuable tool for crafting a visually engaging user interface without the need to manually write extensive HTML, CSS, and python code[25]. Leveraging Bootstrap, developers can incorporate responsive web pages into the application, ensuring optimal performance across various devices, including mobile browsers.

Figure 11 shows the steps to develop a simple web application for proposed NLP tools using Flask framework.

[25] https://github.com/UpekshaIndeewari/geotec_thesis_EO4GEO/tree/main/Website%20-%20Copy

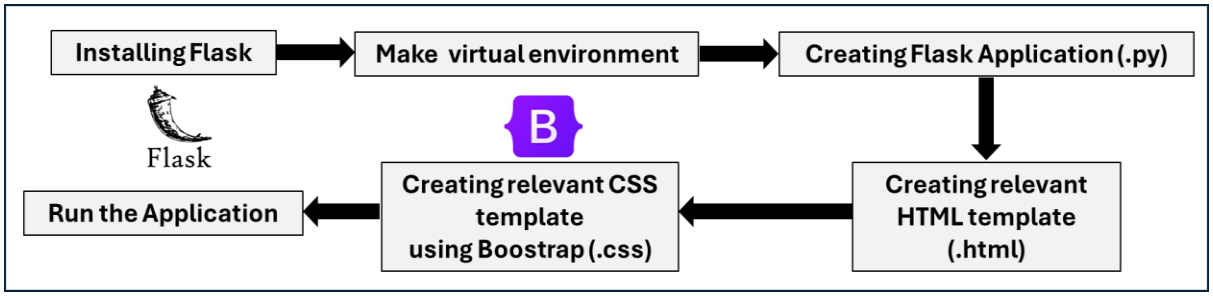


Figure 11: The workflow of development of web application using Flask framework.

- To initiate this development, Flask, a lightweight Python web framework, is first installed.
- Virtual environment is created as a best practice in Python development to manage project dependencies and isolate them from the system-wide Python installation.
- The application file `app.py` was created to define the Flask application and establish routes.
- The HTML (HyperText Markup Language) file named 'index.html is created to define structure and the content on the web page.
- To further customize the visual presentation, a CSS ("Cascading Style Sheets") file named `style.css` is created. This file allows for the addition of custom styles or Bootstrap classes, enhancing the overall appearance and ensures the application's responsiveness of the web application.
- Once the Flask application and styling components are in place, the application is run locally, and users can access the application in their web browser.

The following describes the overview and functionalities of the proposed web application.

Once a user logs in to the 'Home' page (1), if the user is new, he/she can get an overview of the web application through 'Instructions' section (2). If the user needs to go to annotation can go through 'Annotation' section (3). Through 'Home' page, user can go directly go to annotation through 'Click Here to Annotation' button (4) which is shown in figure 12.

'Instruction' section mentioned all the steps of annotation and user can get clear idea about how the tool/s are worked. When user click 'Click Here to Annotation' button (5) in this page, it will directly go to annotation tab shown in figure 13.

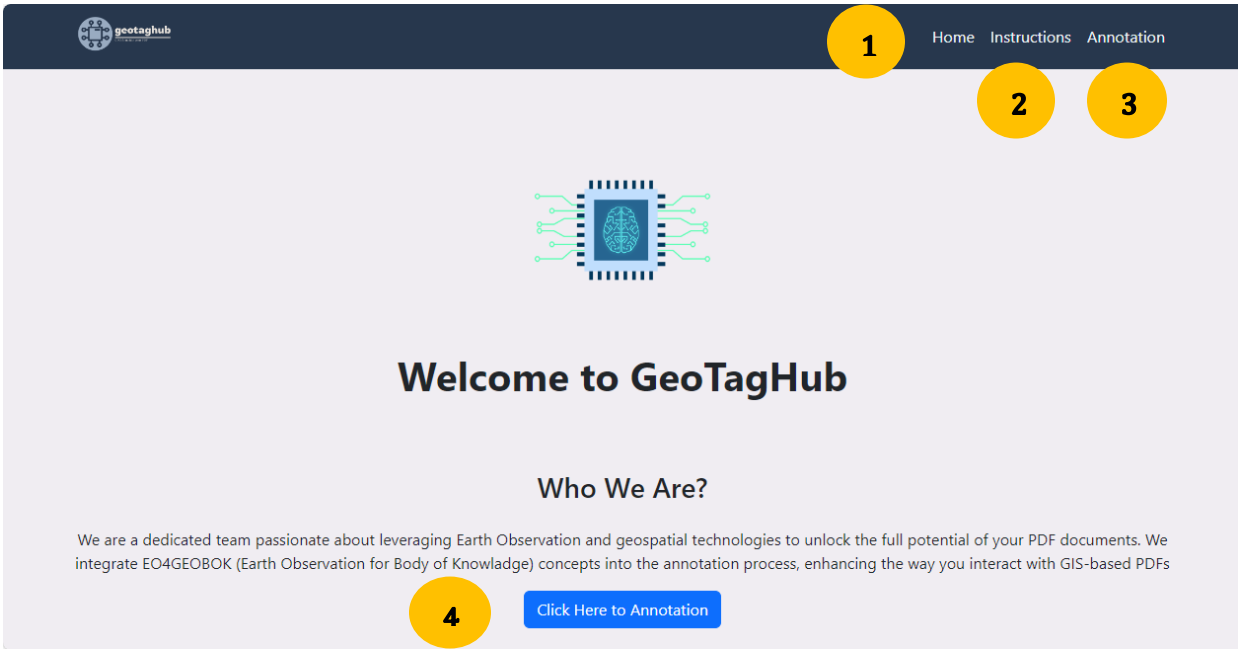


Figure 12: 'Home' page in proposed web application

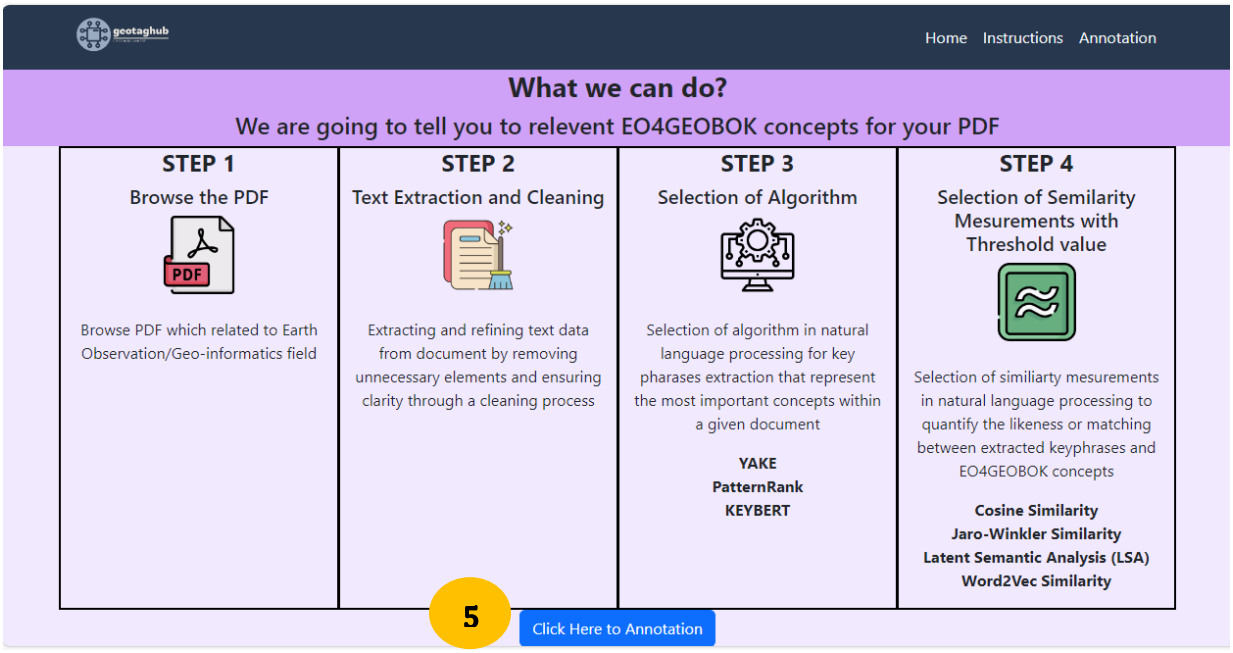


Figure 13: 'Instructions' page in proposed web application

In the annotation page, the user needs to browse the PDF which relevant EO/GI field through 'Choose File' button (6). After clicking 'Extract Text' button (7) the text from abstract to reference was extracted and cleaned. Once it finished, user will get message "*Text Extraction and Cleaning is Done!"* and the title and author details of the PDF are printed. Then the user needs to select key phrase extraction algorithm through drop down list (8) and click 'Extract Key phrases' button (9). Once it finishes it gives the message "*You have selected the algorithm*" and the user needs to click 'Extract EO4GEO

BoK Concepts' button (10). Then it will give message "*You have extracted the EO4GEOBOK Concepts*". In the final stage the user needs to select similarity measure and threshold value through drop down lists (11,12) and click 'Calculate Similarity' button (13). Once it is finished, user will get "DONE!!!, *Your document is matching with following EO4GEO Concepts".* Then relevant matching EO4GEO BoK concepts for given PDF are printed. Above mentioned steps and output of each step are shown in figure 14 and 15.

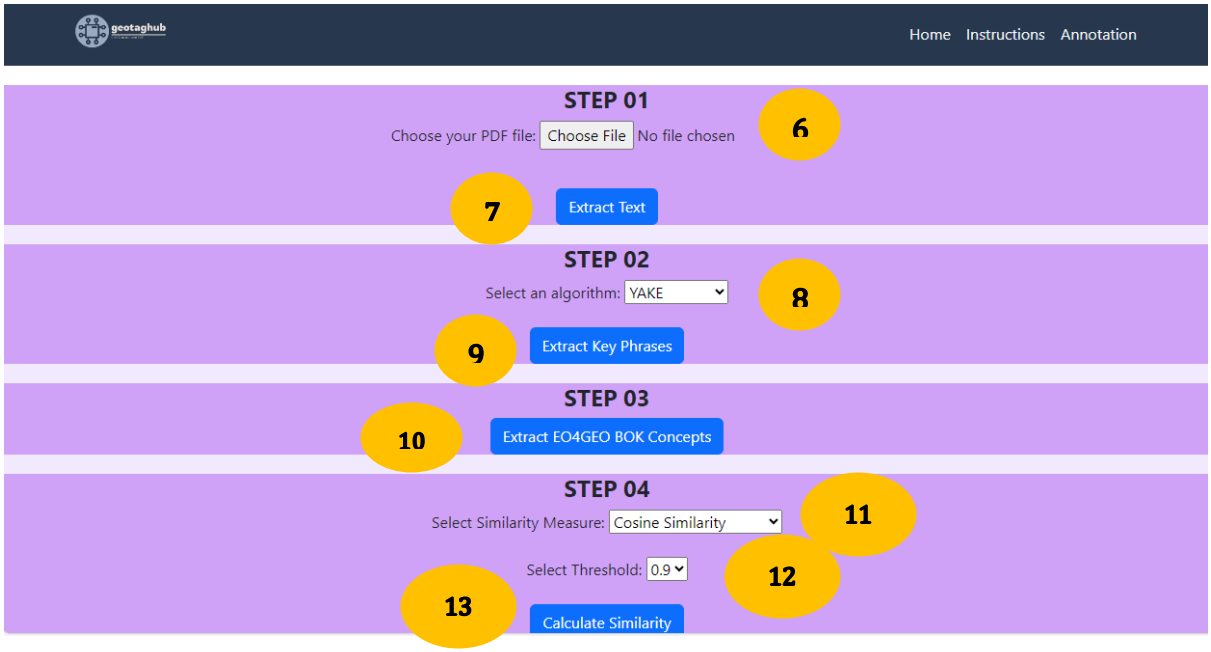


Figure 14: Output of the proposed web application

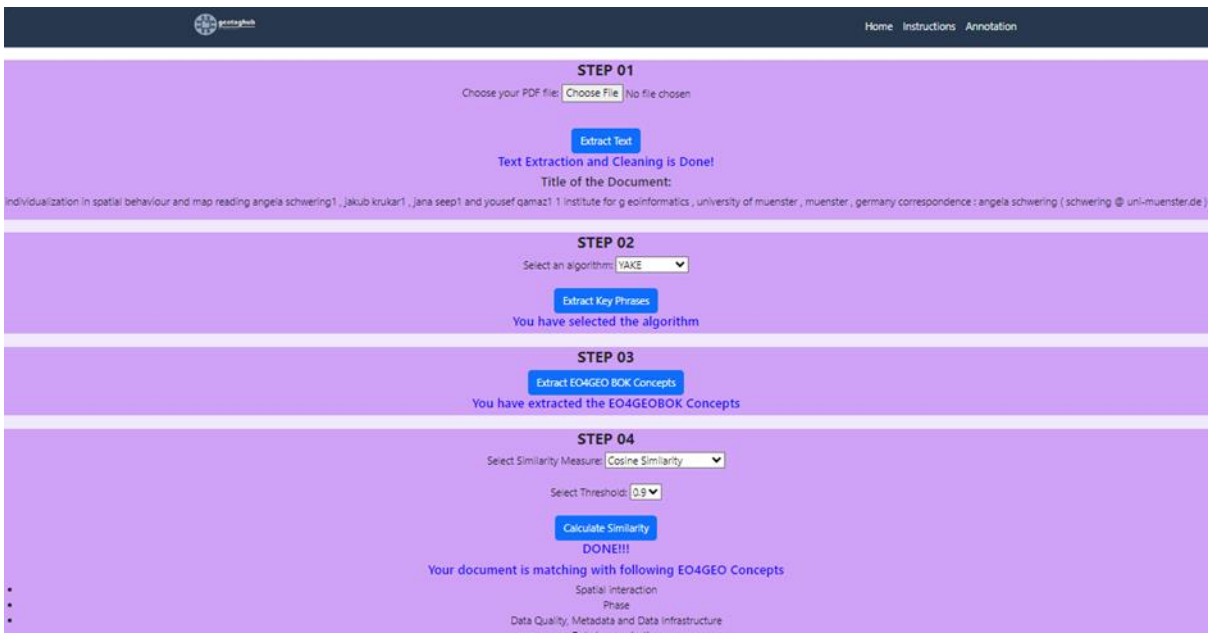


Figure 15: Output of the proposed web application

CHAPTER 04

4 EXPERIMENTAL EVALUATION

The entire experimental evaluation of the performance of the proposed NLP tools is undertaken using extracted EO4GEO BoK concepts given by NLP tools with EO4GEO BoK concepts for the selected articles (see section 3.2.1 Data Collection) and comparing them with annotations performed manually using the BAT tool provided by the authors of the relevant papers (Figure 16). This comprehensive evaluation process involves two main steps: annotation and evaluation.

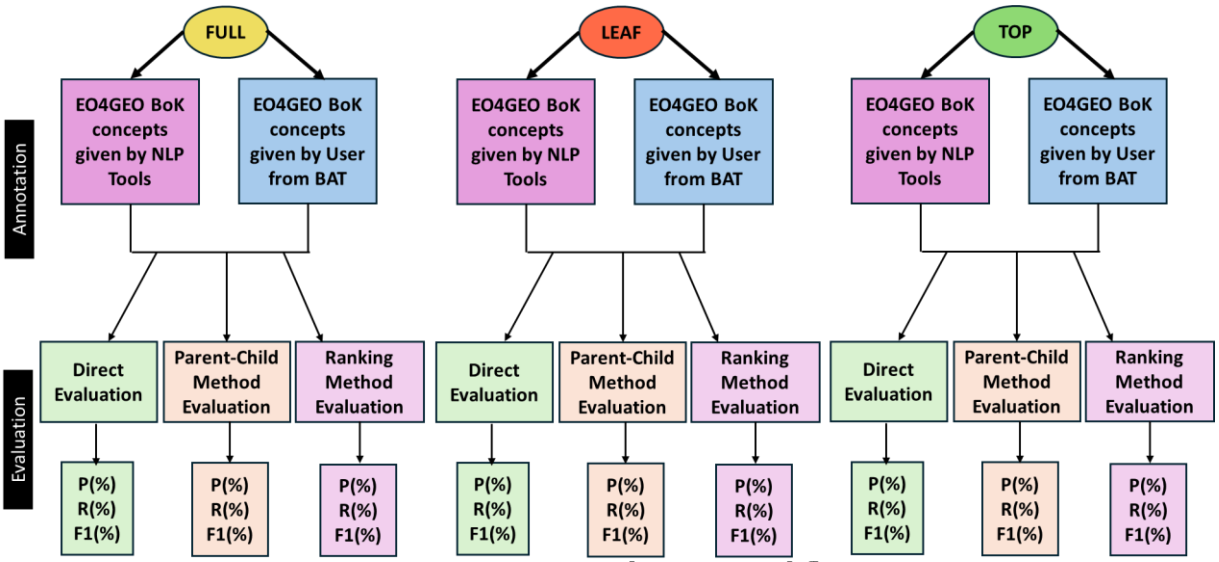


Figure 16: Entire evaluation workflow

4.1 Annotation

The annotation procedure[26] is done with three approaches known as FULL, LEAF and TOP:

- FULL: all 952 EO4GEO BoK concepts are considered for annotation within the PDF documents.
- LEAF: focuses on the bottom level of the EO4GEO BoK, i.e. concepts with no children (346 concepts).
- TOP: involves annotating only with the top-level main concepts, i.e. the 14 EO4GEO BoK concepts that are direct children of the root node (i.e."Geographic Information Science and Technology").

There are several purposes to use FULL, LEAF, and TOP approaches for the annotation, each contributing to a comprehensive assessment of the tools' performance.

[26] https://github.com/UpekshaIndeewari/geotec_thesis_EO4GEO/tree/main/Annotation_approaches

- By annotating with FULL approach, the evaluation captures the tools' capability to handle a wide range of concepts across different levels of the BoK hierarchy. This ensures a thorough examination of the tools' coverage and efficiency in annotating with diverse concepts within the relevant domain.
- By annotating with the LEAF approach, it allows for a more targeted evaluation of the tools' performance in annotating specific, granular concepts within the BoK.
- Same as LEAF approach, TOP approach offers a broader perspective on the tools' performance, focusing on their ability to identify and annotate core concepts within the relevant domain. Evaluating the tools' performance at this level assesses their proficiency in capturing high-level concepts.

In this study, the annotations are classified as follows.

- Annotation 1: Annotations of the selected Agile papers using the BAT tool by one of the original authors of the respective paper.
- Annotation 2: Annotation of same set of Agile papers using the proposed NLP-based tools. For each paper, each of the selected 3 NLP algorithms (YAKE, PatternRank, KeyBert) in combination with 3 matching techniques (Cosine, Jarow-Wrinkler, Word2Vec) were used to generate annotations.

In the context of the annotation process, annotation 1 and 2 are executed for each of the three distinct approaches: FULL, LEAF, and TOP. This entails that for a given Agile paper, both the user and the tool perform annotations three times, each time aligning with one of the specified approaches. This comprehensive approach ensures a thorough evaluation across different levels within the EO4GEO BoK hierarchy. These approaches allow for more understanding of the tools' strengths and weaknesses, their performance in handling varying levels of complexity and granularity within the domain.

4.2 Evaluation

This evaluation is accomplished by using the matching EO4GEO BoK concepts by the proposed NLP-based tool with the BoK Concepts by manual annotation using BAT tool given by the author for the different annotation approaches.

To measure the effectiveness of the proposed NLP tools, key performance metrics[27], namely precision (P%), recall (R%), and F1-score (F%), are calculated. The PDF document is classified using a multi-class classification method, wherein the classes correspond to the total number of EO4GEO BoK concepts within each annotation approach (952 for FULL, 346 for LEAF, and 14 for TOP). The evaluation employs a micro-averaging method for multi-class classification, ensuring equal weight is given to each instance across all classes. Micro-averaging involves aggregating the counts of true positives (TP), false positives (FP), and false negatives (FN) across all classes, subsequently computing precision and recall based on these total counts (EvidentlyAI, 2024).

- Total True Positive (TP) is the sum of true positive counts across all classes.
- Total False Positive (FP) is the sum of false positive counts across all classes.
- Total False Negative (FN) is the sum of false negative counts across all classes.

Following this summation, precision and recall are computed utilizing these cumulative counts.

4.2.1 Evaluation Parameters

4.2.1.1 Precision

Precision is a measure of the accuracy of positive predictions, is calculated by dividing the total true positives by the sum of total true positives and false positives. This ratio provides the proportion of correctly predicted positive instances among all instances predicted as positive (EvidentlyAI, 2024). The formula for calculating the precision is:

$$Precision_{\ Micro\ Averages} = \frac{TP_A + TP_B + \ldots\ldots\ldots + TP_N}{TP_A + FP_A + TP_B + FP_B \ldots\ldots\ldots + TP_N + FP_N} \quad \text{---------- } Equation\ 01$$

Where,

- A, B, ..., N: Classes or categories
- TP_A, TP_B, ..., TP_N: True Positives for each class.
- FP_A, FP_B, ..., FP_N: False Positives for each class

4.2.1.2 Recall

Similarly, recall, which assesses the model's ability to capture all positive instances, is determined by dividing the total true positives by the sum of total true positives and

[27] https://github.com/UpekshaIndeewari/geotec_thesis_EO4GEO/tree/main/Evaluation_matrix

false negatives. This ratio calculates the effectiveness of the model in identifying and correctly classifying all actual positive instances (EvidentlyAI, 2024). The formula for calculating the Recall is:

$$Recall_{\ Micro\ Averages} = \frac{TP_A + TP_B + \ldots\ldots\ldots + TP_N}{TP_A + FN_A + TP_B + FN_B \ldots\ldots\ldots + TP_N + FN_N} \quad \text{---------- } Equation\ 02$$

Where,

- A, B, ..., N: Classes or categories
- TP_A, TP_B, ..., TP_N: True Positives for each class.
- FP_A, FP_B, ..., FP_N: False Positives for each class

4.2.1.3 F1- Score

The F1-score is a metric commonly used in classification tasks to evaluate a model's performance, especially when there is an imbalance between the classes. It is the harmonic mean of precision and recall and provides a balanced measure that considers both false positives and false negatives (EvidentlyAI, 2024). The formula for calculating the F1-score is:

$$F1 - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \text{ ---------- } Equation\ 03$$

The F1-score ranges from 0 to 1, with 1 indicating perfect precision and recall, and 0 indicating poor performance. It is particularly useful in situations where false positives and false negatives have different consequences or when there is an imbalance between the classes. A higher F1-score suggests a better balance between precision and recall, indicating a more robust classifier.

4.2.2 Evaluation Approaches

Once the PDFs are annotated for the specified annotation approaches using both NLP-based tools and user input, the study employs a structured evaluation process divided into three distinct categories: (i) direct matching evaluation method, (ii) parent-child matching evaluation method, and (iii) ranking-based evaluation.

4.2.2.1 Direct Matching Evaluation Method

In the direct evaluation method, the focus is on assessing the results related to EO4GEO BoK concepts provided by both the user and the NLP-based tool. This evaluation is

conducted to directly measure micro-averages, providing a comprehensive understanding of the performance of the annotation approaches. Micro averages are calculated by aggregating the counts of true positives, false positives, and false negatives across all classes and then determining precision, recall, and F1-score based on these aggregated counts. This direct approach offers a concise and overall evaluation of the effectiveness of the proposed NLP tool and user annotations in capturing relevant BoK concepts within the PDF documents.

4.2.2.2 Parent-Child Matching Evaluation

To enhance the performance of the tool, a solution is implemented wherein the results provided by both the user and the NLP-based tool are adjusted based on parent-child combinations within the EO4GEO BoK visualization.

In this context, if a child element is identified in the results generated by the NLP-based tool and the corresponding parent element is identified in the results provided by the user, the child element is replaced with its relevant parent element, and vice versa. Importantly, this replacement strategy is applied selectively, specifically when the two concepts involved have a direct parent-child relationship within the BoK hierarchy. It's essential to note that this adjustment is not applicable when the two concepts under consideration belong to different branches of the BoK hierarchy. In such cases, where there is no direct parent-child relationship, this replacement method is not applied.

This specific approach helps maintain the meaningful structure of the BoK hierarchy. The refined adjustments, focusing on these relationships, aim to fine-tune and optimize the tool's results. The goal is to bring the NLP-based tool's output closer to what the user provided, improving overall performance in identifying BoK concepts.

4.2.2.3 Ranking-based Method

In this approach, the goal is to align the number of annotated concepts provided by the NLP-based tool more closely with the number given by the user. This is important for effective performance evaluation because if the tool outputs a higher number of concepts than the user, it can impact the evaluation results.

To address this, a solution is implemented where the evaluation is conducted in stages. For instance, if the user provides 10 annotated results and the NLP-based tool outputs

30, the evaluation begins with the first 10 annotated results from the NLP-based tool being compared with the user's results. Subsequently, the evaluation extends to the first 20 annotated results, and at each stage, performance parameters are assessed. This staged evaluation allows for a fair comparison, considering an increasing number of annotated results from the NLP-based tool and aligning it with the user's input. The stage that yields the highest performance results is then selected as the representative evaluation point.

This strategy is designed to deal with differences in the number of annotations between the user and the NLP-based tool. By evaluating the results in stages and comparing a matching number of annotations at each step, ensure a fair and accurate assessment. This approach aims to balance the evaluation, preventing the NLP-based tool's higher number of annotations from skewing the results. Ultimately, it helps better to measure how well the tool identifies and matches concepts compared to what the user provided.

CHAPTER 05

5 RESULTS AND DISCUSSION

5.1 Results for Evaluation Parameters

In this study, we explore the performance of various proposed NLP-based tools in annotating documents with EO4GEO BoK concepts. As mentioned above, the chapters employ three distinct unsupervised key phrase extraction methods: YAKE, PatternRank, and KeyBert, Furthermore, four similarity measures are used such as, Cosine, Jaro-Winkler, LSA, and Word2Vec2 to develop proposed NLP-based tools. By combining these techniques, developed twelve NLP-based tools. However, the results produced by the LSA algorithm were discarded because it tends to generate a high number of concepts. Consequently, the evaluation process was conducted using 9 NLP-based tools on randomly selected Agile papers, calculating: precision, recall, and F1-score for three annotation approaches (comparing with expert-based annotation): FULL, LEAF, and TOP. Additionally, three evaluation methods are employed namely, direct, parent-child, and ranking to comprehensively assess the performance of the proposed NLP tools.

To determine the best threshold value, various thresholds were tested, and a threshold value of 0.8 was identified as optimal after analyzing the annotated results for each tool. When considering a threshold of 0.7, the number of concepts detected was the highest, potentially skewing the results. Conversely, a threshold of 0.9 resulted in the lowest number of concepts, also impacting the outcome. Therefore, a threshold of 0.8 was selected as it yielded a moderate number of concepts, approximately the amount a human expert would typically user. In addition, the number of key phrases extracted from proposed NLP algorithms are set as a 100.

The average performance metrics for the developed NLP tools are presented in table 2. This table shows the mean values of precision (P%), recall (R%), and F1-score (F%) across the various annotation approaches and evaluation methods. Through this comprehensive evaluation, we aim to compare and identify the most effective NLP-based tool for annotation with EO4GEO BoK concepts. The results of the proposed NLP tools are discussed under three evaluation approaches. In this evaluation 8 Agile papers were used.

Table 2: Performance of the NLP based tool for annotation with EO4GEO BoK concepts in terms of annotation approaches and evaluation approaches.

Annotation Approach	Technique	Evaluation Method								
		Direct			P-C			Ranking		
		P(%)	R(%)	F(%)	P(%)	R(%)	F(%)	P(%)	R(%)	F(%)
FULL-952	YAKE_Cosine	20.83	14.73	17.26	54.93	18.36	27.52	12.50	12.04	12.27
	YAKE_Jarow	6.41	22.26	9.95	19.34	42.70	26.62	9.11	17.57	12.00
	YAKE_Word2vec	6.73	37.31	11.41	10.10	46.14	16.57	9.85	36.29	15.49
	PATTERN_Cosine	21.39	5.63	8.91	52.70	25.87	34.71	21.39	5.63	8.91
	PATTERN_Jarow	5.37	26.31	8.92	23.04	38.80	28.91	6.83	22.33	10.46
	PATTERN_Word2vec	7.11	45.47	12.30	13.64	53.93	21.78	6.53	37.04	11.10
	KEYBERT_Cosine	42.76	6.93	11.92	73.00	9.91	17.44	42.76	6.93	11.92
	KEYBERT_Jarow	4.15	13.34	6.33	19.57	25.77	22.25	5.59	10.90	7.39
	KEYBERT_Word2vec	6.00	24.03	9.60	21.51	36.29	27.01	16.80	32.31	22.11
LEAF - 346	YAKE_Cosine	13.04	12.71	12.88	13.04	12.71	12.88	13.04	12.71	12.88
	YAKE_Jarow	4.30	25.76	7.37	4.30	25.76	7.37	10.99	22.90	14.85
	YAKE_Word2vec	6.61	45.14	11.54	6.61	45.14	11.54	14.43	30.43	19.58
	PATTERN_Cosine	23.71	18.43	20.74	23.71	18.43	20.74	19.00	13.71	15.93
	PATTERN_Jarow	3.96	22.29	6.73	3.96	22.29	6.73	5.36	17.57	8.22
	PATTERN_Word2vec	3.55	40.43	6.53	3.55	40.43	6.53	12.13	33.29	17.78
	KEYBERT_Cosine	32.14	12.81	18.32	32.14	12.81	18.32	32.14	12.81	18.32
	KEYBERT_Jarow	8.61	28.43	13.22	8.61	28.43	13.22	7.93	23.71	11.88
	KEYBERT_Word2vec	2.35	22.24	4.24	2.35	22.24	4.24	10.94	17.53	13.47
TOP -14	YAKE_Cosine	28.57	9.00	13.69	28.57	9.00	13.69	28.57	9.00	13.69
	YAKE_Jarow	78.57	37.57	50.83	78.57	37.57	50.83	78.57	37.57	50.83
	YAKE_Word2vec	24.57	30.43	27.19	24.57	30.43	27.19	24.57	30.43	27.19
	PATTERN_Cosine	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	PATTERN_Jarow	21.43	20.29	20.84	21.43	20.29	20.84	21.43	20.29	20.84
	PATTERN_Word2vec	25.00	29.57	27.09	25.00	27.43	26.16	25.00	27.43	26.16
	KEYBERT_Cosine	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	KEYBERT_Jarow	38.14	22.14	28.02	38.14	22.14	28.02	38.14	22.14	28.02
	KEYBERT_Word2vec	21.43	13.14	16.29	21.43	13.14	16.29	21.43	13.14	16.29

5.1.1 Direct Matching Evaluation Method

Figure 17 illustrates the variation of precision, recall, and F1-score across the FULL, LEAF, and TOP annotation approaches under the direct matching evaluation method.

- Analyzing the mean precision (P%) values for each annotation approach, the highest precision is achieved by the KEYBERT_Cosine tool for both FULL (42.67%) and LEAF (32.14%) approaches, while for the TOP (78.57%) approach, YAKE_JAROW demonstrates the highest precision.
- Examining the mean recall (R%) values, Pattern_Word2Vec exhibits the highest recall for both FULL (45.47%) and LEAF (45.14%) approaches, whereas YAKE_ JAROW yields the highest recall for the TOP (37.57%) approach.
- Regarding the F1-score (F%) variation, YAKE_Cosine tool demonstrates the highest F1-score for FULL (17.26%), PATTERN_Cosine shows the highest value for LEAF (20.74%) approaches, while YAKE_JAROW performs the best for the TOP (50.83%) approach.

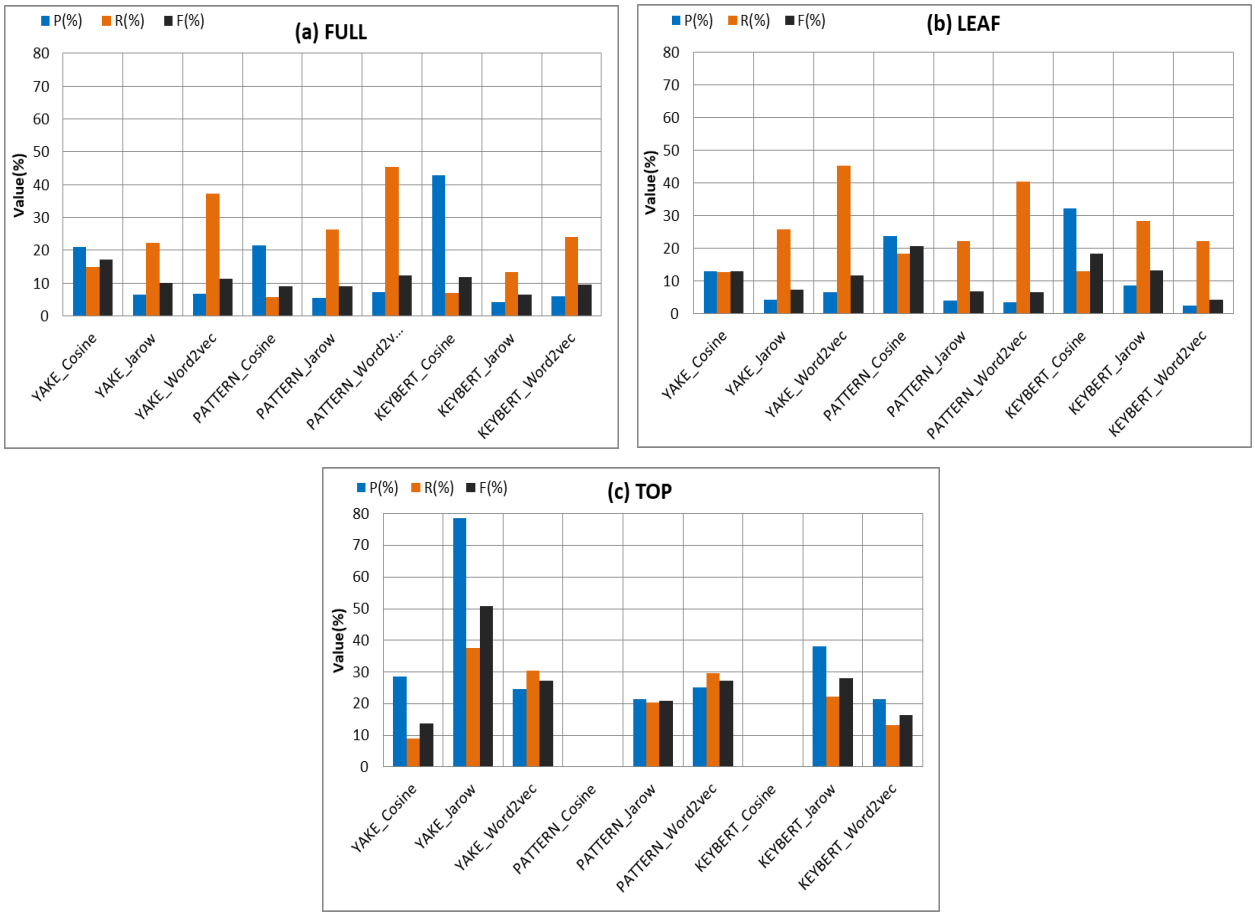


Figure 17: P (%), R (%) and F (%) values for NLP based tool employed for direct matching evaluation method for (a) FULL (b) LEAF and (c) TOP approaches

The overall performance for all annotation approaches in the direct matching evaluation method is shown in Figure 18.

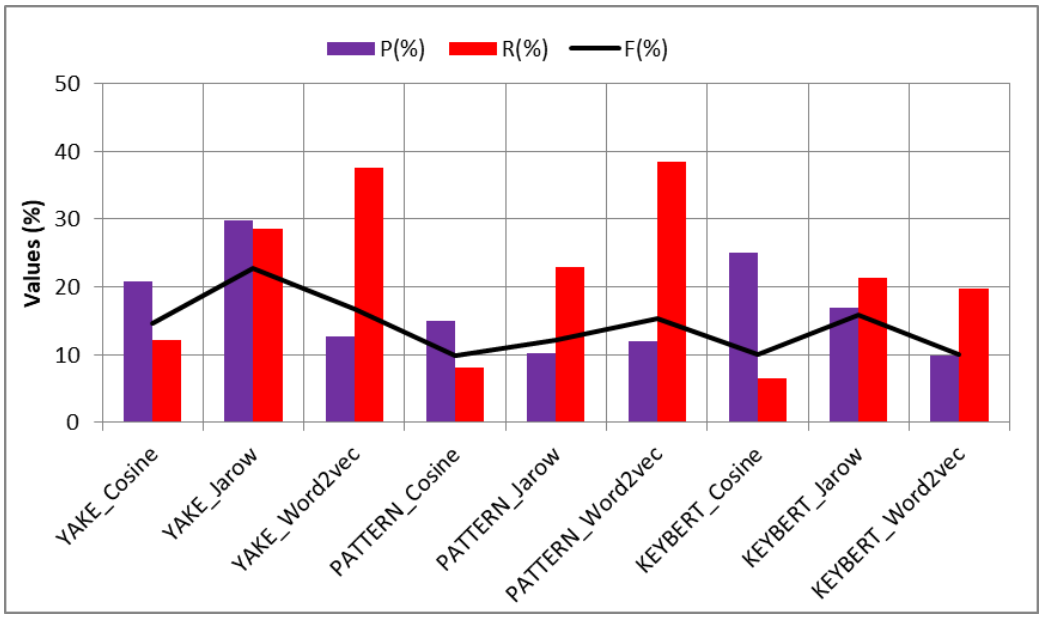


Figure 18: The overall performance for NLP based tool employed for direct evaluation method

For overall performance for direct matching evaluation method, starting with precision, the highest value of 29.76% is achieved by the YAKE_JaroW tool. Moving on to recall, the highest value of 38.49% is obtained by the Pattern_Word2Vec tool. Finally, in terms of F1-score, which provides a balance between precision and recall, the highest value of 22.72% is achieved by YAKE_Jarow. Parent-Child Matching Evaluation Method

5.1.2 Parent-child Matching Evaluation Method

Figure 19 illustrates the variation of precision, recall, and F1-score across the FULL, LEAF, and TOP annotation approaches under the parent-child matching evaluation method.

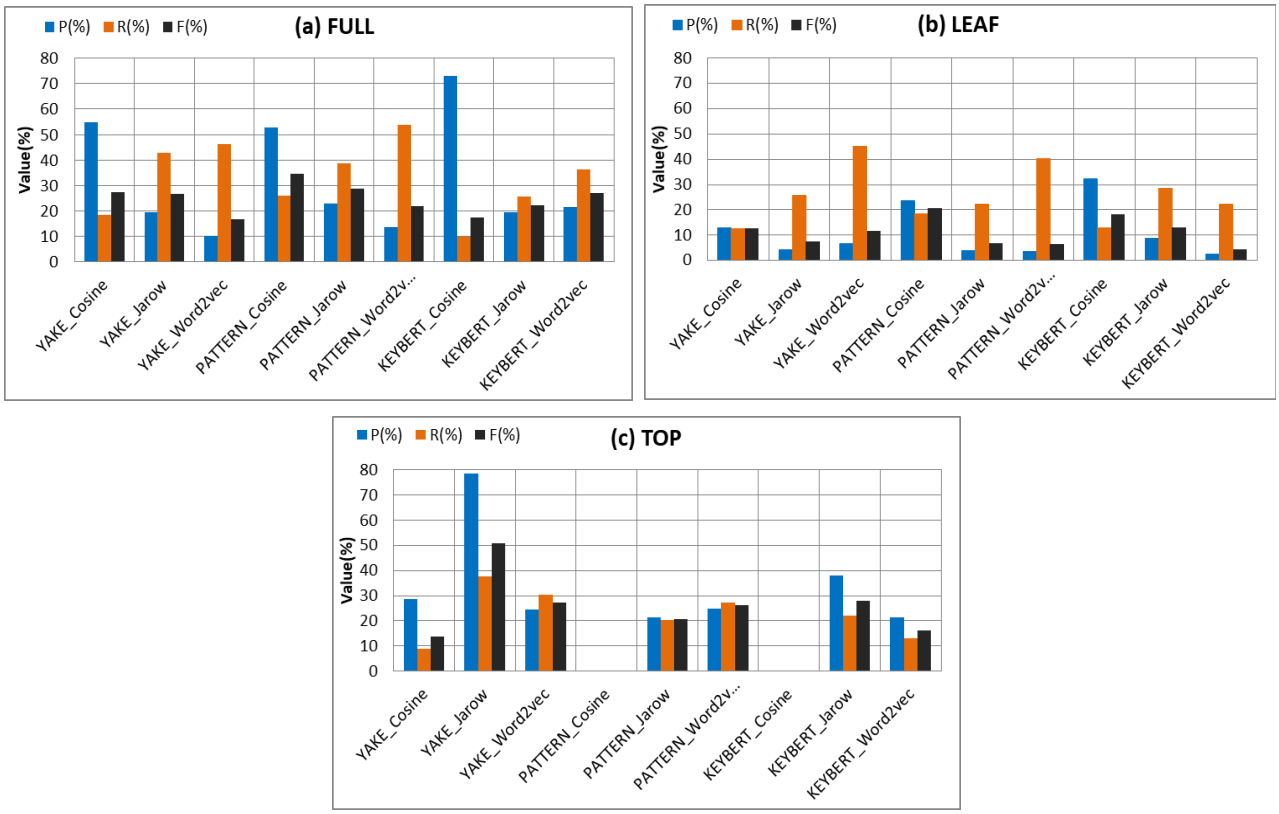


Figure 19: P (%), R (%) and F (%) values for NLP based tool employed for parent- child matching evaluation method for (a) FULL (b) LEAF and (c) TOP approaches.

- Analyzing the mean precision (P%) values for each approach, the highest precision is achieved by the KEYBERT_Cosine tool for both FULL (73%) and LEAF (45%) approaches, while for the TOP (78.57%) approach, YAKE_JAROW tool demonstrates the highest precision.

- Examining the mean recall (R%) values, Pattern_Word2Vec tool exhibits the highest recall for both FULL (53.93%) and LEAF (20.74%) approaches, whereas YAKE_JaroW tool yields the highest recall for the LEAF (50.83%) approach.
- Regarding the F1-score (F%) variation, the PATTERN_Cosine tool demonstrates the highest F1-scores for FULL (34.71%) and LEAF (20.74%) approach, while YAKE_JAROW for TOP (50.83%) approach.

The overall performance for all annotation approaches in parent-child matching evaluation method is shown in Figure 20.

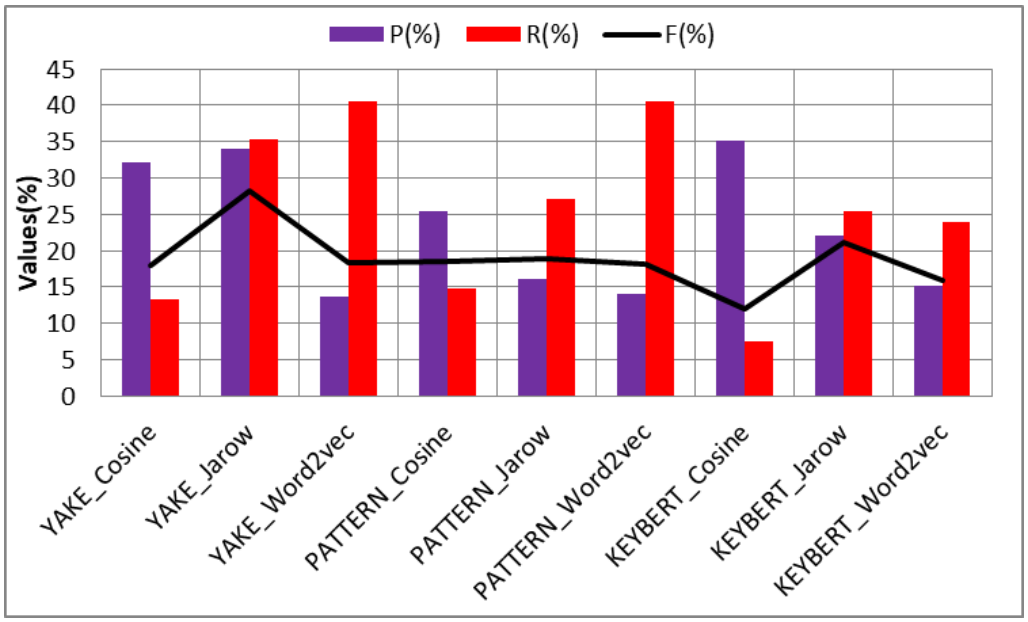


Figure 20: The overall performance for NLP based tool employed for parent-child matching evaluation method

Like for the direct evaluation method, for precision, the highest value of 35.05% is achieved by the KEYBERT_Cosine tool. Moving on to recall, the highest value of 40.6% is obtained by the PATTERN_Word2Vec tool. Finally, in terms of F1-score, which provides a balance between precision and recall, the highest value of 28.28% is achieved by YAKE_Jarow tool.

5.1.3 Ranking-based Evaluation Method

Figure 21 illustrates the variation of precision, recall, and F1-score across the FULL, LEAF, and TOP annotation approaches under the ranking-based evaluation method.

- Analyzing the mean precision (P%) values for each approach, the highest precision is achieved by the KEYBERT_Cosine tool for both FULL (42.76%) and

LEAF (32.14%) approaches, while for the TOP (78.57%) approach, YAKE_JAROW tool demonstrates the highest precision.

- Examining the mean recall (R%) values, PATTERN_Word2Vec tool exhibits the highest recall for both FULL (37.04%) and LEAF (33.29%) approaches, whereas YAKE_JaroW yields the highest recall for the TOP (37.57%) approach.
- Regarding the F1-score (F) variation, the KeyBert_Word2Vec tool demonstrates the highest F1-score for FULL (22.11%), KETBERT_Cosine for LEAF (18.32%) and YAKE_JaroW for TOP (50.83%) approach.

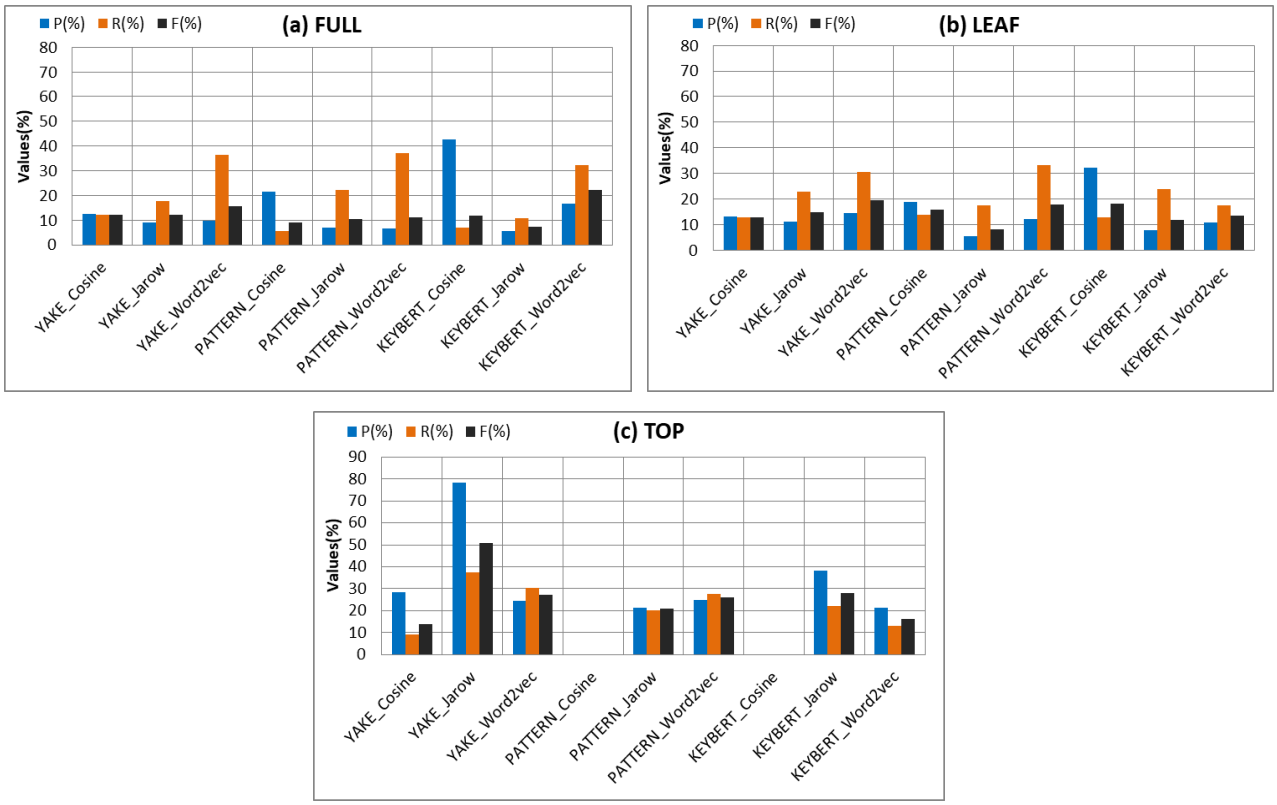


Figure 21: P (%), R (%) and F (%) values for NLP based tool employed for ranking-based evaluation method for (a) FULL (b) LEAF and (c) TOP approaches.

The overall performance for all annotation approaches in ranking-based evaluation method is shown in Figure 22.

In this evaluation approach, precision, the highest value of 32.89% is achieved by the YAKE_JaroW tool. Moving on to recall, the highest value of 38.49% is obtained by the PATTERN_Word2Vec tool. Finally, in terms of F1-score, which provides a balance between precision and recall, the highest value of 22.72% is achieved by YAKE_Jarow tool.

.

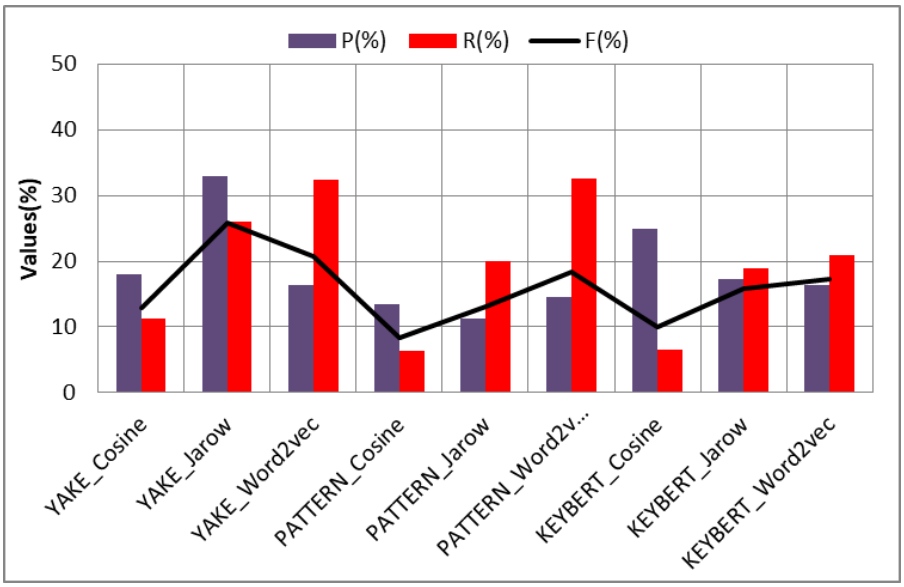


Figure 22: The overall performance for NLP based tool employed for ranking evaluation method

Discussion

According to the overall results given by the proposed NLP-based tools, YAKE_JaroW tool consistently achieves the highest precision values across direct matching (29.76%) and ranking-based evaluation method (32.89%). Also, for parent-child matching approach KEYBERT_Cosine tool achieves the highest precision value (35.05%).

This trend suggests that YAKE_JaroW tool is highly effective in accurately identifying and annotating relevant concepts within the text across different evaluation approaches. The consistent high precision values imply that YAKE_JaroW and KEYBERT_Cosine tools minimize false positives, ensuring that the identified concepts are relevant to the domain or topic being studied. The comparatively higher precision value obtained in the parent-child evaluation method may be attributed to the specific characteristics or criteria of this evaluation approach, which potentially favor a more refined assessment of precision.

On the other hand, PATTERN_Word2Vec tool consistently obtains the highest recall values across all three evaluation methods. For direct evaluation, PATTERN_Word2Vec tool achieves a recall value of 38.49%, for ranking evaluation it achieves 32.59%, and for parent-child evaluation, it attains the highest recall value of 40.60%.

This trend suggests that PATTERN_Word2Vec tool is successful in capturing a larger proportion of relevant concepts present in the text across different evaluation approaches. The consistently high recall values indicate that PATTERN_Word2Vec tool minimizes false negatives, ensuring that a comprehensive set of relevant concepts is identified and annotated. The comparatively higher recall value obtained in the parent-child evaluation method further underscores the effectiveness of this approach in capturing a broader range of relevant concepts. Overall, these findings highlight the strengths of YAKE_JaroW and KEYBERT_Cosine tools in precision-focused tasks and PATTERN_Word2Vec tool in recall-focused tasks across various evaluation methods.

5.1.4 Overall Performance

To evaluate the overall performance, F1-score values are considered because it serves as a comprehensive metric that accounts for both precision and recall, providing a holistic view of a tool's effectiveness in identifying relevant concepts within the text. Figure 23 presents the overall F1-score variations for each evaluation method, highlighting that **YAKE_JAROW** tool achieves the highest values among other NLP=based tools. Moreover, it illustrates that the F1-score values in the parent-child method (28.28%) are comparatively higher than those in the ranking (25.90%) and direct evaluation methods (22.72%), respectively. Furthermore, the results show that the F1-score values are comparatively higher in the parent-child matching method compared to the ranking-based and direct matching evaluation methods.

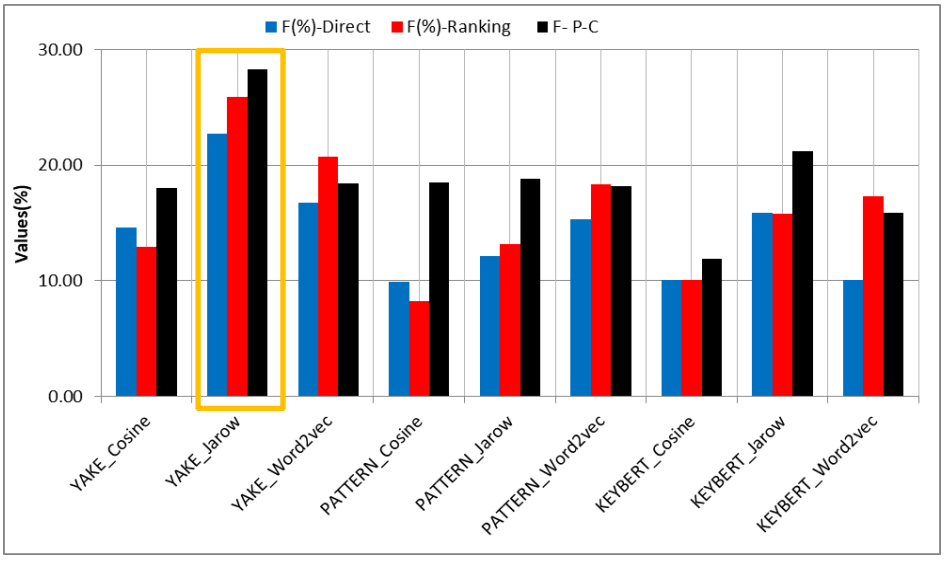


Figure 23: The overall F1-score variation for proposed NLP based tool employed for each evaluation method

Discussion

To select the best NLP-based tool, the overall performance is assessed comprehensively. By considering the F1 score, it shows how well a tool balances the need for accurate identification of relevant concepts (precision) with the need to capture a comprehensive set of such concepts (recall).

In this study, the choice of the best-performing tool is based on its ability to achieve the highest F1 score across different evaluation methods. The tool that consistently attains the highest F1 score demonstrates strong performance in both precision and recall, indicating its effectiveness in accurately identifying relevant concepts while minimizing both false positives and false negatives. **YAKE_JaroW** emerges as the best-performing tool overall, as it consistently achieves the highest F1-score across different evaluation methods. The observed differences in F1-score values across evaluation methods highlight the importance of selecting the most suitable evaluation approach based on the specific requirements and characteristics of the task.

The **parent-child** matching evaluation method stands out for higher F1-score values, than other evaluation approaches. It implies that the parent-child matching evaluation method may provide a more conducive environment for achieving a balance between precision and recall.

5.2 Results for Matching Percentages

In addition to evaluating the parameters for finding performance levels, matching percentages for each NLP-based tool in various evaluation and annotation approaches were analyzed for a total of 8 Agile papers. This analysis aimed to identify cases where author annotations aligned with annotations given by proposed NLP-based tools and where they diverged. It also helps to quantify the number of author annotations that matched well or did not match. It's important to note that these results were not influenced by the selection of the best-performing tools and only highlight significant cases which perfectly match or not.

In this analysis, all data were collected during annotation process. Number of matching concepts given by 09 NLP-based tools align with user annotation concepts were recorded. Then it is converted as a percentage value shown in equation 04.

$$Matching\ Percentage\ (\%) = \frac{Number\ of\ BoK\ concepts\ that\ are\ found\ in\ both\ the\ user\ annotation\ and\ the\ tool\ annotation}{Total\ number\ of\ BoK\ concepts\ given\ by\ user\ annotation} * 100\%$$

For example, if matching percentage is 5/8 means, out of total 08 BoK user annotation concepts, 05 concepts are match between user and tool annotations. According to the above equation all data were recorded and shown in table3. According to the values of table 3, some NLP-based tools were obtaining more than 50% matching percentage values, some are 100%, and some are 0%. The following describes some significant use cases in matching percentages and their reasons for selected Agile paper.

PDF 01 (P1)- "Land use influence on ambient PM2.5 and ammonia concentrations: Correlation analyses in the Lombardy region, Italy"

User annotation BoK Concepts (15) using BAT for FULL Approach:

- TA- Thematic and application domains
- TA13-1-1- Monitor the atmosphere
- TA13-5-1- Monitor urban areas
- TA13-2-2- Monitor health
- TA14-2-2-1-1- Land cover maps
- TA14-2-3- EO-derived attribute products
- IP3-11- Time series analysis
- GD2-2- Remote sensing
- GD2- Data Collection
- GC4- Data Quality, Metadata and Data Infrastructure
- GC1-3- Spatio-temporal problems and applications
- GD- Geospatial Data
- TA12-7- EO for health surveillance
- GS3-4- Use of geospatial information in environmental issues
- GS3-3-Use of geospatial information in research and education
- GS3-4- Use of geospatial information in environmental issues
- GS3-3-Use of geospatial information in research and education

Table 3: Calculated matching percentages (%) for each PDF documents for each evaluation and annotation stages

PDF	NLP Tool	Direct Matching			Ranking			Parent-Child Matching		
		FULL (%)	LEAF(%)	TOP(%)	FULL (%)	LEAF(%)	TOP(%)	FULL (%)	LEAF(%)	TOP(%)
P 01	YAKE_Cosine	53.3	0	0	53.3	0	0	20	0	0
	YAKE_JaroW	13.3	0	33.3	13.3	0	50	26.7	0	33.3
	YAKE_Word2	50	33.3	0	50	50	0	50	33.3	33.3
	Pattern_Cosine	0	0	0	0	0	0	6.7	0	0
	Pattern_JaroW	6.7	0	0	6.7	0	0	20	0	0
	Pattern_Word2	40	33.3	0	40	25	0	53.3	33.3	0
	KeyBert_Cosine	0	0	0	0	0	0	6.67	0	0
	KeyBert_JaroW	6.7	16.7	0	6.7	16.7	50	33.3	16.7	0
	KeyBert_Word2	6.7	0	0	6.7	0	0	33.3	0	0
P 02	YAKE_Cosine	11.1	25	33.3	11.1	25	33.3	11.1	25	33.3
	YAKE_JaroW	33.3	25	33.3	11.1	25	33.3	44.4	25	33.3
	YAKE_Word2	66.7	25	66.7	11.1	25	66.7	44.4	25	66.7
	Pattern_Cosine	11.1	25	0	11.1	25	0	22.2	25	0
	Pattern_JaroW	33.3	25	66.7	33.3	25	66.7	66.7	25	66.7
	Pattern_Word2	55.6	25	66.7	11.1	25	66.7	77.8	25	66.7
	KeyBert_Cosine	11.1	25	0	11.1	25	0	11.1	25	0
	KeyBert_JaroW	11.1	25	0	11.1	25	0	55.6	25	0
	KeyBert_Word2	33.3	25	66.7	11.1	25	66.7	55.6	25	66.7
P 03	YAKE_Cosine	0	0	0	0	0	0	9.5	0	0
	YAKE_JaroW	23.8	0	33.3	23.8	0	33.3	38	0	33.3
	YAKE_Word2	28.6	0	33.3	28.6	0	33.3	28.6	0	33.3
	Pattern_Cosine	0	0	0	0	0	0	9.5	0	0
	Pattern_JaroW	9.5	0	0	9.5	0	0	9.5	0	0
	Pattern_Word2	33.3	0	0	33	0	0	33.3	0	0
	KeyBert_Cosine	0	0	0	0	0	0	0	0	0
	KeyBert_JaroW	4.8	0	0	4.8	0	0	4.8	0	0
	KeyBert_Word2	23.8	0	0	23.8	0	0	23.8	0	0
P 04	YAKE_Cosine	0	0	0	0	0		0	0	0
	YAKE_JaroW	11.1	25	0	11.1	25	0	44.4	25	0
	YAKE_Word2	11.1	0	0	11.1	0	0	44.4	0	0
	Pattern_Cosine	0	0	0	0	0	0	0	0	0
	Pattern_JaroW	22.2	25	0	22.2	25	0	66.7	25	0
	Pattern_Word2	22.2	25	0	22.2	25	0	55.6	25	0
	KeyBert_Cosine	0	0	0	0	0	0	0	0	0
	KeyBert_JaroW	33.3	25	0	33.3	25	0	55.6	25	0
	KeyBert_Word2	0	0	0	0	0	0	44.4	0	0
P 05	YAKE_Cosine	7.7	0	25	0	0	25	15.4	0	25
	YAKE_JaroW	30.8	50	50	30.8	25	50	30.8	50	50
	YAKE_Word2	69.2	50	0	84.6	50	0	76.9	50	0
	Pattern_Cosine	7.7	25	0	0	0	0	7.7	25	0
	Pattern_JaroW	15.4	25	25	15.4	0	25	46	25	25
	Pattern_Word2	41.6	25	50	25	50	50	76.9	25	50
	KeyBert_Cosine	7.7	0	0	7.7	0	0	7.7	0	0
	KeyBert_JaroW	38.5	25	0	38.5	0	0	38.5	25	0
	KeyBert_Word2	23	50	0	30.8	25	0	69.2	50	0
P 06	YAKE_Cosine	16.7	33.3	0	16.7	33.3	0	22.2	33.3	0
	YAKE_JaroW	33.3	33.3	50	66.7	33.3	50	66.7	33.3	50
	YAKE_Word2	22.2	33.3	50	33.3	22.2	0	55.6	33.3	50
	Pattern_Cosine	11.1	33.3	0	11.1	33.3	0	22.2	33.3	0
	Pattern_JaroW	33.3	33.3	50	33.3	33.3	50	33.3	33.3	50
	Pattern_Word2	55.6	33.3	50	55.5	33.3	50	33.3	33.3	50
	KeyBert_Cosine	22.2	33.3	0	22.2	33.3	0	11.1	33.3	0
	KeyBert_JaroW	55.6	33.3	100	33.3	33.3	100	22.2	33.3	100
	KeyBert_Word2	44.4	33.3	0	44.4	33.3	0	44.4	33.3	0
P 07	YAKE_Cosine	0	0	0	0	0	0	16.7	0	0
	YAKE_JaroW	16.7	20	50	16.7	33.3	50	16.7	20	50
	YAKE_Word2	41.7	40	0	41.7	40	0	66.7	40	0
	Pattern_Cosine	0	0	0	0	0	0	0	0	0
	Pattern_JaroW	33.3	20	0	33.3	20	0	66.7	20	0
	Pattern_Word2	0	40	25	0	40	25	0	40	25
	KeyBert_Cosine	0	0	0	0	0	0	0	0	0
	KeyBert_JaroW	25	20	25	25	20	25	41.7	20	25
	KeyBert_Word2	16.7	0	25	16.7	0	25	41.7	0	25
P 08	YAKE_Cosine	4.1	4.3	0	4.1	4.3	0	10.2	4.3	0
	YAKE_JaroW	12.2	13	0	12.2	13	0	24.5	13	0
	YAKE_Word2	14.3	17.4	0	14.3	17.4	0	20.4	17.4	0
	Pattern_Cosine	8.2	8.7	0	8.2	8.7	0	10.2	8.7	0
	Pattern_JaroW	8.2	13	0	8.2	13	0	16.1	13	0
	Pattern_Word2	32.7	17.4	0	40	17.4	0	32.7	17.4	0
	KeyBert_Cosine	4.1	7.6	0	4.1	7.6	0	6.1	7.6	0
	KeyBert_JaroW	6.1	13	0	6.1	13	0	6.1	13	0
	KeyBert_Word2	8.1	4.3	0	8.1	4.3	0	8.1	4.3	0

The following table 4 shows the matching concepts between BAT tool and YAKE_Word2Vec tool under FULL, LEAF and TOP approaches for P1 document. According to the data, matching percentages for FULL, LEAF and TOP approaches are 50%, 33.3% and 0% respectively.

Table 4: Annotation results given for P01 by YAKE_Word2Vec tool and BAT tool for all annotation approaches

FULL		LEAF		TOP	
BAT	**NLP Tool**	**BAT**	**NLP Tool**	**BAT**	**NLP Tool**
TA	TA13-1	TA13-1-1	GS3-4	TA	-
TA13-1	TA13-5-1	TA13-5-1	GS3-3	GD	-
TA13-5-1	TA13-2-2	TA14-2-3		GC	-
TA13-2-2	IP3-11	GS3-4			
TA14-2-2-1-1	GD2-2	GS3-3			
TA14-2-3	GC4	TA12-7			
IP3-11	GS3-4				
GD2-2	GS3-3				
GD2					
GC4					
GC1-3					
GD					
TA12-7					
GS3-4					
GS3-3					
GC					

Here in FULL approach matching percentage gives 50% because half of the BoK concepts given by BAT tool are given by the NLP-based tool (YAKE_Word2Vec). But for LEAF approach percentage value gives less than 50%, the reason for this less value is some of LEAF concepts given by NLP tools are not matching BAT concepts. It is the same for the LEAF approach.

During annotation with NLP tools, there's a particular case where two concepts provided by the BAT and the NLP-tool used may not directly match. However, some of these concepts could still be related as they belong to the same branch (i.e 'Machine learning' and 'Artificial Neural Networks' are not matching but those are in same branch). This situation can significantly impact the matching percentage. Because of this case some NLP tools give 0% for matching percentage in FULL approach.

The following table 5 shows the number of NLP-based tools which have more than 50% matching for each PDF document.

Table 5: Number of NLP-based tools which have more than 50% matching for each PDF document.

PDF	Direct Matching			Ranking			Parent-Child Matching		
	FULL	LEAF	TOP	FULL	LEAF	TOP	FULL	LEAF	TOP
P1	**2**	0	1	1	1	2	**2**	0	0
P2	2	0	4	0	0	4	4	0	4
P3	0	0	0	0	0	0	0	0	0
P4	0	0	0	0	0	0	3	0	0
P5	1	2	2	1	2	2	3	3	2
P6	2	0	5	2	0	4	2	0	5
P7	0	0	1	0	0	1	2	0	1
P8	0	0	0	0	0	0	0	0	0

Figure 24 shows the graphical representation of number of NLP-based tools for each PDF document which have matching percentage more than 50% under different evaluation and annotation approaches.

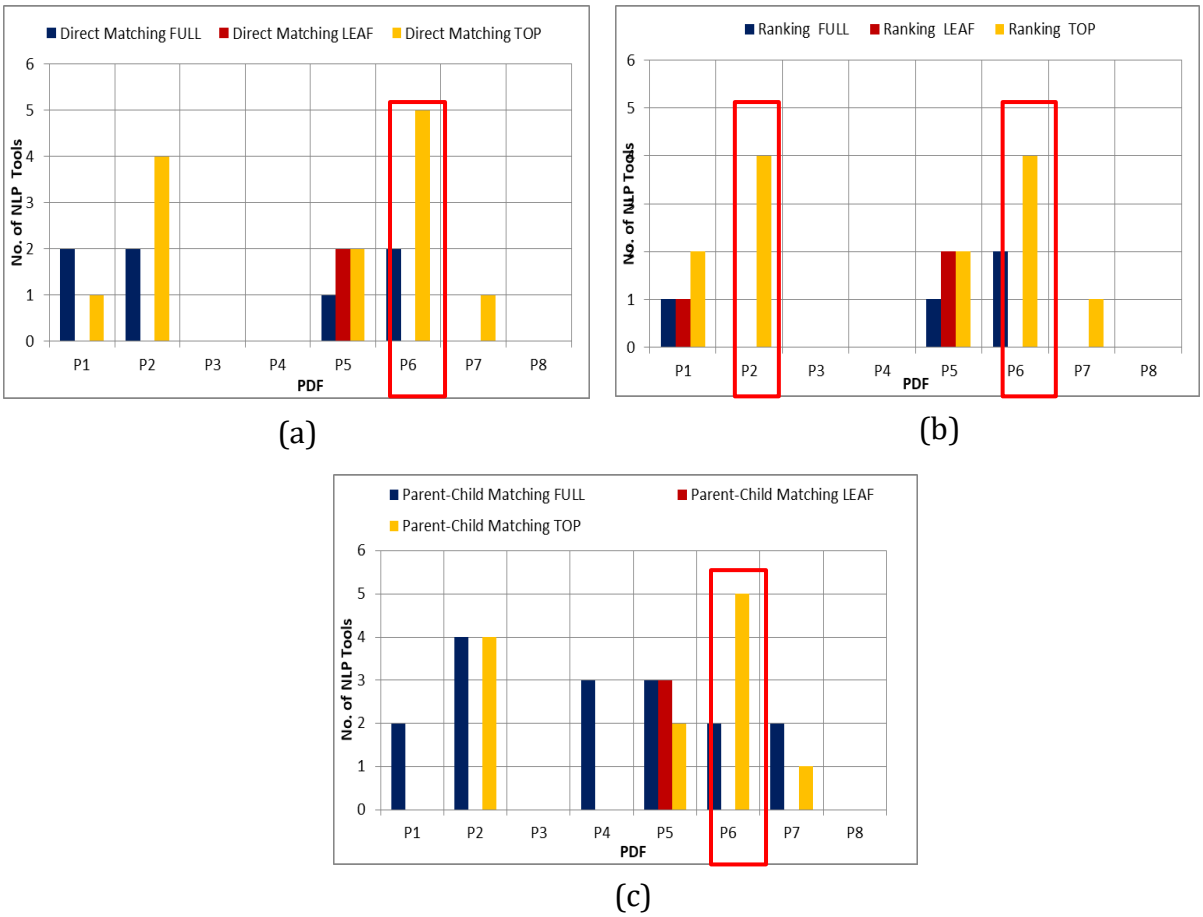


Figure 24: Number of NLP-based tools for each PDF document which having matching percentage more than 50% (a)Direct Matching (b) Ranking-based (c) Parent-child matching

In direct matching evaluation approach, the P6 document indicates that among the NLP tools assessed, five of them exhibit more than a 50% matching rate with the annotations provided by the author. That means, when considering document 6, these five NLP-based tools demonstrate a significant alignment with the user annotations which is same for the parent-child matching evaluation approach. In ranking-based evaluation approach P6 and P2 documents have more than 50% matching. This suggests that these tools are effective in accurately aligning with the user's annotations.

Discussion

Based on the provided results, can identify strengths and weaknesses of the proposed NLP-based tools in aligning with user annotations:

Some NLP-based tools demonstrate matching percentages above 50%, indicating a strong alignment with user annotations (BAT). This suggests that these tools are effective in accurately identifying concepts relevant to the user annotations. Certain tools perform well in specific evaluation approaches. This indicates that these tools may have strengths in particular types of annotation tasks or data sets.

Several tools have matching percentages below 50%, indicating a poor alignment with user annotations in certain cases. This suggests that these tools may struggle to accurately identify relevant concepts or may produce outputs that diverge significantly from the user's annotations. Some tools may perform well in one evaluation approach but poorly in others. This inconsistency suggests limitations in the tool's ability to adapt to different evaluation criteria or data sets. The presence of mismatched concepts, where concepts identified by the tool do not directly match user annotations, indicates a weakness in concept recognition or understanding. Tools that consistently exhibit low matching percentages across multiple documents may have limitations in scope or coverage, failing to capture a broad range of concepts relevant to user annotations.

CHAPTER 06

6 CONCLUSION

6.1 Conclusion

The study addresses a significant gap in the annotation of resources with EO4GEO Body of Knowledge (BoK) concepts by proposing NLP-based tools. After understanding the specific needs and challenges of annotating with EO4GEO BoK concepts, NLP-based tools were designed and developed for annotating with EO4GEO BoK concepts. This involves implementing the identified functionalities into the tools and ensuring their compatibility with the BoK framework. Furthermore, the study presents the proposed NLP-based tools as a web application for user-friendly accessibility. The evaluation of performance of the developed tools and assesses the results to ensure the effectiveness of the annotation process. This involves measuring the efficiency of the tools in annotating resources with EO4GEO BoK concepts. This study covers all the research questions mentioned in section 1.2.

Which NLP-based tools are suitable for extracting key EO/GI knowledge from text?

To achieve the research gap and to fulfill the identified functionalities, 03 key phrase extraction (YAKE, PatternRank, KeyBert) and 04 similarity measures techniques (Cosine, Jarow-Wrinkler, LSA, Word2Vec) in NLP were used to develop 12 proposed NLP-based tools.

How can the identified NLP-based tools be used to associate EO/GI knowledge, in terms of EO4GEO BoK concepts, with text documents?

The identified NLP-based tools can be utilized to associate EO/GI knowledge, specifically in terms of EO4GEO BoK concepts, these proposed tools are (semi)-automating the annotatio. Proposed NLP-tool, first preprocess the text document, then extract important key phrases from the text, extraction of EO4GEO BoK concepts and compare the extracted key phrase from the text documents with the concepts defined in the EO4GEO BoK. Then matching concepts are given with matching scores.

How do the identified NLP-based tools perform in extracting and associating EO/GI knowledge with text documents?

To analysis the performance of each proposed NLP-based tool, experimental evaluation was done with different annotation approaches (FULL, LEAF, TOP) and evaluation methods (direct, parent-child, ranking). Through an experimental evaluation, YAKE_JaroW tool emerged as the most efficient tool, consistently demonstrating good-performance values across different evaluation methods. The identification of YAKE_JaroW as the top-performing tool underscores its potential to significantly enhance the annotation process, facilitating the accurate association of relevant EO4GEO BoK concepts with textual resources.

Furthermore, the adoption of the parent-child matching evaluation method stands out as a key finding of this study. By leveraging its hierarchical structure, this evaluation approach offers enhanced efficiency in assessing the performance of NLP-based tools. The parent-child method optimally balances precision and recall, providing more value to tool effectiveness compared to other evaluation methods.

Overall, the findings of this study contribute valuable insights into the development and evaluation of NLP-based tools for annotating resources with EO4GEO BoK concepts. The identification of YAKE_JaroW as the best-performing tool, coupled with the recommendation of the parent-child matching evaluation method, offers practical guidance for researchers and practitioners in the field. Moving forward, these findings pave the way for the refinement and implementation of more efficient and accurate annotation processes, thereby advancing knowledge discovery and decision-making in geospatial domains.

How can the identified NLP-based tools be made available to the community of EO/GI researchers and practitioners?

The study presents the proposed NLP-based tools as a web application for user-friendly accessibility. This aims to make the tools readily available and easy to use for practitioners and researchers working with EO4GEO BoK concepts.

6.2 Limitations

One significant limitation encountered during the evaluation stage pertains to the annotation results provided by users. Initially, it is challenging to determine the exact number of annotations required from users, and this can lead to variability in the quantity of annotations provided. In some cases, users may offer only a minimal number of annotations, such as 1 or 2, which can significantly impact the evaluation of the tool's performance.

This limitation arises because the effectiveness of the tool is often assessed based on its ability to generate annotations that align closely with those provided by users (BAT). When users provide a limited number of annotations, it restricts the scope of the evaluation and can skew the results. For instance, if a user provides only a few annotations while the proposed tool generates a higher number of annotations, it may affect the evaluation outcomes. Additionally, the smaller number of annotations provided by users can introduce bias into the evaluation process. It may unfairly affect the tool's performance, even if it accurately identifies relevant concepts. Also, if two concepts are not matching but they related to same branch of the BoK makes causes to reduce the performance values.

Moreover, the quality of annotations provided by users can vary, further complicating the evaluation process. Annotations that lack specificity or relevance may not accurately reflect the user's true understanding of the domain, leading to inaccuracies in the assessment of the tool's performance.

6.3 Future Works

In future experiments, the scope of the study will be broadened by utilizing a larger dataset comprising articles related to the EO*GI domain from various sources, rather than limiting it to specific Agile papers. This expansion should lead to more diverse results and a better understanding of the domain.

Moreover, while the current study employed unsupervised approaches for key phrase extraction, the potential benefits of supervised approaches, particularly those involving deep learning, will be explored. This exploration aims to determine whether employing

supervised methods could lead to more accurate recommendations for annotating with EO4GEO BoK concepts.

Furthermore, experiments will be conducted varying the number of key phrases extracted from each paper to determine the optimal number for key phrase generation. This approach aims to refine the methods and improve the quality of key phrase extraction in future studies.

It is also recommended to provide users with a good range to give annotation to obtain better results. Providing users with examples of well-annotated documents and encouraging them to provide enough annotations can help to give good performance results.

Also, besides using FULL, LEAF and TOP evaluation process, first sharing the annotated documents with the authors along with the automated annotations (with ranking scores) can be done. Then encourage them to compare the automated annotations with their understanding of the EO4GEO BOK concepts and identify any similarities and errors in the annotations. Ask them to verify whether the annotated concepts are correctly matched and compare both annotations results with ranking.

Lastly, the possibility of focusing only on the abstract portion of papers to develop tools for key phrase extraction will be explored. These future directions aim to enhance the robustness and effectiveness of research methodologies and tools in the field of EO*GI domain analysis.

7 Bibliography

Ahamed, I., Jahan, M., Tasnim, Z., Karim, T., Reza, S. M. S., & Hossain, D. A. (2021). Spell corrector for bangla language using norvig's algorithm and jaro-winkler distance. *Bulletin of Electrical Engineering and Informatics*, *10*(4), 1997–2005. https://doi.org/10.11591/EEI.V10I4.2410

Alami Merrouni, Z., Frikh, B., & Ouhbi, B. (2020). Automatic keyphrase extraction: a survey and trends. *Journal of Intelligent Information Systems*, *54*(2), 391–424. https://doi.org/10.1007/s10844-019-00558-9

Alenazi, S. R., Ahmad, K., & Olowolayemo, A. (2017). A review of similarity measurement for record duplication detection. *Proceedings of the 2017 6th International Conference on Electrical Engineering and Informatics: Sustainable Society Through Digital Innovation, ICEEI 2017*, *2017-November*, 1–6. https://doi.org/10.1109/ICEEI.2017.8312386

Amur, Z. H., Hooi, Y. K., Soomro, G. M., Bhanbhro, H., Karyem, S., & Sohu, N. (2023). Unlocking the Potential of Keyword Extraction: The Need for Access to High-Quality Datasets. *Applied Sciences (Switzerland)*, *13*(12). https://doi.org/10.3390/app13127228

Aseervatham, S. (2008). A local latent semantic analysis-based kernel for document similarities. *Proceedings of the International Joint Conference on Neural Networks*, 214–219. https://doi.org/10.1109/IJCNN.2008.4633792

Asudani, D. S., Nagwani, N. K., & Singh, P. (2023). Impact of word embedding models on text analytics in deep learning environment: a review. In *Artificial Intelligence Review* (Vol. 56, Issue 9). Springer Netherlands. https://doi.org/10.1007/s10462-023-10419-1

Campos, R., Mangaravite, V., Pasquali, A., Jorge, A., Nunes, C., & Jatowt, A. (2020). YAKE! Keyword extraction from single documents using multiple local features. *Information Sciences*, *509*(September 2019), 257–289. https://doi.org/10.1016/j.ins.2019.09.013

DiBiase, D., DeMers, M., Johnson, A., Kemp, K., Luck, A. T., Plewe, B., & Wentz, E. (2007). Introducing the first edition of geographic information science and technology body of knowledge. *Cartography and Geographic Information Science*, *34*(2), 113–120. https://doi.org/10.1559/152304007781002253

Dibiase, D., DeMers, M., Johnson, A., Kemp, K., Taylor Luck, A., Plewe, B., & Wentz , E. (2006). *Geographic information science and technology body of knowledge* (1st ed.). Association Of American Geographers.

Dupuis, R., Bourque, P., & Abran, A. (2003). SWEBOK guide: An overview of trial usages in the field of education. *Proceedings - Frontiers in Education Conference, FIE*, *3*(November), S3C19-S3C23. https://doi.org/10.1109/FIE.2003.1265987

Dubois, C., Jutzi, B., Olijslagers, M., Pathe, C., Schmullius, C., Stelmaszczuk-Górska, M. A., Vandenbroucke, D., & Weinmann, M. (2021). KNOWLEDGE and SKILLS RELATED to ACTIVE OPTICAL SENSORS in the BODY of KNOWLEDGE for EARTH OBSERVATION

and GEOINFORMATION (EO4GEO BOK). *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *5*(5), 9–16. https://doi.org/10.5194/isprs-annals-V-5-2021-9-2021

de Vos, I. M., den Boogerd, G. L., Fennema, M. D., & Correia, A. (2021). Comparing in context: Improving cosine similarity measures with a metric tensor. *Proceedings of the 18th International Conference on Natural Language Processing (ICON)*, 128–138. https://aclanthology.org/2021.icon-main.17

du Plessis, H., & van Niekerk, A. (2013). A New GISc Framework and Competency Set for Curricula Development at South African Universities. *South African Journal of Geomatics*, *3*(1), 1–12.

Efriyanto, T., & Hayaty, M. (2022). Jaro Winkler Algorithm for Measuring Similarity Online News. *Jurnal Teknik Informatika (JUTIF)*, *3*(4), 975–982. https://doi.org/10.20884/1.jutif.2022.3.4.152

EO4GEO Alliance. (2018). *BoK Matching Tool*. EO4GEO. http://www.eo4geo.eu/tools/bok-matching-tool/

EO4GEO Alliance. (2022). *Living Textbook*. EO4GEO. http://www.eo4geo.eu/tools/living-textbook/

EO4GEO Alliance. (2022). *BoK Annotation Tool*. EO4GEO. http://www.eo4geo.eu/tools/bok-annotation-tool/

EO4GEO Alliance. (2022). *BoK Visualization and Search.* EO4GEO. http://www.eo4geo.eu/tools/bok-visualization-and-search/

EvidentlyAI. (2024). *Accuracy, precision, and recall in multi-class classification*. Www.evidentlyai.com. Retrieved February 16, 2024, from https://www.evidentlyai.com/classification-metrics/multi-class-metrics#:~:text=Micro%2Daveraging%20gives%20equal%20weight%20to%20every%20instance%20and%20shows

Foltz, P. W. (1996). Latent semantic analysis for text-based research. *Behavior Research Methods, Instruments, and Computers*, *28*(2), 197–202. https://doi.org/10.3758/BF03204765

Gomaa, H. W., & Fahmy, A. A. (2013). A Survey of Text Similarity Approaches. *International Journal of Computer Applications*, *68*(13), 13–18. https://doi.org/10.5120/11638-7118

Hameed, N. H., Alimi, A. M., & Sadiq, A. T. (2022). Short Text Semantic Similarity Measurement Approach Based on Semantic Network. *Baghdad Science Journal*, *19*(6), 1581–1591. https://doi.org/10.21123/bsj.2022.7255

Hofer, B., Casteleyn, S., Aguilar-Moreno, E., Missoni-Steinbacher, E. M., Albrecht, F., Lemmens, R., Lang, S., Albrecht, J., Stelmaszczuk-Górska, M., Vancauwenberghe, G., & Monfort-Muriach, A. (2020). Complementing the European earth observation and geographic information body of knowledge with a business-oriented perspective.

In *Transactions in GIS* (Vol. 24, Issue 3, pp. 587–601). https://doi.org/10.1111/tgis.12628

Hu, J., Li, S., Yao, Y., Yu, L., Yang, G., & Hu, J. (2018). Patent keyword extraction algorithm based on distributed representation for patent classification. *Entropy*, *20*(2). https://doi.org/10.3390/e20020104

Ide, N. (2017). Handbook of Linguistic Annotation. *Handbook of Linguistic Annotation*, *June 2017*. https://doi.org/10.1007/978-94-024-0881-2

Imaduddin, H., Widyawan, & Fauziati, S. (2019). Word embedding comparison for Indonesian language sentiment analysis. *Proceeding - 2019 International Conference of Artificial Intelligence and Information Technology, ICAIIT 2019*, 426–430. https://doi.org/10.1109/ICAIIT.2019.8834536

Ikhwan Syafiq, M., Shukor Talib, M., Salim, N., Haron, H., & Alwee, R. (2019). A Concise Review of Named Entity Recognition System: Methods and Features. *IOP Conference Series: Materials Science and Engineering*, *551*(1). https://doi.org/10.1088/1757-899X/551/1/012052

Khan, M. Q., Shahid, A., Uddin, M. I., Roman, M., Alharbi, A., Alosaimi, W., Almalki, J., & Alshahrani, S. M. (2022). Impact analysis of keyword extraction using contextual word embedding. *PeerJ Computer Science*, *8*, 1–16. https://doi.org/10.7717/peerj-cs.967

Koloski, B., Pollak, S., Škrlj, B., & Martinc, M. (2022). Out of Thin Air: Is Zero-Shot Cross-Lingual Keyword Detection Better Than Unsupervised? *2022 Language Resources and Evaluation Conference, LREC 2022*, 400–409.

Lehal, M. S. (2017). *Comparison of Cosine , Euclidean Distance and Jaccard Distance*. *3*(8), 1376–1381.

Lemmens, R., Albrecht, F., Lang, S., Casteleyn, S., Stelmaszczuk-Górska, M., Olijslagers, M., Belgiu, M., Granell, C., Augustijn, E.-W., Pathe, C., Missoni-Steinbacher, E.-M., & Monfort Muriach, A. (2022). *Updating and using the EO4GEO Body of Knowledge for (AI) concept annotation*. AGILE: GIScience Series. https://doi.org/10.5194/agile-giss-3-44-2022

Li, Q., Peng, H., Li, J., Xia, C., Yang, R., Sun, L., Yu, P. S., & He, L. (2022). A Survey on Text Classification: From Traditional to Deep Learning. *ACM Transactions on Intelligent Systems and Technology*, *13*(2). https://doi.org/10.1145/3495162

Mahata, D., Kuriakose, J., Shah, R. R., & Zimmermann, R. (2018). Key2Vec: Automatic ranked keyphrase extraction from scientific articles using phrase embeddings. *NAACL HLT 2018 - 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, *2*, 634–639. https://doi.org/10.18653/v1/n18-2100

Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., & Gao, J. (2020). *Deep Learning Based Text Classification: A Comprehensive Review*. *1*(1), 1–43. http://arxiv.org/abs/2004.03705

Mocnik, F.B. (2023). Data and Coherence Theories of Truth – Examples From a Data-

Driven Geographical Information Science. *AGILE: GIScience Series*, *4*, 1–5. https://doi.org/10.5194/agile-giss-4-33-2023

Monfort, A., Moreno, E. A., & Casteleyn, S. (2020). *BoK Annotation Tool (BAT) EO4GEO Tools User guides*. *1*, 1–13.

NLTK. (2009). *Natural Language Toolkit — NLTK 3.4.4 documentation*. Nltk.org. https://www.nltk.org/

Neves, M., & Ševa, J. (2021). An extensive review of tools for manual annotation of documents. *Briefings in Bioinformatics*, *22*(1), 146–163. https://doi.org/10.1093/bib/bbz130

Otto, M. (2022). *Bootstrap*. Getbootstrap.com. https://getbootstrap.com/

Ouarda, L., Malika, B., & Brahim, B. (2023). Towards a better similarity algorithm for host-based intrusion detection system. *Journal of Intelligent Systems*, *32*(1), 1–18. https://doi.org/10.1515/jisys-2022-0259

Pallets. (2010). *Welcome to Flask — Flask Documentation (3.0.x)*. Flask.palletsprojects.com. https://flask.palletsprojects.com/en/3.0.x/

Panchal, D., Mehta, M., Mishra, A., Ghole, S., & Dandge, M. S. (2022). Sentiment Analysis Using Natural Language Processing. *International Journal for Research in Applied Science and Engineering Technology*, *10*(5), 2262–2266. https://doi.org/10.22214/ijraset.2022.42711

Papagiannopoulou, E., & Tsoumakas, G. (2020). A review of keyphrase extraction. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *10*(2), 1–59. https://doi.org/10.1002/widm.1339

Python Software Foundation. (2016, May 18). *PyPDF2*. PyPI. https://pypi.org/project/PyPDF2/

Rahutomo, F., Kitasuka, T., & Aritsugi, M. (2012). Semantic cosine similarity. *The 7th International Student Conference on Advanced Science and Technology ICAST*, *4*(1), 1. https://www.researchgate.net/publication/262525676

Rehbein, I., Ruppenhofer, J., & Sporleder, C. (2012). Is it worth the effort? Assessing the benefits of partial automatic pre-labeling for frame-semantic annotation. *Language Resources and Evaluation*, *46*(1), 1–23. https://www.jstor.org/stable/41486059

Ronzhin, S., Lemmens, R. L. G., Augustijn, P. W. M., Verkroost, M. J., & Walsh, N. (2018). *Space Education with The Living Textbook, A web-based tool using a Concept Browser*. *February 2022*. https://research.utwente.nl/en/publications/space-education-with-the-living-textbook-a-web-based-tool-using-a

Ronzhin, S., Folmer, E., & Lemmens, R. (2018). *Technological Aspects of (Linked) Open Data*. T.M.C. Asser Press. https://doi.org/10.1007/978-94-6265-261-3_9

Piskorski, J., Stefanovitch, N., Jacquet, G., & Podavini, A. (2021). Exploring Linguistically-Lightweight Keyword Extraction Techniques for Indexing News Articles in a

Multilingual Set-up. *EACL Hackashop on News Media Content Analysis and Automated Report Generation, Hackashop 2021 at 16th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2021 - Proceedings*, 35–44.

Schopf, T., Klimek, S., & Matthes, F. (2022). PatternRank: Leveraging Pretrained Language Models and Part of Speech for Unsupervised Keyphrase Extraction. *International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, IC3K - Proceedings*, *1*, 243–248. https://doi.org/10.5220/0011546600003335

Sitikhu, P., Pahi, K., Thapa, P., & Shakya, S. (2019). A Comparison of Semantic Similarity Methods for Maximum Human Interpretability. *International Conference on Artificial Intelligence for Transforming Business and Society, AITB 2019*. https://doi.org/10.1109/AITB48515.2019.8947433

Stelmaszczuk-Górska, M. A., Aguilar-Moreno, E., Casteleyn, S., Vandenbroucke, D., Miguel-Lago, M., Dubois, C., Lemmens, R., Vancauwenberghe, G., Olijslagers, M., Lang, S., Albrecht, F., Belgiu, M., Krieger, V., Jagdhuber, T., Fluhrer, A., Soja, M. J., Mouratidis, A., Persson, H. J., Colombo, R., & Masiello, G. (2020). BODY of KNOWLEDGE for the EARTH OBSERVATION and GEOINFORMATION SECTOR-A BASIS for INNOVATIVE SKILLS DEVELOPMENT. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, *43*(B5), 15–22. https://doi.org/10.5194/isprs-archives-XLIII-B5-2020-15-2020

Sun, C., Hu, L., Li, S., Li, T., Li, H., & Chi, L. (2020). A review of unsupervised keyphrase extraction methods using within-collection resources. *Symmetry*, *12*(11), 1–20. https://doi.org/10.3390/sym12111864

Tedeschi, S., Conia, S., Cecconi, F., & Navigli, R. (2021). Named Entity Recognition for Entity Linking: What Works and What's Next. *Findings of the Association for Computational Linguistics, Findings of ACL: EMNLP 2021*, 2584–2596. https://doi.org/10.18653/v1/2021.findings-emnlp.220

Tsvetkov, A., & Kipnis, A. (2023). *EntropyRank: Unsupervised Keyphrase Extraction via Side-Information Optimization for Language Model-based Text Compression*. http://arxiv.org/abs/2308.13399

University of Twente. (2021.). *Living Textbook | Home | By ITC, University of Twente*. Ltb.itc.utwente.nl. Retrieved January 30, 2024, from https://ltb.itc.utwente.nl/page/671/dashboard

Vandenbroucke, D., & Vancauwenberghe, G. (2016). Towards a new body of knowledge for geographic information science and technology. *Micro Macro Mezzo Geoinf*, *6*(March 2018), 7–19.

Vetriselvi, T., Albert Mayan, J., Priyadharshini, K. V., Sathyamoorthy, K., Venkata Lakshmi, S., & Vishnu Raja, P. (2022). Latent Semantic Based Fuzzy Kernel Support Vector Machine for Automatic Content Summarization. *Intelligent Automation and Soft Computing*, *34*(3), 1537–1551. https://doi.org/10.32604/iasc.2022.025235

Waters, N.M. (2013). The Geographic Information Science Body of Knowledge 2.0: Toward a New Federation of GIS Knowledge. *Communications in Computer and Information Science, vol 372*. https://link.springer.com/chapter/10.1007/978-3-642-38836-1_11

Wilson, J. P. (2016). *Geographic Information Science & Technology Body of Knowledge 2 . 0 project final report*. *January 2014*, 0–81.

Wang, R., Liu, W., & Mcdonald, C. (2014). *Unsupervised Keyphrase Extraction*. 163–176.

Yildiz, T. (2019). A comparative study of author gender identification. *Turkish Journal of Electrical Engineering and Computer Sciences*, *27*(2), 1052–1064. https://doi.org/10.3906/elk-1806-185

Masters Program in **Geospatial Technologies**

Supported by:
Education and Culture
ERASMUS MUNDUS