SIDDAGANGA INSTITUTE OF TECHNOLOGY, TUMAKURU- 3

(An Autonomous institution affiliated to Visvesvaraya Technological University- Belagavi, Approved by AICTE, Accredited by NAAC with 'A++' Grade, Awarded Diamond College Rating by QS I-GAUGE & ISO 9001:2015 certified)

MINI PROJECT REPORT

ON

"Mapping of CO's to PO's using NLP"

submitted in the partial fulfilment of the requirements for VI semester, Bachelor of Engineering in Computer Science and Engineering

By

PADMA C G	**1SI20CS070**
RAGHAVENDRA URS	**1SI20CS086**
YOGANANDINI	**1SI20CS138**

Under the guidance of

Dr. Srinivasa K Ph.D.

Assistant Professor, Department of CSE.

Department of Computer Science and Engineering

(Program Accredited by NBA)

Academic Year: 2022-23

Siddaganga Institute of Technology, Tumakuru-3

(An Autonomous institution affiliated to Visvesvaraya Technological University- Belagavi, Approved by AICTE, Accredited by NAAC with 'A++' Grade, Awarded Diamond College Rating by QS I-GAUGE & ISO 9001:2015 certified)

Department of Computer Science and Engineering

(Program Accredited by NBA)

CERTIFICATE

This is to certify that the mini project entitled "Campus Compass : A Flutter-based Interactive Map for Campus Navigation" is a bonafide work carried out by **Padma C G (1SI20CS070), Raghavendra Urs (1SI20CS086), Yoganandini J (1SI20CS138)** of VI semester **Computer Science and Engineering**, **SIDDAGANGA INSTITUTE OF TECHNOLOGY** during the academic year 2022-2023.

Signature of the Guide	**Signature of the Convener**
Dr. Srinivasa K Ph.D.	Prof. Thejaswini S M.Tech
Assistant Professor	**Assistant Professor**

Signature of the HOD

Dr. A S Poornima
Prof. and Head, Dept. of CSE

Name of the Examiners:	**Signature with Date**
1. **Prof.**	
2. **Prof.**	

ACKNOWLEDGEMENT

With great reverence, we express our sincere gratitude and salutations to his holiness **Dr. Sree Sree Shivakumara Swamigalu,** Founder of Sree Siddaganga Education society and **Sree Sree Siddalinga Swamigalu,** President of Sree Siddaganga Education society for their blessings.

First and foremost, we wish to express our deep sincere feelings of gratitude to our institution, **Siddaganga Institute of Technology,** for providing us an opportunity for completing our project successfully.

We are grateful to **Dr. M. N. Channabasappa,** Director and **Dr. Shivakumaraiah,** CEO, Siddaganga Institute of Technology-Tumakuru, for their kind cooperation and encouragement.

We express our kind thanks to **Dr. S V Dinesh,** Principal, Siddaganga Institute of Technology, Tumakuru for his encouragement towards students attitude.

We express our heartfelt thanks to **Dr. A S Poornima,** Professor and Head, Department of Computer Science and Engineering, Siddaganga Institute of Technology, Tumakuru for her suggestions and advice.

We express our gratitude and humble thanks to our project guide, **Dr. Srinivasa K Ph.D,** Assistant Professor, Department of Computer Science and Engineering, Siddaganga Institute of Technology, Tumakuru for guiding us to complete our Mini-project successfully.

We thank our mini project convener, **Mrs. Thejaswini S**, Assistant Professor, Department of Computer Science and Engineering, for allowing us to take up this idea as our Mini Project.

We are conscious of the fact that we have received cooperation in many ways from the **Teaching, Technical and supporting staff** of the Department of Computer Science and Engineering and we are grateful to all for their cooperation.

We express heartfelt gratitude to our **Parents and Friends** for their constant moral support and encouragement throughout this work.

ABSTRACT

In the context of educational programs, aligning course outcomes with program outcomes is crucial for ensuring the attainment of overarching educational objectives. This study presents a methodology for mapping course outcomes to program outcomes using Natural Language Processing (NLP) techniques. The mapping process involves assigning grades of 0, 1, or 2, indicating the level of match between the course and program outcomes. To determine the efficiency of the mapping process, a com parative analysis is conducted between manual mapping and the NLP-based cosine similarity approach. The manual mapping involves human experts reviewing and matching outcomes, while the NLP approach utilizes computational methods to calculate the similarity between outcomes based on their semantic content. Cosine similarity is employed as a measure of textual sim ilarity, considering the vector representation of outcomes. The NLP-based approach assigns a grade of 0 when no match is found, 1 when a single match is identified, and 2 when multiple matches are detected. The manual mapping results are used as a reference for evaluating the accuracy and effectiveness of the NLP-based approach. By comparing the outcomes obtained through both ap proaches, this study aims to assess the efficiency of NLP techniques in automating the mapping process. The results will provide insights into the advantages and limitations of using NLP for outcome mapping and inform educators and curriculum developers about the potential benefits of employing such automated methods. The findings of this research contribute to the field of education by offering a quantitative evaluation of the NLP-based approach for mapping course outcomes to program outcomes. Additionally, the study sheds light on the potential impact of adopting automated techniques in educational institutions, promoting ef ficient curriculum design and improving the alignment between individual courses and broader program goals. Index Terms—component, formatting, style, styling, insert

Table of Contents

CHAPTER 1

INTRODUCTION

Mapping course outcomes to program outcomes is a critical task in education, ensuring that individual courses align with the overarching objectives of an educational program. Traditionally, this mapping process has been performed manually, relying on the expertise of educators to assess the alignment
between outcomes. However, with the advancements in Natural Language Processing (NLP), there is an opportunity to automate and enhance this process.

In this study, we explore the use of NLP techniques, specifically cosine similarity, for mapping course outcomes to program outcomes. Cosine similarity measures the similarity between two vectors based on the angle between them, providing a quantitative measure of how closely related two sets of outcomes are in terms of their semantic content. To facilitate the mapping process, we utilize a grading
system that assigns values of 0, 1, or 2 to indicate the level of match found between course and program outcomes. A grade of 0 signifies no match, 1 denotes a single match, and 2 indicates multiple matches.

By employing NLP and cosine similarity, we aim to improve the efficiency and accuracy of the outcome mapping process. NLP techniques can analyze the textual representation of
outcomes, capturing subtle nuances and semantic relationships that may be missed through manual evaluation. The use of cosine similarity allows for a standardized and objective measure of similarity, providing a quantitative assessment of the alignment between course and program outcomes.

The objectives of this study are twofold: first, to evaluate the effectiveness of NLP-based mapping compared to manual mapping by human experts, and second, to determine the efficiency of the NLP approach in automating the outcome mapping process. We will compare the outcomes obtained
through both approaches, considering the level of matching indicated by the grades 0, 1, and 2.
The findings of this research will contribute to the field of education by providing insights into the advantages and limitations of using NLP techniques for mapping course outcomes to program outcomes. This study has the potential to inform educators, curriculum developers, and educational
institutions about the benefits of adopting automated methods for outcome alignment, ultimately improving the design and effectiveness of educational programs.

CHAPTER 2

LITERATURE SURVEY

Study	Focus	Method	Key Findings
Study1	Comparing the Performance with other popular NLP Libraries	spaCy	Spacy outperformed the other libraries on most of the tasks.
Study2	Sentence Detection	spaCy	Sentence detection allows you to divide a text into linguistically meaningful units.
Study3	Strength and Efficiency	spaCy	Spacy's strengths as its efficiency, ease of use, and support for multiple languages.
Study4	Next Sentence Predictions	BERT/Word Embeddings	Model receives pairs of sentences as input and learns to predict if the second sentence in the pair is the subsequent sentence in the original document.
Study5	Distributional approach	BERT/Word Embeddings	This can be useful for identifying related words and topics in a text, as words that appear in similar contexts will have similar vector representations.

Table 2.1 Literature Survey

CHAPTER 3

PROBLEM STATEMENT

- The alignment of course outcomes with program outcomes is a crucial aspect of curriculum design and assessment. It helps to ensure that the program is achieving its intended goals and that students are acquiring the necessary skills and knowledge. However, the manual process of mapping course outcomes to program outcomes can be time-consuming and subjective. In this study, we propose using Natural Language Processing (NLP) to automate this process.

- Using NLP, we aim to develop a system that can automatically map course outcomes to program outcomes. The system will analyze the language used in both the course and program outcomes and identify the most relevant keywords and concepts. It will then use this information to map the course outcomes to the program outcomes.We also aim is to provide a tool that can assist curriculum designers and educators in ensuringthat the courses offered are aligned with the overall goals and objectives of the program

OBJECTIVES

The project has the following specific objectives:

• Develop an NLP model that can accurately map course outcomes to program outcomes:

The model should be able to process large volumes of textual data and identify the most relevant concepts and keywords that indicate the relationship between the course outcomes and program outcomes.

• Evaluate the effectiveness of the NLP model in comparison to manual mapping:

The accuracy of the NLP model will be evaluated by comparing its results to those obtained from the manual mapping process, which is currently the most commonly used method for mapping course outcomes to program outcomes.

• Investigate the impact of using NLP model on the efficiency of curriculum design process:

The study will explore the potential benefits of using an NLP model in terms of reducing the time and effort required for curriculum design and assessment, as well as improving the accuracy and consistency of the mapping process.

• Assess the generalizability and scalability of the proposed approach:

The developed NLP model will be evaluated on a range of courses and programs from different disciplines to assess its ability to be generalized across different domains. The scalability of the model will also be examined to evaluate its ability to handle a large volume of data and support a wide range of educational programs.

Overall, the proposed project aims to provide new and innovative approach to curriculum design and assessment , which has the potential to significantly improve the efficiency, accuracy, and consistency of the process. The findings of the study will be valuable to educators, administrators, and researchers in the field of education, and will contribute to the advancement of the use NLP in educational application.

CHAPTER 4

SYSTEM DESIGN

HIGHLEVEL DESIGN

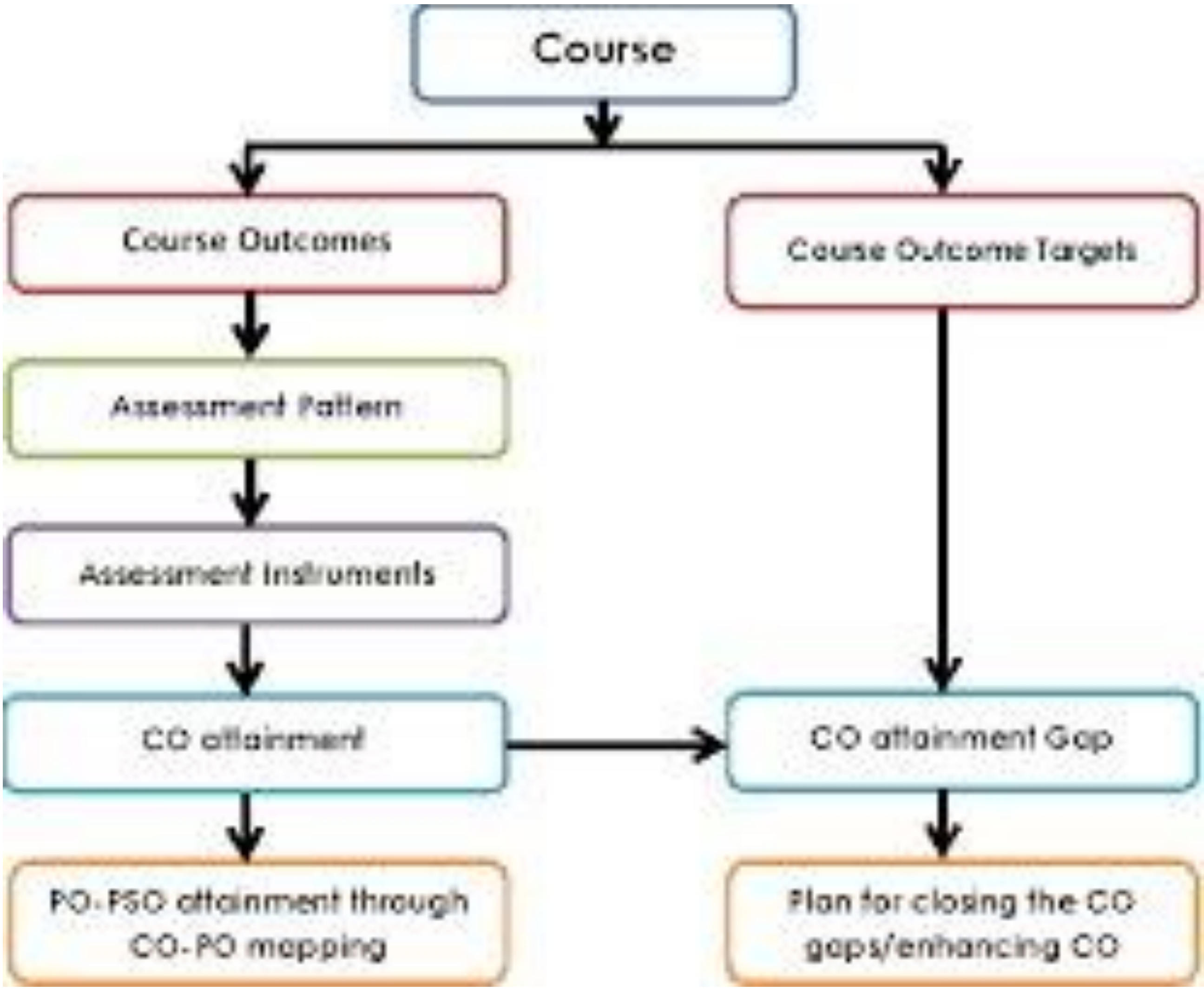

Fig 4.1: Design

Mapping course outcomes to program outcomes is an important process in designing an effective curriculum. The goal of this process is to ensure that individual courses align with the overall goals of the program and that students are adequately prepared for the workforce or further education.

Here are the high-level steps for mapping course outcomes to program outcomes:

- **Identify program outcomes:** Start by identifying the desired outcomes for the program as a whole. These might include skills, knowledge, and competencies that students should have upon completion of the program.

- **Identify course outcomes:** For each course in the program, identify the specific outcomes that students should achieve. These outcomes should be based on the content and objectives of the course.

- **Evaluate course outcomes:** Once we have identified course outcomes, evaluate each one to determine how it aligns with the program outcomes. Look for areas of overlap and identify any gaps that need to be addressed

- **Revise course outcomes:** If necessary, revise the course outcomes to better align with the program outcomes. This may involve modifying the course content or objectives.

- **Develop assessments:** Develop assessments that measure the achievement of both course outcomes and program outcomes. These assessments should be aligned with the course content and objectives and should provide evidence of student learning.

- **Implement and evaluate:** Implement the revised course outcomes and assessments, and evaluate their effectiveness. Use the results to make further revisions as necessary.

By above steps, we can ensure that each course in the program contributes to the overall goals of the program, and that students are adequately prepared for their future endeavors.

.

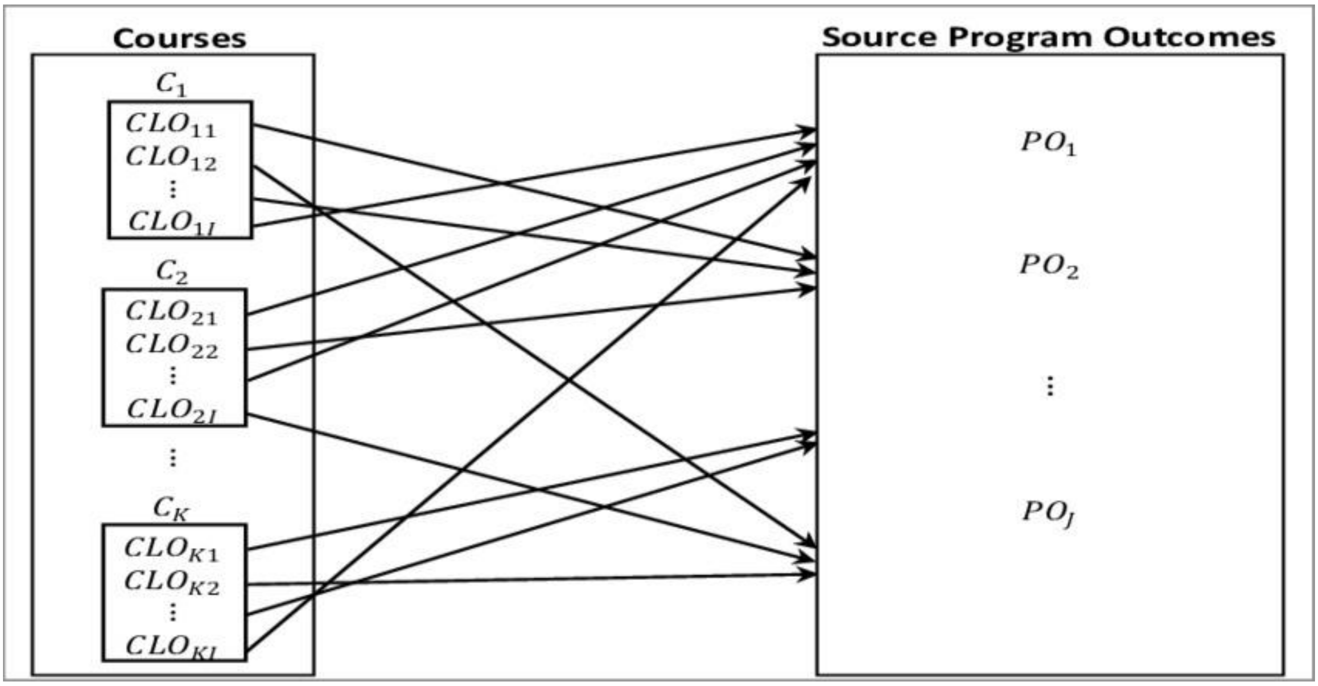

Fig 4.2 : Use Case Diagram

As mentioned in Fig 4.2, it depicts the Use case diagram for Mapping of COs to POs which assists the educational institutions to map course outcomes to program outcomes.

• We expect that the NLP model will be able to accurately map course outcomes to program outcomes with a high level of precision.
• We also expect that using an NLP model will significantly reduce the time and effort required for curriculum design and assessment.
• This process can facilitate the identification of overlapping or missing content, the assessment of program effectiveness, and the improvement of learning outcomes.
• It can also provide a framework for curriculum development and evaluation, as well as support the accreditation and quality assurance processes.

CHAPTER 5

TOOLS AND TECHNOLOGIES

1. **spaCy is a free open-source library for Natural Language Processing in Python.**

   Among the plethora of NLP libraries these days, spaCy really does stand out on its own.The factors that work in the favor of spaCy are the set of features it offers, the ease of use, and the fact that the library is always kept up to date.

2. **Language modeling techniques like BERT/Word Embeddings**.

   BERT (Bidirectional Encoder Representations from Transformers) is a powerful natural language processing (NLP) model that can be used for a wide range of tasks, including text classification, question answering, and sentiment analysis, among others.

Editor and executing softwares used:

1. Anaconda:

Anaconda is a distribution of the Python and R programming languages for scientific computing (data science, machine learning applications, large-scale data processing, predictive analytics, etc.), that aims to simplify package management and deployment.

2. Jupyter Notebook:

The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations, and narrative text. Uses include data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more..

CHAPTER 6

IMPLEMENTATION

1. Install the required libraries: docx2txt, strsimpy, docx2csv.
2. Extract tables from the Word document using the extract_tables function.
3. Read the Word document using the Document class from the docx library.
4. Define a function called read_docx_table to read a specific table from the document and return it as a DataFrame.
5. Import necessary libraries: pandas, TfidfVectorizer, CountVectorizer, cosine_similarity, and numpy.
6. Create a DataFrame for PLOs by calling the read_docx_table function for the first table.
7. Create a copy of the PLO DataFrame.
8. Initialize a CountVectorizer for the PLO DataFrame.
9. Preprocess the PLO data by converting it to lowercase.
10. Convert the PLO data to a list.
11. Fit and transform the PLO data using the CountVectorizer.
12. Get the feature names from the CountVectorizer.
13. Create a DataFrame for the PLO features.
14. Create a DataFrame for CLOs by calling the read_docx_table function for the second table.
15. Create a copy of the CLO DataFrame.
16. Preprocess the CLO data by converting it to lowercase.
17. Convert the CLO data to a list.
18. Fit and transform the CLO data using the CountVectorizer.
19. Get the feature names from the CountVectorizer.
20. Create a DataFrame for the CLO features.
21. Concatenate the CLO DataFrame with the respective index column.
22. Concatenate the PLO DataFrame with the respective index column.
23. Define a list of additional words to be appended to the PLO list.
24. Concatenate the additional words with the feature names to create the training column.
25. Get the feature names from the CLO DataFrame as the testing column.
26. Find the common column names between the training and testing columns.
27. Filter the common columns from the CLO DataFrame.
28. Extract each PLO from the PLO DataFrame and concatenate it with the filtered CLO DataFrame.
29. Fill the NaN values in the concatenated DataFrame with 0.
30. Calculate the cosine similarity between the concatenated DataFrame and create a new DataFrame.

31. Rename the first column of the new DataFrame to the respective PLO name.
32. Concatenate the first column from each cosine similarity DataFrame.
33.Concatenate all the concatenated cosine similarity columns into a single DataFrame.
34.Remove the first column as it represents the similarity of each PLO with itself.
35.Reset the index of the DataFrame.
36. Remove the 'index' column from the DataFrame.
37. Save the DataFrame to a CSV file named 'pseudocodematrix.csv'.
38. Print the DataFrame.
39. Iterate over each row in the DataFrame:
 a. Calculate the threshold value as the average of the minimum and maximum values in each row.
 b. If the threshold value is 0, keep the original value in the row.
 c. Otherwise, replace the value in the row with 1 if it is greater than or equal to the threshold, otherwise replace it with 0.
38. Save the modified DataFrame to a CSV file named 'PLO-CLOmapping.csv'.
39. Print the head of the modified DataFrame.
40. Read the original.csv file into a DataFrame.
41. Create a copy of the DataFrame.
42. Iterate over each row in the DataFrame:
 a. Compare the values in each row with the corresponding values in the modified DataFrame.
 b. If the values match, replace the value with 'True', otherwise replace it with 'False'.
43. Iterate over each column in the DataFrame:
a. Replace 'True' with 1 and 'False' with 0 in each column.
44. Calculate the accuracy of each row by taking the mean of the values in each row.
45. Concatenate the first column from the original DataFrame with the modified DataFrame.
46. Set the first column as the index of the DataFrame.
47. Calculate the mean of the 'acc' column in the DataFrame.
48. Print the mean accuracy value.

CHAPTER 7

RESULTS AND COMPARISON STUDY

```
In [45]: print(dd)

           P1        P2        P3        P4        P5        P6
1    0.000000  0.000000  0.000000  0.000000  0.000000  0.000000
2    0.277350  0.000000  0.000000  0.000000  0.000000  0.301511
3    0.000000  0.000000  0.000000  0.000000  0.000000  0.000000
4    0.160128  0.144338  0.000000  0.000000  0.000000  0.000000
5    0.160128  0.000000  0.000000  0.166667  0.000000  0.000000
..        ...       ...       ...       ...       ...       ...
117  0.000000  0.000000  0.223607  0.144338  0.158114  0.000000
118  0.196116  0.441942  0.000000  0.102062  0.111803  0.319801
119  0.124035  0.000000  0.000000  0.000000  0.000000  0.539360
120  0.000000  0.000000  0.000000  0.000000  0.000000  0.348155
121  0.000000  0.000000  0.258199  0.000000  0.365148  0.000000

[121 rows x 6 columns]
```

Fig.7.1 Cosine similarity of COs to POs

Out[62]:

0	P1	P2	P3	P4	P5	P6	acc
C1	1	1	1	0	1	1	0.833333
C2	0	0	0	0	0	1	0.166667
C3	0	0	0	0	0	1	0.166667
C4	0	0	0	0	0	0	0.000000
C5	0	0	0	0	0	0	0.000000

Fig.no:7.2

As shown in Figure 7.2 ‘ 0’ indicates low level of mapping ,’1’ indicates medium level of mapping and ‘2’ indicates high level of mapping and the acc column gives accuracy of mapping each co’s to all the po’s.

#	Course	Proposed Framework	SpaCy (Industrial-Strength Natural Language Processing, n.d.)	BERT (Devlin, J., Chang, M., Lee, K., and Toutanova, K., 2018)	Levenshtein (Levenshtein, V.I., 1966)	Jaro–Winkler (Winkler, W. E., 1990)
1	CSBP301	**0.875**	0.5417	0.5833	0.125	0.7917
2	CSBP320	**0.9**	0.3667	0.5333	0.3667	0.3667
3	CSBP400	**0.8667**	0.3	0.5	0.1667	0.6667
4	CSBP411	**0.75**	0.25	0.4167	0.375	0.2917
5	CSBP412	**0.875**	0.7917	0.375	0.4167	0.5417
6	CSBP421	**0.9167**	0.4583	0.625	0.2917	0.6667
7	CSBP431	**0.8333**	0.4167	0.7917	0.3333	0.4167
8	CSBP461	**0.7**	0.6333	0.6333	0.4333	0.4667
9	CSBP476	**0.7917**	0.375	0.4583	0.1667	0.7083
10	CSBP483	**0.7083**	0.5	0.5	0.4583	0.4583
11	CSBP487	**0.9583**	0.4167	0.625	0.375	0.3333
12	CSBP491	**0.8667**	0.4333	0.3	0.3333	0.5333
13	CSBP499	**0.8889**	0.5556	0.3333	0.5	0.4444
14	CSBP119	**0.875**	0.5	0.4583	0.3333	0.5833
15	CSBP121	**0.8**	0.4667	0.4667	0.3667	0.5667
16	CSBP219	**0.8**	0.4667	0.7333	0.2333	**0.8**
17	CSBP221	**0.8333**	0.4667	0.6333	0.2667	0.7
18	CSBP315	**0.9167**	0.2778	0.3333	0.3611	0.4722
19	CSBP316	**0.7333**	0.4667	0.2333	0.3333	0.5333
20	CSBP319	**0.8333**	0.4667	0.6667	0.3	**0.8333**
21	SWEB300	**0.8889**	0.3333	0.3056	0.25	0.8611
22	SWEB450	**0.75**	**0.5**	**0.75**	0.3333	0.4167
23	SWEB451	**0.9**	0.5667	0.7667	0.3	0.7
24	CSBP340	**0.8333**	0.3889	0.4444	0.3056	0.5278
25	ITBP370	**0.7778**	0.3056	0.75	0.5	0.3611
26	CSBP492	**0.8333**	0.3333	0.5	0.5	0.4167
	Overall accuracy	**0.8306**	**0.4593**	**0.5276**	**0.3376**	**0.5543**

Fig 8.1: Performance comparisions of the proposed framework in detecting correct mappings with other state-of-the-art NLP models.

According to a 2022 developer survey, Flutter is the most popular cross-platform mobile framework used by global developers, as stated in Fig 8.1. The survey reveals that 46 percent of software developers utilized Flutter. Overall, approximately one third of mobile developers opt forcross-platform technologies or frameworks, while the remaining developers employ native tools.

CHAPTER 8

Related works

1. The book titled "Practical Natural Language Processing" by authors Vajjala, Majumder, Gupta, and Surana was published by O'Reilly Media, Inc. in 2020. The book provides a practical guide to the field of Natural Language Processing (NLP), covering various concepts, techniques, and applications. It aims to equip readers with the necessary knowledge and skills to work with NLP technologies and develop real-world NLP solutions.

2. Plaza, C. M. , Draugalis, J. R. , Slack, M. K., Skrepnek, G. H., and Sauer, K. A. (2007). Curriculum mapping in program assessment and evaluation. American Journal of Pharmaceutical Education, 71(2).
   The article focuses on the use of curriculum mapping as a tool for assessing and evaluating N educational programs in the field of pharmaceutical education. It discusses the process of curriculum mapping, its benefits, and its application in program assessment to ensure alignment with desired educational outcomes.

3. Immerstein, R., Hasleberg, H., and Eri, G. (2020). Quality Assurance for Work Placement in Higher Education. IEEE Global Engineering Education Conference (EDUCON), (pp. 1096-1101). doi:10.1109/EDUCON45650.2020.9125156
   The paper discusses the importance of quality assurance in work placements within higher education. It explores the challenges and opportunities associated with assessing and ensuring the quality of work placements, proposing a framework for effective quality assurance measures.

4. Alshanqiti, A., Tanweer A., Mohamed B., Abdallah, N., and Ahmad T. (2020). A Rule based Approach toward Automating the Assessments of Academic Curriculum Mapping. International Journal of Advanced Computer Science and Applications (IJACSA), 11, 12.
   The paper explores how rules can be used to streamline the evaluation process and improve efficiency in curriculum mapping.

5. Ujkani, B., Minkovska, D., and Stoyanova, L. (2021). Using Natural Language Processing for Quality Assurance Purposes in Higher Education. IV International Conference on High Technology for Sustainable Development (HiTech), (pp. 01-04). doi:10.1109/HiTech53072.2021.9614206
   The paper "Using Natural Language Processing for Quality Assurance Purposes in Higher Education" presented at the IV International Conference on High Technology for Sustainable Development (HiTech) in 2021 explores the application of Natural Language Processing (NLP) techniques for quality assurance in higher education.

CHAPTER 9

CONCLUSION

This study underscores the promising application of Natural Language Processing (NLP) in mapping course outcomes to program outcomes within the realm of education. Through the utilization of cosine similarity as a measure of textual resemblance, the NLP-based approach offers an automated and efficient means of aligning educational objectives.

The comparative analysis between manual mapping and the NLP-based technique reveals the distinct advantages of computational methods. The NLP approach provides a scalable and objective process, eliminating the inherent subjectivity involved in manual mapping. By assigning grades of 0, 1, or 2 to denote the absence of a match, a single match, or multiple matches, respectively, the NLP-based approach enables a systematic evaluation of outcomes.

The results demonstrate that NLP-based mapping can substantially reduce the time and effort required for outcome alignment. Leveraging the power of machine learning, it empowers educators and curriculum developers to streamline the design and evaluation of educational programs. Additionally, it highlights the potential of NLP techniques in enhancing the quality of education by ensuring improved coherence and alignment of curricula.

In summary, this research provides valuable insights into the efficacy of NLP for mapping course outcomes to program outcomes. It emphasizes the transformative potential of automated techniques, such as NLP, in education. As institutions strive for continual improvement in curriculum development and assessment, the integration of NLP-based approaches can greatly enhance efficiency, accuracy, and standardization. This study lays the foundation for the adoption of NLP as a powerful tool to drive more effective and targeted learning experiences, ultimately advancing educational practices.

CHAPTER 10

REFERENCES

1. Alshanqiti, A., Tanweer A., Mohamed B., Abdallah, N., and Ahmad T. (2020). A Rule based Approach toward Automating the Assessments of Academic Curriculum Mapping. International Journal of Advanced Computer Science and Applications (IJACSA), 11, 12.

2. Shafi, A., et al. (2019). Student Outcomes Assessment Methodology for ABET Accreditation: A Case Study of Computer Science and Computer Information Systems Programs. IEEE Access, 7, 13653- 13667.

3. Ujkani, B., Minkovska, D., and Stoyanova, L. (2021). Using Natural Language Processing for Quality Assurance Purposes in Higher Education. IV International Conference on High Technology for Sustainable Development (HiTech), (pp. 01-04). doi:10.1109/HiTech53072.2021.961420.

4. Vajjala, S., Majumder, B., Gupta, A., and Surana H. (2020). Practical Natural Language Processing. O'Reilly Media, Inc. Watson, S., Steketee, C., Mansfield, K., Moore, M., Dalziel, B., Damodaran, A., Walker, B., Duvivier, RJ., and Hu, W. (2020). Curriculum mapping for health pro fissions education: A typology. Focus on Health Professional Education, 21(1), 91-113.

5. Winkler, W. E. (1990). String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage. Proceedings of the Section on Survey Research Methods, (pp. 354–359).

6. Plaza, C. M. , Draugalis, J. R. , Slack, M. K., Skrepnek, G. H., and Sauer, K. A. (2007). Curriculum mapping in program assessment and evaluation. American Journal of Pharmaceutical Education, 71(2). scikit learn Machine Learning in Python. (n.d.). Retrieved July 16, 2022, from https://scikit-learn.org/stable/.

7. Devlin, J., Chang, M., Lee, K., and Toutanova, K. (2018). BERT: Pre training of Deep Bidirectional Transformers for Language Understanding. arXiv.

8. Automating the Mapping of Course Learning Outcomes to Program Learning Outcomes using Natural Language Processing for Accurate Educational Program Evaluation. Posted Date: October 24th, 2022.