Mapping Engineering Course Outcomes to Program Outcomes Using an Intelligent Framework

Kilari Veera Swamy[1], Thirumala Devi S[2], Srilakshmi Aouthu [3]
[1,3] ECE Department, Vasavi College of Engineering, Hyderabad, Telangana, India
[2] ECE Department, KKR & KSR Institute of Technology and Sciences, Guntur, Andhra Pradesh, India
[1] k.veeraswamy@staff.vce.ac.in [2] thirumaladevi.ece@kitsguntur.ac.in [3] a.srilakshmi@staff.vce.ac.in

***Abstract*—Outcome Based Education is an important aspect as per National Education Policy 2020. To accomplish Program Outcomes, Course Outcomes are crucial. A group of subject matter experts typically maps Course Outcomes to Program Outcomes. This is a tedious process. To overcome this an intelligence framework is proposed in this paper. Course Outcome statements along with mapped program outcomes are considered as a data set in this paper. Split the data set as training and testing sets. A training data set is used to train the Machine Learning algorithm. Course Outcomes mapped with Program Outcomes by the expert team are considered input for the system. Machine Learning enables automated processes. Here, text documents are processed using tokenizer, removal of stop words, count words, and numeric data conversion. This numeric data is given as one input to the Machine learning algorithm. Mapped Program outcomes are given as other input to the Machine Learning algorithm. These are denoted as labels. Support Vector Machine is considered to develop intelligence for the system. A testing data set is applied to the intelligence framework to observe the performance of the system. A confusion matrix is formed based on the testing results. Traditional methods take more time to map COs to POs. Impact is more with for large curriculums. Further, it requires significant human effort. In the proposed methodology, COs to POs mapping is implemented fast and efficient manner. Further, mapping can be generated instantly with minimal human input. The accuracy of the system is calculated from the confusion matrix. Experiments indicate that Course outcomes mapped using an intelligence framework give better accuracy.**

***Keywords*—Course Outcomes; Program Outcomes; Machine Learning; Support Vector Machine; Accuracy.**

***JEET Category*—Research**

I. Introduction

Course Outcomes (COs) are the ability of the students after completion of the course. Here, attitude, skills, and knowledge are important parameters. Anderson et al. (2001) stressed the importance of Blooms taxonomy. Accessibility is the important aspect of a CO. COs do not depict the course inadequate detail if the number of COs are less. It may not serve instruction design that very well. A higher number of COs is also not recommended. In general, the number of COs varies between 4 to 6. Assessment is important to judge the attainment of course outcomes. Design the proper assessment techniques to measure the COs attainment. All Engineering colleges mention the COs along with syllabus. Few colleges are even keeping COs mapping with POs. Each course is generally divided into four to five units. No need to map the one CO for one unit. One unit can cover 2 or 3 COs also. Veera Swamy (2021) explained the attainment of graduate attributes using proper aligned courses.

Simpson Elizabeth (1966) explained the importance of educational objectives. Most of the courses do not address any PO other than PO1 (Rao & Kanth, 2019). Hardly any course addresses complex engineering problems and analysis. This is related to PO2. There concedes possibility be few particular courses that address PO7, PO8, PO10 and PO11. Most of the POs are met by mini/major/theme-based projects. Here, rubrics are important. A CO of a course can potentially address a significant number of POs. Here, the challenging task is to conduct instruction and assessment within the available time. Designing assessment techniques is another challenge. Some activities are planned outside the regular time to accomplish specific POs. Program heads should address this carefully. However, the planning of these activities need to be carefully planned by the program head. POs and PSOs are to be addressed through core courses, projects (major and mini), presentations, internships, co-curricular and extra-curricular activities. Varying strengths (1, 2 or 3) are used to map COs and POs. Identify the POs/ PSOs a course is going to address .It depends on the several activities. COs must be written to meet the identified POs/PSOs. Attainment of a PO/PSO depends both on the attainment levels of associated COs and the strengths to which they are mapped. Mapping strength identification is an important aspect. Low (1), Medium (2) and Strong (3) are used for mapping strength. This is very important in engineering course development (Veera Swamy et al. 2024). Several methods can be used to determine the strength of a PO/PSO. Mapping COs and POs higher number of courses is the challenge here.

Mapping COs to POs is done by educators manually, using personal judgment and experience. It purely depends on the expertise and understanding of the person doing the mapping. It requires significant human effort for large curriculums. It varies across individuals. Further, it is prone to human error or bias. Traditional method requires experienced educators with a deep understanding of outcomes and course objectives. To address this problem, an intelligent framework is adapted in this paper. COs mapped with POs by the specialized course group experts are to be considered as the input for the system. Machine Learning (ML) framework is used to train the classifier. Trained classifier is denoted as the model.

Kilari Veera Swamy
ECE Department, Vasavi College of Engineering, Hyderabad-500031, Telangana, India
k.veeraswamy@staff.vce.ac.in

This model is used to map the COs with POs. This phase is denoted as the mapping prediction. Later this is assessed using confusion matrix to calculate the accuracy of the classifier. Here, test data is explored to observe the performance of the model. Harden (2002) presented developments in outcome-based education. Program Outcomes (POs) considered in this work are PO1 to PO12. These are considered from National Board of Accreditation. These are the abilities of the student at the end of the program. In this work, Electronics and Communication Engineering Undergraduate program is considered. Broadly, these 12 POs are related to knowledge, skill, and attitude. Four POs are related to knowledge. Another four POs are related to the skill. Remaining four POs are related to the attitude. Program Specific Outcomes (PSOs) are not considered for simplicity. Further, Program Specific Outcomes varies across programs in a drastic way.

An intelligent framework is presented in chapter II. Results are discussed in chapter III. Conclusions are presented in chapter IV.

II. PROPOSED INTELLIGENCE FRAMEWORK

The sum of the course learning outcomes of constituent courses should be mapped to the program learning outcomes. Typically, 4 to 6 Course Outcomes (COs) are identified per course. POs/PSOs are achieved through various courses. Course content play an important role to meet the specific PO/PSO. POs/PSOs are achieved through course content, learning activity, and assessment technique. An intelligent process of CO-PO mapping framework is shown in Fig.1.

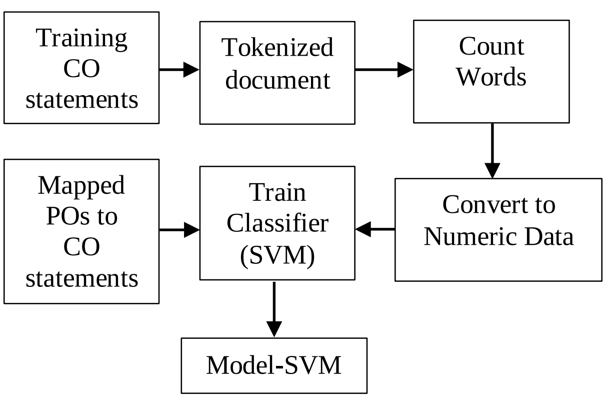

Fig.1. Training Process to develop intelligence

First CO statements and PO mapping strengths are to be gathered. This is divided into two parts. One is called a training data set. Another one is the testing data set. First, CO statements are tokenized. Later words are counted in the CO statements. This information is converted as numeric data. Available Mapped POs are considered as labels. Numeric data and labels are given to train the SVM classifier. Multi classifier SVM is considered in this work. Joachims Thorsten (1998) used SVM for text categorization. Labels vary from PO1 to PO12 with varying mapping strength. Mapping strength varies from 1 to 3. Examples are PO1-3, PO2-3, and PO3-2. The first part is PO number. The second part is mapping strength value. Multiclass classification using SVM extends binary classifiers to handle multiple class labels. It enables accurate classification into predefined categories like text classification. Three main approaches for multiclass classification with SVM are available. These are one vs one, one vs all, and directed acyclic graph. Each one has its own advantages and challenges. The one vs one approach breaks down the multiclass problem into binary subproblems. One vs all trains separate SVM classifiers for each class. Directed acyclic graph offers a hierarchical grouping approach, reducing diversity from the majority class. The testing process of the trained model is shown in Fig.2.

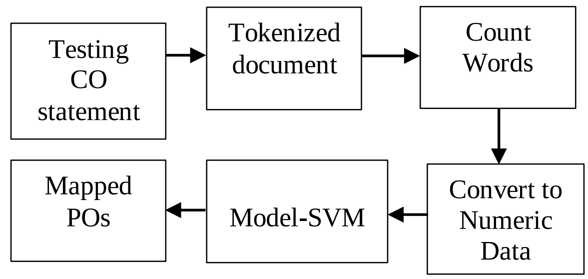

Fig.2. Testing Process to test intelligence

Same as training testing CO statements are processed using tokenizer, counting words and conversion to numeric value. This numeric value is given as input to the Model-SVM for prediction of the POs mapping.

A. Tokenized Document

Collection of words is known as tokens. This is important aspect of the text classification. Detection of complex tokens in text is the challenging task. Theory, lab, and project courses have their own signatures in writing COs. Stop words are to be removed here. Perform word-level preprocessing tasks such as stemming or lemmatization using the normalize words function. Analyze word and n-gram frequencies using specific logic. Use sentence and part-of-speech details for better performance. Entity tags are also an important activity. Use grammatical dependency details for better performance. Analysis of the details about the tokens is also important. Token contains document number, sentence number, line number, type, language, and part of speech. Token text is returned as a string scalar. Document number is the index of document that the token belongs to, returned as a positive integer. Sentence number of tokens in document is returned as a positive integer. The line number of tokens in document is returned as a positive integer. The type of token returns as letters, digits, punctuation, and other. Language of the token returns as English, Japanese, German, Korean etc., Part of speech tag returns as adjective, adverb, auxiliary-verb, coord-conjunction, conjunction, determiner, interjection, noun, numeral, particle, pronoun, proper noun, punctuation, subord-conjunction, symbol, verb and other.

B. Count words

Model concentrates the number of times that words appear in all CO statements. It gives the details of counts, vocabulary, words and documents. Counts indicate that the total COs considered for training/testing. Same number is considered as number of documents considered for training/testing. The number of words depends on the vocabulary. Count words is the important item during training phase and testing phase. After this numeric conversion is initiated.

C. Convert to numeric data

Performance of the Machine Learning is good with numerical data. Each string in the text is converted to feature vector. This is the number of the string. Several techniques are available in the literature to convert text to numeric data. Here everything is represented as numeric data.

D. Train Classifier

Classifier uses binary SVM. Consider N is class labels. It uses one-versus-one coding design. Classes considered are positive and negative. Others are ignored. Generally, the multiclass problem is represented as collection of binary problems. Each row of the coding design corresponds to a distinct class, and each column corresponds to a binary learner. In binary learner:

- '+1' group into positive class.
- '-1' group into negative class.
- '0' group into ignore class.

Classifier consists of multiple binary learners such as support vector machines (SVMs). Awad Mariette and Khanna Rahul (2015) used SVM for classification. Training CO statements are arranged in table. Respective PO mapping is represented as labels. Label has K classes (levels) and can be any of the following:

1. Label possibilities are categorical array, logical vector, numeric vector, string array and cell array of character vectors.
2. The name of a variable is used as the response label. Remaining variables in table are used as predictors.
3. A formula for a character vector or string scalar is given below:
   'Label ~ x1 + x2 + x3 + x4 + x5'
   Label is to be used as the response. Weights can be added for better performance.

Algorithm

- Keep all PO mapping with strength in one column in the table(L)
- Keep all CO statements in another column of the table
- Clean all CO statements
- Generate tokenized document
- Count words
- Convert to numeric data(X)
- Train SVM with X and L using hyper plane concept.
- Generate Model-SVM (M)
- Take test CO statement
- Clean CO statement
- Generate tokenized document
- Count words
- Convert to numeric data(T)
- T is given to the M
- Output is the POs mapping
- Construct confusion matrix
- Calculate accuracy

Detailed flow chart related to Machine Learning is presented in Fig 3.

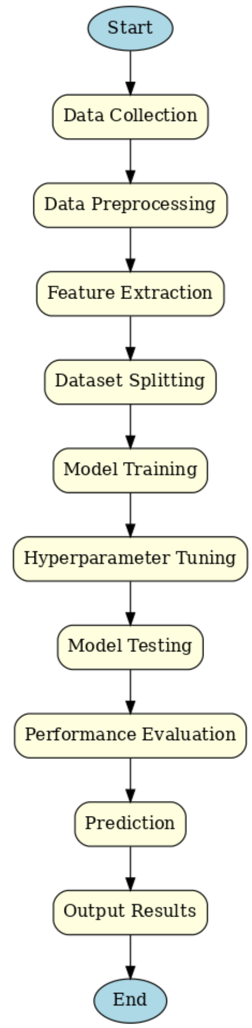

Fig.3. Detailed flow chart of machine learning framework

III. RESULTS AND DISCUSSION

Various CO statements along with PO mapping are considered to train the ML model. Experiments are performed in MATLAB platform. Gatto Marino and Rizzoli Andrea (1993) reviewed the importance of MATLAB. Forty CO statements are considered from theory courses. Twenty CO statements are considered from practical courses. Fifteen CO statements are considered from project related courses. Seventy-five percentage of data set is used for training. Twenty-five percentage of data set is used for testing. Training data is represented as 75x319 sparse double format. Labels are represented as 75x1 string format. First count the words available in the test CO statements. It identifies the frequency of words appear in the test document. Remove the stop words from a collection of words model. Stop words are words such as "a", "the", and "in" which are commonly removed from text before analysis. Create a collection of words model using a string array of unique words and a matrix of word counts. In this case the count is 75x319 sparse double format. Vocabulary in this case is 1x319 string format. The number of unique words is 319. The number of documents is 75. After this, unique words are represented in numeric format. 48 classes are there in 75 CO statements. It indicates that only 48 combinations are there in PO mapping statements. Histogram is shown in Fig 4. X axis is the class. Here are the PO mapping values. Y axis is the frequency.

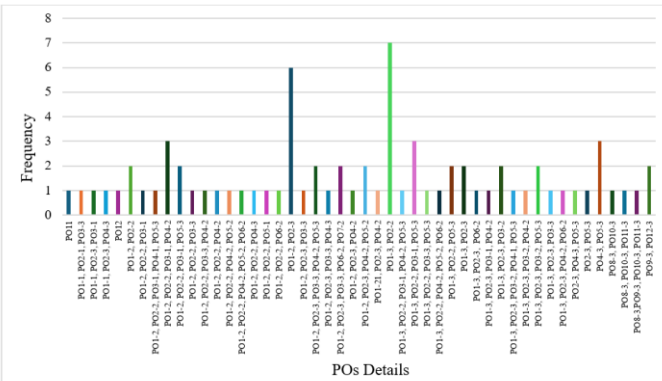

Fig.4. Histogram of training data

Histogram details are important in Machine Learning. Class and frequency information is presented here. After training the ML algorithm with test CO statements and corresponding PO mapping makes the ML model. One CO statement is elaborated for further understanding. CO statement considered is “Apply the basic concepts of Image and Video Processing". Originally it contained 9 tokens. Stop words are removed for better classification performance. After removal of stop words the CO statement is “Apply basic concepts Image Video Processing”. Ony 6 tokens are sufficient to represent the CO statement effectively. Hence, vocabulary size is 1x6 string. Hence numerical representation for the label “PO1-3, PO2-2” is given below.

(1,1) 1
(1,2) 1
(1,3) 1
(1,4) 1
(1,5) 1
(1,6) 1

Here class names are restricted to 48 even though 75 CO statements. The hinge loss is used for "maximum margin" classification, most notably for support vector machines (SVMs). Margin maximization is important in SVM. It uses a model to maximize the margin between different classes. Hyper plane is constructed in such a way that misclassification is reduced. Hence, it should possess the maxim margin property. It reduces the classification error during testing phase. Model is formed using hyper plane concept. This is important aspect in Machine Learning. Number of classes are important while training the model. Numeric input is also important to the corresponding class. Formed model contains the following items as represented in TABLE I.

TABLE I
TEST RESULTS OF CO-PO MAPPING

Details	Property
Binary Learners	1128x1 cell
Binary Loss	hinge
Coding Matrix	48x1128 double
Learner Weights	1x1128 double
Class Name	48x1 cell
Predictor Names	1x319 cell

ML model is tested using various CO statements. Results are presented in TABLE II. One theory, one practical, and one project CO statements are presented.

TABLE II
TEST RESULTS OF CO-PO MAPPING

CO statement	Numeric Data Representation	PO Mapping result
Apply Deep Learning techniques suitable for a given problem	(1,22) 1 (1,38) 1 (1,44) 1 (1,45) 1 (1,46) 1 (1,47) 1 (1,48) 1 (1,49) 1	{'PO1-2 PO2-3'}
Design and implement Neural Networks for classification and regression tasks	(1,9) 2 (1,32) 1 (1,46) 1 (1,196) 1 (1,233) 1 (1,235) 1 (1,236) 1 (1,237) 1 (1,238) 1	{'PO1-2, PO2-3, PO3-3, PO4-2, PO5-3'}
Review the literature to identify problem	(1,4) 1 (1,14) 1 (1,48) 1 (1,298) 1 (1,300) 1	{'PO1-2 PO2-3 PO3-3 PO6-2 PO7-2'}

The first CO statement contains 9 words. Finally, 8 words are considered after removing the stop word ‘a’. The second CO statement contains 10 words. ‘and’ term is repeated twice. Hence, 9 words are considered. The third CO statement contains 6 words. Finally, 5 words are considered after removing the stop word ‘the’. Accordingly, 8, 9 and 5 numeric data representation is presented in TABLE II. PO mapping statements generated after testing the model is presented in TABLE II. CO statements are tested to assess the performance of the model. Accuracy is computed using the confusion matrix. Accuracy is the fundamental metric in assessing machine learning models, reflecting the proportion of correctly predicted instances relative to the total instances. It serves as a foundational measure for evaluating model correctness and remains essential for assessing overall model performance and effectiveness across diverse applications. Confusion matrix details are presented in TABLE III.

TABLE III
CONFUSION MATRIX

	Predicted Positive	Predicted Negative
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

True Positive (TP): Correct item is identified as correct item.
False Positive (FP): Wrong item is identified as correct item.
True Negative (TN): Wrong item is identified as wrong item.
False Negative (FN): Correct item is identified as wrong item.
Accuracy is calculated using "(1)".

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \qquad (1)$$

FP and FN should be zero to get better accuracy. Twelve and twenty-five test CO statements accuracy is presented in Fig 5.

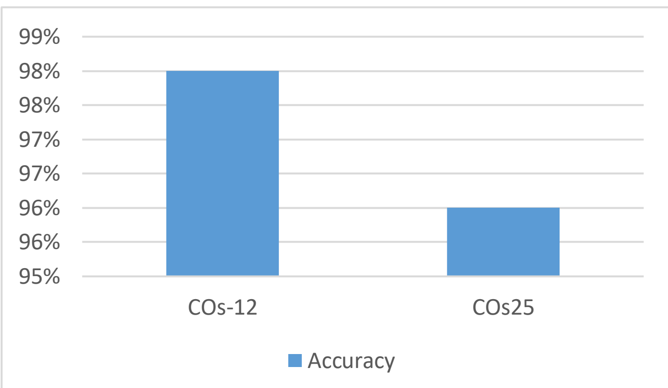

Fig.5. Testing Accuracy with CO statements

Experimental results indicate that the accuracy of the intelligence framework is very good. It varies from 96 to 98 percentage. Number of CO statement mapping to POs is not the matter with an intelligent framework.

CONCLUSIONS

The OBE process defines several key activities at the course level that involve the Course Outcomes, Program Outcomes, mapping of Course Outcomes with Program Outcomes, and attainment of COs/POs. An intelligence framework is proposed using Support Vector Machine. The accuracy of the system is computed. Training data set equivalent numeric representation is the key in this aspect. All the key words are represented in numeric format. Hence, accuracy is 98 percent with twelve CO statements. Accuracy is 96 percent with twenty-five CO statements. This framework avoids manual errors caused by human intervention. Further, it saves lot of time of the faculty. Accuracy is further improved by considering proper training data set. Neural Network based prediction can improve accuracy further. But complexity is more. Further, expanded datasets can also improve accuracy.

ACKNOWLEDGMENT

The authors would like to thank the management authorities of Vasavi College of Engineering and KKR & KSR Institute of Technology and Sciences for their constant support and encouragement.

REFERENCES

Anderson, L.W. (Ed.), Krathwohl, D.R. (Ed.), Airasian,P.W., Cruikshank, K.A., Mayer, R.E., Pintrich, P.R.,Raths, J., & Wittrock, M.C. (2001). A taxonomy for learning, teaching, and assessing: *A revision of Bloom's Taxonomy of Educational Objectives* (Complete edition). New York: Longman.

Kilari Veera Swamy (2021). Methodologies to attain Graduate Attributes at Class Level, *Journal of Engineering Education Transformations*, Volume 34, Special issue, 367–401.

Simpson, Elizabeth J.(1966). The classification of educational objectives: Psychomotor domain. *Illinos Journal of Home Economics*. 10(4):110-144.

Prof. N.J. Rao, Dr. K Rajani Kanth. TALE 2 (2019). *Course Design and Instruction of Engineering Course. https://onlinecourses.nptel.ac.in/noc19_ge17/preview*

Kilari Veera Swamy, Vadhi Radhika and Srilakshmi Aouthu. (2024). Engineering Course Development Process to meet Outcome Based Education, *Journal of Engineering Education Transformations*, Volume 37, Special issue, 67–72.

Awad Mariette and Khanna Rahul (2015). "Support Vector Machines for Classification". In: *Efficient Learning Machines*. Apress. pp. 39–66.

Joachims, Thorsten (1998). "Text categorization with Support Vector Machines: Learning with many relevant features". *Machine Learning: ECML-98*. Lecture Notes in Computer Science. Vol. 1398. Springer. pp. 137–142.

Gatto, Marino; Rizzoli, Andrea (1993). "Review of MATLAB, Version 4.0". *Natural Resource Modeling*. 7 (1). Wiley: 85–88.

R.M.Harden (2002). Developments in outcome-based education. *Medical Teacher*, 24:2, 117-120.

About MathWorks (2022). mathworks.com.