****

**IS450 Text Mining and Language Processing – G2**

**Academic Year 2019/2020 Term 2**

****

**Analysing Students’ Course Feedback  
(Final Report)**

**Group 4 Team Members:**

Tan Wei Long Ryan

Ong De Lin

Narang Aakshi

Lee Jia Ern Janell

Lau Jun Rong

Benedict Then Ji Xiang

Date of Submission: 26 April 2020

**Coding Scripts Repository**

<https://drive.google.com/drive/folders/1k76Kur3iNA9jvEQAKqv-bwZFBjypOKqT?usp=sharing>

**Table of Contents**

[**1.0 Introduction** 1](#_Toc38717960)

[**1.1 Background & Project Motivation** 1](#_Toc38717961)

[**1.2 Objective & Stakeholders** 1](#_Toc38717962)

[**1.3 Business Scenarios** 1](#_Toc38717963)

[**2.0 Data Selection and Pre-processing** 1](#_Toc38717964)

[**2.1 Data Sources** 1](#_Toc38717965)

[**2.2 Exploratory Data Analysis (EDA)** 2](#_Toc38717966)

[2.2.1 Summary of Dataset 2](#_Toc38717967)

[2.2.2 Understanding the Students’ Characteristics 3](#_Toc38717968)

[2.2.3 Understanding the Feedback 5](#_Toc38717969)

[**2.3 Data Pre-Processing** 7](#_Toc38717970)

[2.3.1 Standard Data Pre-Processing Steps 7](#_Toc38717971)

[2.3.2 Data Pre-Processing Steps Specific to our Dataset 7](#_Toc38717972)

[**3.0 Data Challenges** 8](#_Toc38717973)

[**3.1 Short Text** 8](#_Toc38717974)

[**3.2 Single Topic Per Document** 9](#_Toc38717975)

[**3.3 Linguistic Features** 9](#_Toc38717976)

[**4.0 Task 1 (Topic Modelling)** 10](#_Toc38717977)

[**4.1 Objective** 10](#_Toc38717978)

[**4.2 Solution Methodology** 10](#_Toc38717979)

[4.2.1 Solution Overview 10](#_Toc38717980)

[**4.2.1.1 Evaluation of Models** 12](#_Toc38717981)

[**4.2.1.1.1 Qualitative:** 12](#_Toc38717982)

[**4.2.1.1.2 Quantitative:** 13](#_Toc38717983)

[**4.2.1.2 Automatic topic label assigning** 13](#_Toc38717984)

[4.2.2 Baseline Model (LDA) 14](#_Toc38717985)

[**4.2.2.1 Results and Analyses** 14](#_Toc38717986)

[**4.2.2.1.1 Baseline - Unigram topic modelling results** 14](#_Toc38717987)

[**4.2.2.1.2 Baseline - Bigram topic modelling results** 16](#_Toc38717988)

[4.2.3 Data-level Approach: Aggregation of Documents 19](#_Toc38717989)

[**4.2.3.1 Data-level Approach: Aggregation by Vector Space Similarity** 19](#_Toc38717990)

[**4.2.3.1.1 Literature review** 19](#_Toc38717991)

[**4.2.3.1.2 Methodology** 19](#_Toc38717992)

[**4.2.3.1.3 Results and Analyses** 20](#_Toc38717993)

[**4.2.3.2 Data-level Approach: Aggregation by Academic Weeks** 22](#_Toc38717994)

[**4.2.3.2.1 Literature review** 22](#_Toc38717995)

[**4.2.3.2.2 Methodology** 23](#_Toc38717996)

[**4.2.3.2.3 Results and Analyses** 23](#_Toc38717997)

[4.2.4 Model-level approach: Gibbs Sampling Algorithm for a Dirichlet Mixture Model (GSDMM) 25](#_Toc38717998)

[**4.2.4.1 Literature review** 25](#_Toc38717999)

[**4.2.4.2 Methodology** 25](#_Toc38718000)

[**4.2.4.3 Results and Analyses**  **GSDMM** 26](#_Toc38718001)

[**4.3 Results and Analyses** 28](#_Toc38718002)

[*Why are the results useful, pertaining to the use case?* 28](#_Toc38718003)

[**4.4 Gap Analysis and Discussion** 29](#_Toc38718004)

[4.4.1 Limitations of our evaluation methodology 29](#_Toc38718005)

[4.4.2 Discussion on Data Level Approach: Aggregation of Documents 29](#_Toc38718006)

[**4.4.2.1 Aggregation by vector space similarity** 29](#_Toc38718007)

[**4.4.2.2 Aggregation on academic weeks** 31](#_Toc38718008)

[4.4.3 Discussion on Dirichlet Mixture Model (DMM) 31](#_Toc38718009)

[**4.5 Future Work** 31](#_Toc38718010)

[4.5.1 Early warning of crucial chapters through recommender systems 31](#_Toc38718011)

[4.5.2 Implementation of a scalable and continually developing gazette 33](#_Toc38718012)

[**5.0 Task 2 (Classification)** 34](#_Toc38718013)

[**5.1 Objective** 34](#_Toc38718014)

[**5.2 Solution Methodology** 35](#_Toc38718015)

[5.2.1 Solution Overview 35](#_Toc38718016)

[5.2.2 Word Representation 35](#_Toc38718017)

[5.2.3 Feature Engineering 36](#_Toc38718018)

[5.2.4 Models & Hyper-Parameters Considered for Tuning 37](#_Toc38718019)

[5.2.5 Feature Selection Approaches (Term-Frequency) 38](#_Toc38718020)

[5.2.6 Evaluation Metrics (F1 Macro vs F1 Weighted) 38](#_Toc38718021)

[**5.3 Results and Analyses** 39](#_Toc38718022)

[5.3.1 Choosing the Base Classifier 39](#_Toc38718023)

[5.3.2 Model & Feature Evaluation 40](#_Toc38718024)

[**5.4 Error Analysis and Discussion** 41](#_Toc38718025)

[**5.5 Gap Analysis and Discussion** 43](#_Toc38718026)

[5.5.1 Vader Sentiment Analyzer 43](#_Toc38718027)

[5.5.2 StratifyKFold vs KFold 44](#_Toc38718028)

[5.5.3 TFIDF fit\_transform training set & transform testing set 45](#_Toc38718029)

[5.5.4 Cross Validation vs Nested Cross Validation 45](#_Toc38718030)

[**5.6 Future Work** 46](#_Toc38718031)

[5.6.1 Reducing the Number of Classes to Predict 46](#_Toc38718032)

[5.6.2 Developing our own F-Score 46](#_Toc38718033)

[5.6.3 Adding Additional Features besides those from Feedback 46](#_Toc38718034)

[5.6.4 Fine Tuning the Engineering Approach of Topic Modelling Features 46](#_Toc38718035)

[5.6.5 PCA Word Vectors 46](#_Toc38718036)

[**6.0 Conclusion** 46](#_Toc38718037)

[**7.0 Reflections** 47](#_Toc38718038)

[**References** 48](#_Toc38718039)

[**Appendix A** 50](#_Toc38718040)

[**Topic Modelling Results on “ List one topic of the class that you enjoy” column (Raw Results)** 50](#_Toc38718041)

[Baseline - Unigram 50](#_Toc38718042)

[Baseline - Bigram 51](#_Toc38718043)

[Aggregation - Vector Space Similarity 52](#_Toc38718044)

[Aggregation - Academic Weeks 53](#_Toc38718045)

[GSDMM 54](#_Toc38718046)

[**Topic Modelling Results on “What was the most challenging topic of the class?” column (Raw Results)** 55](#_Toc38718047)

[Baseline - Unigram 55](#_Toc38718048)

[Baseline - Bigram 56](#_Toc38718049)

[Aggregation - Vector Space Similarity 57](#_Toc38718050)

[Aggregation - Academic Weeks 58](#_Toc38718051)

[GSDMM 59](#_Toc38718052)

[**Topic Modelling Results on “(Optional) What do you suggest to improve the class?” column (Raw Results)** 60](#_Toc38718053)

[Baseline - Unigram 60](#_Toc38718054)

[Baseline - Bigram 61](#_Toc38718055)

[Aggregation - Vector Space Similarity 62](#_Toc38718056)

[Aggregation - Academic Weeks 63](#_Toc38718057)

[GSDMM 64](#_Toc38718058)

[**Appendix B** 66](#_Toc38718059)

[**Feature Engineering** 66](#_Toc38718060)

[Grades 66](#_Toc38718061)

[Sentiment Score 67](#_Toc38718062)

[Feedback Length Proportion 69](#_Toc38718063)

[Class Type (morning, afternoon, evening) 69](#_Toc38718064)

[Frequency of Suggestion 70](#_Toc38718065)

[Fuzzy Similarity Scores between Feedback & Course 71](#_Toc38718066)

[Topic Modelling 72](#_Toc38718067)

[**Appendix C** 75](#_Toc38718068)

[**Feature Selection Approaches** 75](#_Toc38718069)

[First Approach (All Words) 75](#_Toc38718070)

[Second Approach (Words which occurs more than median word count) 78](#_Toc38718071)

[Third Approach (Words that occur less than median word count) 81](#_Toc38718072)

# 

# **1.0 Introduction**

## **1.1 Background & Project Motivation**

Nearing the end of an academic semester, students in Singapore Management University (SMU) are required to complete a feedback form based on their learning experience as well as their perspective on the performance of the various professors’ teachings across the different modules. The feedback questionnaire consists of both quantitative and qualitative questions such as rating scale, and open-ended inputs. While comprehending quantitative feedback is relatively easy and fast, the same cannot be said for qualitative feedback. Making sense out of the qualitative feedback is tedious and time consuming as it must be manually reviewed.

## **1.2 Objective & Stakeholders**

Through our project, we aim to lighten the workload of the faculty team and improve work efficiency by automating the manual and tedious task of reviewing qualitative feedback. Ultimately, we hope that this will better assist the faculty team in gathering and discovering quality insights that can improve the course content and curriculum structure for future students.

## **1.3 Business Scenarios**

Leveraging on Text Mining techniques, we came up with 2 approaches to extract and understand feedback comments in order to assist our stakeholders in their objectives.

Task 1

Our first approach employs Topic Modelling to extract hot keywords that aim to aid the faculty in identifying common challenging and popular aspects that contributed to the students’ class learning experience.

Task 2

Our second approach uses classification techniques to identify potentially weaker students that might require extra assistance by predicting their assignment grades from their feedback.

# **2.0 Data Selection and Pre-processing**

## **2.1 Data Sources**

For this project, we have made use of 3 data sources which are related to the IS210 BPAS module:

1. **Student Feedback**

This dataset consists of the students’ demographics and their feedback.

1. **Assignment Grades**

This dataset consists of the students’ assignment score for various topics.

1. **BPAS Topic Outline**

This dataset consists of the IS210 BPAS module’s weekly outlines and content descriptions.

There are 11 weekly outlines in total.

A brief snapshot of what information is contained in our dataset is shown below (Refer to Figure 1).

|  |  |  |
| --- | --- | --- |
|  | **Columns** | |
| **IS210 BPAS Weekly Class Feedback Dataset**  **(3099 rows of data)** | 1. Timestamp 2. Name 3. Class Section 4. List one topic of the class that you enjoy | 1. What was the most challenging topic of the class? 2. Overall, the learning experience for the class is 3. (Optional) What do you suggest to improve the class? |
| **IS210 BPAS Assignment Grades**  **(374 rows of data)** | 1. Use Case Model 2. Function model 3. Solution overview model 4. Application model 5. Process Innovation 6. Business Model | 1. Process Architecture 2. Process Model 3. Business goals strategy 4. Workflow diagram 5. Static analysis 6. Recommendations |

*Figure 1: Snapshot of Weekly Class Feedback and Assignment Grades Dataset*

In this report, we will reference these 3 columns in the IS210 BPAS Weekly Class Feedback Dataset as per their alternative names below.

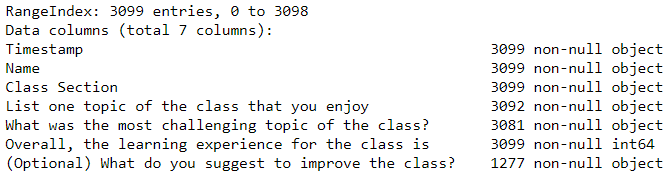
“List one topic of the class that you enjoy” column → “Enjoyable” column

“What was the most challenging topic of the class?” column → “Challenging” column

“(Optional) What do you suggest to improve the class?” column → “Suggestions” column

## **2.2 Exploratory Data Analysis (EDA)**

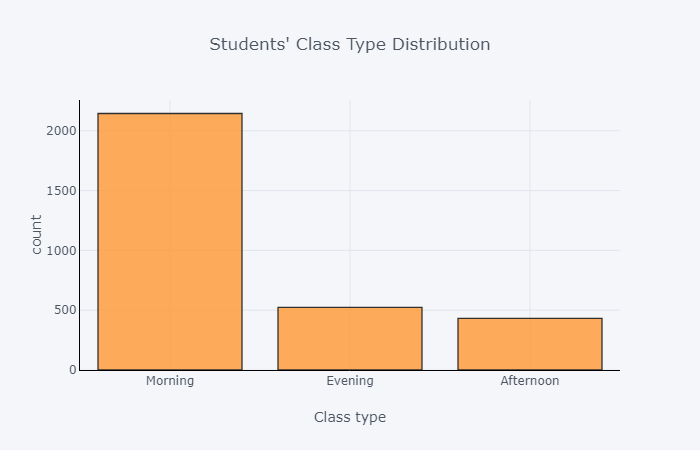
### 2.2.1 Summary of Dataset



*Figure 2: Summary of Weekly Class Feedback Dataset*

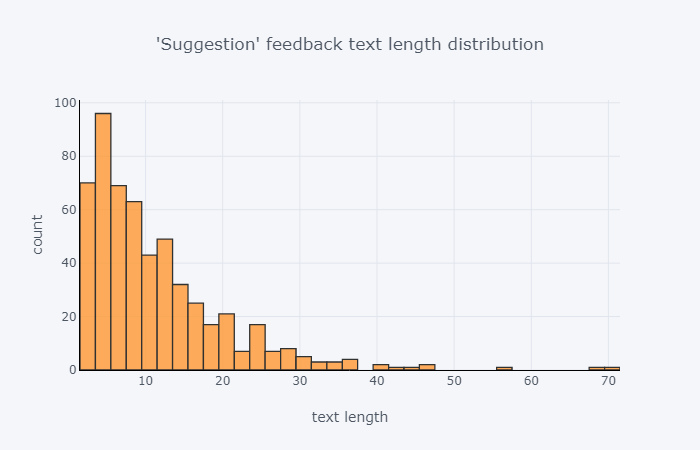
The table above (Refer to Figure 2) shows a summary of our Class Feedback dataset. We observed that we have 3099 rows of data with 7 features. Noticeably, there are 1277 null values in the “Suggestions” column. This means that we have around 40% of empty data for this column.

### 2.2.2 Understanding the Students’ Characteristics



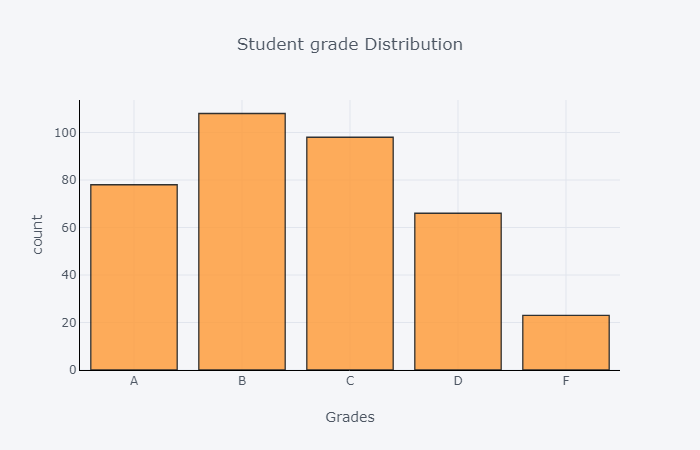
*Figure 3: Student Class Timing Distribution*

Firstly, we observed that most of the feedback is made up of students enrolled in morning classes as compared to afternoon and evening (Refer to Figure 3). This is one characteristic that we will analyse, to determine if their class timings influence the students’ grades

******

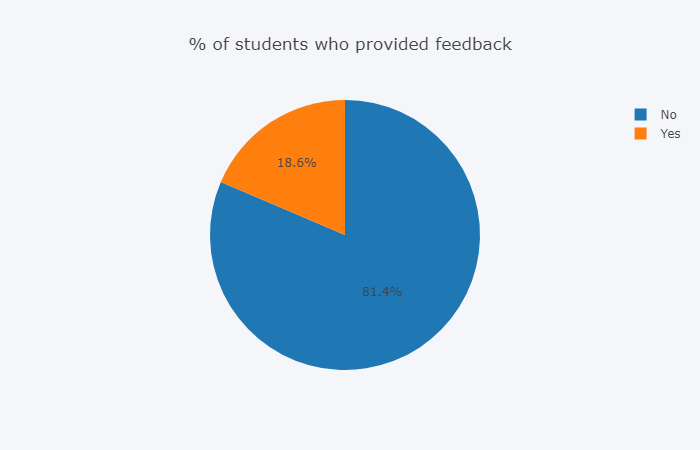
*Figure 4: “Suggestions” column text length distribution*

Secondly, most students **gave relatively short feedback** ranging around 0 to 15 words (Refer to Figure 4). This shows that most students chose to give short and concise feedback when giving their suggestions.



*Figure 5: Students’ grade distribution*

Thirdly, the **grade distribution is approximately normal** and follows a normal bell curve shape (Refer to Figure 5).

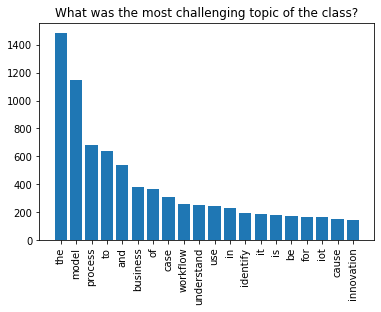


*Figure 6: Percentage of Students who provided feedback*

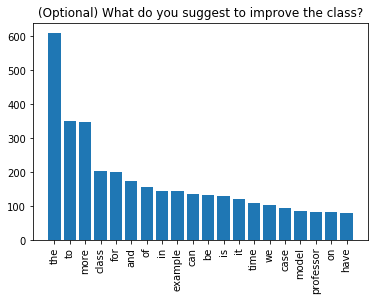
Lastly, only **18% of students provided “Suggestions” feedback** (Refer to Figure 6)**.**

Based on the above descriptive analysis, we can see that we will face a challenge of having limited dataset in terms of quantity and quality that might affect the results of our Text Mining tasks. Further research on the different methods required to help us circumvent this challenge will be explained in the following sections.

### 2.2.3 Understanding the Feedback

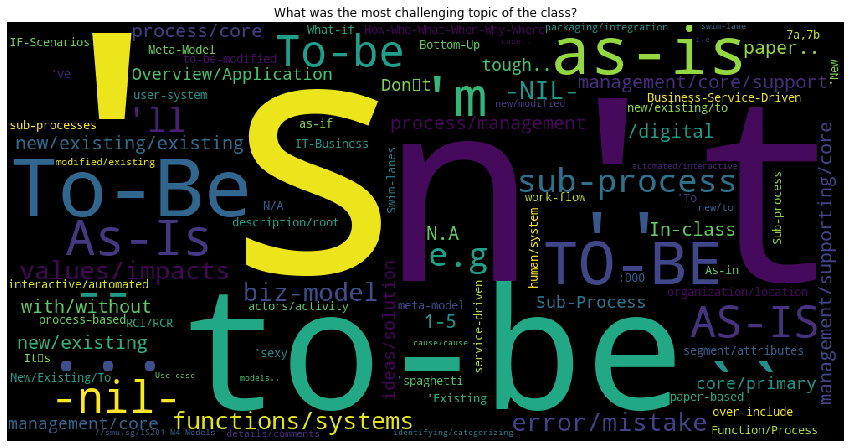


*Figure 7: Top 20 Word Frequencies in “Challenging” column*

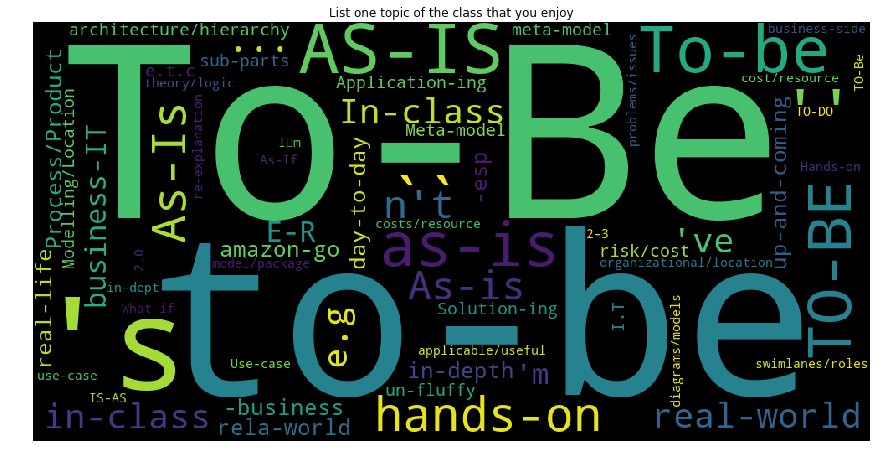
*******Figure 8: Top 20 Word Frequencies in “Suggestions” column*

Figures 7 and 8 above show the top 20 words that were used in the “Challenging” and “Suggestions” columns respectively. This gives us a general idea of the frequently used words written in the students' feedback.

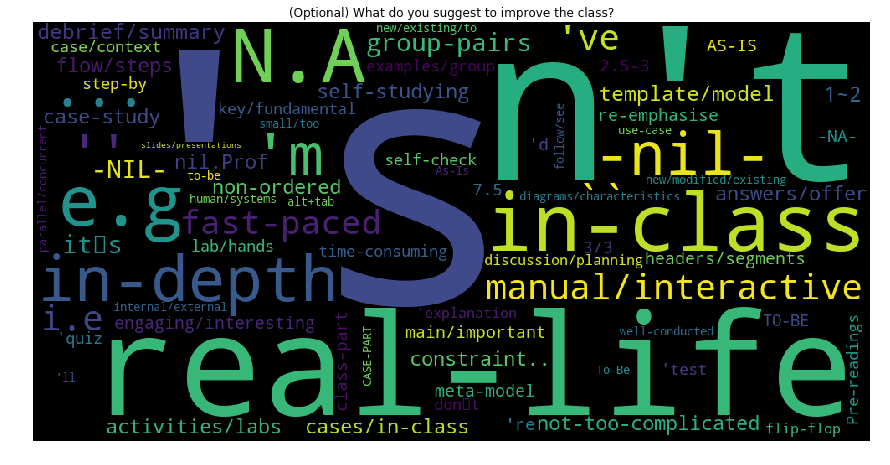
At first glance, both figures tell us that these columns have many stop words such as “the”,” and”, “to” and “of” that should be removed. However, due to considerations based on domain knowledge, we did not remove these stop words immediately as they may be tied to another word to refer to a topic as a whole. Instead, we examined to see if these words could add value to our analysis. Secondly, we picked up some key words such as “example”, “time”, “workflow” and “process”, which tells us that the students may need more time and practice examples to understand the concepts behind the workflow and process diagram taught in this course.

**

*Figure 9: Word Cloud for “Challenging” Column*

**

*Figure 10: Word Cloud for “Enjoyable” Column*

**

*Figure 11: Word Cloud for “Suggestions” column*

Figures 9, 10 and 11 show the commonly used words in the students’ feedback for the “Challenging”, “Enjoyable” and “Suggestions” columns. We observed that there is an overlap in what students found enjoyable and what they found challenging. This can be seen from the words “As-Is” and “To-Be” appearing in both columns (Refer to Figure 9 and Figure 10). Based on domain knowledge, we can link it to the “As-Is” and “To-Be” workflow models that are taught in the course curriculum. This tells us that the stop words such as “as”, “is”, “to” and “be” are of value as they are part of a topic in the syllabus. However, we will still remove these stop words when performing topic modelling in Task 1.

In addition, we can see unusual words such as “‘s”, “n”, “t” being formed (Refer to Figure 11). This tells us that tokenization of words might have affected words such as “shouldn’t”, “don’t” and “aren’t” and suggests that we should expand contractions. Lastly, we can see that students want to have more “in-class”, “real-life” and “interactive” elements in class as they are the commonly used terms used in the “Suggestions” column.

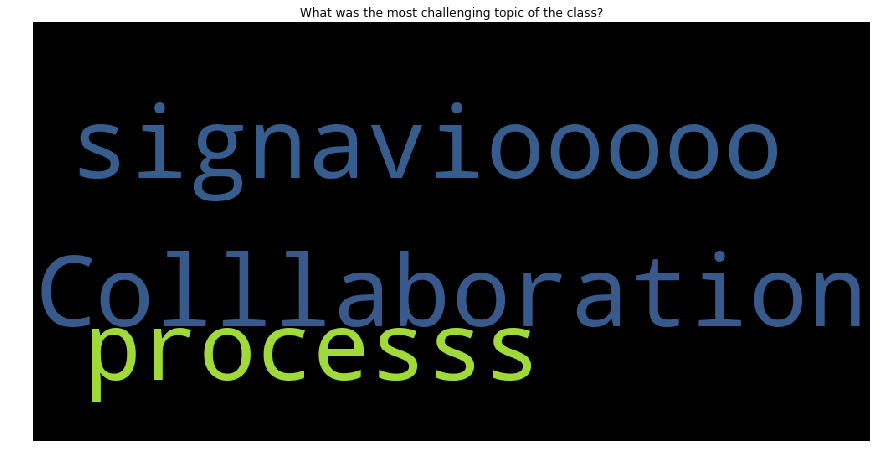
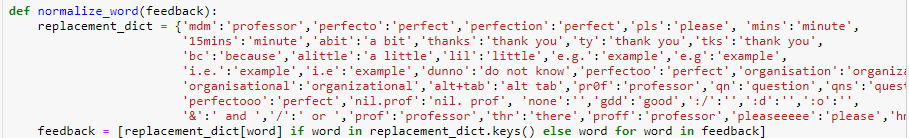
## **2.3 Data Pre-Processing**

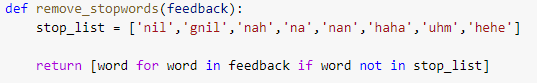
### 2.3.1 Standard Data Pre-Processing Steps

1. Lowercase words
2. Removal of punctuations and symbols
3. Removal of digits
4. Tokenization
5. Lemmatization
6. Removal of short words (1 letter words)

### 2.3.2 Data Pre-Processing Steps Specific to our Dataset

We observed that the feedback provided was mostly written informally. Hence, they had many spelling errors, improper sentence structures, abbreviations and slangs which we had to handle.

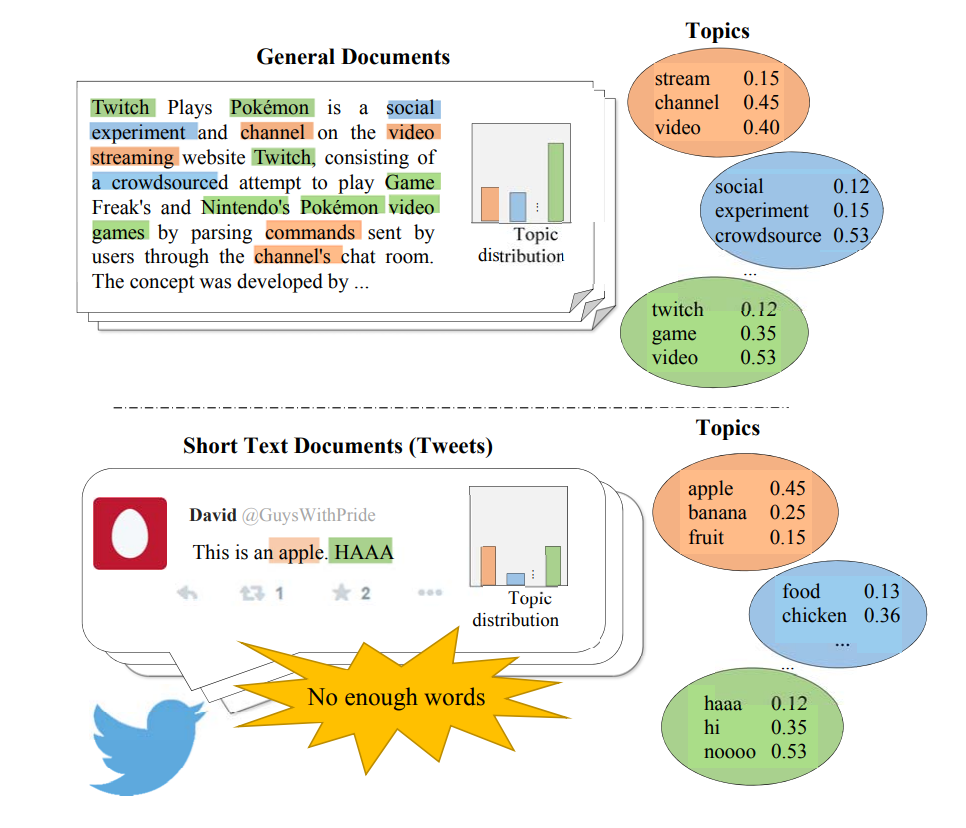
1. **Removal of null feedback**Many students leave words like “nil”, “none” and “nothing as of now” as their feedback. As these words do not add value, these kinds of feedback will require some manual pre-processing and self-declaration of stop words to remove.
2. **Removal of sequence of repeating letters**As seen from the word clouds above, certain students like to “drag” their spellings in their feedback. Hence, during our pre-processing, we must account and remedy such spellings as these words are meaningful for our analysis.
3. **Expand Acronyms **Many of the students’ feedback were written informally. Hence, slangs and acronyms of words were observed in the corpus (Refer to image above). Some examples include “pls”, “ty”, “gd” and “qns”, which are the short forms for the words “please”, “thank you”, “good” and “questions” respectively. Hence, we must identify and convert these terms into their appropriate forms.
4. **Expand Contractions**We observed that some students used contracted words such as I’ll, I don’t, I can’t while some do not. Therefore, for standardization, we expanded contraction words such as ” I’ll” to “I will”, “Don’t” to “I do not” and “I can’t” to “I cannot”, with the main goal of standardizing our corpus and reducing any discrepancies among the same words.
5. **Removing Stop words**

****We observed that some students used words such as “nil”, “gnil”, “nah”. These words do not contribute to any additional insights, thus, we decided to include these words as stop words. As mentioned in Section 2.2.3, the common stop words such as “the”,” and”, “to” and “of” are removed during topic modelling in Task 1. This will be further explained during Task 1. Due to how the students might refer to terms which are domain-specific and contain stop words in them, we did not remove these common stop words for Task 2 because these words could affect the result of our prediction model. Two examples would be “As-is Process” or “To-be Process”.

# **3.0 Data Challenges**

## **3.1 Short Text**

Based on our EDA, we realized that the feedback across all columns are in the form of short texts, with a length of less than 50 words after the text is pre-processed. However, popular topic modelling algorithms like Latent Dirichlet Allocation (LDA) has a key assumption that documents consist of various topics (Clark, 2013). It is also designed to extract topics from a corpus with long text by capturing the word co-occurrences within documents. Without this information, it is challenging to capture topics that are meaningful and coherent using LDA.

**

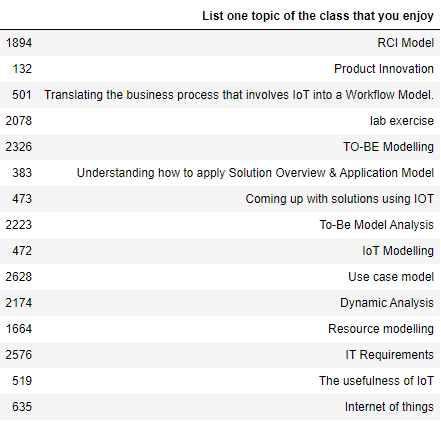
*Figure 12: Sparsity in Word Co-occurrences at Document level (Chen & Kao, 2015)*

For documents with a short text nature, there is sparsity among the word co-occurrences at the document level. Many of the words occur only once and each feedback usually consists of one short idea, unlike a traditional document where there may be multiple latent topics within a single document. Within our dataset, our vocabulary has 1134 words, while 529 words appear only once in the dataset, which takes up around 50% of our vocabulary. A random sample of these words can be found in the below diagram.



## **3.2 Single Topic Per Document**

In Figure 13 below, we observe that from a random sample of 15, most of the feedbacks span no more than 5 words. Out of the 15 random samples, only 3 feedback were more descriptive, and only feedback 501 and 383 could possibly contain more than 1 topic. The rest of the feedback were brief and only mentioned a single topic. This is mainly attributed to the nature of the feedback, where the question specifically asks for one topic, for instance the column “List one topic you enjoyed” and “What was the most challenging topic” all prompts the students to provide a singular and concise answer.

**

*Figure 13: Random sample of 15 feedbacks from the “Enjoyed topic” column*

## **3.3 Linguistic Features**

In addition to these issues, students’ feedback will often lack the linguistic features present in full documents such as word collocations. For instance, a feedback by Student123 wrote “More signavio usage”, which is concise but he may have selectively stripped off the “unnecessary” information that would otherwise make his feedback sentence contain more features, by alternatively saying “I wish to have more signavio usage so that I can be more prepared for my project” instead. Similar to the above issue, this problem may be observed due to the way the question was asked and the nature of the platform, which is a short text question box in Google Forms.

# **4.0 Task 1 (Topic Modelling)**

## **4.1 Objective**

Our main objective for Task 1 is to perform topic modelling to extract popular keywords that will help the faculty identify challenging and enjoyable aspects of the module that contributed to the students’ learning experience. From the output of the topic modelling algorithms, we implemented an automatic topic labelling system which will assign a topic label (from a predefined gazette) to the topic if it contains a certain number of keywords. This allows us to identify at a glance, which topics did the students enjoy or find challenging.

## **4.2 Solution Methodology**

### 4.2.1 Solution Overview

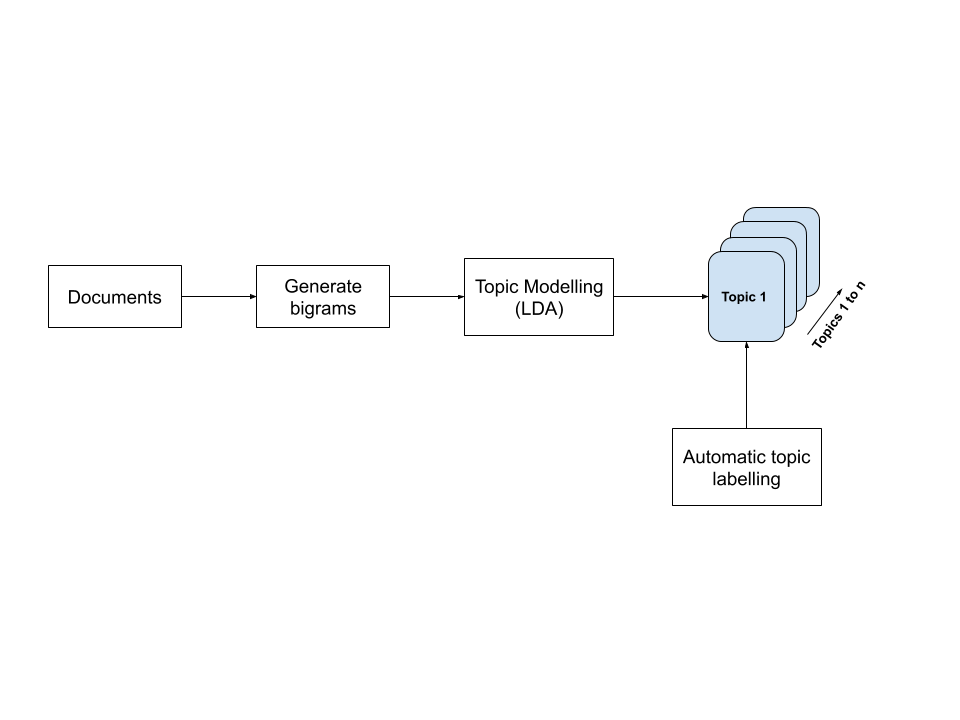
In view of the limitations that arise due to the nature of our dataset, we attempted to utilize different methodologies to generate the optimal topic model for our dataset. The same methodology was executed on 3 columns in the feedback dataset; “Enjoyable”, “Challenging” and “Suggestions”. However, we will only be examining the results of the “Enjoyable” column due to the similarity of the approach used in the other two columns. In this section, we will explain the idea behind each methodology and evaluate the performance of the model before concluding which methodology is superior in this context.

In order to make a comparison between the outputs of the different methods, we will choose a baseline model and compare the other models’ performance to it. After which, we will perform different methods of topic modelling using a data-level approach and model-level approach. In the data-level approach, we examined the aggregation of documents based on vector space similarity and aggregation of documents based on academic weeks. In the model-level approach, we used the Gibbs Sampling Algorithm for a Dirichlet Mixture Model (GSDMM).

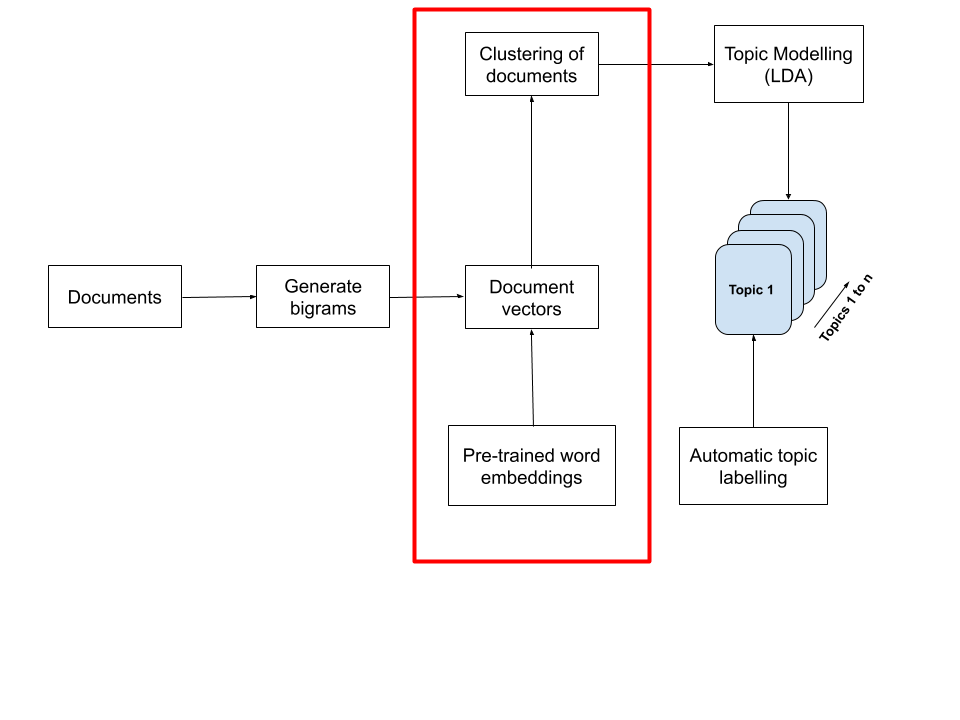
To ensure consistency across our models, we set the same parameters across all the different methods. They are as follows:

* random sample = 0
* number of topics = 10
* number of iterations = 10

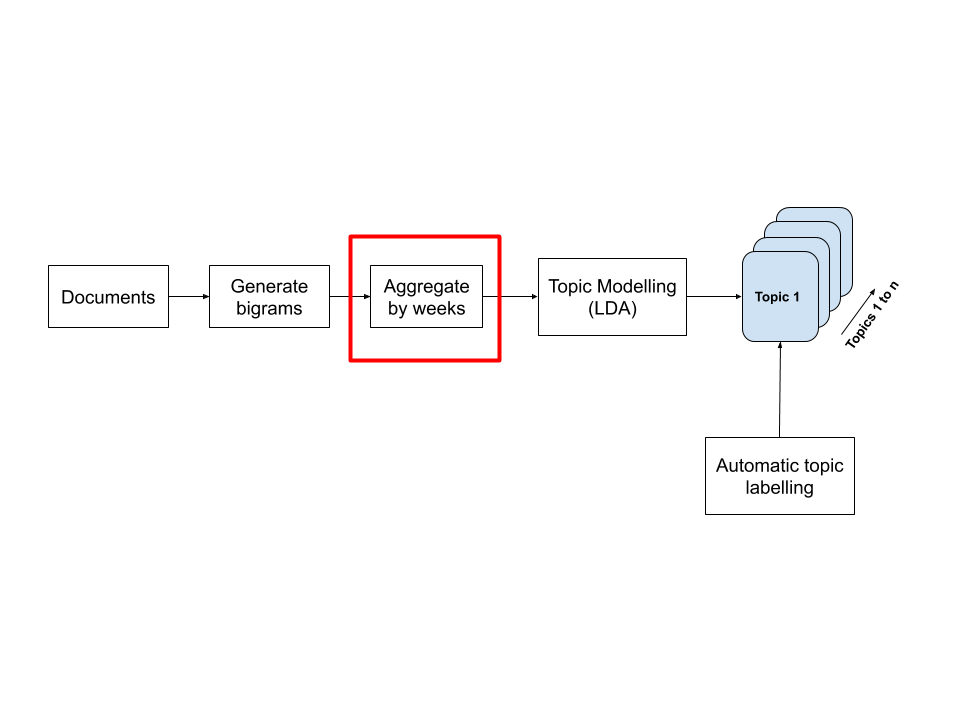
The solution overview models of each proposed approach can be seen from the diagrams below (Refer to Figure 14 to Figure 17), where the boxes highlighted in red are the changes from the baseline approach.

**

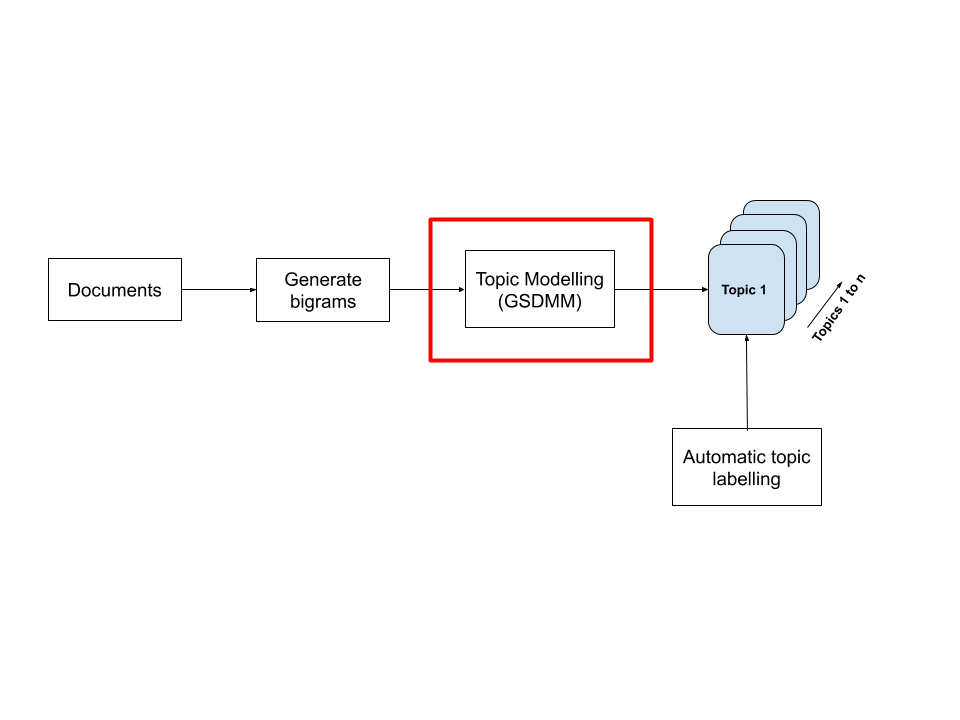
*Figure 14: Baseline approach*

**

*Figure 15: Data Level approach (Aggregating documents by vector space similarity)*

**

*Figure 16: Data Level approach (Aggregating documents by weeks)*

**

*Figure 17: Model level approach (Gibbs sampling Dirichlet Multinomial Mixtures)*

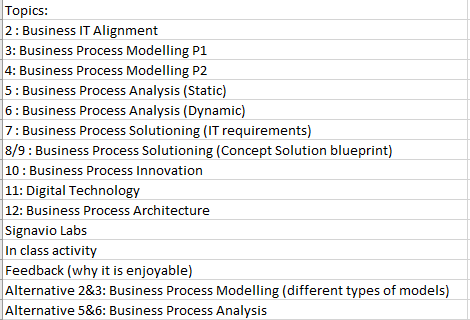
#### **4.2.1.1 Evaluation of Models**

When evaluating the models, we did not choose to use coherence score as the main evaluation method because it mainly depends on the co-occurrences of the text which was identified as one of the limitations of our dataset. Furthermore, there is no fixed way of calculating the coherence score across different models, which could create inconsistency in the calculation of the coherence score values.

Hence, we evaluated the quality of the model using human evaluations, with a mixture of qualitative and quantitative methods to evaluate the outputs of the models. They are as follows:

##### **4.2.1.1.1 Qualitative:**

Based on our predefined dictionary, we manually labelled each of the 10 topics generated for each model. These topics can be seen from Figure 18 below. For topics that did not make sense, we classified them as incoherent and highlighted them in red (refer to Figure 19). Within each manually labelled topic, the words that were incoherent with the topic were coloured in red font and regarded as noise (refer to Figure 20).

**

*Figure 18: Defined dictionary based on the BPAS topic outline and our domain knowledge*

Topic 5

0.285\*"innovation" + 0.114\*"diagram" + 0.111\*"presentation" + 0.043\*"type" + 0.039\*"product" + 0.032\*"different" + 0.027\*"package" + 0.025\*"part" + 0.024\*"technology" + 0.023\*"draw"

*Figure 19.: Example of a topic that was manually labelled as incoherent*

Topic 4 **->** **6:** **Business Process Analysis (Dynamic)**

0.221\*"signavio" + 0.168\*"analysis" + 0.126\*"workflow" + 0.083\*"dynamic" + 0.041\*"canvas" + 0.033\*"path" + 0.027\*"chain" + 0.027\*"value" + 0.026\*"cost" + 0.024\*"methodology"

*Figure 20: Example of identifying the incoherent words within a coherent topic (noise)*

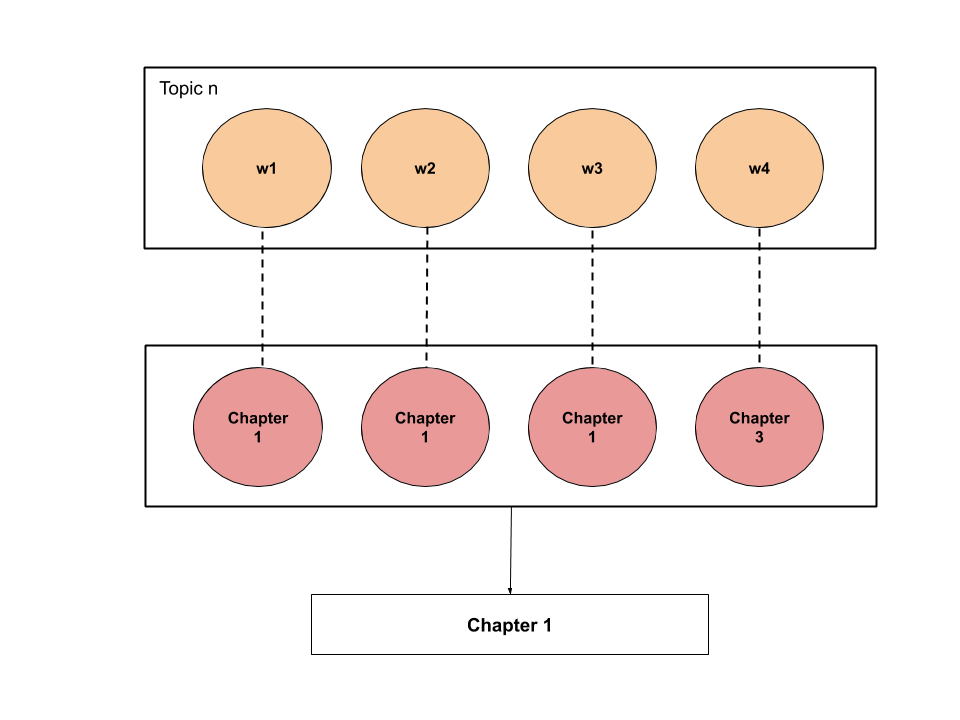
##### **4.2.1.1.2 Quantitative:**

For each model, we counted the number of incoherent topics and calculated its topic purity score using the formula below.

Based on the score, we then evaluated the outputs of all the models before concluding on which is the best model and identified any potential future improvements.

#### **4.2.1.2 Automatic topic label assigning**

A gazette was created based on the weekly topics covered in class. It was created based on the module outline of every chapter and our previous domain knowledge on the module. Keywords unique to every chapter in the course syllabus are then included in the gazette. This allows us to match each word in the topic model output to the words in the gazette, and the most suitable word that is matched is extracted along with the chapter it is tagged to. We achieved this by utilizing **fuzzy matching**, an approximate string-matching algorithm that uses Levenshtein distance to calculate how similar a word is to another word based on the edits required. Each word is tagged to a topic, and the topic that has the highest occurrence amongst all the words will be the assigned label for that topic (refer to Figure 21).

**

*Figure 21: Representation of how the topic labelling system works*

We implemented a selection criterion for the word-to-gazette matching as well as topic assigning to improve the accuracy of the topic assignment. The word from the topic must have a reasonable matching score of 40 and above, else it will be disregarded. For instance, if the topic word “presentation” does not exist in our gazette and is not similar to any other words, we will ignore it. In selecting which chapter is assigned to the topic, we choose the topic that belongs to the most matched keywords, with at least 4 or more keywords being matched to the same topic, similar to a majority vote process.

We applied this technique mainly to organize the “Enjoyable” and “Challenging” topic columns, as only these 2 columns have the most relevant keywords to our gazette. However, it also can prove useful in the “Suggestions” column as there may be several feedbacks that pertain to a specific chapter.

### 4.2.2 Baseline Model (LDA)

In addition to the mentioned common pre-processing steps, we have specifically applied some additional steps that are crucial to perform topic modelling in Task 1.

* Removing stop words plus domain stop words that appear across many topics
* Adding bi-grams to our word corpus
* Filtering out words that occur in less than 5 documents, or more than 30% of the documents.

In Task 1, we removed stop words such as “as”, “is”, “to” and “be” as it generated a lot of noise when we transformed our corpus in a bag of words even though “as-is” and “to-be” is part of the course curriculum. Additionally, these stop words appeared across multiple topics, and hence they were removed along with other words that appeared in many topics as they were not useful in differentiating between topics.

For our baseline model, we have chosen to use Latent Dirichlet Allocation (LDA) as it is simple and one of the most popular topic modelling algorithms. We also examine if generating bigrams on top of our corpus will yield a significant improvement in the representation of our corpus.

#### **4.2.2.1 Results and Analyses**

##### **4.2.2.1.1 Baseline - Unigram topic modelling results**

Topic 0 -> **8/9: Business Process Solutioning (Concept Solution blueprint)**

0.224\*"solution" + 0.158\*"application" + 0.154\*"overview" + 0.110\*"modelling" + 0.072\*"blueprint" + 0.049\*"draw" + 0.030\*"modeling" + 0.027\*"hierarchy" + 0.023\*"concept" + 0.022\*"rule"

Topic 1 -> **Feedback (why it is enjoyable)**

0.166\*"resource" + 0.095\*"enjoy" + 0.068\*"understand" + 0.039\*"make" + 0.039\*"group" + 0.035\*"come" + 0.034\*"topic" + 0.029\*"interesting" + 0.026\*"risk" + 0.026\*"easy"

Topic 2 ->

0.222\*"lab" + 0.118\*"static" + 0.116\*"analysis" + 0.043\*"functional" + 0.036\*"exercise" + 0.035\*"architecture" + 0.035\*"everything" + 0.024\*"orientation" + 0.024\*"performance" + 0.022\*"goal"

Topic 3

0.302\*"collaboration" + 0.110\*"alignment" + 0.027\*"learn" + 0.027\*"course" + 0.025\*"recap" + 0.024\*"example" + 0.022\*"go" + 0.021\*"enjoy" + 0.020\*"difference" + 0.020\*"importance"

Topic 4 -> **6:** **Business Process Analysis (Dynamic)**

0.221\*"signavio" + 0.168\*"analysis" + 0.126\*"workflow" + 0.083\*"dynamic" + 0.041\*"canvas" + 0.033\*"path" + 0.027\*"chain" + 0.027\*"value" + 0.026\*"cost" + 0.024\*"methodology"

Topic 5

0.285\*"innovation" + 0.114\*"diagram" + 0.111\*"presentation" + 0.043\*"type" + 0.039\*"product" + 0.032\*"different" + 0.027\*"package" + 0.025\*"part" + 0.024\*"technology" + 0.023\*"draw"

Topic 6 -> **In class activity**

0.325\*"iot" + 0.094\*"study" + 0.040\*"function" + 0.036\*"executive" + 0.035\*"challenge" + 0.026\*"understand" + 0.023\*"interest" + 0.018\*"food" + 0.017\*"enjoy" + 0.015\*"improve"

Topic 7 -> **Alternative 2&3: Business Process Modelling (different types of models)**

0.120\*"rci" + 0.112\*"location" + 0.087\*"organization" + 0.044\*"triangle" + 0.042\*"learn" + 0.039\*"magic" + 0.033\*"discussion" + 0.030\*"organizational" + 0.029\*"gassmann" + 0.029\*"class"

Topic 8 -> **11: Digital Technology**

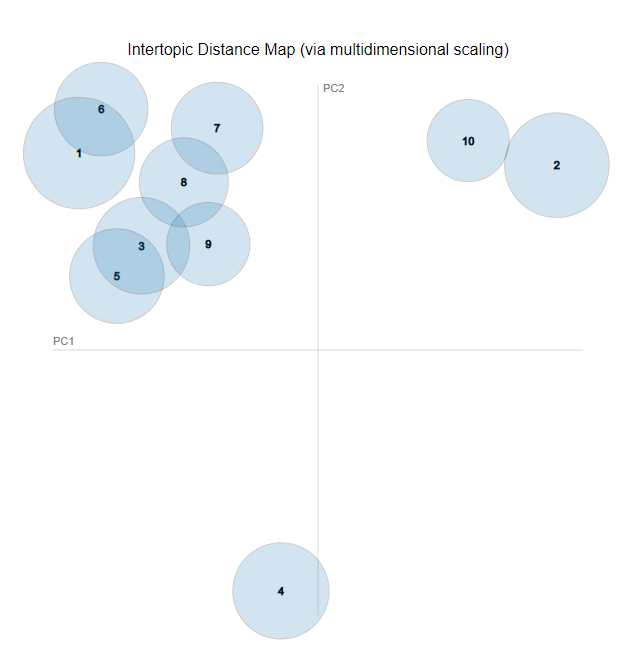
0.216\*"learn" + 0.128\*"different" + 0.057\*"enjoy" + 0.055\*"trend" + 0.031\*"type" + 0.029\*"like" + 0.028\*"digital" + 0.023\*"thing" + 0.019\*"technology" + 0.018\*"various"

Topic 9 -> **5: Business Process Analysis (Static)**

0.096\*"cause" + 0.086\*"root" + 0.082\*"activity" + 0.058\*"class" + 0.055\*"impact" + 0.041\*"management" + 0.040\*"identify" + 0.038\*"create" + 0.034\*"flow" + 0.032\*"issue"

|  |  |
| --- | --- |
| **Average Topic Purity Score:** | 0.87143 |
| **Number of Incoherent Topics:** | 3 |

We observe that the baseline model using unigram performs relatively well as it can create several different coherent topics, although there is some noise in the topics generated. However, some standalone keywords like “analysis” as shown in topic 4 might not really add value in differentiating between topics as it can represent different topics, like static analysis and dynamic analysis, and must be accompanied with other key words to differentiate between the topics. Additionally, we see words appearing together that would make more sense if they were word phrases, such as “root” and “cause” (topic 9), “digital” and “thing” (topic 8). Hence, using key phrases like bigrams might be more applicable for Task 1. From the result above, we can see that there are 3 incoherent topics (topics 2,3 and 5) that contain too many terms belonging to different topics. This is supported with the inter-topic distance map, where it appears that there are many overlaps in the clusters, creating incoherent topics. (Refer to Figure 22)

**

*Figure 22: Distance map of topics (unigram)*

##### **4.2.2.1.2 Baseline - Bigram topic modelling results**

Topic 0 -> **Alternative 2&3: Business Process Modelling (different types of models)**

['root\_cause', 'impact', 'organizational', 'executive', 'location', 'learn', 'different', 'value\_chain', 'iot', 'understand']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 1 -> **In class activity**

['collaboration', 'solution\_overview', 'presentation', 'canvas', 'digital', 'create', 'trend', 'product', 'resource', 'innovation']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 2 -> **In class activity**

['application', 'organization', 'class', 'location', 'activity', 'draw', 'flow', 'rcr', 'discussion', 'work']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 3

['iot', 'resource', 'function', 'trend', 'technology', 'report', 'challenge', 'risk', 'different', 'like']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 4 -> **In class activity**

['study', 'enjoy', 'blueprint', 'learn', 'identify', 'magic\_triangle', 'management', 'gassmann', 'presentation', 'issue']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 5 -> **Signavio Labs**

['signavio', 'innovation', 'learn', 'technique', 'analysis', 'hand', 'tool', 'static', 'face', 'lot']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 6 ->

['rci', 'diagram', 'type', 'different', 'understand', 'package', 'group', 'iot', 'make', 'generate']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 7 -> **Alternative 5&6: Business Process Analysis**

['analysis', 'modelling', 'dynamic\_analysis', 'static', 'path', 'methodology', 'everything', 'impact', 'example', 'revision']

Labelled chapter: 5 : Business Process Analysis (Static)

Keywords appearing in both chapter and topic: ['analysis', 'static', 'path', 'impact']

Topic 8 -> **2: Business IT Alignment**

['learn', 'different', 'alignment', 'concept', 'enjoy', 'type', 'interesting', 'framework', 'importance', 'goal']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 9 -> **Signavio Labs**

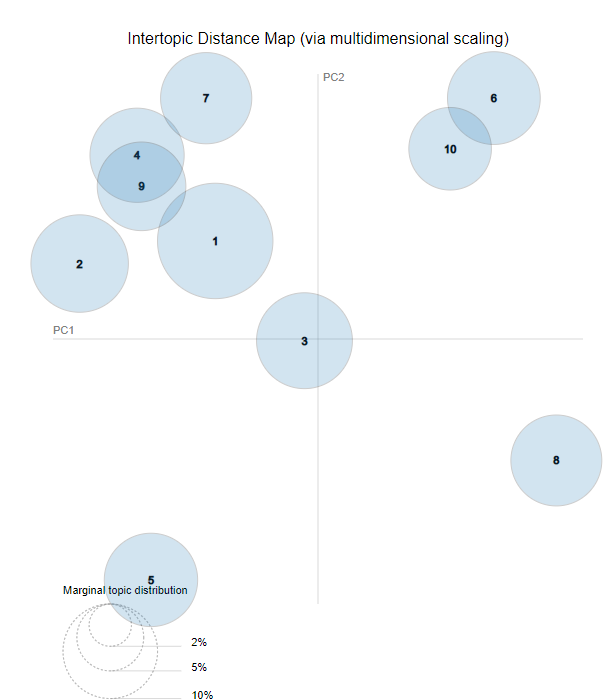
['workflow', 'lab', 'enjoy', 'signavio', 'topic', 'review', 'hierarchy', 'modeling', 'thing', 'cost']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

|  |  |
| --- | --- |
| **Average Topic Purity Score:** | 0.90 |
| **Number of Incoherent Topics:** | 2 |

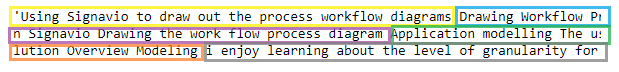
The baseline model including bigrams performs better than the baseline model consisting of only unigrams. This can be seen from its higher topic purity score and lower number of incoherent topics. Furthermore, we can see key phrases being formed such as “magic\_triangle”, “solution\_overview”, “dynamic\_analysis” which helps to differentiate between topics better as these phrases are unique keywords that help to define the topics. This improvement in results can also be seen from the inter-topic distance map where the clusters are more separated now as compared to the unigram model. As such, we will be using bigrams for the rest of the models as it is a better representation of our corpus.

**

*Figure 23: Distance map of topics (bigrams)*

### 4.2.3 Data-level Approach: Aggregation of Documents

In this approach, we aggregate the documents to form a larger pseudo-document of higher word counts, based on how similar the documents are. When the documents are aggregated, they will have a higher word count per document which alleviates the issue of word count sparsity at the document level. Figure 24 is a sample of an aggregated document by weeks. The different colours represent different feedbacks by different students for that week, which are then merged into one document.

**

*Figure 24: Aggregated feedback into a single pseudo-document*

There are several methods proposed to aggregate these documents that have also seen success in the context of social media. However, we will focus on 2 approaches, namely aggregating based on their similarity in a vector space and aggregating by the academic weeks of the feedback pertaining to the lesson of that week.

#### **4.2.3.1 Data-level Approach: Aggregation by Vector Space Similarity**

##### **4.2.3.1.1 Literature review**

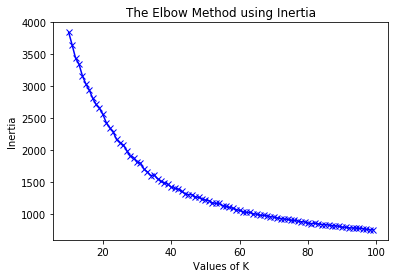
Topic Modeling over Short Texts by Incorporating Word Embeddings (Qiang et al, 2017)

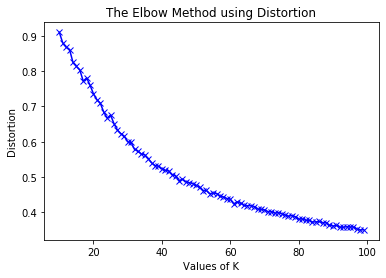
In this approach proposed by Qiang, Wang, Chen and Wu , they targeted the issue of generalizability in aggregating short text documents, since most papers suggested aggregating documents through heuristics such as the hashtags in Twitter or by the user of the post. However, such heuristics may not always be available across all domains. There is also the issue of limited word co-occurrences information, which is apparent in short texts as discussed. Hence, the proposed model, Embedding-Based Topic Modelling (ETM) is to train word embeddings from a large corpus, and leveraging on the semantic knowledge from the embeddings, the documents are clustered and aggregated to form larger documents. The topics are then inferred using a Markov Random Field regularized model, which assigns words that are semantically similar to the same topic. The results are then compared with baseline models such as Biterm Topic Modelling(BTM) and Dirichlet Multinomial Mixture(DMM). Coherence Measure scores on the ETM averaged 72.5% compared to their baseline models of roughly 58%.

##### **4.2.3.1.2 Methodology**

Drawing reference from the approach proposed above, this model leverages the use of semantic structure in our text to form better topics since our dataset has limited word co-occurrences. We first convert our documents into word embeddings to retain their semantics, then cluster them so we can form larger documents out of these clusters, which we will then pass into our LDA model.

Due to the lack of dataset we have to train our own word embeddings, we will leverage on transferred learning to convert our documents into vector representation for our dataset. We achieve this by inferring our documents from the pre-trained word embeddings provided by Google, trained on Google news dataset of about 100 billion words (Google Code, 2013). There are over 3 million words and phrases, and has a 300-dimensional output. However, we only can infer words using this model, as it was trained on words. Hence, we will represent our documents, which comes in the form of sentences. This approach requires us to take the word vectors of all the words in our document, and averaging it to get our document representation.

Next, we will perform clustering on our converted documents. We chose k-means clustering as it was a simple and efficient algorithm to implement. However, it does not perform well on high-dimensional datasets. Hence, we reduced the dimensions of the word vectors using principal component analysis. We are able to obtain a 68% explained variance after reducing it to 30 features, which is a reasonable trade-off to avoid the curse of the dimensionality. In selecting our number of clusters, k, we do not have a clear selection point. we attempted to use both distortion and inertia to determine our k.



The above diagrams show clearly that there is no clear elbow to choose. This may be due to the nature of the dataset having no distinction in the features, as our dataset is highly domain specific and might not make sense when converted into vector space trained on a Google news dataset. Hence, we are unable to form distinguishable clusters. We selected k=20 as there seems to be a small elbow, as well as being a close number to how many chapters we have for the syllabus. The documents belonging to the same clusters are then joined together to form a bigger pseudo-document, so in total we have 20 documents to pass into our LDA model.

##### **4.2.3.1.3 Results and Analyses**

**Aggregated documents – vector space similarity**

Topic 0 -> **Alternative 2&3: Business Process Modelling (different types of models)**

['solution\_overview', 'location', 'rci', 'organization', 'canvas', 'different', 'draw', 'executive', 'function', 'iot']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 1 -> **In class activity**

['presentation', 'diagram', 'draw', 'signavio', 'class', 'study', 'group', 'drawing', 'create', 'discussion']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 2 -> **8/9 : Business Process Solutioning (Concept Solution blueprint)**

['application', 'iot', 'modelling', 'solution\_overview', 'signavio', 'understand', 'blueprint', 'modeling', 'learn', 'create']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 3 -> **Signavio Labs**

['iot', 'signavio', 'trend', 'report', 'static', 'analysis', 'digital', 'management', 'generate', 'workflow']

Labelled chapter: 5 : Business Process Analysis (Static)

Keywords appearing in both chapter and topic: ['trend', 'report', 'static', 'analysis']

Topic 4 ->

['blueprint', 'study', 'concept', 'framework', 'iot', 'understand', 'enjoy', 'analysis', 'interest', 'know']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 5 -> **Alternative 5&6: Business Process Analysis**

['analysis', 'innovation', 'collaboration', 'signavio', 'dynamic\_analysis', 'modelling', 'static', 'rci', 'methodology', 'product']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 6 -> **Feedback (why it is enjoyable)**

['enjoy', 'different', 'learn', 'class', 'iot', 'understand', 'activity', 'topic', 'help', 'like']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 7 -> **Signavio labs**

['analysis', 'different', 'learn', 'dynamic\_analysis', 'static', 'enjoy', 'location', 'iot', 'signavio', 'class']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 8 ->

['learn', 'alignment', 'signavio', 'iot', 'different', 'enjoy', 'type', 'innovation', 'root\_cause', 'find']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 9 -> **Signavio Labs**

['workflow', 'lab', 'resource', 'signavio', 'modeling', 'draw', 'exercise', 'diagram', 'modelling', 'create']

Labelled chapter: 4: Business Process Modelling P2

Keywords appearing in both chapter and topic: ['workflow', 'resource', 'signavio', 'modeling', 'modelling']

|  |  |
| --- | --- |
| **Average Topic Purity Score:** | 0.80 |
| **Number of Incoherent Topics:** | 2 |

The aggregation by vector space similarity model performs worse than the baseline model as it has a lower average topic purity score and an equal number of incoherent topics. The topics obtained generally have more noise, as we see words like “iot” appear across several topics that are not related to digital innovation.

#### **4.2.3.2 Data-level Approach: Aggregation by Academic Weeks**

##### **4.2.3.2.1 Literature review**

Improving LDA Topic Models for Microblogs via Tweet Pooling and Automatic Labeling   
(Mehrotra et al, 2013)

In this paper, the authors investigate different ways to aggregate documents in the form of Tweets to improve topic coherence. The approach, “tweet pooling”, is to merge related tweets into one single document to be presented to the LDA model. This can counter the issue of sparse co-occurrences as the resulting document will have more enriching information compared to just a single tweet. Several heuristics were proposed to aggregate these documents.

One strategy was to pool the tweets according to the author, known as Author-wise pooling. This method results in each document belonging only to one author, with the intuition that each author will only tweet about a fixed set of topics. For example, Elon Musk will mostly tweet about science-related topics while Selena Gomez will mostly talk about the music industry.

Another method is Hashtag-based pooling, which results in merging tweets that have similar ideas or content since hashtags are topical markers themselves. For instance, a tweet with the hashtag “#internationalwomensday” will have content that is related to this event. Likewise, one document will represent only one hashtag as well.

The results that they achieved on Purity scores were slightly higher than their baseline, which was the unpooled tweets in its raw form. The scores were 0.54 and 0.54 for hashtag and author-pooled tweets respectively compared to the 0.49 of the baseline.

##### **4.2.3.2.2 Methodology**

Following the proposed approach, we attempt to aggregate our feedback documents based on the week that was submitted. Using the submitted week to pool our documents is to ensure that feedbacks of similar content and nature are in the same documents, as the feedback submitted during that week is pertaining to the materials of what they have learnt in-class. Hence, we attempt to aggregate by the weeks which feedbacks were submitted, excluding recess week 8 and week 12 where they had a break. Our text corpus is then represented in the form of 10 documents, with each document pertaining to a week of the class and consisting of all feedback submitted in that week. The corpus is then passed into our LDA model.

##### **4.2.3.2.3 Results and Analyses**

**Aggregated documents – weeks**

Topic 0 – **Alternative 2&3: Business Process Modelling (different types of models)**

['location', 'organization', 'alignment', 'different', 'modelling', 'learn', 'organizational', 'enjoy', 'understand', 'activity']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 1 -> **In class activity**

['innovation', 'study', 'canvas', 'presentation', 'product', 'learn', 'gassmann', 'type', 'different', 'class']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 2 -> **Alternative 2&3: Business Process Modelling (different types of models)**

['location', 'innovation', 'organization', 'alignment', 'learn', 'different', 'application', 'enjoy', 'modelling', 'organizational']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 3 -> **In class activity**

['iot', 'trend', 'learn', 'presentation', 'different', 'innovation', 'technology', 'application', 'digital', 'enjoy']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 4 ->

['analysis', 'application', 'blueprint', 'lab', 'path', 'learn', 'modelling', 'understand', 'enjoy', 'technique']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 5 -> **Alternative 2&3: Business Process Modelling (different types of models)**

['collaboration', 'resource', 'workflow', 'different', 'learn', 'enjoy', 'lab', 'understand', 'class', 'diagram']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 6 -> **Alternative 5&6: Business Process Analysis**

['signavio', 'rci', 'static', 'lab', 'report', 'analysis', 'impact', 'learn', 'issue', 'rcr']

Labelled chapter: 5 : Business Process Analysis (Static)

Keywords appearing in both chapter and topic: ['rci', 'static', 'report', 'analysis', 'impact', 'issue', 'rcr']

Topic 7 -> **Alternative 2&3: Business Process Modelling (different types of models)**

['analysis', 'collaboration', 'learn', 'location', 'different', 'resource', 'enjoy', 'workflow', 'modelling', 'organization']

Labelled chapter: 4: Business Process Modelling P2

Keywords appearing in both chapter and topic: ['collaboration', 'resource', 'workflow', 'modelling']

Topic 8 – **Signavio Labs**

['signavio', 'workflow', 'learn', 'draw', 'diagram', 'function', 'lab', 'enjoy', 'different', 'rule']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 9 -

['hierarchy', 'executive', 'review', 'course', 'architecture', 'learn', 'revision', 'exam', 'enterprise', 'topic']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

|  |  |
| --- | --- |
| **Average Topic Purity Score:** | 0.9125 |
| **Number of Incoherent Topics:** | 2 |

The aggregation by academic weeks model performs better than the baseline model as it has a higher average topic purity score and an equal number of incoherent topics. The topics obtained are generally coherent as the keywords within the topic can mostly be linked to the topic label.

### 4.2.4 Model-level approach: Gibbs Sampling Algorithm for a Dirichlet Mixture Model (GSDMM)

This approach, which involves trying a different model: Dirichlet Mixture Model (DMM). It is advantageous in our context as it follows the assumption that each document has only one latent topic. Considering that our feedback has short text counts, it would be reasonable to assume that our feedback holds only one topic, as described and shown in the earlier sections.

#### **4.2.4.1 Literature review**

A Dirichlet Multinomial Mixture Model-based Approach for Short Text Clustering (Yin and Wang, 2014)

The approach proposed by Yin and Wang (2014), is able to tackle the issue of sparse data in short texts, and able to converge quickly as well as being able to detect the number of clusters, requiring only an upper bound on the number of clusters, k.

The paper demonstrated the algorithm using an analogy of a Movie Group Process. There is a class filled with students and each student has a list of their favourite movies. Each student is randomly assigned to tables at the start of the class. The professor repeatedly reads the class list. Each time the student is called, he will select a new table that satisfies the 2 conditions: the new table should have more students than his current table or the table has students with similar lists of movies. As this process continues, some tables will eventually vanish and others will have students that share similar interests.

To state it formally in terms of our problem: the students refer to our documents, which are the weekly feedbacks submitted by each student each week, and the list of movies are the words in the feedback documents. The “tables” that the documents are assigned to are the topics that we define at the start as a parameter.

It is noted that the approach performed better than the baseline model of K-means clustering and DMAFP on metrics like Homogeneity and completeness, scoring 0.853 and 0.896 respectively compared to K-means, which scored 0.692 and 0.775. This can be attributed to the nature of the algorithm, which can infer the number of topics automatically while balancing the trade-offs in homogeneity and completeness.

#### **4.2.4.2 Methodology**

In this approach, we attempt to change the model to suit the task of short text clustering. Following the proposed algorithm, our corpus (generated with bigrams), is fed into the GSDMM model. An existing implementation is found on [GitHub](https://github.com/rwalk/gsdmm) (2017). As GSDMM assumes each feedback document has only 1 topic, it is suitable for our dataset because of the nature of the feedback form and the length distribution as shown in our EDA which conforms to that assumption.

**Parameter selection**

There are 2 parameters that require our tuning and selection for this algorithm. The first parameter, α, influences the probability of a document being assigned to an empty cluster. When α increases, the probability of a document choosing an empty cluster increases, and if we set it too large, it may result in having many clusters with only 1 document. Likewise, if it is an extremely low number, no documents will ever choose a cluster when it is empty and hence it will be discarded once empty. In the paper, the α performed in a stable manner for 3 of their testing datasets across all values 0<α<1. However, the performance will decrease slightly when the α gets larger. Hence, we will select a default α=0.1.

The second parameter, β, influences the probability that the document chooses a cluster based on similarity rather than the number of documents in that cluster. When β increases, the number of clusters found decreases and we will result in a less homogeneous cluster and higher completeness, and vice versa. Since homogeneity is important to us because of how our topic labelling algorithm works (explained later, but general idea is the more words that belong to a topic, the higher the chance to be automatically and correctly labelled ), we will set β to be a lower value than the default (0.1), at β=0.05

#### **4.2.4.3 Results and Analyses**

**GSDMM**

Topic 0 – **In class activity**

['innovation', 'understand', 'presentation', 'value\_chain', 'class', 'enjoy', 'learn', 'different', 'study', 'iot']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 1 -> **In class activity**

['challenge', 'application', 'product', 'study', 'type', 'presentation', 'different', 'learn', 'innovation', 'iot']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 2 -> **Signavio Labs**

['fun', 'activity', 'workflow', 'class', 'lab', 'resource', 'signavio', 'understand', 'enjoy', 'collaboration']

Labelled chapter: 4: Business Process Modelling P2

Keywords appearing in both chapter and topic: ['workflow', 'resource', 'signavio', 'collaboration']

Topic 3 -> **Alternative 5&6: Business Process Analysis**

['rcr', 'methodology', 'technique', 'path', 'static', 'impact', 'root\_cause', 'rci', 'dynamic\_analysis', 'analysis']

Labelled chapter: 5 : Business Process Analysis (Static)

Keywords appearing in both chapter and topic: ['rcr', 'path', 'static', 'impact', 'root\_cause', 'rci', 'analysis']

Topic 4 -> **Feedback (why it is enjoyable)**

['revision', 'enjoy', 'course', 'package', 'review', 'learn', 'signavio', 'resource', 'lab', 'collaboration']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 5: **Signavio Labs**

['lab', 'goal', 'performance', 'internet', 'thing', 'generate', 'analysis', 'report', 'static', 'signavio']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 6 ->

['different', 'hierarchy', 'presentation', 'diagram', 'flow', 'functional', 'enjoy', 'work', 'learn', 'alignment']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 7 -> **Alternative 2&3: Business Process Modelling (different types of models)**

['signavio', 'modelling', 'enjoy', 'resource', 'understand', 'different', 'organizational', 'learn', 'organization', 'location']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 8 -> **Alternative 2&3: Business Process Modelling (different types of models)**

['diagram', 'resource', 'draw', 'collaboration', 'blueprint', 'modelling', 'application', 'solution\_overview', 'workflow', 'signavio']

Labelled chapter: 4: Business Process Modelling P2

Keywords appearing in both chapter and topic: ['resource', 'collaboration', 'modelling', 'workflow', 'signavio']

Topic 9 -> **11: Digital Technology**

['different', 'workflow', 'digital', 'magic\_triangle', 'technology', 'canvas', 'learn', 'iot', 'trend', 'innovation']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

|  |  |
| --- | --- |
| **Average Topic Purity Score:** | 0.92222 |
| **Number of Incoherent Topics:** | 1 |

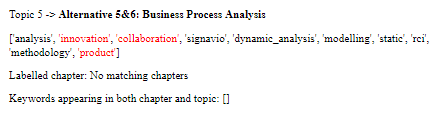
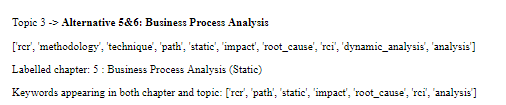
The GSDMM model performs better than the baseline model as it has a high topic purity score as well as a lower number of incoherent topics. The topics obtained generally do not have much noise as the words in the topic can be linked to the assigned topic label.

## **4.3 Results and Analyses**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Score** | **Baseline (unigram)** | **Baseline (bigram)** | **Aggregation - vector similarity** | **Aggregation - academic weeks** | **GSDMM** |
| **Average Topic Purity Score** | 0.87143 | 0.9 | 0.8 | 0.9125 | 0.92222 |
| **Incoherent Topics** | 3 | 2 | 2 | 2 | 1 |

*Figure 25: Overall results across all the models based on our evaluation metrics*

Comparing the overall results across the models (Refer to Figure 25), GSDMM and aggregating the documents by academic weeks model performed better than the baseline model. However, GSDMM is the best performing model as it has the highest average topic purity score and the lowest number of incoherent topics.

**

*Figure 26: Comparison between similar topics returned from the GSDMM model and Aggregation by Vector Space Model*

The strong performance of the GSDMM model can be seen when we examine its topic modelling results against the other models. For comparison, both topics were manually labelled as the same topic, “Alternative 5&6 Business Process Analysis”. However, the topic returned by GSDMM as shown in topic 3 has no identified noise unlike topic 5 which is obtained from the aggregation by vector space model (Refer to Figure 26). This is seen by the GSDMM model’s grouping of related terms such as static and dynamic analysis and acronyms like “rcr” and “rci” together, which are all related terms in the module outline whilst not having any unrelated terms such as “innovation” in it.

### *Why are the results useful, pertaining to the use case?*

The topics obtained can be useful for the faculty to know what topics are enjoyable, so that they are able to extract these aspects and build on them, as having these aspects will likely make the class enjoyable, improving the learning experience for the students. For topics that return content about the syllabus, it may not be that useful as it still has to be taught as it is in the course curriculum but then things outside it such as the medium used to teach the content, are valuable feedback for the faculty to gather feedback regarding their teaching style, so that they can make any necessary tweaks. Moreover, feedback topics that are obtained can contain aspects such as “recaps” and “reviews”, which are components that the faculty can strive to incorporate into their teaching methodology so that the course experience is more enjoyable.

## **4.4 Gap Analysis and Discussion**

### 4.4.1 Limitations of our evaluation methodology

Due to the different approaches we have attempted, we do not have a direct approach to measure the quality of the topics generated quantitatively. We are using an implementation of the GSDMM that is publicly available on GitHub, which is implemented without having any scoring metrics. Hence, we had to use self-defined heuristics such as average topic purity score and number of incoherent topics to measure the quality of individual topics. However, these self-defined heuristics are subjective in nature as it depends on the manual qualitative evaluation of our group members.

Additionally, due to the data sparsity challenge, measures like coherence score are not suitable or practical to implement for evaluations. As coherence score measures the similarities between the top words in the topic, we cannot use co-occurrences of the words as a measure of similarity due to the data sparsity issue. We have considered converting the documents to word vectors to calculate semantic similarity, but that also has its implications due to the limited size of our dataset, which only has 300 records and is too little to be trained on word vectors. Hence, we had to resort to using mostly qualitative-based approaches to evaluate our topics.

As such, our evaluation may not be the most accurate as our domain knowledge may not be enough to help us interpret all the topics correctly, leading us to incorrectly label topics and words as incoherent.

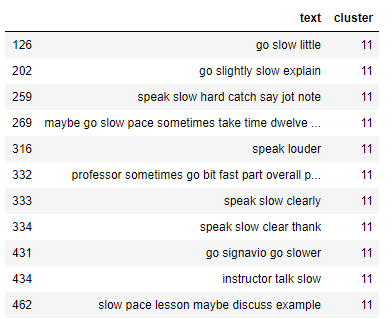
### 4.4.2 Discussion on Data Level Approach: Aggregation of Documents

#### **4.4.2.1 Aggregation by vector space similarity**

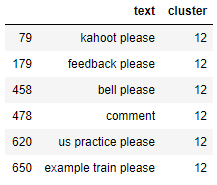
The results of the different approaches vary. In our data level approach, we observe a better performance when aggregating our documents by the weeks instead of aggregation by vector space similarity. One reason is that the documents formed using our clusters may only be meaningful in some clusters while mostly incoherent for the rest. For instance, in cluster 11 (Refer to Figure 27), it aggregated all the feedback that requests for a slower pace of speaking, which seems to align with our intent of having documents of similar meaning into the same cluster. However, in cluster 12 (Refer to Figure 28), it seems like the cluster was formed on the keyword “please” while cluster 2 clusters based on the words related to “time”, such as “Go faster pace” or “Presentation time limit”, which may not be similar topics even though they are related to the notion of “time” (Refer to Figure 29). We can clearly distinguish that one is talking about the pace of the lesson while the other is solely about the in-class presentation segment.

Another possible reason might be due to the quality of our word embeddings. As we are using transferred learning by using pre-trained word vectors, which was trained on external sources like Google News documents, that may not be relevant to our domain, which is academic and specifically on the BPAS module. Hence, when converting our documents to word embeddings, we only can infer words that are in the vocabulary of the model, and many domain-specific words like “as-is model” will not be inferred as they are unlikely to appear in the vocab of our pre-trained model. Hence, the vectors inferred are highly dependent only on known words in the sentence.

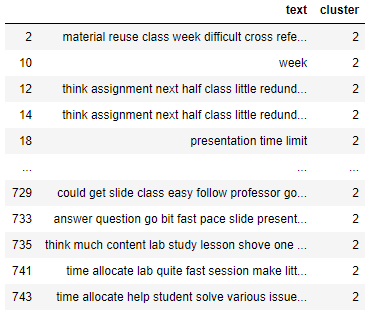
To improve the performance of this approach, we can train our own word embeddings to obtain meaningful and representative word vectors. This allows us to form meaningful clusters which can then improve the issue of having similar words in the same topic even when the documents do not share similar words, as they are now indirectly represented using vector space similarity. However, this requires a sizable amount of data before we are able to properly train the word embeddings.

**

*Figure 27: Cluster 11 documents*

**

*Figure 28: Cluster 12 documents*

**

*Figure 29: Cluster 2 documents*

#### **4.4.2.2 Aggregation on academic weeks**

When aggregating by the weeks, we will not be able to perfectly segregate the topics either. A particular week’s class may not contain just one topic, as they may be a continuation from last week’s chapter or may even be a revision week. However, we have still obtained reasonable results compared to the baseline and aggregation by vector similarity, which indicates high potential to explore aggregation using known characteristics on our dataset. For instance, we can aggregate on both academic weeks as well as the classes which the student belongs to, as different professors and different classes will have different topics of discussion, especially during the case study presentation week.

### 4.4.3 Discussion on Dirichlet Mixture Model (DMM)

DMM is the highest performing approach among our other approaches. Due to the single-topic assumption, it is able to model our data well. We are also able to specify an upper bound for the number of clusters and the algorithm will eliminate clusters that are not populated after the training process, which effectively allows us to achieve a rather optimal number of clusters automatically. The algorithm also converges fast, with the recommended iterations to be set at 5.

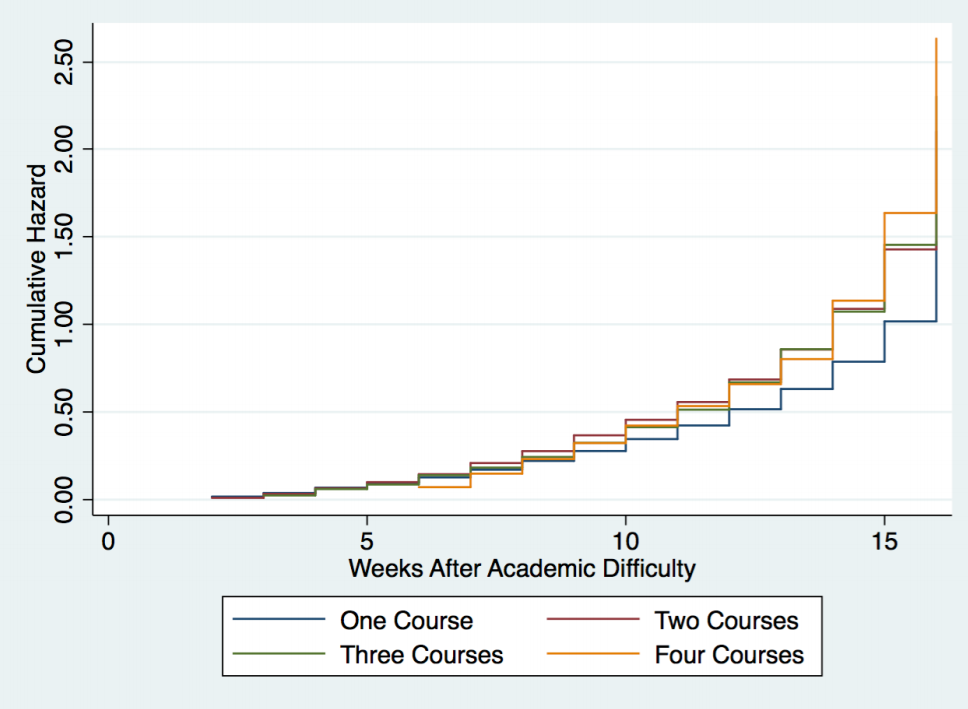
However, since DMM uses word occurrences to determine similarity to predict the topic they belong to, they still suffer from the data sparsity issue. Words that are semantically related are unable to be captured due to low co-occurrences. Hence, by using word embeddings to represent our documents, we can incorporate them into DMM, which has been proposed multiple times (Qiang et al, 2019).In this approach by (Li et al, 2017), they extended the traditional DMM to include word embeddings to capture semantic relatedness between words. The algorithm samples a topic from a document, and words highly related are selected. Auxiliary word embeddings are used to promote semantically related words, which will form better topics as words that are highly semantically related but have little co-occurrences will still be grouped together. Hence, we can leverage word embeddings in this manner to solve the sparsity issue.

## **4.5 Future Work**

### 4.5.1 Early warning of crucial chapters through recommender systems

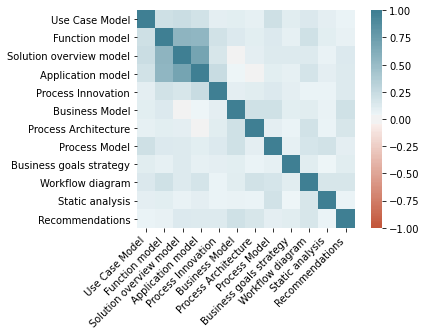
Currently, the use of the topic modelling is only to detect the dominant topics that should be of attention to the professor, which is important especially for the “Challenging” and “Suggestions” columns. The professor can then address the classes or even individuals who mention that this particular topic is challenging, for instance. However, there may be an underlying reason as to why they find it challenging, or even any additional topics they may have trouble with. Viewing which topics students have trouble with is just the first step of assisting students, but to understand and tackle the root issue, we may have to dig deeper.

In a study about the factors influencing the academic performances of students, researchers found that if a student is experiencing difficulties in a prerequisite course, there is an increased likelihood that they will experience difficulties in other similar courses, which is termed as the “snowball effect” (DeMonbrun et al, 2018). In the diagram below, the results suggest that having difficulty in 2 or more courses will exponentially increase the risk of low academic performance.



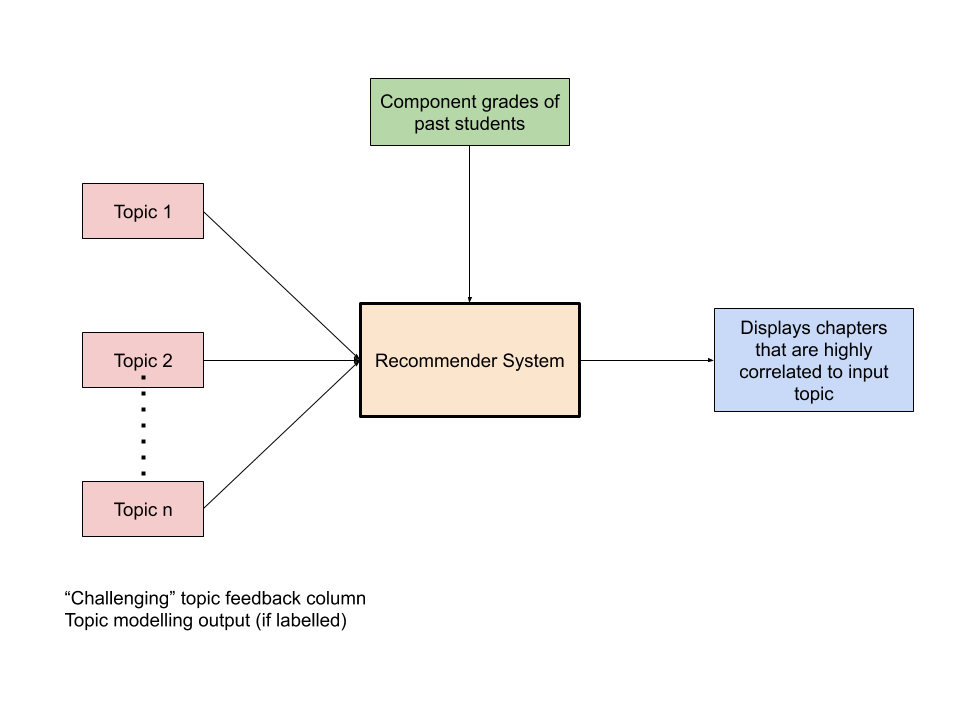
*Figure 30: Course difficulty for students (DeMonbrun et al, 2018)*

Hence, it is essential for us to know when the students are struggling in a key topic that forms a crucial foundation in other topics, so we are able to understand and provide specific resources to allow them to overcome these difficulties before their academic debt load becomes too overwhelming. We first plot a correlation matrix of their component grades pertaining to each chapter and observe that there are correlations between some chapters. In Figure 31, it shows that if you do well in either Function, Solution overview, or Application model, you are likely to do well in the remaining models.

******

*Figure 31: Correlation matrix of component grades*

Hence, after identification of dominant topics through our current topic modelling approach, the chapters in the topics are then passed as input to a recommender system, where it checks for high correlations in grades for the chapters against all other chapters. If the correlation exceeds a certain threshold, the model will alert the teaching team about the potential influence this chapter has on affected chapters. They can then use their domain judgement and take the necessary pre-emptive measures to address the topics for weaker students.

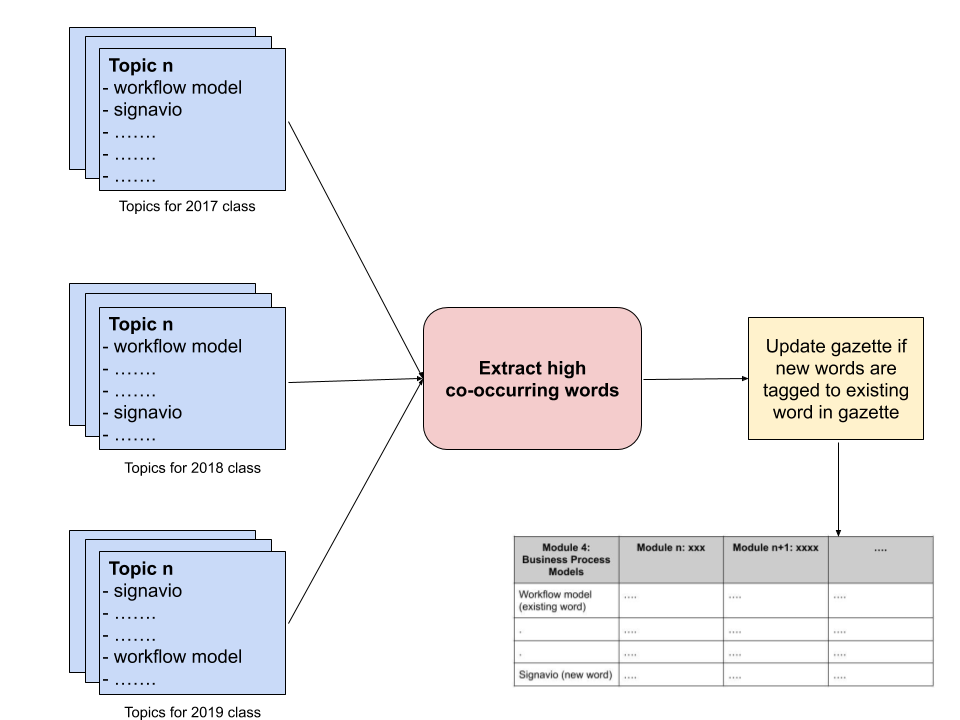


*Figure 32: Process flow of recommender system implementation*

### 4.5.2 Implementation of a scalable and continually developing gazette

In labelling our topics automatically, we used a gazette to match keywords from each chapter. Our current gazette is extracted from the module outlines for each chapter’s slides and supplemented with our domain knowledge by subjectively extracting keywords from each chapter that uniquely defines them. However, this proves to be inefficient, as it will require the manual checking of every chapter’s slides. It will also not scale well if the content of the module increases and if this topic modelling approach is applied on other modules, as it would first require substantial domain knowledge of the module as well as time spent to evaluate the quality of the gazette.

Hence, a gazette that automatically extracts relevant words and updates itself will be important to ensure that the topic models are correctly labelled. Every time the topic modelling algorithm produces new topics from new batches of students, we store these words together in a database. After a few batches of students, a list of highest co-occurring words can be extracted for each topic, and then be updated to the gazette for the corresponding chapter. For instance, in Figure 33 the word “signavio” has been occurring multiple times in the same topic as “workflow diagram” for multiple runs of the module. When we extract the high occurrence words, we know these 2 words are highly related, and so when updating the gazette, we look for the chapter that “workflow diagram” belongs to and add “signavio” to that same chapter. Hence, we do not need to make manual revisions that frequently and may only need human intervention occasionally to evaluate the quality of our gazette.



*Figure 33: Process flow of gazette implementation*

# **5.0 Task 2 (Classification)**

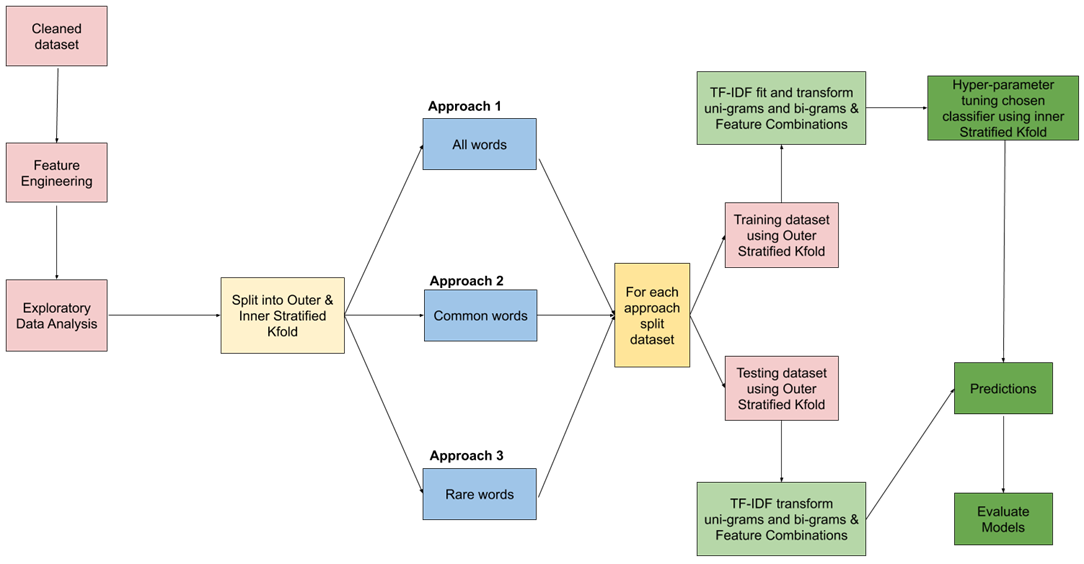
## **5.1 Objective**

The objective of task 2 is to evaluate the correlation between students’ class learning experience and their grades. A student’s class learning experience can be determined by his/her weekly feedback provided. For example, the words they are using, the frequency of their feedback and the sentiment of their feedback. Hence, we built a classifier that aims to predict a student’s grade based on the characteristics of their feedback.

The purpose of this task is to help the faculty identify poor performing students who may potentially require extra assistance in the course. The ability to do so can allow the faculty to intervene and help those in need sooner. Furthermore, it also helps the faculty understand what are the possible factors that lead to a student having a good or bad class learning experience.

## **5.2 Solution Methodology**

### 5.2.1 Solution Overview

****

*Figure 34: Solution overview of Task 2*

**Solution overview approach**  
For this task, after cleaning our dataset and selecting the relevant columns to be used, our first approach was to do feature engineering to create new features that can hopefully segment students better and aid us with our prediction. Next, to select the ideal classifier, we used our raw dataset and tested against numerous classifiers. After identifying the best classifier, we did hyper tuning and feature selection to choose the best parameter settings and features that can give us the best evaluation score. These approaches will be further explained in the following sections.

**Challenge**Out of the three columns (“Enjoyable”, “Challenging” & “Suggestions”), we decided to predict the students’ grades based on the “Suggestions” column. This is because there were little to no variations in their feedback for “Enjoyable” and “Challenging” columns as the students usually mentioned the concept which was being taught within the week. However, even among the “Suggestions” column, out of the 3099 feedbacks from 373 students, 104 students did not provide feedback in the “Suggestions” column. Therefore, the challenge or limitation that we have to work with is the low dataset quantity and being able to have an accurate prediction score of the students’ grades. In addition, accounting for the limitation, dropping students who did not provide feedback in the “Suggestions” column would not be ideal as we cannot afford to decrease our dataset quantity (around 27% of the dataset would be affected). Thus, we decided to aggregate the feedback by the students. Hence, to compensate for this limitation, we utilized other columns by doing feature engineering and leveraged on the results from Task 1 to build our prediction model.

### 5.2.2 Word Representation

We represented unigrams and bi-grams using TF-IDF. This is because it can summarize how often a given word appears within a document and penalize words that appear a lot across documents. This will help us to identify and extract unique characteristics of the students’ feedback to build a more accurate model.

In addition, specific to the BPAS dataset, we also realised that the feedbacks contain a lot of terms that are bi-grams such as “Business Process” , “Function Model”, “Process Model” and “Workflow Diagram”, hence, bi-grams are used to make sure that we capture all this important terms.

### 5.2.3 Feature Engineering

With limited data, we did feature engineering to create new features to aid in our predictive models. 7 new features were derived and created from the existing features in our dataset. They are grades, sentiment score, feedback length proportion, class type, frequency of suggestion, fuzzy similarity scores and topic modelling.

Usually, a sentiment score is provided between -1 to 1. However, due to the limitation of Multinomial Naive Bayes which does not accept negative values as inputs, we incremented the Sentiment Scores by 1.  
(please refer to Appendix A for the full detailed explanation)

|  |  |
| --- | --- |
| **Features** | **Description** |
| 1. Grades | Converted numeric scores into categorical variables to train our classifier. Following SMU’s grading system [A, B, C, D & F]. |
| 1. Sentiment Score | Calculated the polarity score [between -1 to 1] for each students’ feedback. Incremented by 1 due to Multinomial Naive Bayes [Between 0 to 2] |
| 1. Feedback Length Proportion | Refers to the total number of words provided by each student, divided by the total number of words by all students (Standardization). |
| 1. Class Type | Refers to the students attending either 8:15am, 12pm, 3:30pm class which can be classified into  Morning, Afternoon or Evening class. |
| 1. Frequency of Suggestion | Refers to how frequent each student provides feedback within the 13 weeks. [upon 13] |
| 1. Fuzzy Similarity Scores | Refers to how similar the feedback provided with the teaching materials or components of the assignment grades. We formulated the teaching materials or components into a gazette. We used the ratio algorithm of the fuzzy matching where we matched the Feedback with the words in the gazette. |
| 1. Topic Modelling  (From Task 1) | Refers to the number of times a particular topic was mentioned in their “Challenging” and “Enjoyable” topic feedback columns. This allows us to gauge which areas in the syllabus they repeatedly find challenging or enjoyable, which may be latent indicators about their final academic performance |

### 5.2.4 Models & Hyper-Parameters Considered for Tuning

The models we have explored with our dataset are Multinomial Naive Bayes, Support Vector Machine (SVM), Random Forest and KNN Classifier.   
  
For each of the classifiers, we have prepared a few hyperparameters which we would like to tune using RandomizedSearchCV. Since GridSearchCV can be computationally expensive, especially if one is searching over a large hyperparameter space and dealing with multiple hyperparameters, this can be mitigated by using RandomizedSearchCV, in which not all hyperparameter values are tried out and samples from a fixed number of hyperparameter settings are sampled from specified probability distributions. (DataCamp, 2020)

|  |  |
| --- | --- |
| **Classifier** | **Parameters** |
| Multinomial Naive Bayes | alpha: 0.01, 0.03, 0.1, 0.3, 1, 3,10  fit\_prior: True,False |
| Support Vector Machine (SVM) | C: 0.01, 0.03, 0.1, 0.3, 1, 3, 10  gamma: 0.01, 0.03, 0.1, 0.3, 1, 3, 10  kernel: linear, poly, rbf |
| Random Forest | n\_estimators: 200, 500  max\_features: auto, sqrt, log2,None  max\_depth: 3,4,5,6,7,8,9,10,11,12,15,20,30,40,50,70,90,120,150,None  criterion: gini, entropy |
| KNN Classifier | weights: uniform,distance  metric: euclidean,manhattan,minkowski  algorithm: auto, ball\_tree, kd\_tree, brute |

**Multinomial Naive Bayes** requires a small amount of training data to estimate the necessary parameters and are extremely fast compared to more sophisticated methods. However, one of the main limitations is that it treats words independently.

**SVM** is effective in high dimensional spaces and uses a subset of training points in the decision function so it is also memory efficient. However, it does not directly provide probability estimates, these are calculated using an expensive five-fold cross-validation.

**Random Forest** is able to reduce overfitting and is more accurate than decision trees since it uses multiple trees. It requires little data preparation and can handle numerical and categorical data.

**KNN Classifier** is simple to implement, robust to noisy training data, and effective if training data is large. Usually, one would need to determine the value of K and the computation cost is high as it needs to compute the distance of each instance to all the training samples. However, since we already know we have 5 classes, our K value will just be 5.

### 5.2.5 Feature Selection Approaches (Term-Frequency)

|  |  |
| --- | --- |
| **Approach** | **Definition** |
| **All Words** | All words mentioned by students |
| **Common Words** | Common words are words that occur more than the median word count of the training dataset |
| **Rare Words** | Rare words are words that occur less which is lesser than or equals to the median word count of the training dataset. |

We conducted our classification model based on 3 approaches with all the possible combinations of features engineered. The first approach leveraged on all the words mentioned by the students. The second approach only used common words which were mentioned by students. Common words are words that occur more than the median word count of the training dataset. The last approach only used rare words which were mentioned by students. Rare words are words that occur less which is lesser than or equals to the median word count of the training dataset. As there are 6700 vocabularies (unigrams & bi-grams) in our dataset, we want to evaluate whether there are words which are adding noise into our model. Summary of our approach as shown above.

With these 3 feature selection approaches and the new features that we have created, we have a total of 64 combinations to work with and train our model. (The full detailed combinations of features can be seen in Appendix B)

First, we must engineer our features which we have discussed in Section 5.2.3. Instead of doing a normal stratified cross-validation, we decided to employ the use of a nested stratified cross-validation (For more details refer to Section 5.6.1). The outer K-Fold will split the entire dataset into 5 folds and the inner K-Fold will split the training dataset into 3 folds. The outer K-Fold will be used to evaluate the robustness of our model and the inner K-Fold will be used to hyper-parameter tune our model using RandomizedSearchCV. The best model (tuned by RandomizedSearchCV) from the inner K-Fold will then be used to evaluate the testing portion of the outer K-Fold. The results from the outer K-Fold are averaged. The results for all our models can be found in the Appendix. For our evaluation, we decided to look at the best models for each approach.

### 5.2.6 Evaluation Metrics (F1 Macro vs F1 Weighted)

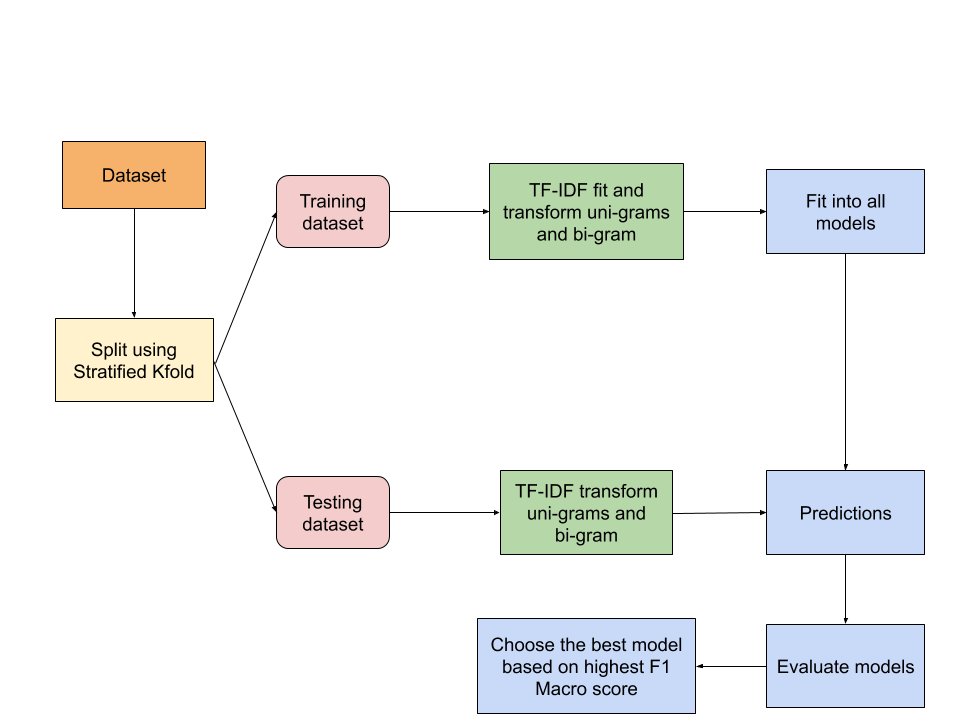
We used the F1 Average Macro metrics score over the F1 Average Weighted metrics score to evaluate our classifier model.

|  |  |
| --- | --- |
| **Evaluation Metrics** | **Reasons** |
| F1 Macro  (**Main Evaluation Metric**) | * **Not bias against imbalanced dataset** * Does not equally treat classes and simply aggregating the individual F1 Scores for each classes |
| F1 Weighted | * **Favour majority classes (Grade A, B & C)** * Calculates the F1 score for each class independently but adds them together using a weight that depends on the number of true labels of each class. |

“Weighted” calculates the F1 score for each class independently but adds them together using a weight that depends on the number of true labels of each class. Hence, favouring the majority class which in our class is made up of students with grade A, B and C (imbalanced dataset). However, our focus would be to correctly predict weaker students with grades C, D F. **Therefore, using “Macro” would be a better metric as it equally treats the classes by not giving any weightage to the individual classes.**

## **5.3 Results and Analyses**

### 5.3.1 Choosing the Base Classifier

****

*Figure 35: Process flow of choosing the base classifier*

As seen in Figure 35, we used the raw dataset to choose our base classifier. First, we split the dataset into 5 folds using Stratified K-Fold so that all classes in the dataset are represented in the training and testing dataset. After splitting, we only vectorize the words seen in our training data and transform the words in the test dataset. This helps to ensure data integrity where the model does not know what is in the test dataset. For each model, we fit the training data into the models. For each model, the F1 Macro and F1 Weighted are averaged after 5 K-Folds. The model which has the highest F1 Macro score is our chosen classifier algorithm for our task.

**Classification chosen**

|  |  |  |  |
| --- | --- | --- | --- |
| **Classifier** | **Average F1 Macro** | **Average F1 Weighted** | **Time Taken (Seconds)** |
| KNN Classifier | 0.182100767618444 | 0.23256944280396538 | 2.0636921809999933 |
| Random Forest | 0.16320820231221167 | 0.21723063023212555 | 4.125800485000013 |
| SVM | 0.1315277912782677 | 0.18327099758161222 | 7.204331434000011 |
| Multinomial Naive Bayes | 0.12917627672261922 | 0.18036887594713322 | 2.257150930999984 |

The table above shows the results of the classifiers. We can observe that the best classifier is the KNN Classifier, followed by Random Forest, SVM and Multinomial Naive Bayes. In addition, KNN Classifier also runs within 6 seconds. Therefore, we will be using KNN Classifier as our baseline model.

The reason why the results are as such could be because KNN classifier takes into consideration the influence and characteristic of the word features around each other while Naive Bayes assumes that the words are independent from each other. Hence, this might explain why KNN classifier is better for our dataset as it contains several word terminologies such as “As-is process”, “To be process”, “Use case model”, “business process model” and “solution blueprint model”.

As for the SVM and Random Forest, these 2 classifiers might not be a good to use as they are prone to overfitting especially when the dataset contains a large number of features. Which in our case, we have over 1000 word features (Varghese, 2018).

### 5.3.2 Model & Feature Evaluation

## 

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Rank** | **Approach** | **Best Features** | **Best F1 Macro** | **Best F1 Weighted** | **Run Time (Seconds)** |
| N/A | Baseline (Raw Text) | N/A | 0.1821 | 0.23256 | 2.0636 |
| 1 | All Words (Cleaned Text) | Feedback + Class Time + Frequency of Suggestion + **Fuzzy Similarity Scores between Feedback & Course + Topic Modelling** | 0.219578 | 0.270086 | 36.089 |
| 3 | Common Words (Cleaned Text) | Feedback + Frequency of Suggestion + **Fuzzy Similarity Scores between Feedback & Course + Topic Modelling** | 0.212915 | 0.260709 | 13.844 |
| 2 | Rare Words (Cleaned Text) | Feedback + Frequency of Suggestion + **Fuzzy Similarity Scores between Feedback & Course + Topic Modelling** | 0.217475 | 0.2652 | 30.50999 |

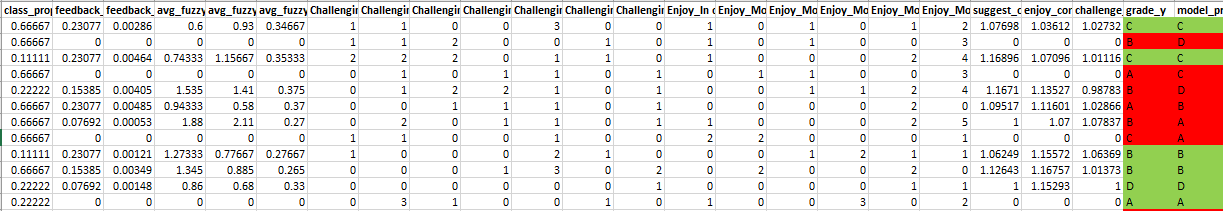
Please refer to Appendix B for the comprehensive results of all our features with the different approaches. Based on the F1 Macro score, the best model we have attained is from the initial approach where we included all words which were mentioned by the students. However, we believed that the best approach is actually the last approach where we only keep words that occur less than or equal to the median word count of the training dataset. Analysis of why we say so is covered under Error Analysis.

Since the KNN classifier takes into consideration the influence and characteristic of the word features around each other, there is a need to find out which words are required and to be kept. However, if resource and time is of concern, the second approach is also decent as the run time is significantly faster and there are minor differences in the F1 Macro score.

Across all three approaches, Frequency of their suggestions, Fuzzy Similarity Scores between feedback & course and Topic Modelling are important features which contributed to the performance of our classifier. This proves that our assumptions about students who find topics which are challenging or enjoyable have an impact on their grades.

## **5.4 Error Analysis and Discussion**

**Understanding Correctly & Incorrectly Classified Data**

****

*Figure 36: Sample of the student’s actual grades and predicted grades*

Figure 36 shows a sample of the student’s actual grades and predicted grades from our final model with the last approach. In the second row, we can observe a student who has no similarity scores in their feedback & course was predicted to score a D but he/she has scored a B.

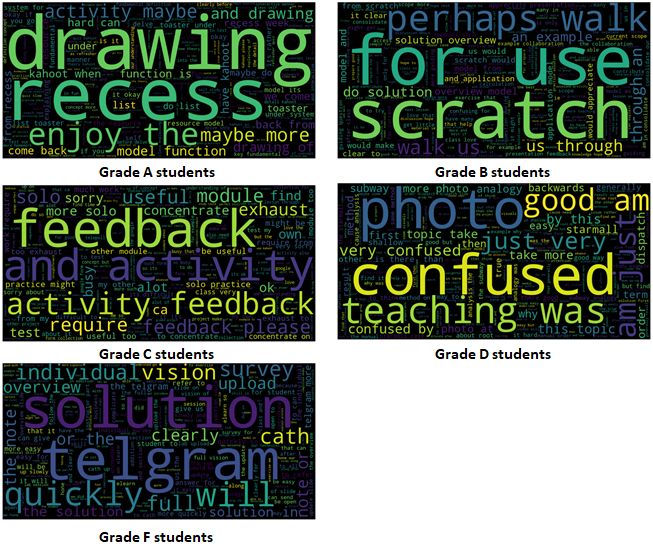
Out of the 12 samples shown above, we can see that it has misclassified half of them. However, the model only over-estimated 2 out of 6 misclassified students who are weaker students but were predicted to be stronger students. This is far better than misclassifying lots of weaker students than stronger students as we would not be able to help these weaker students earlier.

With that, we decided to explore further into the word cloud of each grade class. Based on the F1 model, our best approach is to use all the words, **we would generate a word cloud using all their words in the suggestions** to understand why. (Refer to Figure 37)



*Figure 37: Suggestions Word clouds of students from different grades*

From Figure 37, we can observe that there are no distinctive or specific words used by different students across grades. In addition, there are a lot of common words between each student. For example, ‘class’, ‘more’, for’ and ‘example’. This might be noise in our data which could contribute to the misclassification of some grades. Therefore, to rectify this issue, we excluded commonly used words and **generated another word cloud** (Refer to Figure 38). We define less common by considering those words which are less than or equal to the median word count**.** This way of defining is our last approach.



*Figure 38: Word clouds of students from different grades*

From Figure 38, we can see that there are different keywords used across the students with different grades. For example, we can observe that Grade A students used course topics and teaching materials words [‘model’, ‘kahoot, “resource”’] while Grade D students used words such as “teaching” and “confused”.

After evaluating the students’ word clouds across the grades, we can tell that their feedback has some distinction between them. Therefore, we believed that one possible reason as to why our classifier is misclassifying some of the students could be due to the noise from the common words mentioned by the students.

## **5.5 Gap Analysis and Discussion**

### 5.5.1 Vader Sentiment Analyzer

We used **Vader Sentiment Analyzer** on our cleaned dataset to create a new feature “sentiment score”. Vader is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media. As the students’ Feedback is similar to how people write in social media, it seems to be a suitable fit to use Vader for our sentiment analysis.

Based on Vader’s documentation, compound scores that are more than equals to 0.5 are deemed as positive, compound scores that are less than or equals to -0.5 are deemed as negative and compounds scores which are between -0.5 and 0.5 are deemed as neutral. However, we will treat those that negative scores are negative sentiments and positive scores are positive sentiments as some of the feedback are not provided and we have to fill them with 0.

The tables below show some results of the Vader Sentiment Analyzer as well as our own evaluation on certain feedback comments in the suggestion column. (Those rows highlighted in red are the misclassified feedbacks by Vader).

**Positive Feedback by Vader (Raw dataset)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **(Optional) What do you suggest to improve the class?** | **Vader’s Evaluation** | **Team’s Evaluation** |
| Student353 | Today's class was great, good pace, thank you! :) keep up the good work | 0.9402 | Positive |
| Student179 | THANK YOU PROF! You've been great and entertaining and made a otherwise dry class very interesting | 0.9263 | Positive |
| Student223 | I would like to value the learning experience as 2.5-3. I think more practice can be released before class so we can have a better prepared. I'm a slow learner but always want to try more exercise to understand the topic better. thank you prof! | 0.9134 | Negative |
| Student306 | Give even more chances for class part :D Thank you!!! :) | 0.91 | Positive |

**Negative Feedback by Vader (Raw dataset)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **(Optional) What do you suggest to improve the class?** | **Vader’s Evaluation** | **Team’s Evaluation** |
| Student310 | its very hard to follow the class sometimes because the screen quality is very poor. | -0.6232 | Negative |
| Student226 | a few of errors in the given solutions. Quite misleading | -0.5709 | Negative |
| Student97 | None, it was great! | -0.5553 | Positive |
| Student7 | None. All my questions have been answered promptly and I have no doubts of what I learned in class. Just need to make it more concrete with revision. | -0.5267 | Positive |

From the tables above, negative feedbacks that were predicted positive by Vader might be due to the words “thank you prof”. Seems like when a feedback ends with a “Thank you”, Vader will give a high positive score even though the context of the feedback might be negatively skewed.

On the other hand, positive feedbacks that were predicted negative by Vader might be due to the word “None”. Seems like Vader will give a high negative score when it spots “none” even though the context of the feedback might be positively skewed.

We manually tested a small sample size of 50 feedback to evaluate the accuracy of the Vader sentiment analysis and it managed to correctly classify around 90% of the positive sentiment feedback and around 70 to 80% of the negative sentiment.

**Positive Feedback by Vader (Cleaned dataset)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **cleaned\_(Optional) What do you suggest to improve the class?** | **Vader’s Evaluation** | **Team’s Evaluation** |
| Student345 | great that professor is improve think he is really good can try speak louder little and clear well do | 0.9239 | Positive |
| Student106 | love our in class discussion and exercise but it is the best if we can access more teaching material ie practice solution sample for calculation | 0.9072 | Positive |
| Student223 | would like to value the learn experience as think more practice can be release before class so we can have good prepared slow learner but always want to try more exercise to understand the topic well thank you professor | 0.8807 | Positive |

**Negative Feedback by Vader (Cleaned dataset)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **cleaned\_(Optional) What do you suggest to improve the class?** | **Vader’s Evaluation** | **Team’s Evaluation** |
| Student352 | to have the answer upload as well very difficult to revise after class as there are no correct answer if my answer are wrong | -0.7178 | Negative |
| Student310 | its very hard to follow the class sometimes because the screen quality is very poor | -0.6232 | Negative |
| Student7 | all my question have been answer promptly and have no doubt of what learn in class just need to make it more concrete with revision | -0.5719 | Positive |
| Student191 | for the part on the explanation of the practice to be model with tool it was bit difficult to follow as no slide was provide and it was rather fast | -0.5719 | Negative |

After using Vader on both the unclean and clean feedback, we found that Vader is able to predict the sentiment of the feedback more accurately as it managed to classify all the positive sentiments and 80% of the negative sentiments correctly within our sample dataset. Therefore, we decided to use Vader on our cleaned text.

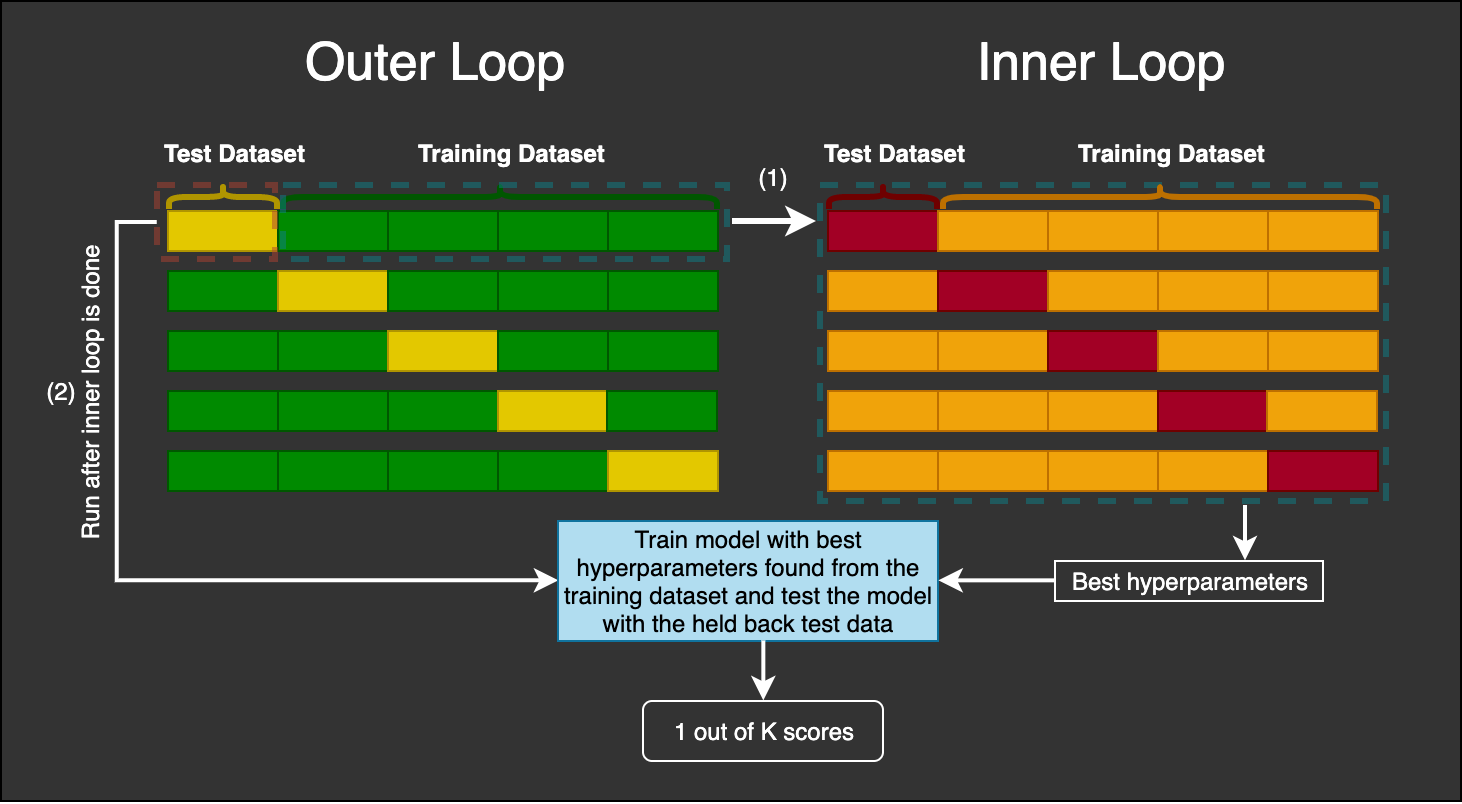
### 5.5.2 StratifyKFold vs KFold

Originally, we did a normal cross validation and realized that there are occasions where the model is not able to predict the minority classes. This is because our dataset is considered to be imbalanced and when we do a simple cross validation, we might be building models that are trained on data where there are so few or even none of the minority classes. Therefore, the models could not predict the minority classes effectively. This is where stratified cross-validation could minimize the impact of such an issue as it also allows for randomization but also makes sure all classes are represented in the training dataset for unbalanced data. This is even more important since the dataset we have has very limited samples where it is very likely to have very extreme differences in distributions (StackExchange, 2014).

### 5.5.3 TFIDF fit\_transform training set & transform testing set

Another mistake that we made while conducting our task was that we conducted TF-IDF using the entire dataset before splitting into the training and testing dataset. However, if we do so, we would also be carrying information from the training dataset over to the test dataset which is not correct. This is because there will only be some words that are only in the training dataset and some words in the testing dataset which has not been seen in the training dataset. However, since we have vectorized our entire dataset before splitting, all the words have been vectorized and are included as part of the features into the model. Therefore, the correct way was to split our dataset into training and testing dataset before vectorizing and should only fit on the train data. Afterwards, we then transform the words in the training and testing data. This will ensure the true unseen nature of the test dataset (Bhattacharya, 2019).

### 5.5.4 Cross Validation vs Nested Cross Validation

****

**Figure xxx (Hansen, 2019)**

Originally, we did a normal cross-validation with RandomizedSearchCV and fit the model for each pair of different hyperparameter sets in each cross-validation fold. However, this approach has a biased estimation of the true error of the model. This is because in a normal cross-validation, we only have a training and testing set, which we find the best hyperparameters for. If we were to evaluate the model as such, it may cause significant bias since we found the best hyperparameters for the model which was tuned on the same data and there might be scenarios of information leakage. Therefore, researchers have found nested cross-validation could be used to estimate the true error of the model, that is almost unbiased (Wainer & Cawley, 2018).

As shown in the image above, we will have two loops. The inner loop is basically normal cross-validation with a search function (in our project, we used RandomizedSearchCV). The outer loop supplies the inner loop with the training dataset, and the test dataset in the outer loop is held back. Furthermore, nested cross-validation is useful for projects where data is limited. An example provided was in biology as there is usually not a lot of data to go with machine learning projects (Hansen, 2019). This is similar to our project as we only have a limited amount of data of 373 students.

## 

## **5.6 Future Work**

### 5.6.1 Reducing the Number of Classes to Predict

It is easier to predict using lesser classes instead of using 5 classes. The classes can be broken down into 3 classes, ‘Above Average’, ‘Average’ & ‘Below Average’.

### 5.6.2 Developing our own F-Score

Currently, we did not engineer our own evaluation score. However, we felt that we could place more emphasis on recall score. This is because the cost of misclassifying weaker students is more expensive than classifying stronger students (Marbouti et al, 2016).

### 5.6.3 Adding Additional Features besides those from Feedback

Current features could just be indicators of a student’s learning attitude about the course which could not be a good indicator of a student’s performance. Adding other features which could capture other characteristics of a student learning behaviour could be helpful. This was proven by a study by (Shahiri et al, 2015). They found that cumulative GPA & the students’ demographic are strong indicators to predict a student’s performance. External assessments are also great features to have.

### 5.6.4 Fine Tuning the Engineering Approach of Topic Modelling Features

We can also explore ways on how to better engineer our features for topic modelling. For example, if we are able to extract important keywords which students are feeling challenging, it could be a strong indicator that the student could score very low, especially if the topic is a prerequisite of the other chapters or if it takes a significant portion of the assignment.

### 5.6.5 PCA Word Vectors

One possible improvement that we can do is to conduct Principal Component Analysis (PCA) on the word vectors. PCA removes irrelevant, noisy and redundant features from the feature vector and thereby improves the performance of the classifier for text categorization. However, one disadvantage of PCA is that it affects the interpretability and readability of our results as the dimensions are condensed (Taloba, 2018).

# **6.0 Conclusion**

Through our insights from our 2 use case scenarios, we hope that it has helped the faculty team to improve the students’ learning experience. Although there is certainly potential in improving the methodologies of both tasks, we hope that our analysis on this topic will help to fuel the discovery of more accurate insights of students’ learning experiences in SMU and contribute to the mutual party learning between the students and the professors to better both their experiences in SMU.

# **7.0 Reflections**

**Ryan Tan**I learned about how it is important to know the motivation behind your tasks as it will affect the way you implement it. From things such as the pre-processing steps, all of it will change depending on your motivation and intended outcomes, and there are many ways to go about doing the same thing, such as using different methods to carry out topic modelling.

**Delin**I enjoyed myself throughout the course of the project. I have also learned about the importance of data leakage and how to properly validate the models’ performance. The theme that we chose for our project is interesting as we get to understand if there are any correlation between the students’ learning journey and the grades they will ultimately achieve. I also enjoyed the process of working in the team as we are all task-oriented and get work done efficiently. We also shared ideas on how to fix any issues when we faced some difficulty in our tasks.

**Akshi**This project taught me the importance to explore beyond what is taught in our course on different text analysis techniques. Every technique must be adapted to the domain we apply to, ours being student course feedback in this case. To be able to apply learnt concepts and adapt them in our context was useful. With determined and hardworking teammates, this project has been a great learning experience.

**Janell Lee**Through this project, I learned that there is no "perfect" approach when it comes to creating the framework for analysing text. However, through extensive experimentation, a better approach can be created. Even so, new methods and approaches are found as research in the various text analysis domains progress, e.g. Text Summarisation, which makes it important for us to maintain a posture of learning and keep ourselves updated with technology.

**Junrong**The process of finding research papers and trying to validate our approach. I implemented various ways of modelling our solution and implementation.

**Benedict Then**After going through this project, I realised how important it is to pre-process the corpus as there may be many invalid words being used, especially short forms and slangs written by students. In addition, doing feature engineering to create new features to aid our classifier was new to me, which I have learnt a lot of different ways to do so. Lastly, I have enjoyed the process of this project and learnt a lot from my hardworking teammates.

# 

# **References**

Bhattacharya, M. (2019). 3 Things You Need To Know Before You Train-Test Split. Retrieved from https://towardsdatascience.com/3-things-you-need-to-know-before-you-train-test-split-869dfabb7e50

Chen, G. Kao, H. (2015). Word Co-occurrence Augmented Topic Model in Short Text. Retrieved from https://www.aclweb.org/anthology/O15-3003.pdf

Clark, S. (2013). Topic Modelling and Latent Dirichlet Allocation. Retrieved from https://www.cl.cam.ac.uk/teaching/1213/L101/clark\_lectures/lect7.pdf

Cross Validated. (2014). Why use stratified cross validation? Why does this not damage variance related benefit? Retrieved from https://stats.stackexchange.com/questions/117643/why-use-stratified-cross-validation-why-does-this-not-damage-variance-related-b

DataCamp.(2020). Hyperparameter tuning with RandomizedSearchCV. Retrieved from https://campus.datacamp.com/courses/supervised-learning-with-scikit-learn/fine-tuning-your-model?ex=11

DeMonbrun, R et al. (2018). The Snowball Effect: Exploring the Influence of Changes in Academic Performance on Student Success in Co-enrolled Courses. Retrieved from https://lib.dr.iastate.edu/cgi/viewcontent.cgi?article=1007&context=edu\_conf

D’agnese, A. (2015). Does class time affect student performance? Retrieved from https://sites.psu.edu/siowfa15/2015/10/19/does-class-time-affect-student-performance/

Github. (2017). GSDMM: Short text clustering. Retrieved from https://github.com/rwalk/gsdmm

Google Code Archive. (2013). word2vec. Retrieved from https://code.google.com/archive/p/word2vec/

Hansen, C. (2019). Nested Cross-Validation Python Code. Retrieved from https://mlfromscratch.com/nested-cross-validation-python-code/#/

Jones, V. (2015). Because Im Happy. Retrieved from https://www.gse.harvard.edu/news/uk/15/03/because-i%E2%80%99m-happy

Li, C et al. (2017). Topic Modeling for Short Texts with Auxiliary Word Embeddings. Retrieved from https://www.ntu.edu.sg/home/axsun/paper/sigir16text.pdf

Marbouti, F et al. (2016). Models for early prediction of at-risk students in a course using standards-based grading. Retrieved from https://www.sciencedirect.com/science/article/pii/S0360131516301634

Mehrotra, R et al. (2013). Improving LDA Topic Models for Microblogs via Tweet Pooling and Automatic Labeling. Retrieved from http://users.cecs.anu.edu.au/~ssanner/Papers/sigir13.pdf

Qiang, J et al. (2017). Topic Modeling over Short Texts by Incorporating Word Embeddings. Retrieved from https://arxiv.org/pdf/1609.08496.pdf

Qiang, J et al. (2019). Short Text Topic Modeling Techniques, Applications, and Performance: A Survey. Retrieved from https://arxiv.org/pdf/1904.07695.pdf

Shahiri, A et al. (2015). A Review on Predicting Student's Performance Using Data Mining Techniques. Retrieved from https://www.sciencedirect.com/science/article/pii/S1877050915036182

StackExchange. (2017). Micro Average vs Macro average Performance in a Multiclass classification setting. Retrieved from https://datascience.stackexchange.com/questions/15989/micro-average-vs-macro-average-performance-in-a-multiclass-classification-settin

Svinicki, M. (2001). Encouraging Your Students to Give Feedback. Retrieved from https://www.jcu.edu.au/\_\_data/assets/pdf\_file/0016/105091/jcu\_121460.pdf

Taloba, A et al. (2018). A Comparative Study on using Principle Component Analysis with Different Text Classifiers. Retrieved from https://arxiv.org/pdf/1807.03283.pdf

Varghese, D. (2018). Comparative Study on Classic Machine learning Algorithms. Retrieved from https://towardsdatascience.com/comparative-study-on-classic-machine-learning-algorithms-24f9ff6ab222

Wainer, J. Cawley G. (2018). Nested cross-validation when selecting classifiers is overzealous for most practical applications. Retrieved from https://arxiv.org/pdf/1809.09446.pdf

Yin, J. Wang, J. (2014). A Dirichlet Multinomial Mixture Model-based Approach for Short Text Clustering. Retrieved from http://dbgroup.cs.tsinghua.edu.cn/wangjy/papers/KDD14-GSDMM.pdf

# 

# **Appendix A**

## **Topic Modelling Results on “ List one topic of the class that you enjoy” column (Raw Results)**

### Baseline - Unigram

Topic 0

0.224\*"solution" + 0.158\*"application" + 0.154\*"overview" + 0.110\*"modelling" + 0.072\*"blueprint" + 0.049\*"draw" + 0.030\*"modeling" + 0.027\*"hierarchy" + 0.023\*"concept" + 0.022\*"rule"

Topic 1

0.166\*"resource" + 0.095\*"enjoy" + 0.068\*"understand" + 0.039\*"make" + 0.039\*"group" + 0.035\*"come" + 0.034\*"topic" + 0.029\*"interesting" + 0.026\*"risk" + 0.026\*"easy"

Topic 2

0.222\*"lab" + 0.118\*"static" + 0.116\*"analysis" + 0.043\*"functional" + 0.036\*"exercise" + 0.035\*"architecture" + 0.035\*"everything" + 0.024\*"orientation" + 0.024\*"performance" + 0.022\*"goal"

Topic 3

0.302\*"collaboration" + 0.110\*"alignment" + 0.027\*"learn" + 0.027\*"course" + 0.025\*"recap" + 0.024\*"example" + 0.022\*"go" + 0.021\*"enjoy" + 0.020\*"difference" + 0.020\*"importance"

Topic 4

0.221\*"signavio" + 0.168\*"analysis" + 0.126\*"workflow" + 0.083\*"dynamic" + 0.041\*"canvas" + 0.033\*"path" + 0.027\*"chain" + 0.027\*"value" + 0.026\*"cost" + 0.024\*"methodology"

Topic 5

0.285\*"innovation" + 0.114\*"diagram" + 0.111\*"presentation" + 0.043\*"type" + 0.039\*"product" + 0.032\*"different" + 0.027\*"package" + 0.025\*"part" + 0.024\*"technology" + 0.023\*"draw"

Topic 6

0.325\*"iot" + 0.094\*"study" + 0.040\*"function" + 0.036\*"executive" + 0.035\*"challenge" + 0.026\*"understand" + 0.023\*"interest" + 0.018\*"food" + 0.017\*"enjoy" + 0.015\*"improve"

Topic 7

0.120\*"rci" + 0.112\*"location" + 0.087\*"organization" + 0.044\*"triangle" + 0.042\*"learn" + 0.039\*"magic" + 0.033\*"discussion" + 0.030\*"organizational" + 0.029\*"gassmann" + 0.029\*"class"

Topic 8

0.216\*"learn" + 0.128\*"different" + 0.057\*"enjoy" + 0.055\*"trend" + 0.031\*"type" + 0.029\*"like" + 0.028\*"digital" + 0.023\*"thing" + 0.019\*"technology" + 0.018\*"various"

Topic 9

0.096\*"cause" + 0.086\*"root" + 0.082\*"activity" + 0.058\*"class" + 0.055\*"impact" + 0.041\*"management" + 0.040\*"identify" + 0.038\*"create" + 0.034\*"flow" + 0.032\*"issue"

### Baseline - Bigram

Topic 0

['root\_cause', 'impact', 'organizational', 'executive', 'location', 'learn', 'different', 'value\_chain', 'iot', 'understand']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 1

['collaboration', 'solution\_overview', 'presentation', 'canvas', 'digital', 'create', 'trend', 'product', 'resource', 'innovation']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 2

['application', 'organization', 'class', 'location', 'activity', 'draw', 'flow', 'rcr', 'discussion', 'work']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 3

['iot', 'resource', 'function', 'trend', 'technology', 'report', 'challenge', 'risk', 'different', 'like']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 4

['study', 'enjoy', 'blueprint', 'learn', 'identify', 'magic\_triangle', 'management', 'gassmann', 'presentation', 'issue']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 5

['signavio', 'innovation', 'learn', 'technique', 'analysis', 'hand', 'tool', 'static', 'face', 'lot']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 6

['rci', 'diagram', 'type', 'different', 'understand', 'package', 'group', 'iot', 'make', 'generate']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 7

['analysis', 'modelling', 'dynamic\_analysis', 'static', 'path', 'methodology', 'everything', 'impact', 'example', 'revision']

Labelled chapter: 5 : Business Process Analysis (Static)

Keywords appearing in both chapter and topic: ['analysis', 'static', 'path', 'impact']

Topic 8

['learn', 'different', 'alignment', 'concept', 'enjoy', 'type', 'interesting', 'framework', 'importance', 'goal']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 9

['workflow', 'lab', 'enjoy', 'signavio', 'topic', 'review', 'hierarchy', 'modeling', 'thing', 'cost']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

### Aggregation - Vector Space Similarity

Topic 0

['solution\_overview', 'location', 'rci', 'organization', 'canvas', 'different', 'draw', 'executive', 'function', 'iot']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 1

['presentation', 'diagram', 'draw', 'signavio', 'class', 'study', 'group', 'drawing', 'create', 'discussion']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 2

['application', 'iot', 'modelling', 'solution\_overview', 'signavio', 'understand', 'blueprint', 'modeling', 'learn', 'create']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 3

['iot', 'signavio', 'trend', 'report', 'static', 'analysis', 'digital', 'management', 'generate', 'workflow']

Labelled chapter: 5 : Business Process Analysis (Static)

Keywords appearing in both chapter and topic: ['trend', 'report', 'static', 'analysis']

Topic 4

['blueprint', 'study', 'concept', 'framework', 'iot', 'understand', 'enjoy', 'analysis', 'interest', 'know']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 5

['analysis', 'innovation', 'collaboration', 'signavio', 'dynamic\_analysis', 'modelling', 'static', 'rci', 'methodology', 'product']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 6

['enjoy', 'different', 'learn', 'class', 'iot', 'understand', 'activity', 'topic', 'help', 'like']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 7

['analysis', 'different', 'learn', 'dynamic\_analysis', 'static', 'enjoy', 'location', 'iot', 'signavio', 'class']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 8

['learn', 'alignment', 'signavio', 'iot', 'different', 'enjoy', 'type', 'innovation', 'root\_cause', 'find']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 9

['workflow', 'lab', 'resource', 'signavio', 'modeling', 'draw', 'exercise', 'diagram', 'modelling', 'create']

Labelled chapter: 4: Business Process Modelling P2

Keywords appearing in both chapter and topic: ['workflow', 'resource', 'signavio', 'modeling', 'modelling']

### Aggregation - Academic Weeks

Topic 0

['location', 'organization', 'alignment', 'different', 'modelling', 'learn', 'organizational', 'enjoy', 'understand', 'activity']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 1

['innovation', 'study', 'canvas', 'presentation', 'product', 'learn', 'gassmann', 'type', 'different', 'class']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 2

['location', 'innovation', 'organization', 'alignment', 'learn', 'different', 'application', 'enjoy', 'modelling', 'organizational']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 3

['iot', 'trend', 'learn', 'presentation', 'different', 'innovation', 'technology', 'application', 'digital', 'enjoy']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

­Topic 4

['analysis', 'application', 'blueprint', 'lab', 'path', 'learn', 'modelling', 'understand', 'enjoy', 'technique']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 5

['collaboration', 'resource', 'workflow', 'different', 'learn', 'enjoy', 'lab', 'understand', 'class', 'diagram']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 6

['signavio', 'rci', 'static', 'lab', 'report', 'analysis', 'impact', 'learn', 'issue', 'rcr']

Labelled chapter: 5 : Business Process Analysis (Static)

Keywords appearing in both chapter and topic: ['rci', 'static', 'report', 'analysis', 'impact', 'issue', 'rcr']

Topic 7

['analysis', 'collaboration', 'learn', 'location', 'different', 'resource', 'enjoy', 'workflow', 'modelling', 'organization']

Labelled chapter: 4: Business Process Modelling P2

Keywords appearing in both chapter and topic: ['collaboration', 'resource', 'workflow', 'modelling']

Topic 8

['signavio', 'workflow', 'learn', 'draw', 'diagram', 'function', 'lab', 'enjoy', 'different', 'rule']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 9

['hierarchy', 'executive', 'review', 'course', 'architecture', 'learn', 'revision', 'exam', 'enterprise', 'topic']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

### GSDMM

Topic 0

['innovation', 'understand', 'presentation', 'value\_chain', 'class', 'enjoy', 'learn', 'different', 'study', 'iot']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 1

['challenge', 'application', 'product', 'study', 'type', 'presentation', 'different', 'learn', 'innovation', 'iot']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 2

['fun', 'activity', 'workflow', 'class', 'lab', 'resource', 'signavio', 'understand', 'enjoy', 'collaboration']

Labelled chapter: 4: Business Process Modelling P2

Keywords appearing in both chapter and topic: ['workflow', 'resource', 'signavio', 'collaboration']

Topic 3

['rcr', 'methodology', 'technique', 'path', 'static', 'impact', 'root\_cause', 'rci', 'dynamic\_analysis', 'analysis']

Labelled chapter: 5 : Business Process Analysis (Static)

Keywords appearing in both chapter and topic: ['rcr', 'path', 'static', 'impact', 'root\_cause', 'rci', 'analysis']

Topic 4

['revision', 'enjoy', 'course', 'package', 'review', 'learn', 'signavio', 'resource', 'lab', 'collaboration']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 5

['lab', 'goal', 'performance', 'internet', 'thing', 'generate', 'analysis', 'report', 'static', 'signavio']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 6

['different', 'hierarchy', 'presentation', 'diagram', 'flow', 'functional', 'enjoy', 'work', 'learn', 'alignment']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 7

['signavio', 'modelling', 'enjoy', 'resource', 'understand', 'different', 'organizational', 'learn', 'organization', 'location']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 8

['diagram', 'resource', 'draw', 'collaboration', 'blueprint', 'modelling', 'application', 'solution\_overview', 'workflow', 'signavio']

Labelled chapter: 4: Business Process Modelling P2

Keywords appearing in both chapter and topic: ['resource', 'collaboration', 'modelling', 'workflow', 'signavio']

Topic 9

['different', 'workflow', 'digital', 'magic\_triangle', 'technology', 'canvas', 'learn', 'iot', 'trend', 'innovation']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

## **Topic Modelling Results on “What was the most challenging topic of the class?” column (Raw Results)**

### Baseline - Unigram

Topic 0

0.130\*"application" + 0.115\*"solution" + 0.102\*"collaboration" + 0.084\*"overview" + 0.045\*"study" + 0.029\*"value" + 0.028\*"draw" + 0.027\*"chain" + 0.025\*"part" + 0.022\*"time"

Topic 1

0.171\*"analysis" + 0.129\*"signavio" + 0.058\*"impact" + 0.053\*"rci" + 0.051\*"dynamic" + 0.038\*"report" + 0.032\*"understand" + 0.031\*"static" + 0.026\*"path" + 0.026\*"technique"

Topic 2

0.132\*"function" + 0.066\*"management" + 0.066\*"come" + 0.041\*"different" + 0.037\*"trend" + 0.037\*"understand" + 0.036\*"core" + 0.031\*"support" + 0.026\*"executive" + 0.026\*"digital"

Topic 3

0.260\*"workflow" + 0.078\*"modelling" + 0.065\*"class" + 0.051\*"activity" + 0.026\*"lane" + 0.025\*"understand" + 0.025\*"swim" + 0.020\*"exam" + 0.019\*"everything" + 0.019\*"draw"

Topic 4

0.155\*"iot" + 0.151\*"cause" + 0.112\*"root" + 0.043\*"architecture" + 0.041\*"component" + 0.038\*"description" + 0.038\*"issue" + 0.036\*"find" + 0.020\*"recommendation" + 0.018\*"identify"

Topic 5

0.130\*"different" + 0.056\*"understand" + 0.044\*"alignment" + 0.044\*"type" + 0.031\*"level" + 0.029\*"presentation" + 0.028\*"present" + 0.020\*"attribute" + 0.018\*"remember" + 0.016\*"identify"

Topic 6

0.110\*"diagram" + 0.093\*"package" + 0.066\*"workflow" + 0.051\*"rcr" + 0.039\*"understand" + 0.038\*"bit" + 0.028\*"function" + 0.026\*"time" + 0.025\*"rci" + 0.025\*"still"

Topic 7

0.172\*"identify" + 0.145\*"innovation" + 0.034\*"topic" + 0.028\*"challenging" + 0.026\*"modeling" + 0.026\*"cause" + 0.023\*"cost" + 0.020\*"issue" + 0.017\*"template" + 0.016\*"root"

Topic 8

0.095\*"challenge" + 0.076\*"understand" + 0.063\*"lab" + 0.041\*"work" + 0.037\*"iot" + 0.035\*"flow" + 0.030\*"question" + 0.026\*"concept" + 0.025\*"innovation" + 0.022\*"teach"

Topic 9

0.115\*"functional" + 0.096\*"orientation" + 0.058\*"difference" + 0.048\*"canvas" + 0.038\*"new" + 0.038\*"differentiate" + 0.032\*"triangle" + 0.032\*"system" + 0.031\*"understand" + 0.025\*"exist"

### Baseline - Bigram

Topic 0

['function', 'application', 'solution\_overview', 'understand', 'identify', 'attribute', 'present', 'come', 'difference', 'error']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 1

['root\_cause', 'issue', 'iot', 'differentiate', 'identify', 'architecture', 'description', 'different', 'apply', 'work']

Labelled chapter: 2 : Business IT Alignment

Keywords appearing in both chapter and topic: ['differentiate', 'identify', 'description', 'different']

Topic 2

['modelling', 'management', 'understand', 'type', 'core', 'different', 'support', 'digital', 'twin', 'categorise']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 3

['analysis', 'understand', 'innovation', 'different', 'dynamic\_analysis', 'report', 'static', 'technique', 'path', 'go']

Labelled chapter: 5 : Business Process Analysis (Static)

Keywords appearing in both chapter and topic: ['analysis', 'report', 'static', 'path']

Topic 4

['iot', 'challenge', 'study', 'identify', 'root\_cause', 'impact', 'value\_chain', 'simulation', 'face', 'exam']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 5

['workflow', 'collaboration', 'diagram', 'identify', 'draw', 'activity', 'rcr', 'determine', 'learn', 'rule']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 6

['lab', 'package', 'time', 'level', 'alignment', 'presentation', 'exercise', 'executive', 'class', 'need']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 7

['signavio', 'functional\_orientation', 'class', 'activity', 'system', 'resource', 'cost', 'difference', 'risk', 'external']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 8

['part', 'question', 'identify', 'think', 'challenging', 'topic', 'challenge', 'concept', 'organization', 'location']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 9

['rci', 'canvas', 'understand', 'new', 'lane', 'component', 'swim', 'swim\_lane', 'exist', 'teach']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

### Aggregation - Vector Space Similarity

Topic 0

['function', 'package', 'understand', 'diagram', 'identify', 'come', 'confuse', 'determine', 'solution\_overview', 'modelling']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 1

['application', 'architecture', 'iot', 'blueprint', 'modelling', 'solution\_overview', 'framework', 'modeling', 'signavio', 'understand']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 2

['understand', 'different', 'management', 'workflow', 'differentiate', 'activity', 'type', 'alignment', 'iot', 'class']

Labelled chapter: 2 : Business IT Alignment

Keywords appearing in both chapter and topic: ['understand', 'different', 'management', 'differentiate', 'alignment']

Topic 3

['understand', 'analysis', 'innovation', 'different', 'challenge', 'iot', 'difference', 'find', 'root\_cause', 'class']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 4

['study', 'understand', 'iot', 'different', 'identify', 'root\_cause', 'challenge', 'signavio', 'workflow', 'find']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 5

['workflow', 'identify', 'collaboration', 'diagram', 'management', 'understand', 'differentiate', 'alignment', 'function', 'system']

Labelled chapter: 2 : Business IT Alignment

Keywords appearing in both chapter and topic: ['identify', 'management', 'understand', 'differentiate', 'alignment']

Topic 6

['lab', 'package', 'presentation', 'exam', 'class', 'recommendation', 'present', 'paper', 'signavio', 'study']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 7

['signavio', 'modelling', 'functional\_orientation', 'activity', 'iot', 'modeling', 'rci', 'gassmann', 'value\_chain', 'rcr']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 8

['understand', 'different', 'challenge', 'iot', 'workflow', 'identify', 'difference', 'find', 'activity', 'issue']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 9

['solution\_overview', 'rci', 'canvas', 'understand', 'iot', 'teach', 'rcr', 'executive', 'root\_cause', 'different']

Labelled chapter: 5 : Business Process Analysis (Static)

Keywords appearing in both chapter and topic: ['rci', 'canvas', 'rcr', 'executive', 'root\_cause']

### Aggregation - Academic Weeks

Topic 0

['function', 'package', 'identify', 'diagram', 'activity', 'understand', 'new', 'exist', 'modify', 'find']

Labelled chapter: 7 : Business Process Solutioning (IT requirements)

Keywords appearing in both chapter and topic: ['function', 'package', 'diagram', 'activity', 'find']

Topic 1

['iot', 'application', 'innovation', 'workflow', 'understand', 'challenge', 'different', 'identify', 'study', 'architecture']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 2

['application', 'understand', 'modelling', 'management', 'identify', 'different', 'difference', 'alignment', 'component', 'core']

Labelled chapter: 2 : Business IT Alignment

Keywords appearing in both chapter and topic: ['understand', 'management', 'identify', 'different', 'alignment']

Topic 3

['analysis', 'signavio', 'lab', 'understand', 'impact', 'identify', 'technique', 'path', 'simulation', 'different']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 4

['iot', 'challenge', 'trend', 'architecture', 'study', 'understand', 'different', 'digital', 'workflow', 'identify']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 5

['workflow', 'collaboration', 'understand', 'different', 'draw', 'activity', 'system', 'diagram', 'identify', 'signavio']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 6

['workflow', 'understand', 'signavio', 'identify', 'presentation', 'application', 'rci', 'class', 'different', 'iot']

Labelled chapter: 2 : Business IT Alignment

Keywords appearing in both chapter and topic: ['understand', 'identify', 'presentation', 'different']

Topic 7

['presentation', 'present', 'iot', 'workflow', 'group', 'class', 'understand', 'innovation', 'really', 'time']

Labelled chapter: 2 : Business IT Alignment

Keywords appearing in both chapter and topic: ['presentation', 'present', 'understand', 'time']

Topic 8

['rci', 'issue', 'signavio', 'description', 'rcr', 'identify', 'find', 'static', 'understand', 'impact']

Labelled chapter: 5 : Business Process Analysis (Static)

Keywords appearing in both chapter and topic: ['rci', 'issue', 'rcr', 'static', 'impact']

Topic 9

['innovation', 'canvas', 'identify', 'product', 'gassmann', 'challenge', 'study', 'understand', 'come', 'proposition']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

### GSDMM

Topic 0

['digital', 'path', 'lab', 'technique', 'report', 'static', 'dynamic\_analysis', 'signavio', 'understand', 'analysis']

Labelled chapter: 5 : Business Process Analysis (Static)

Keywords appearing in both chapter and topic: ['path', 'report', 'static', 'analysis']

Topic 1

['get', 'signavio', 'paper', 'need', 'collaboration', 'cost', 'class', 'exam', 'understand', 'workflow']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 2

['draw', 'diagram', 'collaboration', 'solution\_overview', 'architecture', 'modelling', 'function', 'application', 'iot', 'workflow']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 3

['find', 'teach', 'face', 'come', 'root\_cause', 'different', 'innovation', 'identify', 'challenge', 'iot']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 4

['determine', 'recommendation', 'level', 'analysis', 'rcr', 'issue', 'impact', 'identify', 'root\_cause', 'rci']

Labelled chapter: 5 : Business Process Analysis (Static)

Keywords appearing in both chapter and topic: ['recommendation', 'analysis', 'rcr', 'issue', 'impact', 'root\_cause', 'rci']

Topic 5

['differentiate', 'resource', 'different', 'support', 'activity', 'system', 'understand', 'core', 'identify', 'management']

Labelled chapter: 2 : Business IT Alignment

Keywords appearing in both chapter and topic: ['differentiate', 'different', 'understand', 'identify', 'management']

Topic 6

['work', 'application', 'function', 'solution\_overview', 'functional\_orientation', 'collaboration', 'package', 'difference', 'workflow', 'understand']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 7

['part', 'topic', 'need', 'activity', 'time', 'understand', 'class', 'signavio', 'workflow', 'lab']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 8

['identify', 'type', 'technology', 'differentiate', 'study', 'iot', 'product', 'understand', 'different', 'innovation']

Labelled chapter: 2 : Business IT Alignment

Keywords appearing in both chapter and topic: ['identify', 'differentiate', 'understand', 'different']

Topic 9

['little', 'canvas', 'component', 'level', 'concept', 'different', 'attribute', 'identify', 'alignment', 'understand']

Labelled chapter: 2 : Business IT Alignment

Keywords appearing in both chapter and topic: ['different', 'identify', 'alignment', 'understand']

## **Topic Modelling Results on “(Optional) What do you suggest to improve the class?” column (Raw Results)**

### Baseline - Unigram

Topic 0

0.090\*"example" + 0.065\*"concept" + 0.049\*"explain" + 0.039\*"question" + 0.035\*"week" + 0.030\*"give" + 0.024\*"take" + 0.023\*"topic" + 0.022\*"instead" + 0.022\*"would"

Topic 1

0.071\*"example" + 0.070\*"give" + 0.065\*"model" + 0.059\*"could" + 0.058\*"time" + 0.052\*"presentation" + 0.036\*"study" + 0.029\*"well" + 0.028\*"perhaps" + 0.020\*"draw"

Topic 2

0.199\*"example" + 0.072\*"model" + 0.070\*"real" + 0.059\*"life" + 0.046\*"video" + 0.034\*"maybe" + 0.032\*"learn" + 0.027\*"need" + 0.024\*"different" + 0.020\*"provide"

Topic 3

0.079\*"professor" + 0.062\*"maybe" + 0.049\*"class" + 0.039\*"little" + 0.038\*"good" + 0.032\*"slow" + 0.031\*"pace" + 0.028\*"step" + 0.028\*"think" + 0.020\*"really"

Topic 4

0.100\*"slow" + 0.090\*"professor" + 0.082\*"speak" + 0.052\*"signavio" + 0.036\*"louder" + 0.032\*"diagram" + 0.030\*"lab" + 0.023\*"bit" + 0.022\*"catch" + 0.021\*"time"

Topic 5

0.173\*"class" + 0.078\*"discussion" + 0.074\*"activity" + 0.039\*"teaching" + 0.038\*"lesson" + 0.028\*"group" + 0.021\*"interesting" + 0.021\*"team" + 0.020\*"hand" + 0.019\*"think"

Topic 6

0.102\*"good" + 0.052\*"student" + 0.048\*"class" + 0.038\*"practice" + 0.036\*"quiz" + 0.034\*"work" + 0.033\*"question" + 0.030\*"kahoot" + 0.023\*"well" + 0.022\*"think"

Topic 7

0.083\*"model" + 0.063\*"clear" + 0.043\*"thank" + 0.041\*"explanation" + 0.034\*"interactive" + 0.034\*"quite" + 0.030\*"system" + 0.026\*"understand" + 0.026\*"concept" + 0.025\*"today"

Topic 8

0.100\*"time" + 0.089\*"class" + 0.056\*"give" + 0.047\*"activity" + 0.046\*"slide" + 0.042\*"part" + 0.022\*"help" + 0.021\*"please" + 0.019\*"model" + 0.015\*"especially"

Topic 9

0.088\*"lab" + 0.085\*"class" + 0.031\*"question" + 0.029\*"could" + 0.028\*"exercise" + 0.028\*"fast" + 0.028\*"less" + 0.027\*"slide" + 0.027\*"make" + 0.026\*"bit"

### Baseline - Bigram

Topic 0

['example', 'model', 'give', 'concept', 'explain', 'week', 'understand', 'like', 'slide', 'could']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 1

['class', 'good', 'pace', 'thank', 'exercise', 'instructor', 'today', 'bit', 'fast', 'well']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 2

['real', 'real\_life', 'life', 'break', 'example', 'class', 'student', 'make', 'easy', 'lab']

Labelled chapter: 5 : Business Process Analysis (Static)

Keywords appearing in both chapter and topic: ['real', 'real\_life', 'class', 'make']

Topic 3

['professor', 'would', 'speak', 'clear', 'provide', 'slow', 'thank', 'good', 'example', 'louder']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 4

['model', 'practice', 'kahoot', 'part', 'class', 'quiz', 'please', 'slide', 'question', 'project']

Labelled chapter: 5 : Business Process Analysis (Static)

Keywords appearing in both chapter and topic: ['model', 'practice', 'part', 'class', 'please']

Topic 5

['think', 'class', 'professor', 'like', 'study', 'lesson', 'today', 'hand', 'would', 'maybe']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 6

['time', 'class', 'activity', 'give', 'group', 'slightly', 'teaching', 'speak', 'professor', 'teach']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 7

['slide', 'class', 'professor', 'question', 'follow', 'slow', 'could', 'little', 'pace', 'teach']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 8

['discussion', 'class', 'lab', 'could', 'slow', 'time', 'signavio', 'allocate', 'know', 'bit']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 9

['student', 'good', 'question', 'answer', 'video', 'class', 'could', 'understanding', 'learn', 'well']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

### Aggregation - Vector Space Similarity

Topic 0

['example', 'model', 'give', 'explain', 'real', 'concept', 'life', 'real\_life', 'provide', 'understand']

Labelled chapter: 5 : Business Process Analysis (Static)

Keywords appearing in both chapter and topic: ['model', 'real', 'concept', 'real\_life']

Topic 1

['lab', 'class', 'time', 'discussion', 'go', 'exercise', 'less', 'allocate', 'student', 'part']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 2

['thank', 'clear', 'explanation', 'example', 'class', 'model', 'time', 'activity', 'lab', 'slow']

Labelled chapter: 7 : Business Process Solutioning (IT requirements)

Keywords appearing in both chapter and topic: ['clear', 'example', 'activity', 'slow']

Topic 3

['video', 'class', 'time', 'professor', 'give', 'go', 'could', 'good', 'think', 'part']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 4

['kahoot', 'quiz', 'recap', 'think', 'time', 'us', 'class', 'give', 'question', 'could']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 5

['example', 'model', 'us', 'professor', 'give', 'could', 'would', 'maybe', 'understand', 'question']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 6

['professor', 'slow', 'speak', 'class', 'pace', 'practice', 'bit', 'go', 'louder', 'signavio']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 7

['good', 'time', 'us', 'could', 'professor', 'example', 'think', 'give', 'question', 'would']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 8

['class', 'activity', 'time', 'group', 'part', 'give', 'break', 'discussion', 'week', 'good']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 9

['slide', 'study', 'class', 'diagram', 'lab', 'interactive', 'go', 'upload', 'quiz', 'give']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

### Aggregation - Academic Weeks

Topic 0

['example', 'class', 'model', 'time', 'could', 'professor', 'good', 'slide', 'give', 'discussion']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 1

['class', 'example', 'model', 'time', 'activity', 'professor', 'give', 'need', 'part', 'good']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 2

['class', 'example', 'model', 'time', 'lab', 'activity', 'slow', 'us', 'good', 'give']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 3

['lab', 'class', 'professor', 'go', 'time', 'give', 'slide', 'slow', 'example', 'good']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 4

['study', 'class', 'time', 'presentation', 'give', 'example', 'could', 'would', 'understand', 'concept']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 5

['example', 'class', 'professor', 'could', 'slide', 'give', 'us', 'lab', 'discussion', 'good']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 6

['class', 'lab', 'time', 'slow', 'example', 'go', 'professor', 'give', 'good', 'slide']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 7

['class', 'example', 'slow', 'lab', 'professor', 'discussion', 'go', 'time', 'good', 'could']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 8

['professor', 'presentation', 'class', 'project', 'thank', 'make', 'interesting', 'question', 'good', 'could']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 9

['model', 'class', 'example', 'week', 'time', 'activity', 'teaching', 'good', 'us', 'make']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

### GSDMM

Topic 0

['group', 'less', 'could', 'discussion', 'lab', 'give', 'teaching', 'activity', 'class', 'time']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 1

['quite', 'interactive', 'would', 'group', 'activity', 'good', 'discussion', 'time', 'study', 'class']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 2

['part', 'send', 'short', 'lecture', 'hope', 'attention', 'lab', 'us', 'end', 'class']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 3

['study', 'give', 'maybe', 'good', 'video', 'model', 'life', 'real\_life', 'real', 'example']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 4

['today', 'great', 'lab', 'good', 'pace', 'bit', 'speak', 'slow', 'class', 'professor']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 5

['miss', 'sure', 'need', 'week', 'time', 'take', 'slide', 'upload', 'lab', 'class']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 6

['time', 'teach', 'professor', 'exercise', 'give', 'discussion', 'part', 'practice', 'question', 'class']

Labelled chapter: 5 : Business Process Analysis (Static)

Keywords appearing in both chapter and topic: ['exercise', 'part', 'practice', 'class']

Topic 7

['slide', 'time', 'could', 'professor', 'go', 'class', 'give', 'us', 'model', 'example']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 8

['maybe', 'interactive', 'check', 'make', 'understanding', 'test', 'question', 'self', 'kahoot', 'quiz']

Labelled chapter: No matching chapters

Keywords appearing in both chapter and topic: []

Topic 9

['model', 'pace', 'quite', 'really', 'slow', 'signavio', 'hard', 'professor', 'example', 'class']

Labelled chapter: 5 : Business Process Analysis (Static)

Keywords appearing in both chapter and topic: ['model', 'pace', 'really', 'class']

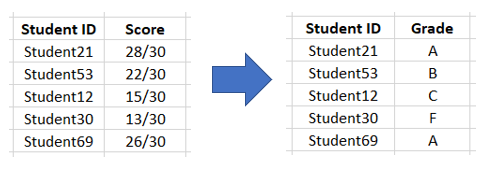
# 

# **Appendix B**

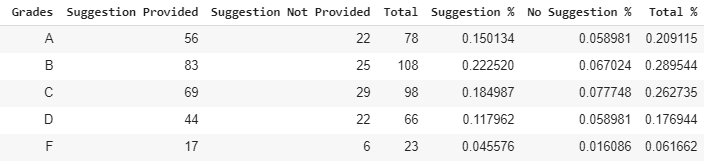
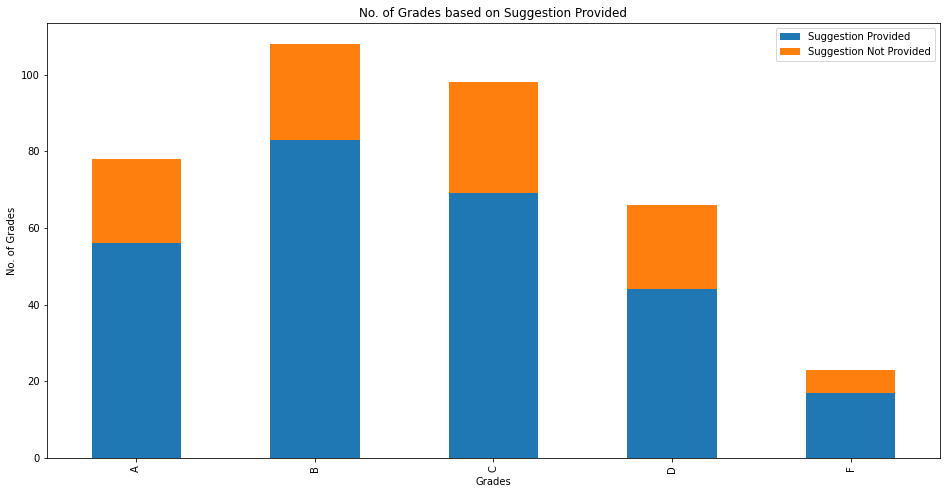
## **Feature Engineering**

This session consists of the full explanation of the feature engineering processes that we did. 7 new features were derived and created from the existing features in our dataset. They are grades, sentiment score, feedback length proportion, class type, frequency of suggestion, fuzzy similarity scores and topic modelling.

### Grades

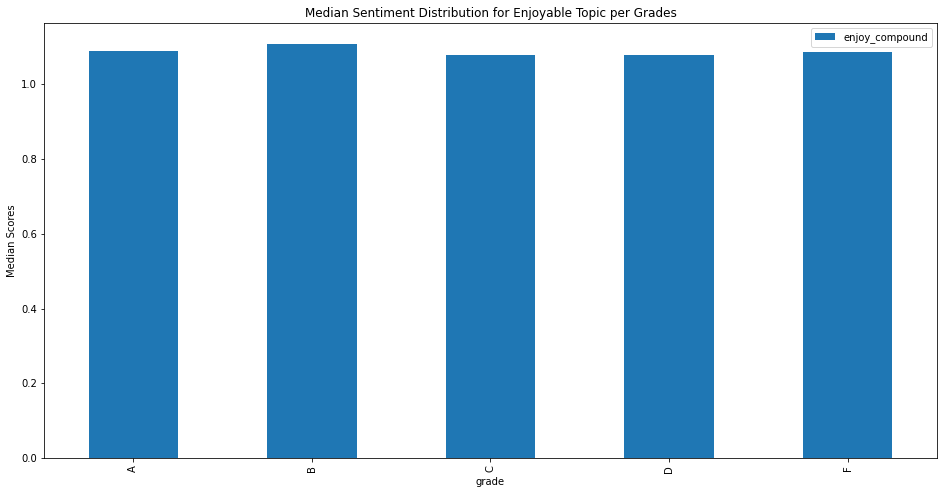


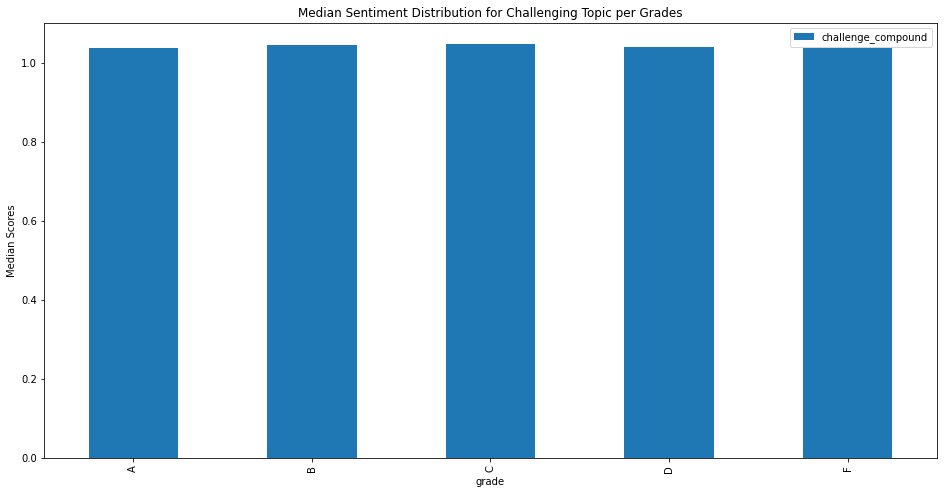
As we only have the total numeric score of each student, we converted these scores into categorical variables to train our classifier. We followed the SMU’s grading system and assigned the grades (A, B, C, D & F) accordingly.

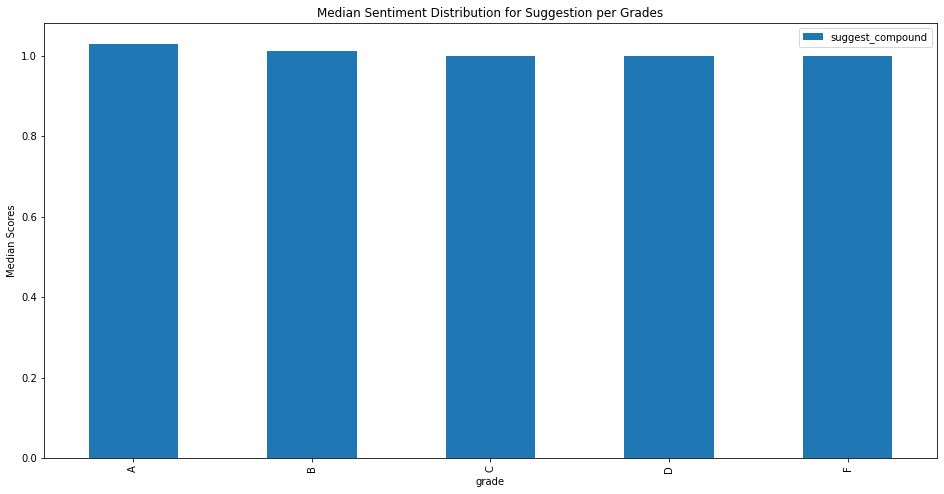


From the charts shown above, we can observe that there 20.9115% of the students received A, 28.9544% of the students received B, 26.2735% of the students received C, 17.6944% of the students received D and 6.1662% of the students received F. This follows a normal bell-curve which schools tend to follow when grading students. To evaluate whether our chosen classification model is effective, it must be able to have a F1 Score of more than 28.9544%, which is the median of the distribution.

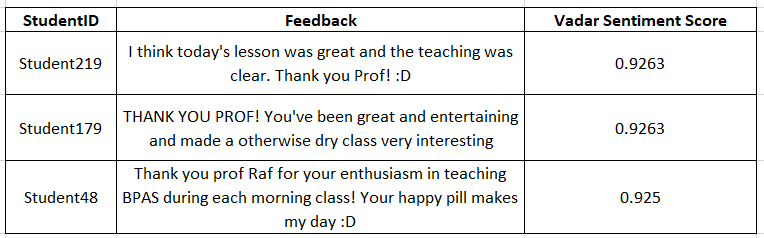
### Sentiment Score





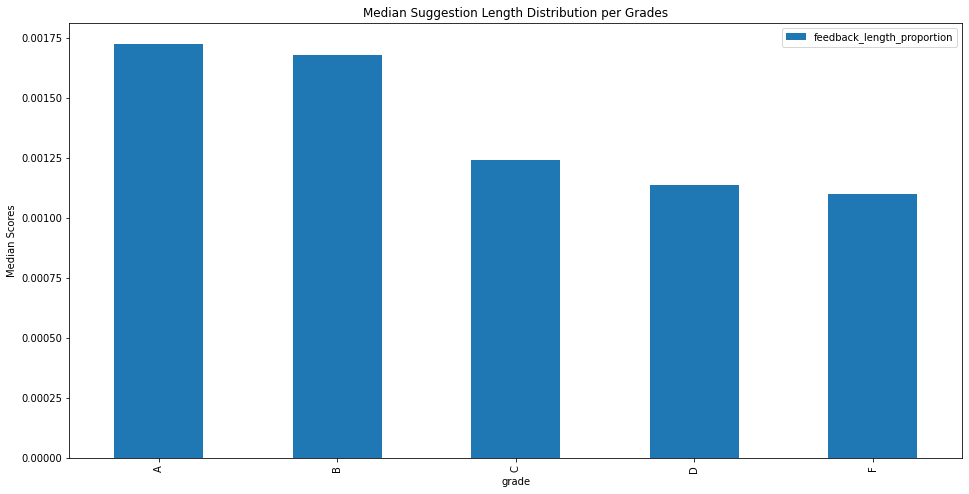


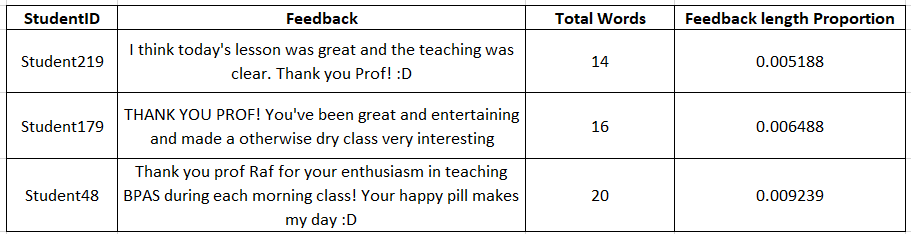
**Sentiment scores** are reflected based on the student’s average sentiment score of enjoyable, challenging & Feedback provided. The purpose of doing a sentiment analysis is to evaluate if there is a correlation or impact between the students’ sentiment towards the course and their course grades. This is inspired from a post in research study[[1]](#footnote-1) in Harvard where it illustrates that happy students tend to have better grades (Jones, 2015).



To find out the sentiment score of the students’ feedback, we used **Vader Sentiment Analyzer** to predict sentiment scores for their Feedbacks. We conducted sentiment analysis on individual feedback and calculated the polarity score for each student before fitting it into our predictive model. However, due to the limitation of Multinomial Naive Bayes, it is unable to take in negative values. Therefore, to rectify this, we increased the range of the polarity scores by 1 before fitting into our models. To illustrate the inherent values of the sentiment score, the graphs shown above are the distribution of sentiment scores before the increment.

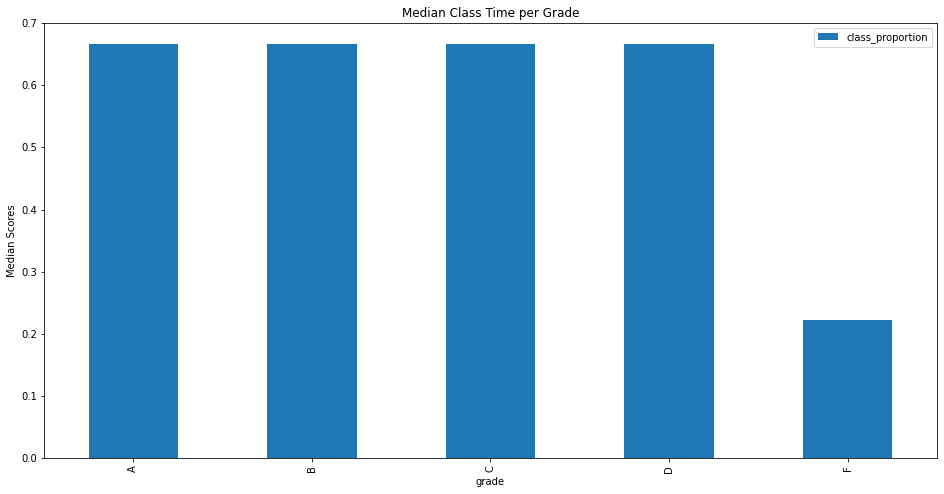
### Feedback Length Proportion

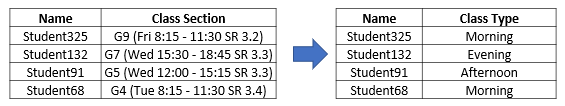


****

**Feedback Length Proportion** refers to the total number of words provided by each student (after text cleaning). In order to standardize the score, we divided it by the total number of words provided by all students (after text cleaning). This is only based on the cleaned text from the Suggestion column. This was inspired from a research article (Svinicki, 2001) where it states that students who give feedback are more motivated and self-regulated. From the graph above, we can observe that those who received higher grades tend to write longer suggestions than those who get lower grades. Hence, we would like to evaluate if the Feedback Length Proportion of a student has any correlation to their grades.

### Class Type (morning, afternoon, evening)

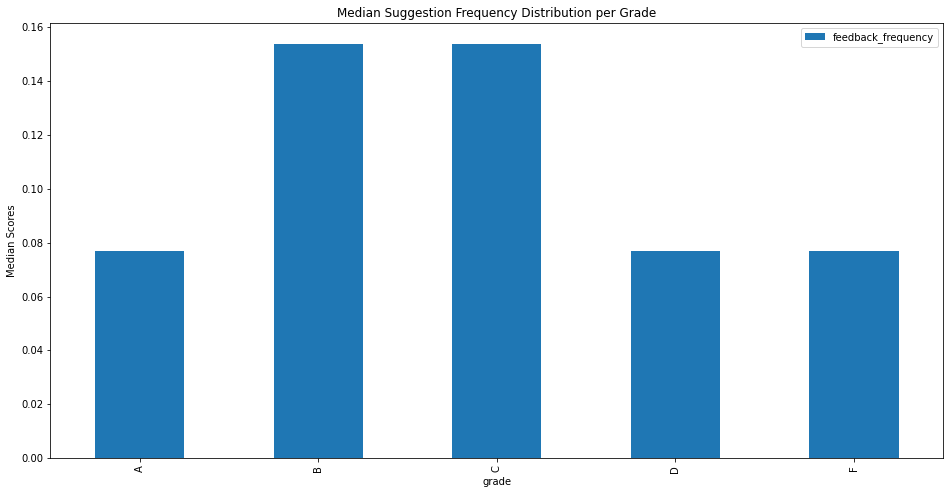




**Class Type** refers to the students attending either morning, afternoon or evening class. [8:15am, 12pm, 3:30pm]

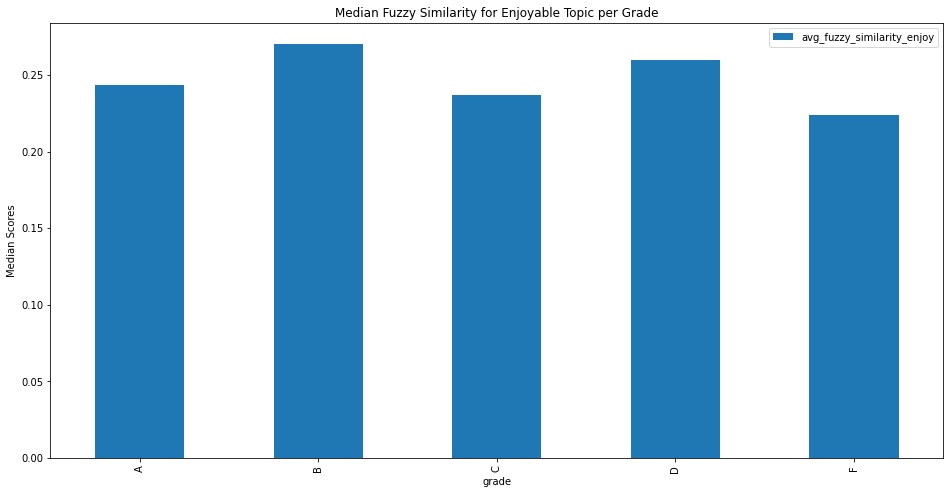
We convert the class types into numeric scores [the number of a particular class type/total number of classes] to fit into our model for training. The purpose of feature engineering is to evaluate if the student’s class time has any correlation or impact towards their grade. This was inspired from various research studies (D’agnese, 2015) claiming that the student’s class time affects their productivity, attentiveness and alertness. Hence, we would like to see if this is true in our case.

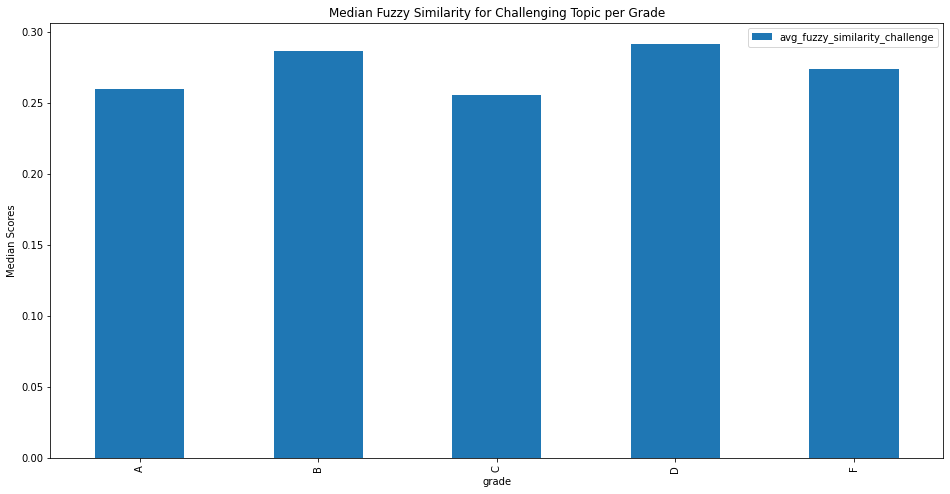
### Frequency of Suggestion

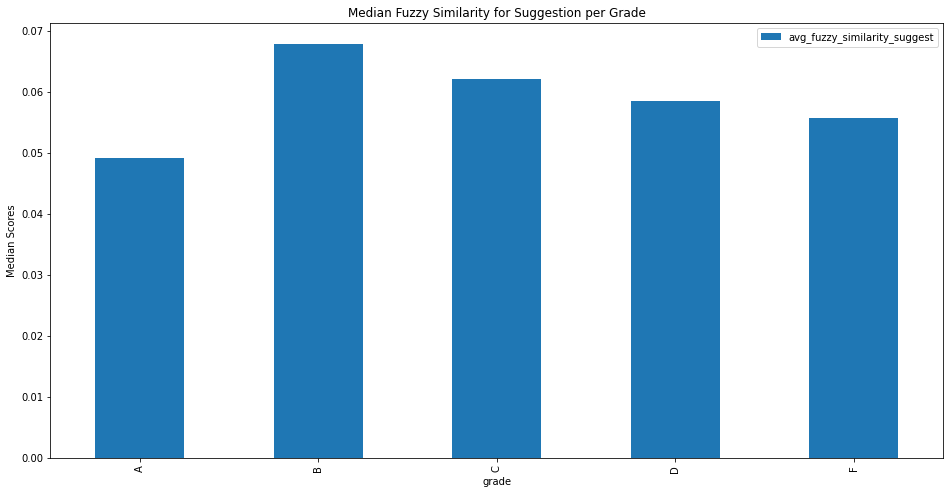


**Suggestion Frequency** refers to the number of times each student has provided a suggestion (after text cleaning) out of the whole week. This is also based on the suggestion column. The reason why we engineered this feature is similar to the Feedback Length Proportion where students who give more feedback are deemed to be more motivated and self-regulated. From the graph above, we can observe that those who received B and C usually provide frequent suggestions to improve the course. Hence, we would like to evaluate is there is a correlation of the number of feedbacks provided and the grade

### Fuzzy Similarity Scores between Feedback & Course

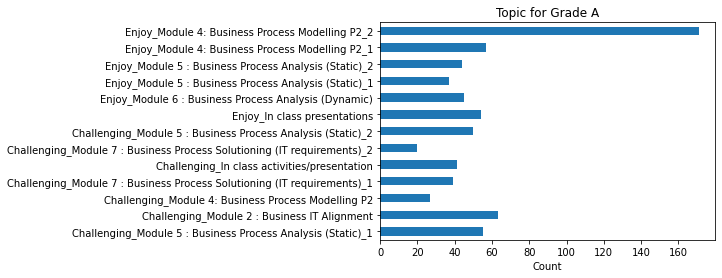


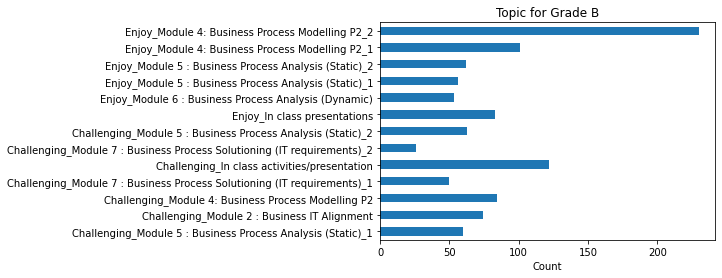


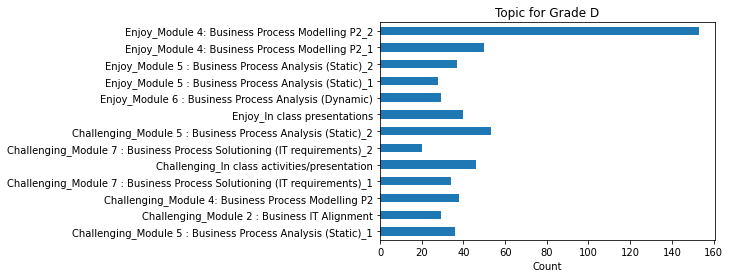
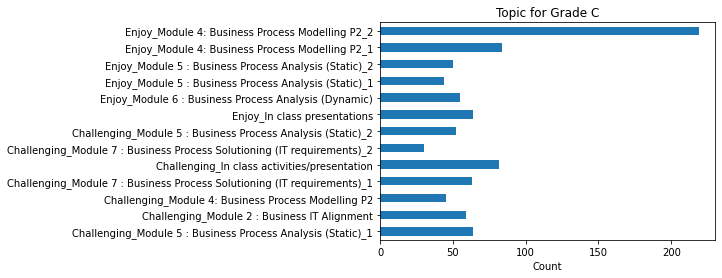


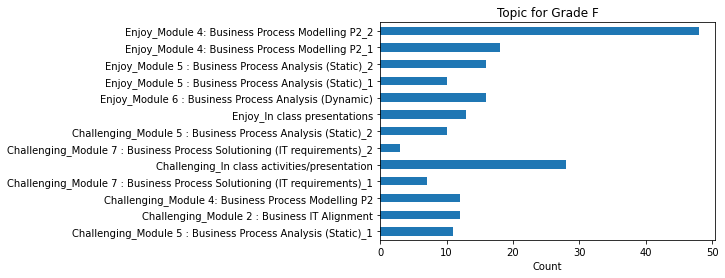
Fuzzy Similarity refers to how similar the feedback provided with the teaching materials or components of the assignment grades. We formulated the teaching materials or components into a gazette. We used the ratio algorithm of the fuzzy matching where we matched the Feedback with the words in the gazette. For each matching term we find, we multiply the fuzzy similarity scores according to the number of words in the term. For example, if the highest score comes from a term of 5 words, the feedback will be assigned a perfect similarity score. However, if the highest score comes from a term of 1 word, the feedback will be assigned 20% of its similarity score. This is because it is more meaningful when a student typed the technical terms in its full form rather than just a part of it. This score will then be averaged and assigned to each student. The reason why we did this is because we assumed that students who mentioned anything related to the teaching materials or the assignment, wanted to do well for the course and might affect the grade they get. We did this for the three columns, Challenging Topic, Enjoying Topic & Feedback.

### Topic Modelling









This feature is built on the idea that if a student finds a particular topic challenging, especially an important topic that builds on the foundation for other important chapters, then that may have a significant influence on the student’s academic performance in that module. The same can also be said for modules which the student is strong in. This is backed up by the study (DeMonbrun et al, 2018) mentioned in the “Future works” section of Task 1, where it was mentioned that students experiencing difficulties in some courses may have a “snowball effect” on other courses.

Putting in context, we want to know how many times a particular topic was mentioned in their “Challenging” and “Enjoyable” topic feedback columns. This allows us to gauge which areas in the syllabus they repeatedly find challenging or enjoyable, which may be latent indicators about their final academic performance. For instance, a student may subconsciously indicate that he is weak in drawing diagrams, that may be conveyed indirectly by mentioning that he finds drawing collaboration diagrams challenging, and then in future weeks he again mentions that solution overview diagrams are challenging. They may appear in the same topic generated from our topic models, which leads us to discover this from the distribution of relevant topics in their feedback, which is obtained from our topic modelling task.

The topic models are first generated in Task 1, after which we can infer a dominant topic from every document. We aggregated all the students by summing the counts of the topics for every student, before removing those topics that are irrelevant or incoherent. The distributions of these topics are then plotted against students of different grades. We can observe, for “A” students, they have a distinctive difference in distribution than the “F” students. For “A” students, their topic 4 (Finding in class activities or presentation challenging) is roughly the same as the rest of the topics. However, for “F” students, the same topic had a much higher count among other topics. Hence, we can infer that students who found in class activities or presentation challenging, may have a higher chance of receiving lower overall grades, which makes this a strong predictor for our model.

# 

# **Appendix C**

## **Feature Selection Approaches**

The tables below show the full evaluation score of all the possible feature combinations across the 3 feature selection approaches that we took.

* The first approach uses every word in the corpus to train and test the classifier.
* The second approach uses only words that occur more than the median count
* The third approach uses only words that occur less than the median count

### First Approach (All Words)

|  |  |  |  |
| --- | --- | --- | --- |
| **Best Features** | **F1 Macro** | **F1 Weighted** | **Run Time (Seconds)** |
| Feedback | 0.092339 | 0.11242 | 35.77208 |
| Feedback + Class Time | 0.105631 | 0.120764 | 35.52531 |
| Feedback + Length of Feedback | 0.135188 | 0.174691 | 36.19272 |
| Feedback + Sentiment Score | 0.153082 | 0.19868 | 31.60345 |
| Feedback + Frequency of Suggestion | 0.133598 | 0.17335 | 34.6199 |
| Feedback + Fuzzy Similarity Scores between Feedback & Course | 0.145659 | 0.174592 | 32.33676 |
| Feedback + Topic Modelling | 0.164009 | 0.198608 | 36.59881 |
| Feedback + Class Time + Length of Feedback | 0.149366 | 0.185522 | 36.35903 |
| Feedback + Class Time + Sentiment Score | 0.159659 | 0.206248 | 31.65619 |
| Feedback + Class Time + Frequency of Suggestion | 0.175235 | 0.215877 | 34.89802 |
| Feedback + Class Time + Fuzzy Similarity Scores between Feedback & Course | 0.119505 | 0.155261 | 30.89379 |
| Feedback + Class Time + Topic Modelling | 0.174695 | 0.214245 | 36.99092 |
| Feedback + Length of Feedback + Sentiment Score | 0.158549 | 0.205806 | 32.67653 |
| Feedback + Frequency of Suggestion + Length of Feedback | 0.143159 | 0.182571 | 35.2452 |
| Feedback + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course | 0.145757 | 0.174753 | 31.83317 |
| Feedback + Length of Feedback + Topic Modelling | 0.166578 | 0.211306 | 37.67401 |
| Feedback + Frequency of Suggestion + Sentiment Score | 0.164225 | 0.207193 | 30.62709 |
| Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.131032 | 0.170535 | 29.98687 |
| Feedback + Sentiment Score + Topic Modelling | 0.183319 | 0.232054 | 37.15045 |
| Feedback + Frequency of Suggestion + Fuzzy Similarity Scores between Feedback & Course | 0.155106 | 0.194725 | 32.92623 |
| Feedback + Frequency of Suggestion + Topic Modelling | 0.181993 | 0.228926 | 36.84811 |
| Feedback + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.201943 | 0.252175 | 35.90155 |
| Feedback + Class Time + Length of Feedback + Sentiment Score | 0.167204 | 0.215204 | 30.95917 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback | 0.163627 | 0.203833 | 33.9845 |
| Feedback + Class Time + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course | 0.124542 | 0.160645 | 31.40232 |
| Feedback + Class Time + Length of Feedback + Topic Modelling | 0.177893 | 0.224597 | 37.30705 |
| Feedback + Class Time + Frequency of Suggestion + Sentiment Score | 0.151776 | 0.195793 | 32.19821 |
| Feedback + Class Time + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.123311 | 0.157267 | 30.70424 |
| Feedback + Class Time + Sentiment Score + Topic Modelling | 0.169057 | 0.214694 | 34.95952 |
| Feedback + Class Time + Frequency of Suggestion + Fuzzy Similarity Scores between Feedback & Course | 0.136791 | 0.176176 | 30.98108 |
| Feedback + Class Time + Frequency of Suggestion + Topic Modelling | 0.190388 | 0.230067 | 37.52606 |
| Feedback + Class Time + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.207817 | 0.256356 | 37.2782 |
| Feedback + Frequency of Suggestion + Length of Feedback + Sentiment Score | 0.167164 | 0.210699 | 31.69324 |
| Feedback + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.136255 | 0.16162 | 30.69072 |
| Feedback + Length of Feedback + Sentiment Score + Topic Modelling | 0.158011 | 0.204033 | 36.95676 |
| Feedback + Frequency of Suggestion + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course | 0.149286 | 0.190836 | 32.27456 |
| Feedback + Frequency of Suggestion + Length of Feedback + Topic Modelling | 0.187211 | 0.234911 | 37.73335 |
| Feedback + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.207971 | 0.259484 | 37.10265 |
| Feedback + Frequency of Suggestion + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.150683 | 0.193324 | 30.39043 |
| Feedback + Frequency of Suggestion + Sentiment Score + Topic Modelling | 0.188971 | 0.239715 | 36.70581 |
| Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.155224 | 0.194622 | 37.06139 |
| Feedback + Frequency of Suggestion + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.192004 | 0.236022 | 36.08026 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Sentiment Score | 0.149108 | 0.192355 | 32.34246 |
| Feedback + Class Time + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.122173 | 0.154797 | 31.7737 |
| Feedback + Class Time + Length of Feedback + Sentiment Score + Topic Modelling | 0.17366 | 0.219716 | 35.79259 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course | 0.136744 | 0.176445 | 31.38951 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Topic Modelling | 0.191582 | 0.232159 | 35.84652 |
| Feedback + Class Time + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.20848 | 0.258622 | 37.15002 |
| Feedback + Class Time + Frequency of Suggestion + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.128676 | 0.161269 | 30.28197 |
| Feedback + Class Time + Frequency of Suggestion + Sentiment Score + Topic Modelling | 0.176193 | 0.2243 | 37.36489 |
| Feedback + Class Time + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.157631 | 0.199617 | 37.97935 |
| Feedback + Class Time + Frequency of Suggestion + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.219578 | 0.270086 | 36.08971 |
| Feedback + Frequency of Suggestion + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.165197 | 0.196885 | 30.48096 |
| Feedback + Frequency of Suggestion + Length of Feedback + Sentiment Score + Topic Modelling | 0.173629 | 0.219031 | 36.60464 |
| Feedback + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.167848 | 0.210933 | 36.53069 |
| Feedback + Frequency of Suggestion + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.192004 | 0.236022 | 36.62348 |
| Feedback + Frequency of Suggestion + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.148311 | 0.18757 | 37.79226 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.130057 | 0.158269 | 31.68951 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Sentiment Score + Topic Modelling | 0.158919 | 0.203404 | 38.2887 |
| Feedback + Class Time + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.157583 | 0.199573 | 38.44286 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.21909 | 0.270301 | 37.45836 |
| Feedback + Class Time + Frequency of Suggestion + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.160549 | 0.20385 | 38.6966 |
| Feedback + Frequency of Suggestion + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.162571 | 0.206285 | 37.6951 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.160549 | 0.20385 | 38.23531 |

### Second Approach (Words which occurs more than median word count)

|  |  |  |  |
| --- | --- | --- | --- |
| **Best Features** | **F1 Macro** | **F1 Weighted** | **Run Time (Seconds)** |
| Feedback | 0.093 | 0.120146 | 13.16808 |
| Feedback + Class Time | 0.118946 | 0.146287 | 12.03519 |
| Feedback + Length of Feedback | 0.141155 | 0.178356 | 14.00284 |
| Feedback + Sentiment Score | 0.160056 | 0.209787 | 13.89282 |
| Feedback + Frequency of Suggestion | 0.159048 | 0.205484 | 11.61198 |
| Feedback + Fuzzy Similarity Scores between Feedback & Course | 0.140789 | 0.169047 | 12.12279 |
| Feedback + Topic Modelling | 0.187317 | 0.235804 | 12.88417 |
| Feedback + Class Time + Length of Feedback | 0.137528 | 0.173944 | 13.24844 |
| Feedback + Class Time + Sentiment Score | 0.154795 | 0.198482 | 11.80169 |
| Feedback + Class Time + Frequency of Suggestion | 0.15613 | 0.203052 | 12.37282 |
| Feedback + Class Time + Fuzzy Similarity Scores between Feedback & Course | 0.126338 | 0.163231 | 13.72604 |
| Feedback + Class Time + Topic Modelling | 0.16392 | 0.211384 | 13.27489 |
| Feedback + Length of Feedback + Sentiment Score | 0.158959 | 0.203859 | 12.24153 |
| Feedback + Frequency of Suggestion + Length of Feedback | 0.142477 | 0.182933 | 13.47887 |
| Feedback + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course | 0.144201 | 0.173575 | 12.58838 |
| Feedback + Length of Feedback + Topic Modelling | 0.1889 | 0.237677 | 14.15667 |
| Feedback + Frequency of Suggestion + Sentiment Score | 0.171009 | 0.213613 | 13.76968 |
| Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.143859 | 0.172428 | 10.36105 |
| Feedback + Sentiment Score + Topic Modelling | 0.163468 | 0.208604 | 14.37321 |
| Feedback + Frequency of Suggestion + Fuzzy Similarity Scores between Feedback & Course | 0.165696 | 0.193393 | 13.89869 |
| Feedback + Frequency of Suggestion + Topic Modelling | 0.17534 | 0.220711 | 12.92971 |
| Feedback + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.210754 | 0.262433 | 13.02539 |
| Feedback + Class Time + Length of Feedback + Sentiment Score | 0.164386 | 0.210529 | 12.78051 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback | 0.148244 | 0.186407 | 13.77631 |
| Feedback + Class Time + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course | 0.11019 | 0.145993 | 12.94651 |
| Feedback + Class Time + Length of Feedback + Topic Modelling | 0.171463 | 0.217705 | 13.90866 |
| Feedback + Class Time + Frequency of Suggestion + Sentiment Score | 0.158484 | 0.204824 | 12.02964 |
| Feedback + Class Time + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.121739 | 0.155022 | 12.12496 |
| Feedback + Class Time + Sentiment Score + Topic Modelling | 0.18835 | 0.239593 | 12.67142 |
| Feedback + Class Time + Frequency of Suggestion + Fuzzy Similarity Scores between Feedback & Course | 0.131485 | 0.168373 | 12.73514 |
| Feedback + Class Time + Frequency of Suggestion + Topic Modelling | 0.178658 | 0.225271 | 14.21877 |
| Feedback + Class Time + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.200462 | 0.248357 | 13.62392 |
| Feedback + Frequency of Suggestion + Length of Feedback + Sentiment Score | 0.166707 | 0.208819 | 12.89853 |
| Feedback + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.14396 | 0.172124 | 11.44179 |
| Feedback + Length of Feedback + Sentiment Score + Topic Modelling | 0.166096 | 0.212181 | 15.60737 |
| Feedback + Frequency of Suggestion + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course | 0.161382 | 0.188732 | 13.42112 |
| Feedback + Frequency of Suggestion + Length of Feedback + Topic Modelling | 0.204123 | 0.253279 | 12.74816 |
| Feedback + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.207354 | 0.25647 | 13.7662 |
| Feedback + Frequency of Suggestion + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.163482 | 0.193218 | 12.78572 |
| Feedback + Frequency of Suggestion + Sentiment Score + Topic Modelling | 0.171825 | 0.214121 | 12.72052 |
| Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.162553 | 0.20478 | 13.00389 |
| Feedback + Frequency of Suggestion + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.212915 | 0.260709 | 13.84443 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Sentiment Score | 0.144602 | 0.189081 | 12.67134 |
| Feedback + Class Time + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.126067 | 0.161379 | 11.65589 |
| Feedback + Class Time + Length of Feedback + Sentiment Score + Topic Modelling | 0.173134 | 0.220879 | 13.85233 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course | 0.142887 | 0.183862 | 12.30735 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Topic Modelling | 0.186043 | 0.233349 | 13.94119 |
| Feedback + Class Time + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.202522 | 0.251371 | 13.83042 |
| Feedback + Class Time + Frequency of Suggestion + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.129956 | 0.158165 | 12.76759 |
| Feedback + Class Time + Frequency of Suggestion + Sentiment Score + Topic Modelling | 0.190813 | 0.241605 | 13.83784 |
| Feedback + Class Time + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.159939 | 0.202892 | 13.56433 |
| Feedback + Class Time + Frequency of Suggestion + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.207026 | 0.258227 | 14.20159 |
| Feedback + Frequency of Suggestion + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.16585 | 0.195845 | 12.2139 |
| Feedback + Frequency of Suggestion + Length of Feedback + Sentiment Score + Topic Modelling | 0.184523 | 0.230534 | 14.0131 |
| Feedback + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.159939 | 0.201327 | 13.70241 |
| Feedback + Frequency of Suggestion + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.212023 | 0.261609 | 13.23526 |
| Feedback + Frequency of Suggestion + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.155677 | 0.196505 | 13.3109 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.128479 | 0.156689 | 13.18208 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Sentiment Score + Topic Modelling | 0.190003 | 0.239558 | 13.60843 |
| Feedback + Class Time + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.161256 | 0.204695 | 13.5716 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.207431 | 0.257827 | 14.49233 |
| Feedback + Class Time + Frequency of Suggestion + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.160694 | 0.20405 | 14.31374 |
| Feedback + Frequency of Suggestion + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.165615 | 0.208905 | 13.63433 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.160706 | 0.204435 | 13.79335 |

### Third Approach (Words that occur less than median word count)

|  |  |  |  |
| --- | --- | --- | --- |
| **Best Features** | **F1 Macro** | **F1 Weighted** | **Run Time (Seconds)** |
| Feedback | 0.097056 | 0.134113 | 26.77667 |
| Feedback + Class Time | 0.099467 | 0.124925 | 25.88925 |
| Feedback + Length of Feedback | 0.119932 | 0.148666 | 27.50138 |
| Feedback + Sentiment Score | 0.188896 | 0.234672 | 22.9513 |
| Feedback + Frequency of Suggestion | 0.14105 | 0.180412 | 27.4455 |
| Feedback + Fuzzy Similarity Scores between Feedback & Course | 0.143645 | 0.184387 | 23.00474 |
| Feedback + Topic Modelling | 0.153795 | 0.197596 | 29.50584 |
| Feedback + Class Time + Length of Feedback | 0.139625 | 0.173686 | 27.37805 |
| Feedback + Class Time + Sentiment Score | 0.167702 | 0.215713 | 25.23188 |
| Feedback + Class Time + Frequency of Suggestion | 0.163727 | 0.208722 | 27.1921 |
| Feedback + Class Time + Fuzzy Similarity Scores between Feedback & Course | 0.172048 | 0.193676 | 23.54957 |
| Feedback + Class Time + Topic Modelling | 0.186175 | 0.225574 | 28.19806 |
| Feedback + Length of Feedback + Sentiment Score | 0.178664 | 0.220076 | 23.98466 |
| Feedback + Frequency of Suggestion + Length of Feedback | 0.131169 | 0.163022 | 28.43349 |
| Feedback + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course | 0.130597 | 0.168262 | 21.86104 |
| Feedback + Length of Feedback + Topic Modelling | 0.184944 | 0.233265 | 29.79314 |
| Feedback + Frequency of Suggestion + Sentiment Score | 0.198375 | 0.240937 | 24.35371 |
| Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.150752 | 0.175352 | 22.99623 |
| Feedback + Sentiment Score + Topic Modelling | 0.190238 | 0.237371 | 28.35827 |
| Feedback + Frequency of Suggestion + Fuzzy Similarity Scores between Feedback & Course | 0.133282 | 0.166456 | 21.87251 |
| Feedback + Frequency of Suggestion + Topic Modelling | 0.185945 | 0.223097 | 27.73404 |
| Feedback + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.212912 | 0.263685 | 29.07971 |
| Feedback + Class Time + Length of Feedback + Sentiment Score | 0.172304 | 0.212756 | 24.81161 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback | 0.156207 | 0.195883 | 26.73269 |
| Feedback + Class Time + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course | 0.158299 | 0.180633 | 23.92016 |
| Feedback + Class Time + Length of Feedback + Topic Modelling | 0.173065 | 0.218078 | 28.50052 |
| Feedback + Class Time + Frequency of Suggestion + Sentiment Score | 0.169684 | 0.21595 | 23.87246 |
| Feedback + Class Time + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.148828 | 0.178688 | 22.31276 |
| Feedback + Class Time + Sentiment Score + Topic Modelling | 0.203818 | 0.251092 | 28.95478 |
| Feedback + Class Time + Frequency of Suggestion + Fuzzy Similarity Scores between Feedback & Course | 0.172378 | 0.203081 | 23.12955 |
| Feedback + Class Time + Frequency of Suggestion + Topic Modelling | 0.194094 | 0.232688 | 29.02033 |
| Feedback + Class Time + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.205021 | 0.253298 | 28.72355 |
| Feedback + Frequency of Suggestion + Length of Feedback + Sentiment Score | 0.183444 | 0.229018 | 24.18428 |
| Feedback + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.15085 | 0.176203 | 24.86165 |
| Feedback + Length of Feedback + Sentiment Score + Topic Modelling | 0.186212 | 0.230942 | 28.46893 |
| Feedback + Frequency of Suggestion + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course | 0.130298 | 0.166895 | 22.71377 |
| Feedback + Frequency of Suggestion + Length of Feedback + Topic Modelling | 0.178423 | 0.214039 | 28.43868 |
| Feedback + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.216353 | 0.268767 | 29.07056 |
| Feedback + Frequency of Suggestion + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.127274 | 0.162952 | 23.33399 |
| Feedback + Frequency of Suggestion + Sentiment Score + Topic Modelling | 0.191544 | 0.239432 | 28.56304 |
| Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.187187 | 0.233752 | 28.00374 |
| Feedback + Frequency of Suggestion + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.217475 | 0.2652 | 30.50999 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Sentiment Score | 0.168352 | 0.214351 | 24.69737 |
| Feedback + Class Time + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.155877 | 0.185312 | 23.78704 |
| Feedback + Class Time + Length of Feedback + Sentiment Score + Topic Modelling | 0.200337 | 0.247648 | 29.09228 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course | 0.140464 | 0.17193 | 21.99298 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Topic Modelling | 0.196737 | 0.234972 | 27.94793 |
| Feedback + Class Time + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.201229 | 0.249676 | 27.82864 |
| Feedback + Class Time + Frequency of Suggestion + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.153504 | 0.185544 | 22.95472 |
| Feedback + Class Time + Frequency of Suggestion + Sentiment Score + Topic Modelling | 0.186163 | 0.232486 | 27.18013 |
| Feedback + Class Time + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.168501 | 0.21242 | 28.26508 |
| Feedback + Class Time + Frequency of Suggestion + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.186065 | 0.232518 | 28.79519 |
| Feedback + Frequency of Suggestion + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.125741 | 0.161855 | 22.38871 |
| Feedback + Frequency of Suggestion + Length of Feedback + Sentiment Score + Topic Modelling | 0.199662 | 0.248317 | 28.80842 |
| Feedback + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.177655 | 0.222165 | 27.66254 |
| Feedback + Frequency of Suggestion + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.204039 | 0.2491 | 29.18801 |
| Feedback + Frequency of Suggestion + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.177229 | 0.222353 | 29.92272 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course | 0.139385 | 0.174285 | 22.80803 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Sentiment Score + Topic Modelling | 0.191714 | 0.240703 | 27.76969 |
| Feedback + Class Time + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.162353 | 0.203903 | 27.93001 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.195607 | 0.244766 | 28.86696 |
| Feedback + Class Time + Frequency of Suggestion + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.171181 | 0.21708 | 28.74445 |
| Feedback + Frequency of Suggestion + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.177284 | 0.223361 | 26.39979 |
| Feedback + Class Time + Frequency of Suggestion + Length of Feedback + Sentiment Score + Fuzzy Similarity Scores between Feedback & Course + Topic Modelling | 0.165904 | 0.208553 | 29.15324 |

# 

1. [↑](#footnote-ref-1)