32933 RESEARCH PROJECT

DISCOVER RELATIONSHIP BETWEEN DOCUMENT CORPUS VIA NATURAL LANGUAGE *PROCESSING*

Supervisor Dr Jun Li
12976433 Shuting Xu
13023484 Jingyi Lin
12866152 Wenxi Tu

AUTUNM SEMESTER 2020 YEAR
WORD COUNT: 3066

Contents

Introduction

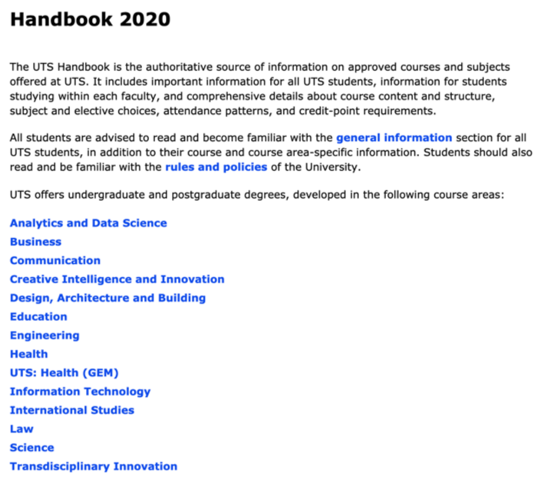
The corpus refers to a large electronic text library that has been scientifically sampled and processed. Through using computer analysis tools, researchers can conduct relevant language theory and applied research (McEnery, 2012). Theoretically, there are different relationships between these corpora but not implicit or obvious. Therefore, in order to reduce the difficulty of understanding document corpus and improve efficiency, people usually use natural language processing (NLP) technology. As an interactive technology, NLP realizes the effective communication between humans and computers in natural language. This technology can make data more meaningful by understanding the context of the data, which facilitates text analysis and data mining (Chowdhury, 2003). In addition, it also helps to discover relationships between document corpora.

This report will propose to use NLP technology to represent the document classification. The data used in this project will be the University of Technology Sydney (UTS) subject outline in session. Through the preprocessing, TFIDF and cosine similarity are implemented to discover the relationship between different documents. Then T-SNE is applied to help locate the data. Data-driven documents (D3) is visualization tool which achieved intuitive presentation and enhanced the interactive user experience.

This paper firstly illustrated our inspiration in literature review part, and introduced our system design thinking with data preparation, followed by implement part which includes the techniques and operation, after that are prototype display and discussion, to summary the paper, a brief conclusion is at the final part.

Literature review

As current students in UTS, we found the guide of choosing course is the handbook in UTS, but the courses are classified by faculty and sorted by alphabetic (Figure1), and the corresponding course guide are very general and limited, students cannot find the relationship between different courses. Especially students may feel the selected are similar after enrolling. Some courses have such relationship like requisite and anti-requisite, but students must check the detailed outlines to figure them out.


Handbook 2020

The UTS Handbook is the authoritative source of information on approved courses and subjects offered at UTS. It includes important information for all UTS students, information for students studying within each faculty, and comprehensive details about course content and structure, subject and elective choices, attendance patterns, and credit-point requirements.

All students are advised to read and become familiar with the **general information** section for all UTS students, in addition to their course and course area-specific information. Students should also read and be familiar with the **rules and policies** of the University.

UTS offers undergraduate and postgraduate degrees, developed in the following course areas:

Analytics and Data Science
Business
Communication
Creative Intelligence and Innovation
Design, Architecture and Building
Education
Engineering
Health
UTS: Health (GEM)
Information Technology
International Studies
Law
Science
Transdisciplinary Innovation

Figure1 sorted faculty in UTS

In terms of the coordinators, they may cannot recognize the relationship among courses due to the large number of courses existing in UTS. At the same time, coordinators should ensure that his or her course description is closed to its relative courses when writing new outlines.

As to UTS, some courses with different name are duplicate even they are from different faculty, so UTS can check the relationship among all courses and reduce the educational expense by canceling or modifying duplicate courses.

System requirement

The system requires the content of course outline with unique ID, name. We need to design the system which can process the raw data, with a model to calculate the similarity based on certain NLP mechanism, then use to handle those course data for visualization.

To present the relationship of courses specifically and reasonably, the requirement of our project is listed below:
1, all courses in all faculties would be presented in our prototype.
2, similar courses have certain visual link between them, and users can compare them with simple operation.
3, the relationship of requisite and anti-requisite is important for selecting course, so that should be highlighted in our prototype.
4, the prototype should be easy to recognize and understand

Selection of dataset

We crawled UTS subject outline as the resource of dataset.

The reason can be listed as below:
1, subject outline is formal document, which has normalized structure and specific meaning.
2, those subjects can be divided into several faculties in UTS, so faculty can be one feature. We assumed some courses from different faculty may have potential connection, so analyzing the similarity between them and point them out is meaningful.

Implementation

System Structure

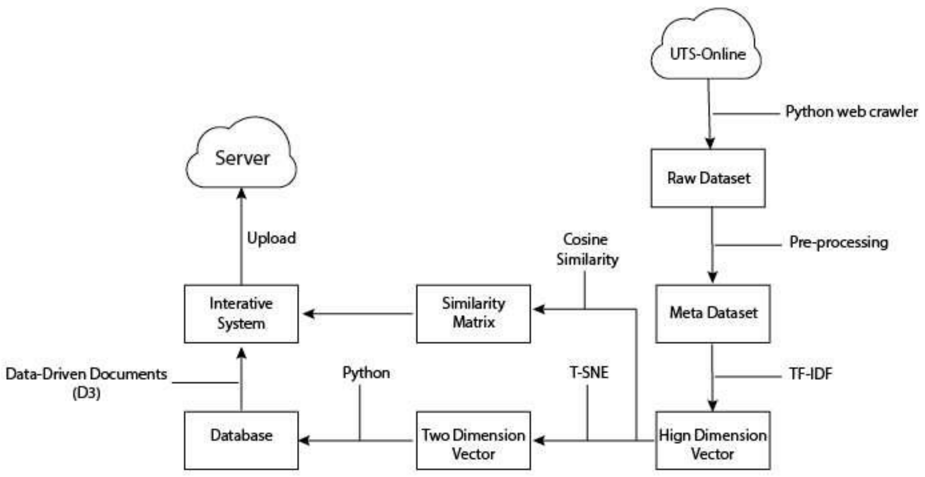


Figure 2 System architect process

After a comprehensive analysis of course document similarity, our team designed a prototype via the relative theory as figure. After data pre-processing, our prototype based on TFIDF to extract the unique words in each document from the whole corpus and convert them into vectors in same long dimension, then T-SNE algorithm is deployed to reduce the vectors into 2-dimension for visualization. Then use data-driven documents (D3) to visualize the processed data to discover the relationships between the documents and enhancing the interactive user experience.

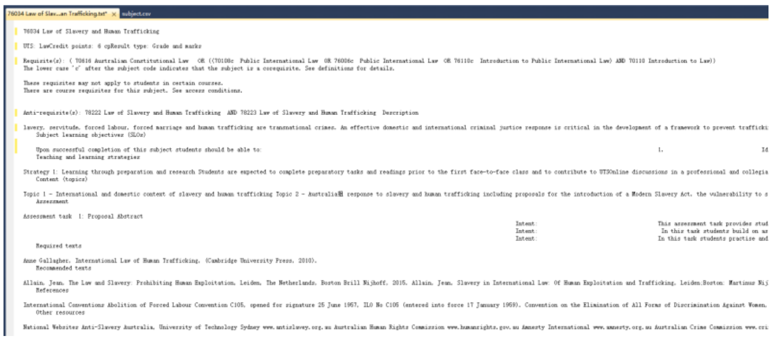
Through this prototype, we can handle following problems:
1, to convert the keywords in individual course outlines into vectors
2, to draw points by reducing the vectors dimension
3, the distance of points presents the similarity among them.

The project was implemented in python which requires various type of libraries and toolkits.

It is very essential to pick up the useful column from our dataset, since the columns has different sense, only the contents that can present the course would be remained and we would clear other unused content.

Data preparation

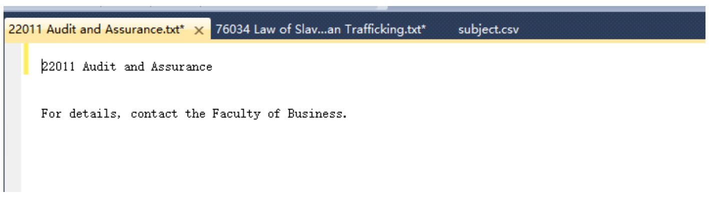
According to the requirements of the project, our group collected data related to the subject. These data will include the subject description, outline, requirements, etc. (Figure3)



```
76034 Law of Slavery and Human Trafficking

UTS: LawCredit points: 6 cpResult type: Grade and marks

Requisite(s): ( 70616 Australian Constitutional Law   OR ((70108c  Public International Law  OR 76006c  Public International Law  OR 76110c  Introduction to Public International Law) AND 70110 Introduction to Law))
The lower case 'c' after the subject code indicates that the subject is a corequisite. See definitions for details.

These requisites may not apply to students in certain courses.
There are course requisites for this subject. See access conditions.


Anti-requisite(s): 78222 Law of Slavery and Human Trafficking  AND 78223 Law of Slavery and Human Trafficking  Description

lavery, servitude, forced labour, forced marriage and human trafficking are transnational crimes. An effective domestic and international criminal justice response is critical in the development of a framework to prevent trafficki
    Subject learning objectives (SLOs)

    Upon successful completion of this subject students should be able to:                                                    1.                              Id
    Teaching and learning strategies

Strategy 1: Learning through preparation and research Students are expected to complete preparatory tasks and readings prior to the first face-to-face class and to contribute to UTSOnline discussions in a professional and collegia
    Content (topics)

Topic 1 - International and domestic context of slavery and human trafficking Topic 2 - Australia responses to slavery and human trafficking including proposals for the introduction of a Modern Slavery Act, the vulnerability to s
    Assessment

Assessment task  1: Proposal Abstract
                                                                                     Intent:                                This assessment task provides stud
                                                                                     Intent:                                 In this task students build on as
                                                                                     Intent:                                In this task students practise and

    Required texts

Anne Gallagher, International Law of Human Trafficking, (Cambridge University Press, 2010).
    Recommended texts

Allain, Jean, The Law and Slavery: Prohibiting Human Exploitation, Leiden, The Netherlands, Boston Brill Nijhoff, 2015, Allain, Jean, Slavery in International Law: Of Human Exploitation and Trafficking, Leiden;Boston: Martinus Nij
    References

International Conventions Abolition of Forced Labour Convention C105, opened for signature 25 June 1957, ILO No C105 (entered into force 17 January 1959). Convention on the Elimination of All Forms of Discrimination Against Women,
    Other resources

National Websites Anti-Slavery Australia, University of Technology Sydney www.antislavery.org.au Australian Human Rights Commission www.humanrights.gov.au Amnesty International www.amnesty.org.au Australian Crime Commission www.cri
```

Figure 3 course outline

Some courses have been cancelled or renamed so the introduction content is relatively simple. The specific content is only ‘for details, contact the Faculty of Business’.


22011 Audit and Assurance.txt*
76034 Law of Slav...an Trafficking.txt*
subject.csv

```
22011 Audit and Assurance

For details, contact the Faculty of Business.
```

Figure 4 useless course outline

This part of data that is not meaningful, so, we will delete them, thereby reducing the amount of calculation. However, not all remaining words can be used, so these data need to be pre-processed by NLP.

Data preprocessing

NLP (nature language processing) is a technology that uses computers to process language and characters. According to the project requirements, we use this technology to remove stop-words and lemma.

Stop words are usually a common word, such as: we, are and other words that are not helpful for classification. According to subject content, words such as “UTS” and “description which appear in each document will be deleted in this operation. NLP pre-processing mainly consists of three parts:

Tokenization	Divide text characters into small and usable communication units. “hello world”to” hello”,”word”
Standardization	Place all text in the same level environment.
Noise removal	Unified format. For example: remove extra spaces, HTML tags

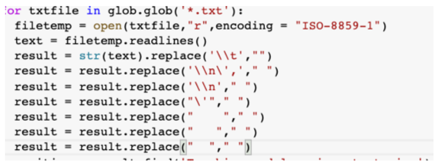
Table 1. NLP pre-processing main parts

Through the pre-processing of these three parts, the data text will be unified in the form of small units and stored in the environment of the same level. Since the subject data in this project is more in line with the requirements, the operation steps in NLP are reduced to following 7 steps.

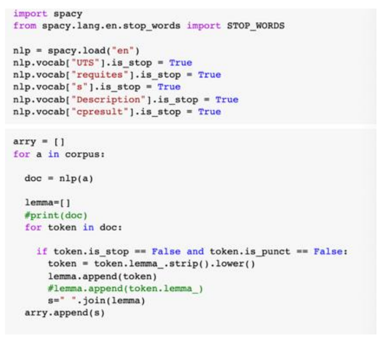
Step	Example
Remove HTML tags	http://, .com
Remove extra spaces	“ “
Remove special characters	“! @ # $ % ^”
Lowercase all text	“Able” to “able”
Delete number	“6”, “7”
Remove stop words	“the”, “it”
Lemmatization	“eating” to “eat”

Table 2. 7 steps of preprocessing

The content has removed HTML tags in crawler part.
Then we used python to remove special characters and space, spacy toolkit to remove stop words and lemmatize words (Figure5).



```
or txtfile in glob.glob('*.txt'):
 filetemp = open(txtfile,"r",encoding = "ISO-8859-1")
 text = filetemp.readlines()
 result = str(text).replace('\\t',"")
 result = result.replace('\\n\',',"  ")
 result = result.replace('\\n'," ")
 result = result.replace("\'"," ")
 result = result.replace("    "," ")
 result = result.replace("   "," ")
 result = result.replace("  "," ")
```



```
import spacy
from spacy.lang.en.stop_words import STOP_WORDS

nlp = spacy.load("en")
nlp.vocab["UTS"].is_stop = True
nlp.vocab["requites"].is_stop = True
nlp.vocab["s"].is_stop = True
nlp.vocab["Description"].is_stop = True
nlp.vocab["cpresult"].is_stop = True
```

```
arry = []
for a in corpus:

  doc = nlp(a)

  lemma=[]
  #print(doc)
  for token in doc:

    if token.is_stop == False and token.is_punct == False:
      token = token.lemma_.strip().lower()
      lemma.append(token)
      #lemma.append(token.lemma_)
      s=" ".join(lemma)
  arry.append(s)
```

Figure 5 python code of preprocessing

The input content:

“92015 Fundamentals ~~of~~ Mental Health Nursing (Graduate Entry) ~~UTS~~: Health ~~Requisite(s):~~ 92017 Health Assessment ~~and~~ Nursing Therapeutics ~~There are~~ course requisites ~~for this~~ subject. ~~See~~ access conditions. ~~Description~~ Students explore ~~a~~ range ~~of~~ issues related ~~to the~~ promotion ~~of~~ health ~~and the~~ nursing care ~~of people in need of~~ mental health care.”

The output content:
92015 fundamental mental health nursing graduate entry health 92017 health assessment nursing therapeutics course requisite subject access condition student explore range issue relate promotion health nursing care people mental health care.

This step can reduce the computation complexity and enhance the accuracy of next implementation.

TFIDF

After NLP pre-processes the data, we will use TF-IDF (term frequency anti-document frequency) technology to mine common keywords. TF-IDF is a common weighted technique used for information retrieval and data mining (Salton &Buckley, 1988). This technique will use a statistical method to assess the importance of a word to one of the documents in a collection or corpus (Figure6).

1. Calculate TF = number of occurrences / total number of words
2. Calculate IDF = log (total number of corpus files / number of files containing words + 1)

TFIDF

For a term i in document j:

$$w_{i,j} = tf_{i,j} \times \log\left(\frac{N}{df_i}\right)$$

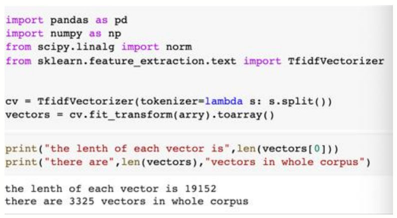
tf_{ij} = number of occurrences of i in j
df_i = number of documents containing i
N = total number of documents

Figure 6 TFIDF technique

In addition, after mining the common keywords by this technology, our group will also use these keywords to discover the correlation between various documents. For example, word data does not appear frequently in the entire database, but often appears in some documents. Through the TF-IDF technology, we can filter out important meaning in each sentence, and justify how similar is between any two documents. (Oren,2002)

e.g.:“Information technology” is common in IT faculty but not in other faculty, so “information technology” has higher TFIDF in IT related course contents.

To implement TFIDF, TfidfVectorizer in Sklearn is used.
First step is to tokenize and split the corpus into “tokens” and then transfer them into vectors with TFIDF features (Figure7).



```
import pandas as pd
import numpy as np
from scipy.linalg import norm
from sklearn.feature_extraction.text import TfidfVectorizer


cv = TfidfVectorizer(tokenizer=lambda s: s.split())
vectors = cv.fit_transform(arry).toarray()

print("the lenth of each vector is",len(vectors[0]))
print("there are",len(vectors),"vectors in whole corpus")

the lenth of each vector is 19152
there are 3325 vectors in whole corpus
```

Figure 7 python code of TFIDF

Cosine similarity

To measure the how similar among those documents, cosine similarity is used as a special metric which can determine the similarity of vectors in regardless of the vector size. In multi-dimensional space, the distance between each two vectors can be far apart by Euclidean distance and the smaller angle between two vectors represents the higher similarity(Figure8).

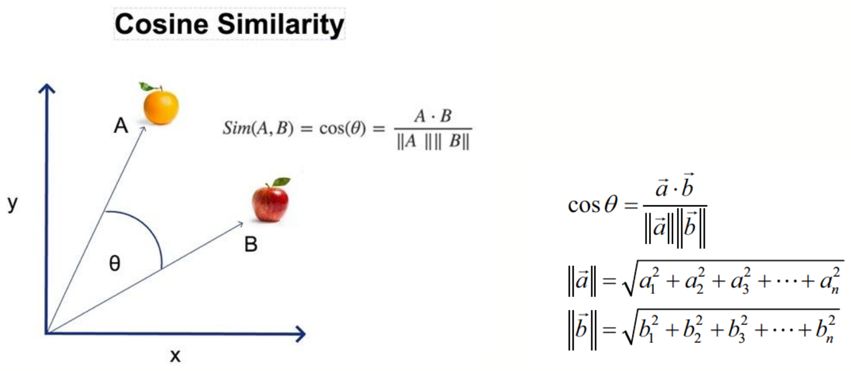
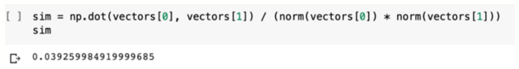

Cosine Similarity
A
y
θ
B
x
$Sim(A,B) = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|}$
$\cos\theta = \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\|\|\vec{b}\|}$
$\|\vec{a}\| = \sqrt{a_1^2 + a_2^2 + a_3^2 + \cdots + a_n^2}$
$\|\vec{b}\| = \sqrt{b_1^2 + b_2^2 + b_3^2 + \cdots + b_n^2}$

Figure 8 cosine similarity

To implement it in our project, we calculated the similarity by such code below.

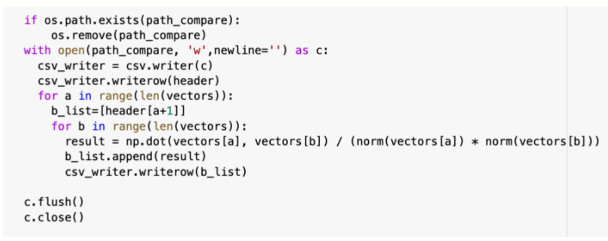


```
[ ]  sim = np.dot(vectors[0], vectors[1]) / (norm(vectors[0]) * norm(vectors[1]))
     sim

     0.039259984919999685
```

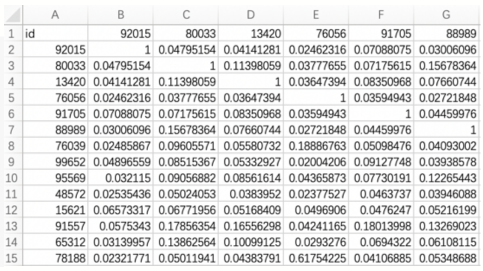
Figure 9 python code of cosine similarity

After 3325*3325 times calculation, we created a csv with all similarity(Figure10).

According to manual check and comparison, based on our understanding to original course content, we defined the similarity>0.4 as threshold.
If the similarity is more than that, the two courses are similar at our standard.



```
if os.path.exists(path_compare):
    os.remove(path_compare)
with open(path_compare, 'w',newline='') as c:
  csv_writer = csv.writer(c)
  csv_writer.writerow(header)
  for a in range(len(vectors)):
    b_list=[header[a+1]]
    for b in range(len(vectors)):
      result = np.dot(vectors[a], vectors[b]) / (norm(vectors[a]) * norm(vectors[b]))
      b_list.append(result)
      csv_writer.writerow(b_list)

c.flush()
c.close()
```



	A	B	C	D	E	F	G
1	id	92015	80033	13420	76056	91705	88989
2	92015	1	0.04795154	0.04141281	0.02462316	0.07088075	0.03006096
3	80033	0.04795154	1	0.11398059	0.03777655	0.07175615	0.15678364
4	13420	0.04141281	0.11398059	1	0.03647394	0.08350968	0.07660744
5	76056	0.02462316	0.03777655	0.03647394	1	0.03594943	0.02721848
6	91705	0.07088075	0.07175615	0.08350968	0.03594943	1	0.04459976
7	88989	0.03006096	0.15678364	0.07660744	0.02721848	0.04459976	1
8	76039	0.02485867	0.09605571	0.05580732	0.18886763	0.05098476	0.04093002
9	99652	0.04896559	0.08515367	0.05332927	0.02004206	0.09127748	0.03938578
10	95569	0.032115	0.09056882	0.08561614	0.04365873	0.07730191	0.12265443
11	48572	0.02535436	0.05024053	0.0383952	0.02377527	0.0463737	0.03946088
12	15621	0.06573317	0.06771956	0.05168409	0.0496906	0.0476247	0.05216199
13	91557	0.0575343	0.17856354	0.16556298	0.04241165	0.18013998	0.13269023
14	65312	0.03139957	0.13862564	0.10099125	0.0293276	0.0694322	0.06108115
15	78188	0.02321771	0.05011941	0.04383791	0.61754225	0.04106885	0.05348688

Figure 10 document similarity csv

Dimensionality Reduction Technique T-SNE

The output vectors of TFIDF has too high dimension that is exclusive, so Dimensionality Reduction technique is required to improve that for further visualization steps (Laurens van der M & Geoffrey E 2018).

T-Distributed Stochastic Neighbor Embedding(T-SNE) is a typical non-linear dimensionality reduction technique used for converting high dimensional dataset to lower dimensional format with minimizing structural data loss, and the output can be 2D (cartesian coordinates) which is suitable for visualization purpose.

The T-sne starts at computing the high dimensional Euclidean distance between two given point to get the pairwise similarity, that is the probability

$p_{j|i}$ if x_j can be the neighbor of x_i" (Josh S, 2017).

"x_i " is regarded as a Gaussian center with variance σi.
"x_k "is the point needed to compare with "x_i "

$$p_{j|i} = \frac{\exp(-\left\|x_i - x_j\right\|^2 / 2\sigma_i^2)}{\sum_{k \neq i} \exp(-\left\|x_i - x_k\right\|^2 / 2\sigma_i^2)}$$

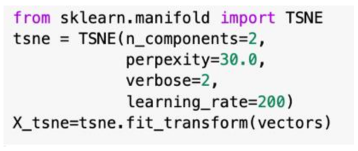
As for the low-dimension, the point "y_j", "y_i", "y_k" are referred to the "x_j"," x_i"," x_k".

$$\boldsymbol{q_{ij}} = \frac{(\mathbf{1} + \|\mathbf{y_i} - \mathbf{y_j}\|^\mathbf{2})^{-\mathbf{1}}}{\sum_{\boldsymbol{k \neq i}} (\mathbf{1} + \|\mathbf{y_i} - \mathbf{y_k}\|^\mathbf{2})^{-\mathbf{1}}}$$

Parameter	definition
$p_{j\|i}$	The high dimensional Euclidean distance/ probability of "x_j" is the neigbour of "x_i"
x_i , x_j , x_k	Given point j,i,k in high dimension
σi	Gaussian kernel,the density of the data
$q_{j\|i}$	The distance in cartesian coordinates(low-dimension)/ probability of "y_j" is the neigbour of "y_i"
y_i , y_j , y_k	Given point j,i,k in low dimension

Table 3 parameter and definition of T-SNE formula

In our project, we got a series of vectors with the length up to 20950 which is impossible for human to observe. So TSNE from sklearn is used to reduce the vectors (Krijthe, J.H., 2015). The code is shown below(Figure11).



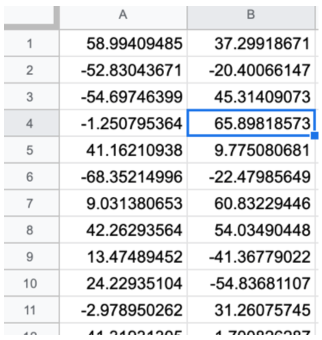
```
from sklearn.manifold import TSNE
tsne = TSNE(n_components=2,
            perpexity=30.0,
            verbose=2,
            learning_rate=200)
X_tsne=tsne.fit_transform(vectors)
```

Figure 11 python code of T-SNE

Parameter	definition	Value
n_components	The targeted dimension	2
perplexity	The number of nearest neigbors to the given points	30
Learning_rate	To define the points clustering method	200
verbose	Mention the number of epochs	2

Table 4 parameter and definition of T-SNE

Then we wrote the 2D result into csv (Figure12) and visualized it with "seaborn" toolkit(Figure13).



	A	B
1	58.99409485	37.29918671
2	-52.83043671	-20.40066147
3	-54.69746399	45.31409073
4	-1.250795364	65.89818573
5	41.16210938	9.775080681
6	-68.35214996	-22.47985649
7	9.031380653	60.83229446
8	42.26293564	54.03490448
9	13.47489452	-41.36779022
10	24.22935104	-54.83681107
11	-2.978950262	31.26075745

Figure 12 python code of 2D result

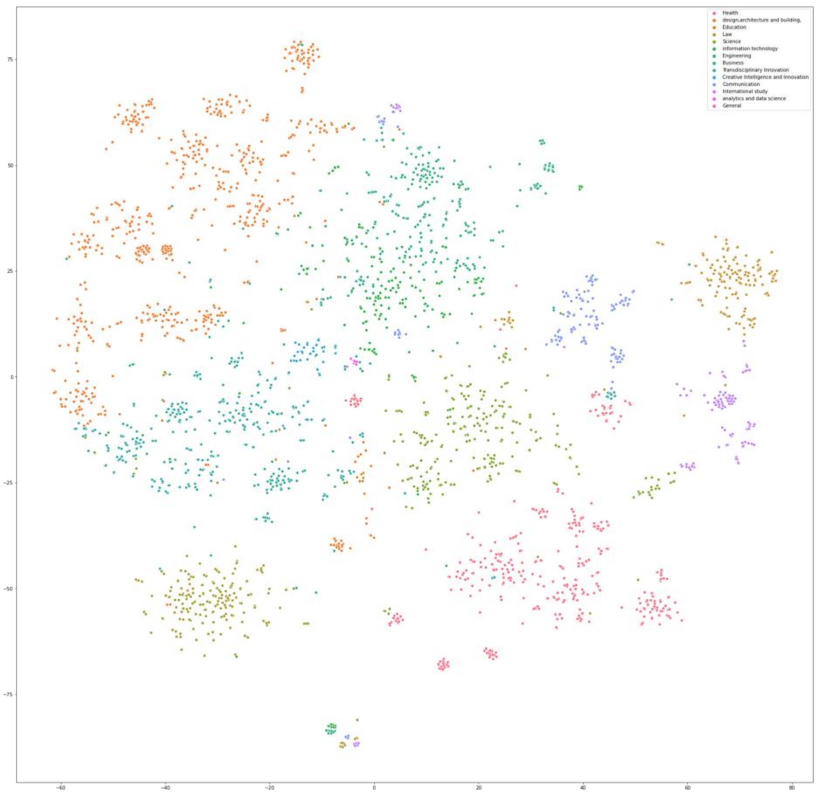
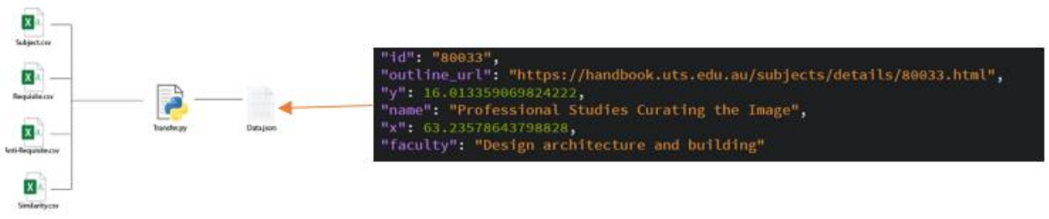


Figure 13 seaborn result

After all relative data elicited and processed as a csv file, we use python to collect the data and put as a node into JavaScript Object Notation (JSON) (Figure14).. Comparing with traditional database such as MySQL, JSON do no need to send to request server to get the result. It can be use in Asynchronous JavaScript and XML(AJAX) to get the data asynchronous from server, which will improve the effective of system especially when the user need retrieve data frequently. The JSON file will contain all information and relationship of subject. Last but not least, we use D3 to architect our interactive system.



```
"id": "80033",
"outline_url": "https://handbook.uts.edu.au/subjects/details/80033.html",
"y": 16.013359069824222,
"name": "Professional Studies Curating the Image",
"x": 63.23578643798828,
"faculty": "Design architecture and building"
```

Figure 14 use json as database

Testing against requirement

Compared to the previous part, we have satisfied each requirement:

1, crawl to get raw data with ID,name,content, requisite etc from UTS website.
2, the user interface is multi-function,including required links and color to represent system requirement.
3, the general result is acceptable, and the lag of manual operation has been minimized.
4, Due to the corpus used in the project, the meaning expression is straightforward and explicit. For this reason, TF-IDF technology is relatively suitable to classify the course outlines, and the classification result is reliable.

Prototype

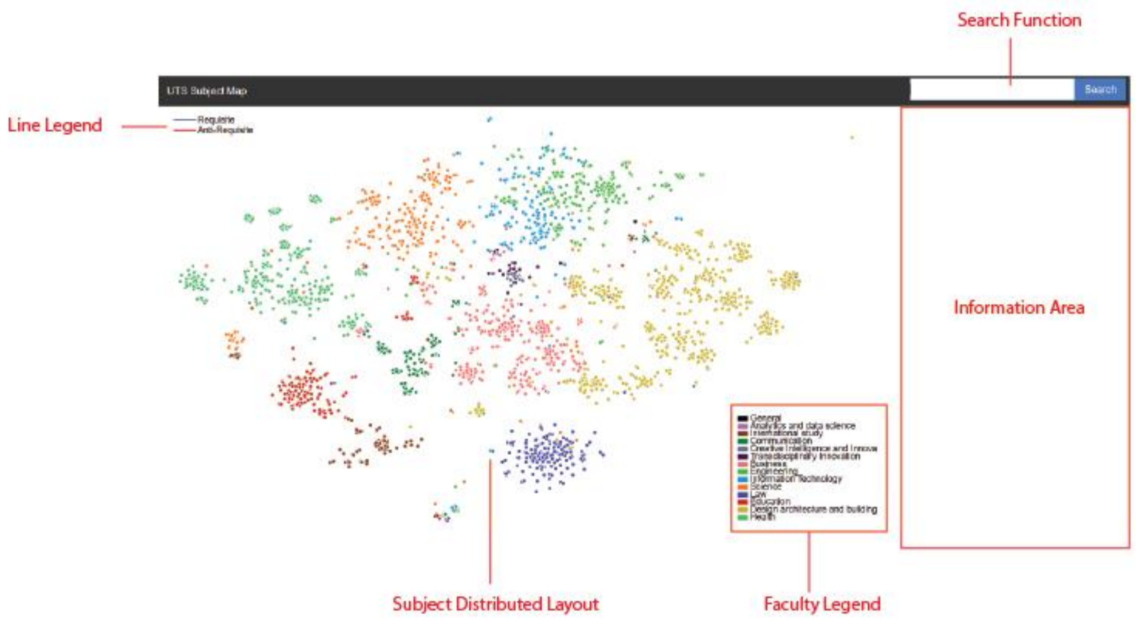


Figure 16 prototype main layout

The Figure demonstrate the general layout of the interactive system (Figure16). In the system, the main part is subject distributed layout with line legend and faculty legend. The different faculty can be easily distinguished by their own presented color. In the top right, it is a search bar. The user can input the key word to find certain subject in the system. In the right side, it is an information area which will show the detail information about selected subject. The information includes the identification code, subject name, origin document URL, the name of requisite, anti-requite and the high similarity subject. In order to view the relationship of different subject clearly, the user can use scroller to control zoom-in and zoom-out. Due to the high similarity subject is close, the user can drag the certain subject to separate them from subject cluster to get a better observation of relationship.

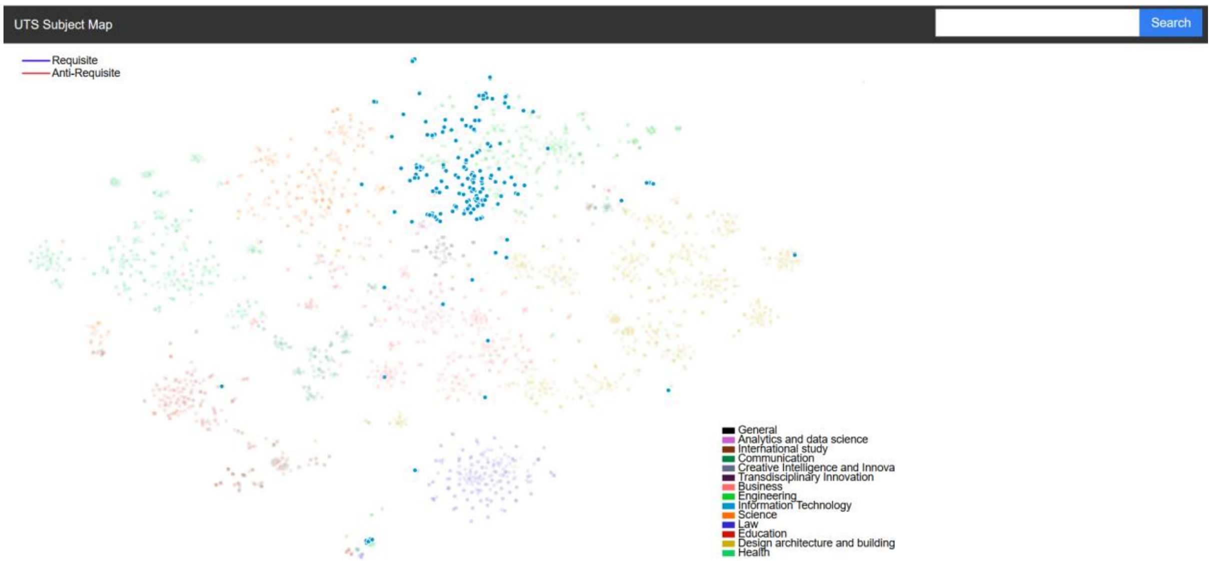
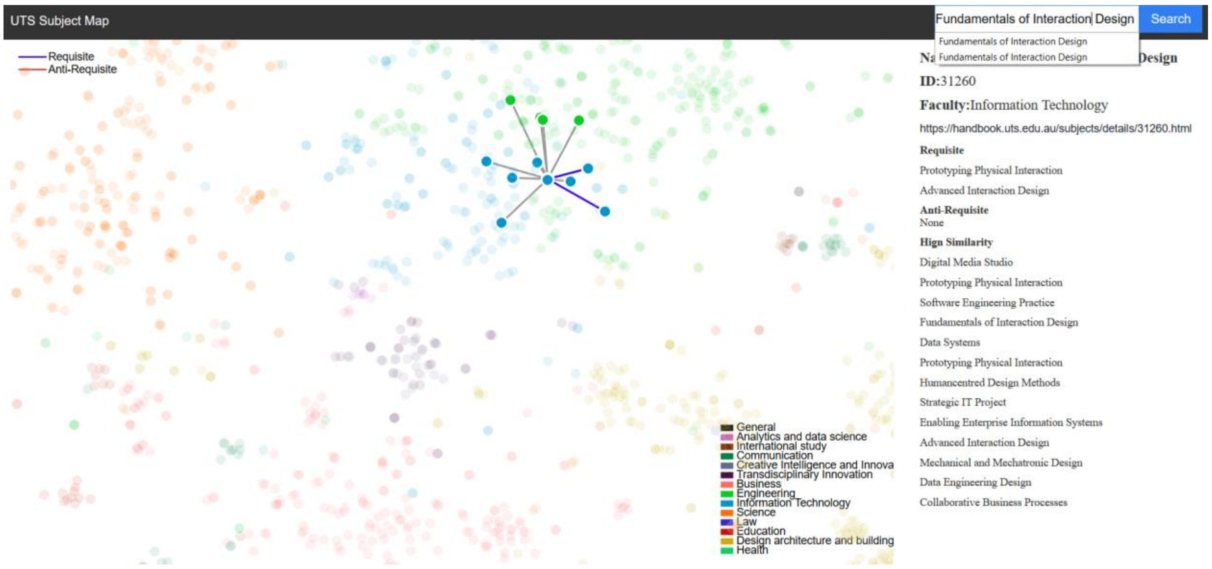


Figure 17. Certain Faculty Distributed

The user can click the faculty legend in the right bottom to get the distributed information of certain faculty (Figure17). The result will be showed as Figure. So that the user can get a big picture about the faculty and then compare the similarity between different faculties.



Fundamentals of Interaction Design

Fundamentals of Interaction Design

Fundamentals of Interaction Design

Na **Design**

ID:31260

Faculty: Information Technology

https://handbook.uts.edu.au/subjects/details/31260.html

Requisite

Prototyping Physical Interaction

Advanced Interaction Design

Anti-Requisite

None

Hign Similarity

Digital Media Studio

Prototyping Physical Interaction

Software Engineering Practice

Fundamentals of Interaction Design

Data Systems

Prototyping Physical Interaction

Humancentred Design Methods

Strategic IT Project

Enabling Enterprise Information Systems

Advanced Interaction Design

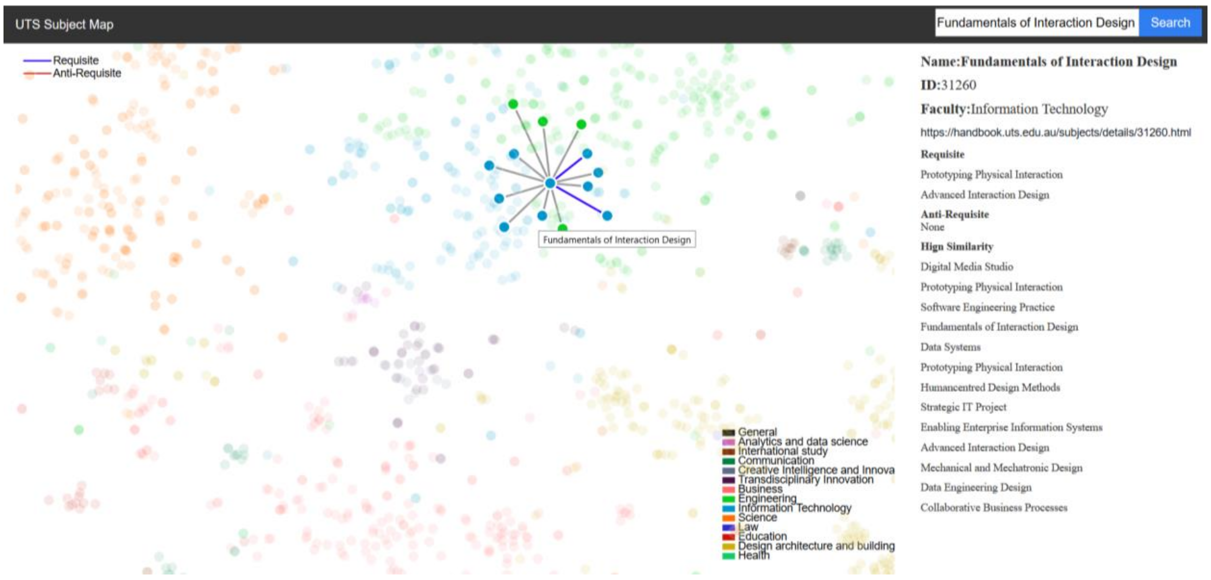
Mechanical and Mechatronic Design

Data Engineering Design

Collaborative Business Processes

Figure 18.Search subject function

When user want to find a subject, they can use the search fuction locating on the top right to find it. According to the input, the search function will show the match result as a list (Figure18). The user do not need to know and input whole subject name. They can choose from the drop-down list and then click search button. And then, as figure show, the relationship of searched subject will display in the main area and the detail information will show in the right side simultaneously.


UTS Subject Map
Fundamentals of Interaction Design
Search
Requisite
Anti-Requisite
Fundamentals of Interaction Design
General
Analytics and data science
International study
Communication
Creative Intelligence and Innova
Transdisciplinary Innovation
Business
Engineering
Information Technology
Science
Law
Education
Design architecture and building
Health

Name:Fundamentals of Interaction Design

ID:31260

Faculty: Information Technology

https://handbook.uts.edu.au/subjects/details/31260.html

Requisite

Prototyping Physical Interaction

Advanced Interaction Design

Anti-Requisite
None

Hign Similarity

Digital Media Studio

Prototyping Physical Interaction

Software Engineering Practice

Fundamentals of Interaction Design

Data Systems

Prototyping Physical Interaction

Humancentred Design Methods

Strategic IT Project

Enabling Enterprise Information Systems

Advanced Interaction Design

Mechanical and Mechatronic Design

Data Engineering Design

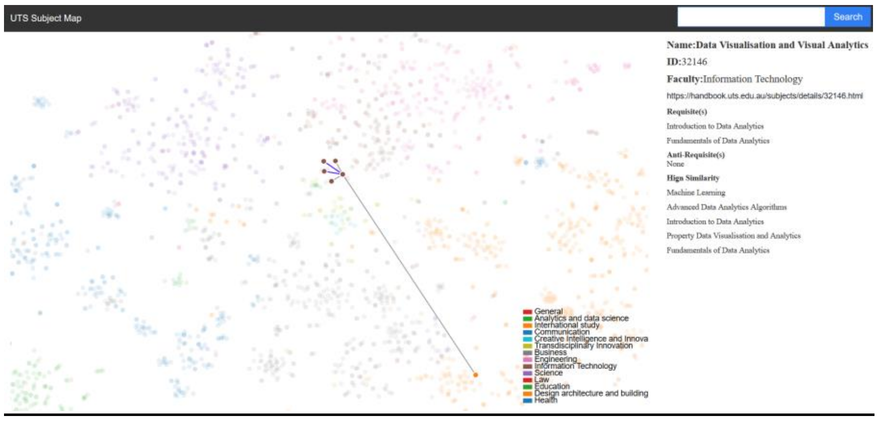
Collaborative Business Processes

Figure 19 Drag and Hover to observe the relationship

The position of each subject is based on the content of subject outline. When two subject outlines have high similarity content, in the system the two subject nodes will overlap. To avoid difficulty of observation, the system allows the users to drag the subject node to the place where they want. In addition, the user can use mouse hover the node to see the name of the subject (Figure19).

Discussion

Finding1
As a whole, the prototype can reflect the distribution of the courses in different faculty, and the distances among them is their relevance. Users can get the detailed information of certain courses by search engine or clicking from the prototype, dragging point to avoid overlap. And the lines link highly similar courses and colored lines is requisite condition so users can compare the existing courses and have intuitive analysis from our prototype.



Name:Data Visualisation and Visual Analytics

ID:32146

Faculty: Information Technology

https://handbook.uts.edu.au/subjects/details/32146.html

Requisite(s)

Introduction to Data Analytics

Fundamentals of Data Analytics

Anti-Requisite(s)
None

Hign Similarity

Machine Learning

Advanced Data Analytics Algorithms

Introduction to Data Analytics

Property Data Visualisation and Analytics

Fundamentals of Data Analytics

Figure 20 Finding1 layout

In our exploration, the points group shown as (Figure 20)has weird link at first sight, The "Data Visualization and Visual Analysis"in IT faculty has high similarity with other IT courses about data, that is reasonable, but a course point located in design faculty has high similarity with "Data Visualization and Visual Analysis", so we did further research and found this course is "Property Data Visualization and Analytics".
From our perspective(Figure 21), to some extent the "Property Data Visualization and Analytics" in Design faculty can be partly replaced by "Data Visualization and Visual Analysis "in IT faculty.

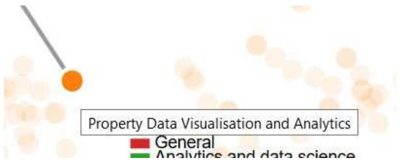

Property Data Visualisation and Analytics
General
Analytics and data science

Figure 21 Finding1 observation

Finding2

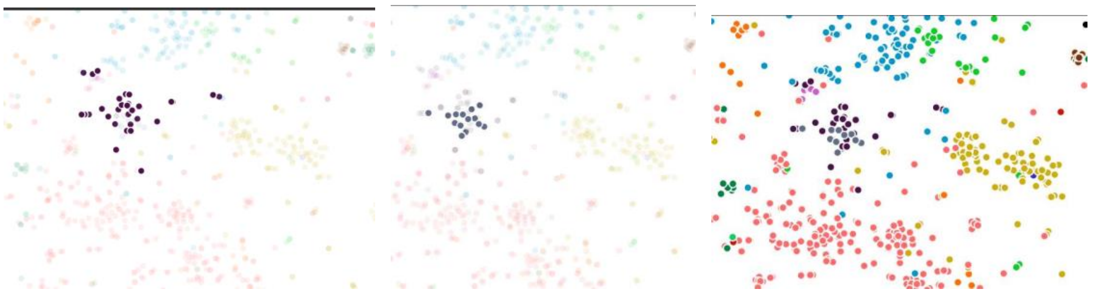
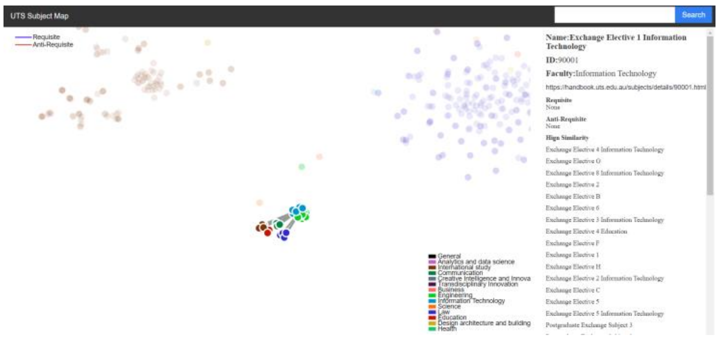


Figure 22 Distribution of Transdisciplinary Innovation&Creative Intelligence and innovation & overlaped points

General
Analytics and data science
International study
Communication
Creative Intelligence and Innova
Transdisciplinary Innovation
Business
Engineering
Information Technology
Science
Law
Education
Design architecture and building
Health

The points of “Transdisciplinary Innovation”and”Creative Intelligence and Innovation” (Figure22) are distributed too close and has many overlaps as shown in Figure_.So we regarded them have high similarity,and the two faculty can combine to avoid resource waste.

Finding 3



Name:Exchange Elective 1 Information Technology

ID:90001

Faculty: Information Technology

https://handbook.uts.edu.au/subjects/details/90001.html

Requisite
None

Anti-Requisite
None

Hign Similarity

Exchange Elective 4 Information Technology
Exchange Elective G
Exchange Elective 8 Information Technology
Exchange Elective 2
Exchange Elective B
Exchange Elective 6
Exchange Elective 3 Information Technology
Exchange Elective 4 Education
Exchange Elective F
Exchange Elective 1
Exchange Elective H
Exchange Elective 2 Information Technology
Exchange Elective C
Exchange Elective 5
Exchange Elective 5 Information Technology
Postgraduate Exchange Subject 3

Figure 23 finding 3

From figure 23, we can notice this cluster including many faculties, but the relative courses have the same name, “exchange elective “Those exchange elective are duplicate and can be compressed into few “exchange elective” to satisfy teaching requirement.

Finding 4

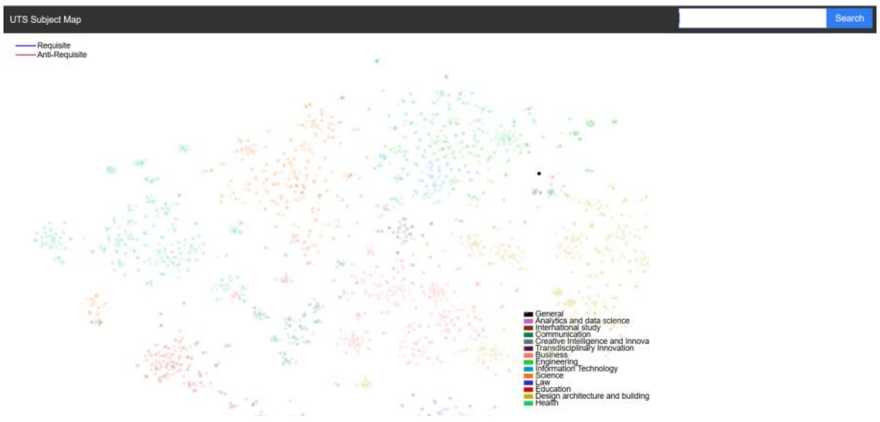


Figure 24 finding 4

We noticed that there is a faculty named “General” but there is only one course in it (Figure 24). We checked the UTS faculty introduction and there are 13 faculty which does not include “General”, so we consider this as a remaining error.

Conclusion

The overall result is satisfied, and the basic mechanism is explicit and straightforward, which can support the prototype well with the formal style documents like course outline. And the clear and informative user interface is benefit and advisable for users to view or search courses, understand their relationship.

According to our findings, there are many duplicate courses which can be replaced or combined with other faculty, so it would avoid university resource waste, and make users understand the relationship among all courses.

Nevertheless, the prototype still remained some drawbacks to be solved.
1, the user interface of the prototype can be more aesthetic and harmonious by improve the original JaveScript and adding more designed element.
2, the position of some points is exclusive and unreasonable, they maybe noise, or the wrongly dimension reduction. So, the data prepossessing and tuning parameter of model in extra experiments to promote the result is an indispensable part.
3, the standard to evaluate the model is based on our or peer's common sense and experience, that is not accurate enough, professional assessment is important in our research project.

Reference

Chowdhury, G. G. (2003). Natural language processing. Annual review of information science and technology, Vol.37,No.1, 51-89.

McEnery, T. (2012). *Corpus linguistics*, Vol. 978019. Oxford University Press Inc.

Laurens van der Maaten& Geoffrey E (2018), "Visualizing Data using t-SNE" *Computing & Informatics*, Vol. 33, No. 3, pp. 652–666, viewed 16 June 2020

Salton, G. & Buckley, C. (1988). Term-weighing approache sin automatic text retrieval. In Information Processing & Management, Vol.24, No.5, pp. 513-523

Oren, Nir. (2002). Reexamining tf.idf based information retrieval with Genetic Programming. In Proceedings of SAICSIT pp. 1-10.

N. Biggs. (1974) Algebraic graph theory. In Cambridge Tracts in Mathematics, Vol 67, pp.1-3. Cambridge University Press.

Josh Starmer (2017), ‘StatQuest: t-SNE, Clearly Explained’, video recording, YouTube, viewed 18 June 2020, <https://www.youtube.com/watch?v=NEaUSP4YerM >

Marine Genomics (2020) t-Distributed Stochastic Neighbor Embedding (t-SNE): A tool for eco-physiological transcriptomic analysis Vol.51

Krijthe, J.H., (2015). Rtsne: t-Distributed Stochastic Neighbor Embedding using a Barnes-Hut implementation, version 0.13. <https://github.com/jkrijth>, viewed 20 June 2020

Acknowledgement

First and foremost，we would like to express our deepest appreciation to our supervisor Dr Jun Li, a responsible professor, who offered us valuable guidance and instruction if we need. Especially when we got confused to visualize the large size vectors, he based on his knowledge and suggested that T-sne algorithm is effective to solve the problem, which turns to be suitable in visualization. Thanks again for Dr Jun Li’ s support, we got tremendous improvement in our research project.