Back To Work Connect

1st Archana Uday Mahajan
MSc. Data Analytics
National College of Ireland
Dublin, Ireland
x20198825@student.ncirl.ie

2nd Pritish Mehta
MSc. data Analytics)
National College of Ireland
Dublin, Ireland
x20184409@student.ncirl.ie

3rd Rutuja Dinesh Mehta
MSc. Data Analytics
National College of Ireland
Dublin, Ireland
x20129751@student.ncirl.ie

4th Savin Vishwas Karkada
MSc. Data Analytics
National College of Ireland
Dublin, Ireland
x20184727@student.ncirl.ie

***Abstract*—The following study primarily deals with processing and clustering of textual data that represents course and job description gathered by an organisation, that is focused towards matching courses and course description with respect to the relevant job titles. The necessary preprocessing steps are carried out to prepare the data to enable efficient clustering techniques to be applied. Further certain analyses are done to retrieve all the relevant courses when queried for job titles. An array of word embedding techniques were considered to vectorize the textual data and was finalised with TF-IDF vectorizer. The numerical vectors are clustered as per the value of k using the elbow method to arrive at an optimal cluster size value. Alternative techniques like Silhouette Method were also considered to understand the optimum cluster size, and the ideal k value was considered as per the requirements of the study and feasibility of the dataset used. The clustered data can be queried with respect to job titles so that relevant courses titles and its description can be attained as an output. Further certain evaluation metrics are applied to quantitatively measure the efficacy of the applied algorithm.**

***Index Terms*—Clustering, Word Embedding, Vectorization, TF-IDF, Doc2Vec, Elbow, Silhouette.**

I. Introduction

As there is a rise in the need for effective skill sets to perform corporate jobs, there exists a gap between the industry relevant skills that match the job description and the skills of working individuals. Technology has been evolving in great tempo, the work that adheres to these enhanced technologies constantly improves and changes. Industries have been on a constant lookout for professionals that stay relevant to industry requirements so that the delivered work can be efficiently managed. However, there are certain working professionals who do not possess these skills due to a variety of reasons such as a brief pause from work, change of job positions, adaptation of advanced tools and technologies in the company etc. This can bring about certain issues in terms of efficiency in the deliverable due to an imbalance between skills and requirements. The following study deals with a similar use case where the organization aims to enable working individuals to obtain industry relevant skills by mapping their previous roles in which the individuals worked, the sector in which they possess their expertise and the current skills sets that they hold. The most vital aspect of the study is to understand the data the organization has collected over the years in regards to the profile of their customers. These textual information has been procured through multiple sources, mainly through the company's website where the customers are required to consensually input the data regarding the profile that they currently hold. They primarily have the position in which the applicants are in, the number of years the applicant has served in different positions and the brief description of the kind of work delivered during the stint at that particular position. Apart from this the resumes of the applicants are also extracted to obtain relevant information regarding the profile. The company has actively partnered with institutions to identify suitable course structures that are highly in demand for the job titles that the applicants provide. These courses are differ and cover a vast area of industry settings including technical and non technical education. The courses offered are majorly from reputed institutes in Ireland and also from abroad. They are delivered both on campus and and online setting as per the need of the applicant. These courses are carefully picked after extensive research in terms of the outcomes of the course to ensure suitable transfer of knowledge that are industry relevant and adheres to the latest technology demands of the related companies. Through this any user who is on a lookout to enhance existing skills or gain new skills that are highly in demand to build the existing portfolio will be matched with courses that are relevant to their job titles. The courses provided by the institutions largely focus on practical understanding of the area of study and also provide up to date course materials as described in the description of the course so that the user can suitably tailor make the profile to pertinently suit the requirements of the jobs. The matched courses not only include the ones that focuses on the core aspects of the subjects but also include courses that provide sufficient skills to make a transition of jobs that are closely related to the current job title of the user. Thus, through the following study, effective solutions that can be adapted to enable efficient matching of courses with job titles are identified, implemented and studied.

II. Related Work

M. Badawy, M. A. Mahmood, A. A. Abd El-Aziz and H. A. Hefny [1] used automatic text mining approach for selecting the academic course based on the course content for reducing the manual efforts for selection of the relevant course. The data was collected and the process of document pre-processing such as transforming the cases, removing the stop words and tokenization were carried out. The set of keywords were formed and the similarity between the sentences were determined. Term Frequency (TF) was used for topic ranking. The said approach could have been applied to the larger dataset for gaining for accurate results.

A domain independent algorithm for text-segmentation was used by Y. Tu, Y. Xiong, W. Chen and C. Brinton [2] to access the education content. The present corpus of documents was trained. Tropical distributions and word embeddings were used for calculating input text features. Similarity between the textual distribution and detect distribution was calculated and the clustering was performed. The method was tested on two datasets and it outperformed the used LDA method by reducing the error rate by 81%. The research says that the segmentation accuracy and quality could have been improved if domain dependent algorithms was used.

Y. Safali, G. Nergız, E. Avaroglu and E. Dogan [3] used deep learning for classification of academic studies in Doc2Vec modelling technique. The title of the course and the summary helped in determining the field of the study. The study was repeated in 9 different categories with the use of repeated neural networks (Rnn's) and LSTM architectures.

A Deep Word-Topic Latent Dirichlet Allocation(DWT-LDA) was used in the research work by S. Limwattana and S. Prom-on [4] for LDA training with word embedding. A word topic assignment also used neural network with application to Collapsed Gibbs Sampling process as an alternative method. The metric topic coherence framework and topic diversity were used for the evaluation, and the comparison was done with the traditional LDA. Since, the method used in this report has the dependency on the original LDA, other methods would have helped in getting more accurate results.

It is essential to capture the intention of the people's need accurately from the abundant sources available and return them the match best of their needs. A similar work was done of Mongolian information retrieval by Siriguleng [5]. All the Mongolian grammatical features were combined and a model was built based on LDA three-tier Bayesian structure for mining the hidden topic and feature word distribution. The user queries were expanded according to the model Word2vec for obtaining similarity between user query keyboards semantically. Then the topic model was further expanded to vocabulary. So, after the calculation of similarity amongst the query and the document topic is done, the highest relevant document is reverted back to the user. Though the proposed model gave satisfying results, it would be advisable to use neural networks for identification of the queries and identifying them for the better semantic association amongst the query words and sets of documents.

The relevant job opportunities available according to the courses undertaken was given by A. Fortino, Q. Zhong, W. C. Huang and R. Lowrance [6]. Term Frequency-Inverse Document Frequency (TD-IDF) was compared with Latent Semantic Indexing (LSI) was gaining job description with respect to course and curricula description. It was concluded that TD-IDF performed better than the LSI. Other methods can also be used for saving the program cost and time.

The people's demand are rising up and they require the related information as quickly as possible. Since, it is essential for quickly classifying the information and apply the relevant filter, feature and non-semantic selection with application to traditional text classification models was used by R. Wang and Y. Shi [7]. Word2Vec failed to give the essential words, so the TF-IDF approach was used here. But, weighted Word2Vec model with TF-IDF model would have given more accurate results.

R. K. Ibrahim, S. R. M. Zeebaree, K. Jacksi, M. A. M. Sadeeq, H. M. Shukur and A. Alkhayyat [8] provided a way of grouping documents based on textual similarity. Text synopses were found out and the unnecessary words were stopped using the NLTK dictionary. TF-IDF was used for building the vectors and the clusters were formed using the K-Means clustering approach. The results yield were satisfying but improvements can be done in the time consumption of all the datasets without pre-processing.

Dimension Reduction technique were used by R. Kumbhar, S. Mhamane, H. Patil, S. Patil and S. Kale [9] for clustering the unstructured text documents. TF-IDF with singular value decomposition (SVD) and TD-IDF with non-negative matrix factorization (NMF) were used as two different dimension reductionality techniques and later the K-Means clustering was applied. The comparison of all the techniques was done with respect to homogeniety score, completeness score and adjusted rand score. The results would have been improved with the application of Non-negative Matrix Factorization and Singular Vector Decomposition.

J. Song, X. Huang, S. Qin and Q. Song [10] studied the classification of imbalance text data. SMOTE algorithm of over-sampling and under-sampling was used here with K-Means for solving the class imbalance problem, within-class and between-class. This approach helped in reducing the noise and also resolved the problem of the shortage of the samples. This method succeeded in improvements of classification problem of the imbalance dataset. Further, it can be better improved for giving best classification of the datasets.

With the increase in organizational competition in the market, there is also the need for educational institutes and candidates to stay updated with their skills. S. Gottipal et al., in their paper [11], have come up with an approach to analyze job listings from Glassdoor that uses NLP to mine trending skills for Data Science and prepare a data science curriculum and also recommend featured skills to the course designers. They have used models like TF-IDF, Lexicon Mapping, and N-Grams to study high-frequency phrases from the dataset of job descriptions on Glassdoor. The results have been put forward through visualizations and classified as technical skills by job roles and locations.

In paper [12], S. Mukherjee et al., have developed a method to determine the SOC code for the US work visa applications that is dependent on the job description and its comparison to the US Bureau of Labor Statistic's definitions. Their system is based on NLP and predictive models to assess the best model w.r.to quality and time taken for the prediction. They have used TF-IDF n-grams and Doc2Vec for the NLP and KNN, GNB, LR, LinearSVC, DT, and RF for machine learning models. They found out that the best-suited model is a trade-off between time and accuracy in the real world. If there are time limitations, then random forest or doc2vec-based SVC-RBF would be better suited, and if not, then TF-IDF n-gram-based SVC-RBF would be preferred. There are many extensions to this work, including students' t-test stats or using a confidence matrix to rank SOC codes.

Due to the online recruitment process, it has become necessary to correctly map jobs to the resume for both the recruiter and the applicant. R. Shaikh et al., in their paper [13], have come up with a solution that makes use of NLP, i.e., POS tagging, tokenization, and lemmatization of the data for an autonomous text classification of the data. For calculating resume scores, they used Phrase Matcher based on data given by the recruiter and provided top skills to them. To categorize the data of the resume, a Word Order Similarity was used. One limitation of this system is that it does not categorize data by year of experience.

The paper [14] combines SVM and K-NN algorithms to find similarities in job titles based on the description and their industry. Its main merit is to solve the complex issue of selecting the right candidate for a job by improving time efficiency and accuracy. This model improved the memory usage and execution time by 98.75%. The limitation of this model is that it only uses the basic info of the applicants, which can be improved to allow additional features, such as specific job skills, in the future.

Because of the manual work, recruitment is a tedious task. To solve this issue, M. Almelu et al., in their paper [15], have proposed a web application to make this process easier and simpler. It ranks the resumes and compares them with the job descriptions using NLP. It also helps in eliminating the bias that is created due to human intervention. This process reduces 85% of human time and reads 30 resumes per minute.

In the paper [16], R. Agarwal et al., proposed a system to recommend a hospital to a patient by looking at parameters like the hospital location, affordability, specialization, and doctor experience. They have used NLP for this purpose along with ML algorithms, based on the reviews based on surveys and posts on online platforms, and a web interface to better facilitate navigation to a patient. For future work, this project could be expanded to more regions; as of now, it is only limited to the hospitals to Mumbai region.

A. Fiallos, in the paper [17], has proposed a method for educators to academic disciplines' curricula, with the use of recommendations of educational materials. This system uses NLP through domain ontologies constructed from digital texts. This method uses the removal of stopwords, stemming, and POS. Also, ML techniques like LDA and HLDA can be used. This work is still in the research phase and can be implemented to get the best results. This will help the student to study specific academic topics during theie learning.

The widespread use of the internet has given rise to getting to know the behavior of buyers and sellers using predictive analysis. It is tough to find new buyers, handle the current ones, and manage inventory. Companies like Flipkart and Amazon use AI and chatbots to predict this behavior. A novel technique to self-serve the seller to assist and target the correct buyer has been proposed in the paper [18], by V. Vivek et al., with the use of NLP and ML, named CIB-PA, which is a call based interface. This has been deployed by integrating text analysis using call recording, Bot integrations, Persona Assistance and IasS. CIB-PA has outperformed several other systems.

It is challenging to derive document representation for short texts using NLP due to the noise and sparsity of the text. One of the solutions is latent topic models. Z. Liu et al., in their paper [19], have proposed a model that includes the use of CME-DMM, a collaborative and embedding framework that from short texts will capture latent topics, which lessens the sparsity of the short texts. They use attention mechanisms to accommodate each other and, in turn, solve the probability of word generation of latent topics. It then iteratively fine-tunes latent topic distribution.

In recent times, the online platform has become a significant source for ordinary people to share their political views on essential matters, of which Twitter is the most used platform. It has become necessary to analyze topics and extract words aptly describe them. In their paper [20], GM Harshvardhan et al. proposed an LDA and multidimensional scaling system to measure inter-topic distances through relevance and saliency. It can be further extended to resolve issues like the context of the topic.

III. Data Mining Methodology

In recent years, the interest in Knowledge Discovery in Database (KDD) has been rising tremendously. The data collection by the organization of their products, customers and other business areas are called as data warehouses. These huge databases are mined further are getting high quality interesting business insights which adds as an advantage for investigating the behaviour of the customers, planning direct marketing strategies, or detecting frauds. The KDD process has the feedback loop, this iterative nature helps in returning to the previous stage from one stage by making some minor adjustments to some of the parameters and decisions. The methodology used here gears the decisions that must be taken and the available options at each stage for accomplishing the given task. The detailed stages of KDD are given below :

- Data Selection and Data Cleaning
- Pre-processing
- Transformation
- Data Mining
- Evaluation of results
- Interpretation of results

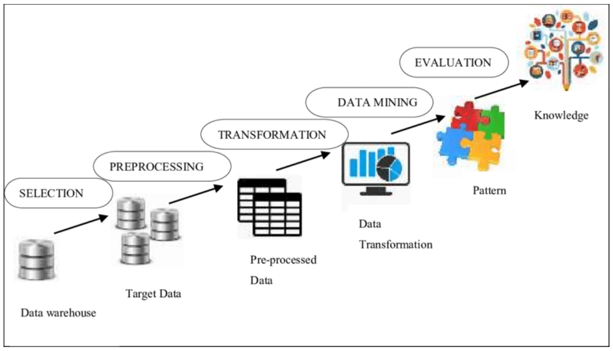


Fig. 1. Knowledge Discovery in Database

Each of the stage explained here, has multiple smaller tasks, which would be repetitive as per the needs. The detailed description to each of the stages used in KDD has been explained below.

In Figure 2 we can see the outline of this project. These include all the steps of the K-DD process model. The first step was to collect data from various files, then clean and merge then into one file called the Final_Dataset.csv and use this for the pre-processing. Stop-word removal, stemming and tokenization were the methods used to pre-process the data and bring them in the required format. The next step was to use NLP to transform the data via TF-IDF and Doc2Vec methods. The vectors created were then fed as input to the K-NN machine learning model to train the dataset, whose output were the clusters formed of the vectors. In the final output, the model took as input the job title and gave the related courses in response. These steps are further explained in the methodology and evaluation in detail.

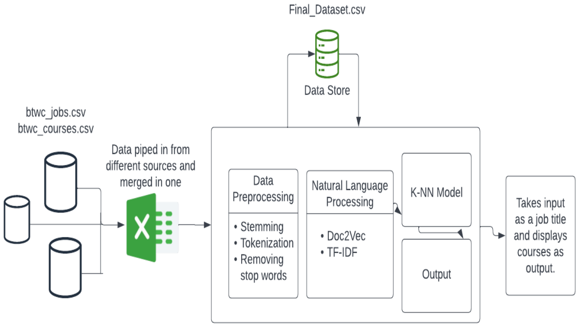


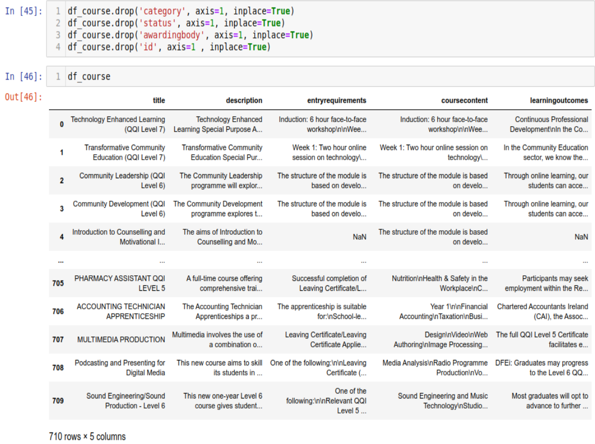
Fig. 2. Project Outline

A. Data Selection and Data Cleaning

The dataset used in this project belongs to the corporation called BackToWorkConnect, which includes two major files btwc_jobs and btwc_course, which consists the data about the job titles, descriptions and the course titles and descriptions, respectively.

Data cleaning is the first step of the K-DD process model, and a very important part which includes the fixing and removal of corrupt, unwanted, incorrect, and/or incomplete data from the dataset. It also includes combining of various data-sources into one file if required.

For this project the first step was to import the files in the jupyter notebook titled Cleaned_Code.ipynb and store them in a pandas data-frame. Initially, the unwanted columns were dropped from the data-frames namely, category, status, awardingbody, and id.



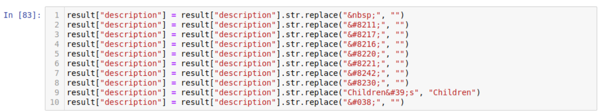
```
In [45]: df_course.drop('category', axis=1, inplace=True)
         df_course.drop('status', axis=1, inplace=True)
         df_course.drop('awardingbody', axis=1, inplace=True)
         df_course.drop('id', axis=1 , inplace=True)

In [46]: df_course
Out[46]:
```

	title	description	entryrequirements	coursecontent	learningoutcomes
0	Technology Enhanced Learning (QQI Level 7)	Technology Enhanced Learning Special Purpose A...	Induction: 6 hour face-to-face workshop\n\nWee...	Induction: 6 hour face-to-face workshop\n\nWee...	Continuous Professional Development\n\nIn the Co...
1	Transformative Community Education (QQI Level 7)	Transformative Community Education Special Pur...	Week 1: Two hour online session on technology\...	Week 1: Two hour online session on technology\...	In the Community Education sector, we know the...
2	Community Leadership (QQI Level 6)	The Community Leadership programme will explor...	The structure of the module is based on develo...	The structure of the module is based on develo...	Through online learning, our students can acce...
3	Community Development (QQI Level 6)	The Community Development programme explores t...	The structure of the module is based on develo...	The structure of the module is based on develo...	Through online learning, our students can acce...
4	Introduction to Counselling and Motivational I...	The aims of Introduction to Counselling and Mo...	NaN	The structure of the module is based on develo...	NaN
...	...	...	...	...	...
705	PHARMACY ASSISTANT QQI LEVEL 5	A full-time course offering comprehensive trai...	Successful completion of Leaving Certificate/L...	Nutrition\nHealth & Safety in the Workplace\nC...	Participants may seek employment within the Re...
706	ACCOUNTING TECHNICIAN APPRENTICESHIP	The Accounting Technician Apprenticeships a pr...	The apprenticeship is suitable for:\nSchool-le...	Year 1\n\nFinancial Accounting\nTaxation\nBusi...	Chartered Accountants Ireland (CAI), the Assoc...
707	MULTIMEDIA PRODUCTION	Multimedia involves the use of a combination o...	Leaving Certificate/Leaving Certificate Applie...	Design\nVideo\nWeb Authoring\nImage Processing...	The full QQI Level 5 Certificate facilitates e...
708	Podcasting and Presenting for Digital Media	This new course aims to skill its students in ...	One of the following:\n\nLeaving Certificate (...	Media Analysis\nRadio Programme Production\nVo...	DFEi: Graduates may progress to the Level 6 QQ...
709	Sound Engineering/Sound Production - Level 6	This new one-year Level 6 course gives student...	One of the following:\n\nRelevant QQI Level 5 ...	Sound Engineering and Music Technology\nStudio...	Most graduates will opt to advance to further ...

710 rows × 5 columns

Fig. 3. Dropping unused columns

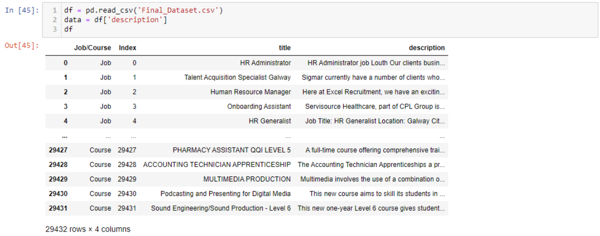
After which the corrupt characters were removed from the job_description like &nsbp, &8211, &8217, Children&#39;s and so on.



```
In [83]: result["description"] = result["description"].str.replace("&nbsp;", "")
         result["description"] = result["description"].str.replace("&#8211;", "")
         result["description"] = result["description"].str.replace("&#8217;", "")
         result["description"] = result["description"].str.replace("&#8216;", "")
         result["description"] = result["description"].str.replace("&#8220;", "")
         result["description"] = result["description"].str.replace("&#8221;", "")
         result["description"] = result["description"].str.replace("&#8242;", "")
         result["description"] = result["description"].str.replace("&#8230;", "")
         result["description"] = result["description"].str.replace("Children&#39;s", "Children")
         result["description"] = result["description"].str.replace("&#038;", "")
```

Fig. 4. Removal of corrupt data

The last step was to merge the job_description and job_course data-frames into a single data-frame called df, which included the job and course title and their descriptions and store it in a file named Final_Dataset.csv.



```
In [45]: df = pd.read_csv('Final_Dataset.csv')
         data = df['description']
         df
```

Out[45]:

	Job/Course	Index	title	description
0	Job	0	HR Administrator	HR Administrator job Louth Our clients busin...
1	Job	1	Talent Acquisition Specialist Galway	Sigmar currently have a number of clients who...
2	Job	2	Human Resource Manager	Here at Excel Recruitment, we have an excitin...
3	Job	3	Onboarding Assistant	Servisource Healthcare, part of CPL Group is...
4	Job	4	HR Generalist	Job Title: HR Generalist Location: Galway Cit...
...	...	...	...	...
29427	Course	29427	PHARMACY ASSISTANT QQI LEVEL 5	A full-time course offering comprehensive trai...
29428	Course	29428	ACCOUNTING TECHNICIAN APPRENTICESHIP	The Accounting Technician Apprenticeships a pr...
29429	Course	29429	MULTIMEDIA PRODUCTION	Multimedia involves the use of a combination o...
29430	Course	29430	Podcasting and Presenting for Digital Media	This new course aims to skill its students in ...
29431	Course	29431	Sound Engineering/Sound Production - Level 6	This new one-year Level 6 course gives student...

29432 rows × 4 columns

Fig. 5. Merging data into df data-frame

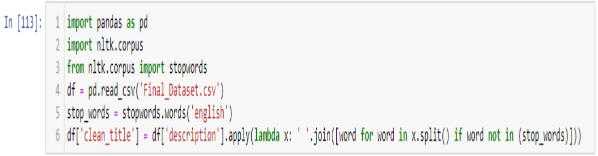
B. Data Pre-processing

The next step to the model, is the pre-processing of data, which includes manipulation of the data to ensure that we get a superior performance and thus makes it a necessary part.

Below given are the methods applied in this project for the purpose of data pre-processing:

1) Stop-Word Removal

This process includes the removal of words that appear commonly across the text like articles, and pronouns. In this project, stopwords from nltk package was used to remove the stopwords from the Final_Dataset.csv file. All the stop-words from the English language were removed.

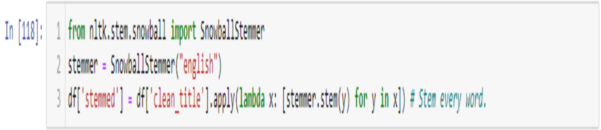


```
In [113]: import pandas as pd
          import nltk.corpus
          from nltk.corpus import stopwords
          df = pd.read_csv('Final_Dataset.csv')
          stop_words = stopwords.words('english')
          df['clean_title'] = df['description'].apply(lambda x: ' '.join([word for word in x.split() if word not in (stop_words)]))
```

Fig. 6. Stop-Word Removal

2) Stemming

Stemming reduces the modification of a word to their root forms, which is caused to communicate gender, tense, aspect, voice, and case of grammar. The purpose is to remove the redundancy in the data that is produced due to these words. SnowballStemmer from nltk package was used to stem the words to their root form in this project.

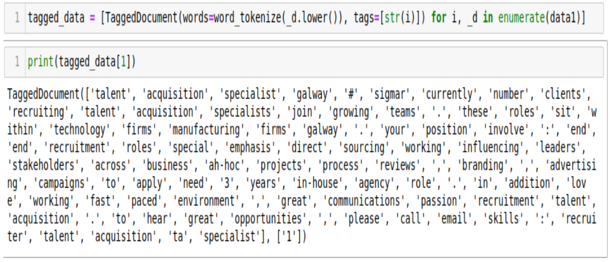


```
In [118]: from nltk.stem.snowball import SnowballStemmer
          stemmer = SnowballStemmer("english")
          df['stemmed'] = df['clean_title'].apply(lambda x: [stemmer.stem(y) for y in x]) # Stem every word.
```

Fig. 7. Stemming

3) Tokenization

The process of tokenization includes chopping up the character sequence into small pieces called tokens and also removing punctuations at the same time. These are later grouped together to form useful semantic. The text was tokenized using word_tokenize method and saved in a tagged_data data-frame. This will assure that the character sequence will match the same sequence while quering.



```
tagged_data = [TaggedDocument(words=word_tokenize(_d.lower()), tags=[str(i)]) for i, _d in enumerate(data1)]
```

```
print(tagged_data[1])
```

```
TaggedDocument(['talent', 'acquisition', 'specialist', 'galway', '#', 'sigmar', 'currently', 'number', 'clients',
'recruiting', 'talent', 'acquisition', 'specialists', 'join', 'growing', 'teams', '.', 'these', 'roles', 'sit', 'w
ithin', 'technology', 'firms', 'manufacturing', 'firms', 'galway', '.', 'your', 'position', 'involve', ':', 'end',
'end', 'recruitment', 'roles', 'special', 'emphasis', 'direct', 'sourcing', 'working', 'influencing', 'leaders',
'stakeholders', 'across', 'business', 'ah-hoc', 'projects', 'process', 'reviews', ',', 'branding', ',', 'advertisi
ng', 'campaigns', 'to', 'apply', 'need', '3', 'years', 'in-house', 'agency', 'role', '.', 'in', 'addition', 'lov
e', 'working', 'fast', 'paced', 'environment', ',', 'great', 'communications', 'passion', 'recruitment', 'talent',
'acquisition', '.', 'to', 'hear', 'great', 'opportunities', ',', 'please', 'call', 'email', 'skills', ':', 'recrui
ter', 'talent', 'acquisition', 'ta', 'specialist'], ['1'])
```

Fig. 8. Tokenization

C. Data Transformation

The discovery of the content, structure, language and word selection for the classification of job and course description could be easily done by the Natural Language Processing (NLP). NLP is very useful in extracting text features. It becomes easy to understand how the courses and jobs are described by an intelligent entity. This would help in building the models for analyzing and generating language for linking up the courses with the relevant job titles.

After the text pre-processing is done, the suitable features are extracted from the chunk of huge dataset as they would decrease the accuracy and the overall classifier's performance. Here, in this project, two linguistic text extraction methods are used as given below :

1) Term Frequency - Inverted Document Frequency (TF-IDF)

The relevancy of a word to a documents in a set of documents is given by TF-IDF. This method multiplies two metrics : the number of times a word appears in the document, and the frequency of the word in a document calculated inversely in a collection of documents.

The Term Frequency is calculated by just looking at the instances the word appears in a document. Inverse Document Frequency is how commonly or rarely a word is in a set of documents. The calculation id done by taking the logarithm of the total number of documents divided by the document number. The closer the value is to 0, more common the word is.

$$tf\,idf\,(t, d, D) = tf\,(t, d)\;.\;idf\,(t, D)$$

Fig. 9. TF-IDF Calculation

where,

$$tf\,(t, d) = \log\,(1 + freq\,(t, d))$$

Fig. 10. TF-IDF Calculation

$$idf\,(t, D) = \log\left(\frac{N}{count\,(d \in D : t \in d)}\right)$$

Fig. 11. TF-IDF Calculation

2) Doc2Vec :

Vector representation of word embedding is done easily by Doc2Vec which helps us for document embeddings. The vector representation of the text paragraphs is extracted from the whole text documents. The neural networks are used for creating vectors of different lengths from documents, paragraphs or sentences. Semantic vectors are obtained using this method. The similar words would be adjacent to each other. Doc2Vec has two training, CBOW and Skip-Gram. CBOW method has the output word vector to the input context vector and vice versa with the Skip-Gram.

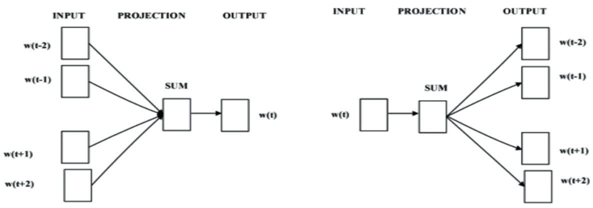


Fig. 12. CBOW model (left) and Skip-Gram(right)

D. Data Mining

The process of data mining is to find anomalies, correlations and/or patterns in a given data-set to predict the outcomes. These predictions can be used to increase sales and revenue, reduce the risks, improve relations between customers and organizations, and many such reasons by using a wide range of techniques.

The technique used to predict the courses from a given job title in this project is done by the K-Nearest Neighbors Algorithm which is explained below in detail:

K-NN Model:

The K-NN Model assumes that similar things are close to each other. Based on this theory it forms clusters of similar things by calculating the distance between two points. To calculate the difference different methods like the Eucledian, Minkowski or Manhattan are used.

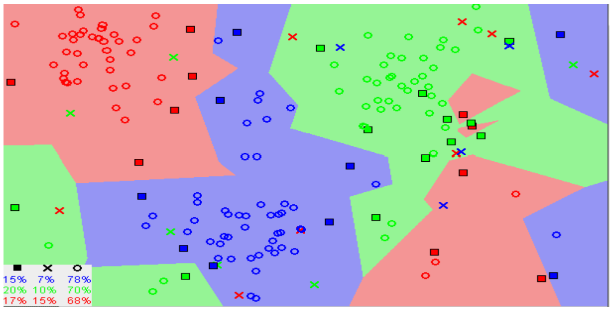


Fig. 13. K-NN Model

The steps to the KNN model are given below:

1) Load Data

2) Initialize the k that fit your data-set

3) Calculate the distance between query and current points and add it to an ordered collection

4) Sort the collection from small to largest

5) Choose first k points and get their labels

6) Return mean for regression and mode for classification

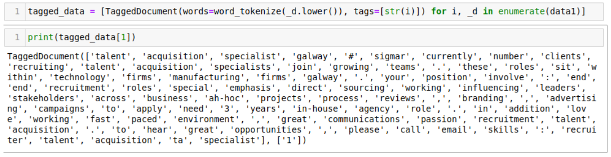
IV. Evaluation

For the purpose of this study we have evaluated two methods for word embeddings. Since we have the dataset which contains the job and course titles and description our first step will be to vectorize the textual data to perform further analysis. Additionally, we have used 2 methods for vecotrizing the textual data which are :-

A. Doc2Vec

Doc2Vec is an algorithm which uses neural networks to create vectors of different lengths from sentences, paragraphs or documents. The advantage of using this approach is that it gives us the vectors that are semantic. In other words the words which have a similar meaning will be adjacent to each other in the vector space.

The evaluate the performance of the doc2vec model we have first tag the data using the inbuilt function within gensim library which is used to process unstructured data generally text. We used the TaggedDocument which creates a tag along with the sentence, paragraph or a document. Figure 14 illustrates that taggeddocument that was created for the study.



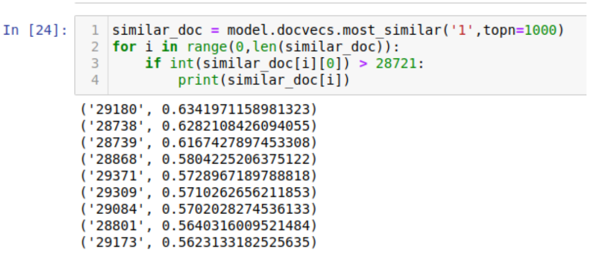
```
tagged_data = [TaggedDocument(words=word_tokenize(_d.lower()), tags=[str(i)]) for i, _d in enumerate(data1)]
```

```
print(tagged_data[1])
```

```
TaggedDocument(['talent', 'acquisition', 'specialist', 'galway', '#', 'sigmar', 'currently', 'number', 'clients',
'recruiting', 'talent', 'acquisition', 'specialists', 'join', 'growing', 'teams', '.', 'these', 'roles', 'sit', 'w
ithin', 'technology', 'firms', 'manufacturing', 'firms', 'galway', '.', 'your', 'position', 'involve', ':', 'end',
'end', 'recruitment', 'roles', 'special', 'emphasis', 'direct', 'sourcing', 'working', 'influencing', 'leaders',
'stakeholders', 'across', 'business', 'ah-hoc', 'projects', 'process', 'reviews', ',', 'branding', ',', 'advertisi
ng', 'campaigns', 'to', 'apply', 'need', '3', 'years', 'in-house', 'agency', 'role', '.', 'in', 'addition', 'lov
e', 'working', 'fast', 'paced', 'environment', ',', 'great', 'communications', 'passion', 'recruitment', 'talent',
'acquisition', '.', 'to', 'hear', 'great', 'opportunities', ',', 'please', 'call', 'email', 'skills', ':', 'recrui
ter', 'talent', 'acquisition', 'ta', 'specialist'], ['1'])
```

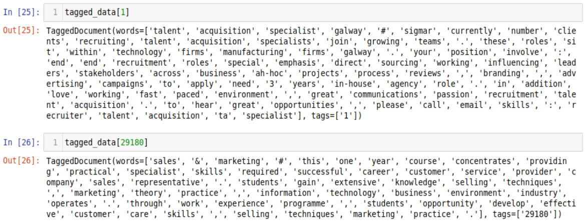
Fig. 14. Elbow Method

Further, we trained the Doc2Vec model with the obtained TaggedDocument data with size of the vectors being 15, min count being 10 and the number of epochs being 100. Where the vector size determines the dimensionality of the feature vectors, min count ignores all words with total frequency lower than this and finally the number of iterations the model is being trained for is set by the value passed in the epochs. After running the model we use the method within docvecs called most_similar which gives us the document/paragraph that is most similar to the paragraph/argument passed while calling this method. We tried to pass the argument of the first tagged document to check if it is able to accurately display courses for the particular job description. The output of which is shown in Figures 15 and 17



```
In [24]:  similar_doc = model.docvecs.most_similar('1',topn=1000)
          for i in range(0,len(similar_doc)):
              if int(similar_doc[i][0]) > 28721:
                  print(similar_doc[i])

('29180', 0.6341971158981323)
('28738', 0.6282108426094055)
('28739', 0.6167427897453308)
('28868', 0.5804225206375122)
('29371', 0.5728967189788818)
('29309', 0.5710262656211853)
('29084', 0.5702028274536133)
('28801', 0.5640316009521484)
('29173', 0.5623133182525635)
```

Fig. 15. Elbow Method



```
In [25]:  tagged_data[1]

Out[25]:  TaggedDocument(words=['talent', 'acquisition', 'specialist', 'galway', '#', 'sigmar', 'currently', 'number', 'clie
          nts', 'recruiting', 'talent', 'acquisition', 'specialists', 'join', 'growing', 'teams', '.', 'these', 'roles', 'si
          t', 'within', 'technology', 'firms', 'manufacturing', 'firms', 'galway', '.', 'your', 'position', 'involve', ':',
          'end', 'end', 'recruitment', 'roles', 'special', 'emphasis', 'direct', 'sourcing', 'working', 'influencing', 'lead
          ers', 'stakeholders', 'across', 'business', 'ah-hoc', 'projects', 'process', 'reviews', ',', 'branding', ',', 'adv
          ertising', 'campaigns', 'to', 'apply', 'need', '3', 'years', 'in-house', 'agency', 'role', '.', 'in', 'addition',
          'love', 'working', 'fast', 'paced', 'environment', ',', 'great', 'communications', 'passion', 'recruitment', 'tale
          nt', 'acquisition', '.', 'to', 'hear', 'great', 'opportunities', ',', 'please', 'call', 'email', 'skills', ':', 'r
          ecruiter', 'talent', 'acquisition', 'ta', 'specialist'], tags=['1'])

In [26]:  tagged_data[29180]

Out[26]:  TaggedDocument(words=['sales', '&', 'marketing', '#', 'this', 'one', 'year', 'course', 'concentrates', 'providin
          g', 'practical', 'specialist', 'skills', 'required', 'successful', 'career', 'customer', 'service', 'provider', 'c
          ompany', 'sales', 'representative', '.', 'students', 'gain', 'extensive', 'knowledge', 'selling', 'techniques',
          ',', 'marketing', 'theory', 'practice', ',', 'information', 'technology', 'business', 'environment', 'industry',
          'operates', '.', 'through', 'work', 'experience', 'programme', ',', 'students', 'opportunity', 'develop', 'effecti
          ve', 'customer', 'care', 'skills', ',', 'selling', 'techniques', 'marketing', 'practice', '.'], tags=['29180'])
```

Fig. 16. Elbow Method

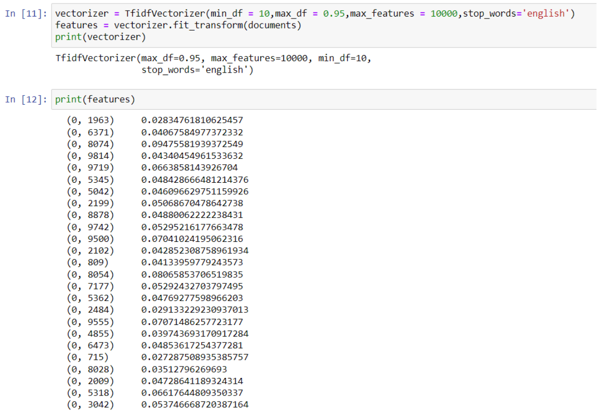
From the figure we can see that the model was not able to semantically link the paragraphs together. Additionally, we tried various iteration using different values of vector_size, min_count and epochs. However the results for multiple iterations gave a similar output where both the documents did not have any similarities. Hence, we tried another approach to perform vectorization which is explained in the next section.

B. *TF-IDF*

TF-IDF an abbreviation for Term Frequency - Inverse Document Frequency is one of the widely used word embedding methods to transform textual data into numerical entities. TF-IDF is largely taken into consideration to vectorize text data when the corpus present do not require extensive semantic relationship to be established between the documents but enable the textual data to arrive at position where retrieval of queries become easier. In the current study the idea is obtain relevant course titles and its description when the relating job titles are provided as inputs. The outputs provided are the weighted values of the particular token with respect to the number of times the word has appeared in the documents, proportionally enhancing its weighted factor.

As the name suggests, working of TF-IDF can be bifurcated into two parts. Firstly Term Frequency means taking into account the frequency in which the term has been used in the document. This results in providing a higher weighted average for the token making it more significant than the closely related tokens. The terms that do not posses weights that are not high enough, represents the term being farther away from the queried term in terms of similarity relationship. At the same time there can appear an ambiguity in establishing the relationship as the most commonly used definite and indefinite articles might attain the weighted strength as they can be found being used innumerable times. This incorrect allocation of weights will provide no semantic relationship making the required keywords left unattended. The counter this issue the latter part Inverse Document Frequency factor comes into picture. This factor diminishes the improperly emphasised terms such as the indefinite and definite articles to allot lesser weighted value while significantly enhancing the weighted value of the keywords that are critical to establish a relationship between the tokens. In the current study the data used have job titles and course titles along with their descriptions that briefly address the type of job and course. As there is sufficient information to enable quality relationship between the keywords a straightforward vectorizer will suffice to embed the tokens into a numerical array. Thus TF-IDF was opted to carry out this operation. Certain parameters are tweaked in the TF-IDF vectorizer to adhere to the requirements of the data.

The parameters such as min df and max df are exclusively given as inputs to make the vectorization more robust. The min df represents that the vocabulary constructed during the word embedding process will take no notice of the values that are below the input limit. This is to allow only the terms that have higher weighted values into the vocabulary so that all the terms present in the bag of words contribute in establishing the relationship between the keywords. Similarly max df eliminates the terms that have higher weighted values than the input threshold value so that the built vocabulary do not include higher weighted values that can disrupt the semantic stability by improper allocation of weights to the words. The values for min df and max df were given as 10 and 0.95 respectively. The parameter, max features allows to set a size for the vocabulary through which the keywords having specified weightage are passed into as per the allocation of threshold values. The size value was set to 1000. The resulting numerical output after applying TF-IDF is as follows.



```
In [11]: vectorizer = TfidfVectorizer(min_df = 10,max_df = 0.95,max_features = 10000,stop_words='english')
         features = vectorizer.fit_transform(documents)
         print(vectorizer)

         TfidfVectorizer(max_df=0.95, max_features=10000, min_df=10,
                         stop_words='english')

In [12]: print(features)

           (0, 1963)    0.02834761810625457
           (0, 6371)    0.04067584977372332
           (0, 8074)    0.09475581939372549
           (0, 9814)    0.04340454961533632
           (0, 9719)    0.0663858143926704
           (0, 5345)    0.048428666481214376
           (0, 5042)    0.046096629751159926
           (0, 2199)    0.05068670478642738
           (0, 8878)    0.04880062222238431
           (0, 9742)    0.05295216177663478
           (0, 9500)    0.07041024195062316
           (0, 2102)    0.042852308758961934
           (0, 809)     0.04133959779243573
           (0, 8054)    0.08065853706519835
           (0, 7177)    0.05292432703797495
           (0, 5362)    0.04769277598966203
           (0, 2484)    0.029133229230937013
           (0, 9555)    0.07071486257723177
           (0, 4855)    0.039743693170917284
           (0, 6473)    0.04853617254377281
           (0, 715)     0.027287508935385757
           (0, 8028)    0.03512796269693
           (0, 2009)    0.04728641189324314
           (0, 5318)    0.06617644809350337
           (0, 3042)    0.053746668720387164
```

Fig. 17. TF-IDF output for the data

Additionally, to evaluate the K-Means clustering technique we require a number of clusters as parameters. To get the optimum number of clusters is very important as if the cluster is too high then each point in the cluster space then the distortion will decrease but on the other hand if the number of clusters are too small then the points can be clustered in a wrong way. Moreover to find the optimal number of cluster we will be using few techniques.

1) Elbow Method: After performing the vectorization using ”TfidfVectorizer” we will then use the features obtained from the vectorization to calculate the number of optimum clusters within the vector space. Figure 18 represents the graphical representation of the number of clusters and the SSE (sum of square error).

From the figure we can see that the number of optimum clusters is 7 which is shown by the dotted black line. Where the error is flattening by a small margin. Hence we first took k=7 clusters. To Evaluate further we have performed Silhouette Method as well to compare the results with the Elbow Method.

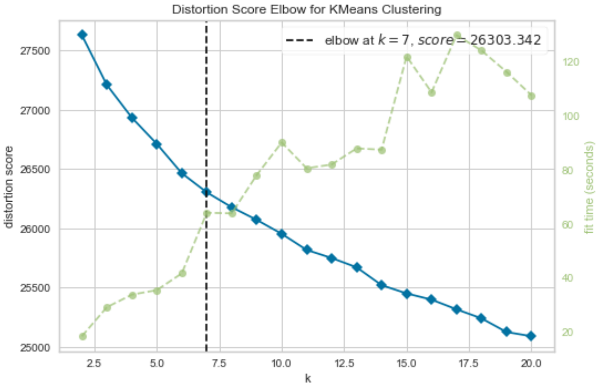


Fig. 18. Elbow Method

2) Silhouette Method: To Further evaluate our model we have performed Silhouette Method. However the method did not give satisfactory results. The results shows that the number of optimum clusters should be 2 which is too small and will not be able to recommend courses for a job title accurately.

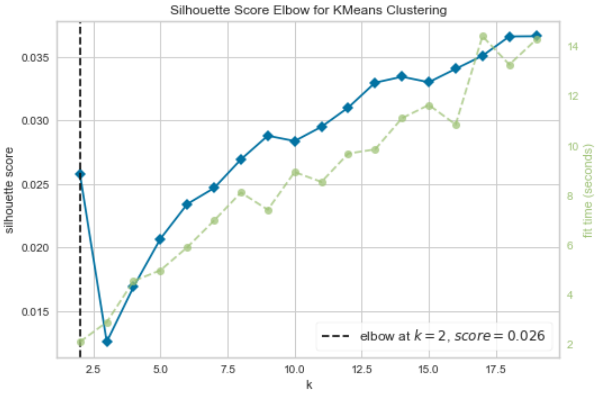


Fig. 19. Silhouette

Additionally, we performed two more methods

3) Calinski-Harabasz Index: Calinski-Harabasz uses a different approach by assuming that the clusters which are compact properly separated are good clusters. It is the ratio between the sum of the dispersion between the clusters and sum of dispersion within the clusters. Additionally, from the graph shown in Fig 20 illustrates that the optimum cluster size should be 3 clusters. Since the clusters will give a generalized out and will not able to grab the relevant courses for the given job title hence this test failed for our study.

4) Davies-Bouldin Index: Lastly, we performed on last test to confirm the optimum number of clusters. It measures the ”goodness of fit” for each given cluster. Hence if the index gives a lower similarity the cluster will be better. Figure 21

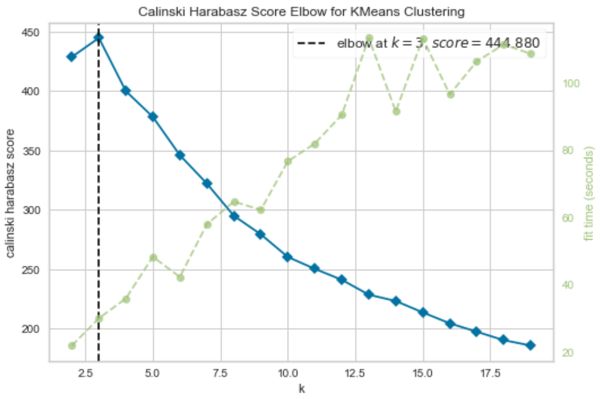


Fig. 20. Calinski-Harabasz Index

depicts that the score is less at around 5 clusters. Hence that will be our optimum number of clusters.

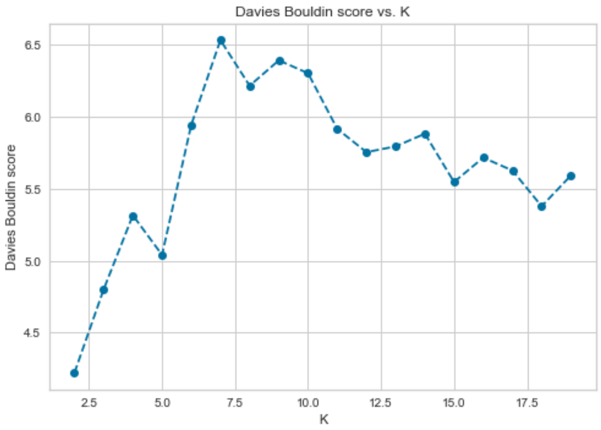


Fig. 21. Davies-Bouldin Index

For the purpose of this study we have proceeded with the output of the Elbow Method as we performed various tests and found out that taking k=7 gave the optimal results.

V. Conclusions and Future Work

The above study that majorly dealt with suitably recommending relevant courses to job titles was carried out after implementing various vectorization and clustering techniques. The data cleaning was implemented using python and excel function to achieve suitable data frame in order to effectively implement clustering techniques. The cleaned data was pre processed to eliminate the stop words that so not contribute in establishing any semantic relationship between the keywords. The data was further was further vectorized to transform the textual data into numerical vectors in order to pass the data into the clustering algorithms. TF-IDF vectorization was identified to be suitable for the dataset considering the requirement of the data and the level of semantic complexity required to establish relationship. Doc2vec was however discarded after careful consideration of the vectors. Clustering the data was found feasible in order to effectively segregate related data into clusters so that the retrieval process becomes easier and more efficient when compared to other recommendation algorithms. The clustering algorithm - K means clustering was chosen after extensive literature survey that suggested the use of this technique to cluster textual data relating to the dataset and similar usecases. Various methods such as elbow method, silhouette method and many more were adapted to identify the optimal K value of clusters. Although these methods gave optimal clusters, the clusters sizes had to be revamped in order to meet the requirements of the dataset and the project. As the data used for the study had certain limitations in terms of the presence of job titles, the cluster size had to be gradually decreased in order to effectively distribute all the data points evenly into all the clusters so that no query is left unanswered. These clusters were printed into csv files to cross verify the consistency of the clusters. An input data is taken and the vector is pointed towards the corresponding cluster. Then the course titles and course description in the particular clusters relating to the query is printed.

The implementation of K means clustering was finalised as per the literature review and the feasibility of the dataset. However, other methods that take into consideration the semantic aspect of the corpus can be integrated as word embedders to evaluate the results. Contextual word embedders such as ELMo have been known to provide higher degree of semantic quality in the output vectors. Thus the adaptation of contextual embedders might increase the the semantic relationships leading to better cluster outputs. Further a recommendation system like engine is also another functional technique that can be employed to yield better results. Methods such as LUCIN using collaborative functions could be an alternative method to approach the problem. Topic modelling however has been one of the primary go to methods to tackle such issues. The use of topic modelling to classify into relevant topics could also provide effective outcomes. Although these methods have been previously used in a variety of usecases, the approach to the problem highly depends on the volume and kind of data available. Thus to suit the requirements of the project the above steps were considered and superior results were gained.

References

[1] M. Badawy, M. A. Mahmood, A. A. Abd El-Aziz and H. A. Hefny, ”A Text Mining Approach for Automatic Selection of Academic Course Topics based on Course Specifications,” 2018 14th International Computer Engineering Conference (ICENCO), 2018, pp. 162-167, doi: 10.1109/ICENCO.2018.8636148.

[2] Y. Tu, Y. Xiong, W. Chen and C. Brinton, ”A Domain-Independent Text Segmentation Method for Educational Course Content,” 2018 IEEE International Conference on Data Mining Workshops (ICDMW), 2018, pp. 320-327, doi: 10.1109/ICDMW.2018.00053.

[3] Y. Safali, G. Nergız, E. Avaroğlu and E. Doğan, ”Deep Learning Based Classification Using Academic Studies in Doc2Vec Model,” 2019 International Artificial Intelligence and Data Processing Symposium (IDAP), 2019, pp. 1-5, doi: 10.1109/IDAP.2019.8875877.

[4] S. Limwattana and S. Prom-on, ”Topic Modeling Enhancement using Word Embeddings,” 2021 18th International Joint Conference on Computer Science and Software Engineering (JCSSE), 2021, pp. 1-5, doi: 10.1109/JCSSE53117.2021.9493816.

[5] Siriguleng, ”Mongolian Information Retrieval Method Based on Word2vec and Topic Model,” 2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), 2019, pp. 1217-1220, doi: 10.1109/IAEAC47372.2019.8997588.

[6] A. Fortino, Q. Zhong, W. C. Huang and R. Lowrance, ”Application of Text Data Mining To STEM Curriculum Selection and Development,” 2019 IEEE Integrated STEM Education Conference (ISEC), 2019, pp. 354-361, doi: 10.1109/ISECon.2019.8882067.

[7] R. Wang and Y. Shi, ”Research on application of article recommendation algorithm based on Word2Vec and Tfidf,” 2022 IEEE International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA), 2022, pp. 454-457, doi: 10.1109/EEBDA53927.2022.9744824.

[8] R. K. Ibrahim, S. R. M. Zeebaree, K. Jacksi, M. A. M. Sadeeq, H. M. Shukur and A. Alkhayyat, ”Clustering Document based Semantic Similarity System using TFIDF and K-Mean,” 2021 International Conference on Advanced Computer Applications (ACA), 2021, pp. 28-33, doi: 10.1109/ACA52198.2021.9626822.

[9] R. Kumbhar, S. Mhamane, H. Patil, S. Patil and S. Kale, ”Text Document Clustering Using K-means Algorithm with Dimension Reduction Techniques,” 2020 5th International Conference on Communication and Electronics Systems (ICCES), 2020, pp. 1222-1228, doi: 10.1109/ICCES48766.2020.9137928.

[10] J. Song, X. Huang, S. Qin and Q. Song, ”A bi-directional sampling based on K-means method for imbalance text classification,” 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS), 2016, pp. 1-5, doi: 10.1109/ICIS.2016.7550920.

[11] S. Gottipati, K. J. Shim and S. Sahoo, ”Glassdoor Job Description Analytics – Analyzing Data Science Professional Roles and Skills,” 2021 IEEE Global Engineering Education Conference (EDUCON), 2021, pp. 1329-1336, doi: 10.1109/EDUCON46332.2021.9453931.

[12] S. Mukherjee, D. Widmark, V. DiMascio and T. Oates, ”Determining Standard Occupational Classification Codes from Job Descriptions in Immigration Petitions,” 2021 International Conference on Data Mining Workshops (ICDMW), 2021, pp. 647-652, doi: 10.1109/ICDMW53433.2021.00085.

[13] R. Shaikh, N. Phulkar, H. Bhute, S. K. Shaikh and P. Bhapkar, ”An Intelligent framework for E-Recruitment System Based on Text Categorization and Semantic Analysis,” 2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA), 2021, pp. 1076-1080, doi: 10.1109/ICIRCA51532.2021.9544102.

[14] E. Mankolli and V. Guliashki, ”A Hybrid Machine Learning Method for Text Analysis to Determine Job Titles Similarity,” 2021 15th International Conference on Advanced Technologies, Systems and Services in Telecommunications (TELSIKS), 2021, pp. 380-385, doi: 10.1109/TELSIKS52058.2021.9606341.

[15] M. Alamelu, D. S. Kumar, R. Sanjana, J. S. Sree, A. S. Devi and D. Kavitha, ”Resume Validation and Filtration using Natural Language Processing,” 2021 10th International Conference on Internet of Everything, Microwave Engineering, Communication and Networks (IEMECON), 2021, pp. 1-5, doi: 10.1109/IEMECON53809.2021.9689075.

[16] R. Agrawal, H. Bajaj, R. Gupta and A. P. I. Sidddavatam, ”Healthcare recommendation system using patients' review,” 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI), 2021, pp. 1758-1762, doi: 10.1109/ICOEI51242.2021.9452960.

[17] A. Fiallos, ”Assisted curricula design based on generation of domain ontologies and the use of NLP techniques,” 2017 IEEE Second Ecuador Technical Chapters Meeting (ETCM), 2017, pp. 1-6, doi: 10.1109/ETCM.2017.8247474.

[18] V. Vivek; T. R. Mahesh; C. Saravanan; K. Vinay Kumar, ”5 A Novel Technique for User Decision Prediction and Assistance Using Machine Learning and NLP: A Model to Transform the E-commerce System,” in Big Data Management in Sensing: Applications in AI and IoT , River Publishers, 2021, pp.61-76.

[19] Z. Liu, T. Qin, K. Chen and Y. Li, ”Collaboratively Modeling and Embedding of Latent Topics for Short Texts,” in IEEE Access, vol. 8, pp. 99141-99153, 2020, doi: 10.1109/ACCESS.2020.2997973.

[20] G. Harshvardhan, M. K. Gourisaria, A. Sahu, S. S. Rautaray and M. Pandey, ”Topic Modelling Twitterati Sentiments using Latent Dirichlet Allocation during Demonetization,” 2021 8th International Conference on Computing for Sustainable Global Development (INDIACom), 2021, pp. 811-815.