National College of Ireland

# Recommendation Framework on English Speaking Podcasts using Textual Information Analysis

MSc Research Project
Data Analytics


Surabhi Singh
Student ID: X21148325

School of Computing
National College of Ireland

Supervisor: Noel Cosgrave

**National College of Ireland**
**Project Submission Sheet**
**School of Computing**

| | |
|---|---|
| **Student Name:** | Surabhi Singh |
| **Student ID:** | X21148325 |
| **Programme:** | Data Analytics |
| **Year:** | 2023 |
| **Module:** | MSc Research Project |
| **Supervisor:** | Noel Cosgrave |
| **Submission Due Date:** | 31/01/2023 |
| **Project Title:** | Recommendation Framework on English Speaking Podcasts using Textual Information Analysis |
| **Word Count:** | 6481 |
| **Page Count:** | 16 |

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

**ALL** internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

| | |
|---|---|
| **Signature:** | Surabhi Singh |
| **Date:** | 31/01/2023 |

## PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

| | |
|---|---|
| Attach a completed copy of this sheet to each project (including multiple copies). | ☐ |
| **Attach a Moodle submission receipt of the online project submission**, to each project (including multiple copies). | ☐ |
| **You must ensure that you retain a HARD COPY of the project**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. | ☐ |

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

| **Office Use Only** | |
|---|---|
| Signature: | |
| Date: | |
| Penalty Applied (if applicable): | |

# Recommendation Framework on English Speaking Podcasts using Textual Information Analysis

Surabhi Singh
X21148325

**Abstract**

Podcasts are known as an audio style of talk show which can be tuned by listeners on demand. They have gained a lot of popularity in the digital era and are now available on prominent music streaming platforms. Most recently, several studies are being conducted to identify a suitable approach to increase the engagement of listeners with the platform through the podcast. Creating a recommendation system using a re-ranking approach based on ratings and reviews of the listener has been one of the noticeable methods. In these systems extracted features like listener's ratings and reviews are passed to a decoder-like long short-term memory (LSTM) model and embedded in multi-directional vector space using the DistMult algorithm. One major challenge faced in this approach is the limitation of the system to only recommend podcasts that are high on rating or have higher reviews. This discourages producers to create content in different genres as they may not be tuned by active listeners. To overcome this challenging research, suggest creating models based on semantic features like podcast description, textual documentation of audio content, etc. Although few models have been created to conduct textual analysis, the area is still in the early stages of development. The research proposed, provides a novel framework to perform textual analysis on a description or podcast and its episodes using word embeddings and cosine similarity. The framework provides a list of the top 9 podcasts for each of the genres based on the user's current choice of the podcast being consumed. The proposed model contains a customized Word2Vec model applied on the corpora derived from podcast descriptions, along with cosine similarity to identify a list of the most significant podcast titles to input data. The model is trained over a secondary dataset consisting of around 4300 podcasts in 19 genres and was collected by performing a query to connect with Spotify's Web API. The author has also used PorterStemmer and WordNeLemmatizer techniques available in Natural Language processing to normalize the description for faster textual analysis. To create a significant framework, the cosine similarity matrix proposed model is compared with three of the existing models – 'Count Vectorizer, TF-IDF, and GloVe' presented by various researchers. Details about these models are given in related works. According to the comparison analysis, Count Vectorizer is considered as a baseline and hence performs significantly lower than all models. The proposed Customized Word2Vec shows the best performance with similarity at 9 percent higher than GloVe considering the hat proposed model was inspired by the GloVe framework. The model is developed to provide a framework design to prioritize recommendations based on the user's own choice of listening type of podcast over popular suggestions.

# 1 Introduction

Podcasts are a new generation source of content consumption, says 'Podcast Insights ', a well–known online news magazine. As per their analysis, more than 2 million people are now consuming podcasts on daily bases through their phones and laptops. These estimates are identified latest by January 2021 and suggest a whooping growth of 55 percent in the field of audio content consumption. The magazine has identified various features of the population of active listeners to podcasts. As per the magazine, the majority of active listeners (i. e., those who have listened to more than 10 podcasts) belong to the European Union and the United States of America, with which the EU grabbing the top spot with 82 percent active listeners and the USA with 64 percent. They also described that range of ages actively listening to a podcast is wide, from 15 years to 56 years. This can be inferred from the wide variety of genres provided by different producers in the podcast domain.

Such a massive potential demand and availability of variety made many researchers identify the challenges in the field and present solutions on how to overcome them. One of the major issues identified is low levels of listener engagement with the podcast. For most beginners' podcasts are suggested through word-of-mouth publicity. Once the new user enters the environment they get flooded with various options to choose from for the next podcast. Many users are pushed through the platform to listen to the podcast which is popular due to its rating and positive reviews. This suggestion may work for few times, but not receiving an option to choose podcasts that align with the listener's interest may subsequently demotivate the consumer to keep listening to these shows. Therefore, researchers have tried to identify the most suitable solution for this problem and many of them suggest, creating a recommendation system for a podcast based on the user's interest.

The highly talked about process flow for developing the recommendation system is either User Collaboration based or User Content-based. The collaborated method works successfully when the item is being recommended through the collective choice of all the participating users, which is suggested through rating and reviews or rating of the podcast. This method is not optimal as discussed before because the system function by suggesting items that are liked by the majority of participants and demotivate the users to flow path through their interest. The second approach is Content based, which will suggest podcasts based on his current podcast libraries or recently listened to podcasts. The methodologies have an advantage over their predecessors, as it accounts for each user's choice of interest and trains the system accordingly to provide desired results.

In the content-based approach, researchers have suggested various frameworks such as using the LSTM model applied with the DistMult algorithm applied to genres to find the closest and most relevant item in the group. The model is effective but as it is applied to genre features, the list provided is still vast for a user to choose from. The same model applied to semantic information like title, producer name, geography, and demography gives a cohesive list of podcasts. The challenge here is the limitation provided by the semantic information on the type of content available. If a feature like geography is considered, it will recommend only podcasts available in that region that may not align with the user's interest to listen to podcasts globally available. Therefore, a new approach with a wider scope was required to create a framework most suitable for the listener.

The proposed project is using a novel concept of recommending items based on natural language techniques. As the podcast is audio content, the user will always have a prefer-

ence for the type of language to listen to. This infers that language is a feature that will remain consistent in all suggested podcasts. Some research conducted on this method suggest converting the audio content of each episode of a podcast and conducting textual analysis on them to find similarity. The method is highly time-consuming and lacks a focal point in the analysis as many podcasts have new topics to discuss in each episode. Therefore, the proposed project is presenting an NLP-based model trained on word embedding created using podcast description – a semantic feature providing the complete gist of a podcast and using cosine similarity on vector space to provide a list of podcasts according to the user's interest. The input parameter given to the model consists of single podcast titles in each genre. The model uses customized Word2Vec word embedding where vectors are created through tokenized technique applied on the preprocess dataset consisting of a bag of words relevant to the podcast. The input parameter given to the model consists of single podcast titles in each genre. These words are created using Stemming and Lemmatizing techniques to normalize the words. The Word2Vec model is developed using Skip-gram techniques to create word embedding using the context of the word in each sentence. The final embedding is then passed through cosine similarity to get a list of podcast titles that are similar to input podcasts.
To understand the performance of the proposed model, a comparison analysis is done with other proposed models suggested by various researchers which also use a similar concept of textual analysis. These models are Count Vectorizer, TF-IDF, and GloVe. The complete analysis for considering these models in the analysis is given in the Related work section. Finally, through experiment done on a dataset of 4300 podcasts captured from the Spotify website using Web API, results in providing a list of top 9 podcasts for given genres. As per the performance matrix, the Count Vectorizer model had the lowest accuracy, and the proposed model – the custom Word2Vec model had the highest with GloVe being at the 2nd position. The experiment successfully should that NLP techniques can be a useful tool to create a substantial recommendation system without any limitations through the process flow

# 2 Related Work

## 2.1 Semantic Feature Selection

In a content-driven recommendation system for the podcast, accessing its information is a requirement (Jones et al.; 2021). As suggested by the author, podcasts can be considered spoken information with a wide variety of genres and document styles. The greatest challenges for creating a search or recommendation system for podcasts consist of understanding the consumption patterns, listening habits, and any potential feedback system from consumers. As of current capacity, major platforms use data-driven variables to suggest popular podcasts. These variables can be ratings, view counts, review counts, etc. Although these systems are effective, they showcase majorly a skewed data and fore go the remaining unpopular podcasts.
Another method used is by creating a K-nearest neighbor model using cross domain metadata with pre-dominantly demographic feature to recommend podcast (Nazari et al.; 2020). This method has a significant advantage for a new user where popular podcasts are suggested in various genres based on their demography. The technique developed by the author uses various cross-domain KNN models created to compare user embedding with demographic data and music metadata. The goal was to create a most effective

top-m list of podcasts based on a combination of features (i.e., location, music interest) that may impact a user's choice. The methodology significantly provides an effective solution to filter podcasts based on users' content and choices. But still, it presents a wide list of choices for the user to scout. Also, using demography as a feature to search popular podcasts can limit the choices for the user to only podcasts that are available in that region.
(Rezapour et al.; 2021) has pitched the idea of using an automatic episode summary generated transcripts and genres as entities and feeding them into summarization models to suggest the appropriate style of podcast. The BERT token classification models used are abstract-driven and supervised by a given description from a podcast producer. ROUGE-L scores are compared to the generated summaries against the creator's descriptions. The ROUGE precision and F1 scores showed a 9 percent improvement in relevant summary creation when using genre and titles as named entities, compared to its baseline model. Although, the models created to show biased results towards summaries under performed, lacking in suggesting appropriate podcasts. This limited the usability of models for developing a recommendation system. Through these research papers, it can be agreed that developing a podcast recommendation system requires a category-aware classification model with genres and podcast descriptions of the most significant features.

## 2.2 COSINE Similarity

A research paper by (Reddy et al.; 2021) investigated the linguistic factors that may be responsible for predicting suitable podcasts and increasing the e engagement of consumers. The most probable factors considered were vocabulary diversity, distinctiveness, emotions, and syntax of various genres of podcasts. The analysis was conducted using the creator's descriptions and audio transcripts to build a topic-focused - 'Latent Dirichlet Allocation (LDA) model based on transcripts of 100+ topics and find features with the highest predictability. They trained a logistic regression classifier using various representations like a non-stylistic LDA topic distribution, bag-of-n-grams with TF-IDF scoring, and two feed-forwards neural networks. The proposed LDA model has a significant engagement of 70 percent with linguistic features. But as the scarcity of common words reduced the accuracy level dropped by 14 percent. Also, the models found it difficult to differentiate between words with closed semantic similarities.
The concept of using the textual description given by podcast producers and identifying text similarity is also proposed by (Wijewickrema et al.; 2019). The model is performed on journal articles published in two genres – social science and medicine. The authors have used three types of textual similarity – BM25, Cosine, and Unigram of two datasets each containing one genre. The textual similarity is compared using the NDCG values of the model. In the case of overall performance BM25 and Cosine similarity tie–up at the top position and Unigram is at the last position. But the score of identifying explicit technical vocabulary which is the most relevant part of the study to create better search results, Cosine Similarity has outperformed all of them.
Another research has shown that it can be reasoned to say that Cosine Similarity is a significant measure to compare textual information (Korenius et al.; 2007). The authors have conducted a principal component analysis (PCA) on Cosine and Euclidean similarity involving the mean-correlation of data. The textual language group used in the analysis is Finno-Ugric, which is considered to have single words with a different inflection, unlike

other European languages. This form of vocabulary with various meanings of a single word is proposed to be significant in identifying the accuracy of both textual similarity models. The results of the analysis showed a marginal improvement of 10 percent when Euclidean similarity is used with PCA compared to Cosine Similarity being used alone. Therefore, both textual similarities can be used to compare the performance of the model and understand the impact part by each of them.
A recommendation model created by (Liu and Guo; 2019) on House purchasing metadata uses Cosine similarity in user feedback information to predict the most suitable houses as per the consumer's needs. The model proposed used the user's information retrieved from a grid platform containing details of all the necessities in an ideal house and Cosine similarity is used to compare the user's data from retail descriptions given on open-house websites. Although the model is applied for a collaboration-based recommendation system, it is observed that cosine similarity measures have significantly performed well in showcasing houses to users with a similarity of more than 0.75. In a conclusion, it is found that most research including the most recent one, still backs textual analysis as the most preferable method to predict the choices of the user for audio content consumption. In terms of textual analysis, finding similarity leads to a better pathway for a recommendation, and Cosine similarity is a technique suggested as the most reliable one among all the techniques.

## 2.3 Model Consideration and Word Embedding

After identifying the suitable concepts for developing a recommendation system for the podcast, a suitable analytical framework is explored. Research done by (Dobkin and Satheesh; 2020), shared light on the symmetric pattern in podcast data, i.e., the user should ideally consume a complimentary podcast derived from their previous podcasts. The authors noted that many podcast listeners are loyal consumers of certain genres which were proven by analyzing ad conversion rates for each genre. Therefore, the proposed model consists of the customized algorithm - Word2Vec to indicate symmetric complement relationships in podcasts. Although the research identified the quantitative factors like podcast genres and descriptions to indicate similarities in choices for the user but contained gaps in proving the sub-filtration of genres to capture focused users – a suggestion given by the focus group to the research team.
Another paper published by (Galety et al.; 2022) suggested combining Count Vectorizer with Cosine Similarity to predict a user's choice in music using a context-aware recommendation system. In comparison to individual algorithms, the combined model outperformed two-fold. Both the Precision and Recall values were increasing with the number of recommendations for the proposed model. The paper by (Marappan and Bhaskaran; 2022), also suggests the use of a count vectorizer with cosine similarity and compares its performance within the KNN model. The said model performs moderately better with a combination of two algorithms. And henceforth, the suggested model is used as a baseline for the model framework in this proposal.
Another study done by (Liu; 2019) trained a shallow neural model with the Word2Vec model – a framework proposed by (Mikolov et al.; 2013) that used a 1.6 billion words dataset to train a high-quality word vector with lower computation cost. (Mikolov et al.; 2013) used a new concept called – the 'Continuous Skip-gram Model' which maximizes the classification of a word using another word in the same sentence to understand its context in a sentence. This CSGM model is the core of the Word2Vec model developed

by the authors. They used the DistBelief distribution framework to train the model over one trillion words and generated a whopping accuracy of 58.9 percent. The study hence became the baseline used in (Liu; 2019) research to create a recommendation system with domain–specific word embedding.
The models created using the Word2Vec framework performed significantly better than their baseline, showcasing positive evidence for using the Word2Vec model for podcast recommendation systems using textual embeddings. Although, unlike the researchers, the author has decided to create a custom train Word2Vec embedding using Skip gram framework to train the small dataset as its fast processing compared to the self-train model. Also, Cosine Similarity will be used with the model to better identify the textual information from the metadata. To further understand the performance of the proposed model, multiple frameworks are considered as part of the performance comparison analysis.
To start with, a paper published by (Saravanan et al.; 2022) showcased the use of the Term Frequency-Inverse Document Frequency model to analyze the bibliometric and textual information on review papers talking about the impact on the supply chain due to COVID-19. A total of 574 research papers were considered in the database to identify keywords and four major clusters of distinctive topics for text analysis. The proposed TD-IDF model showed a significant result in unveiling topics that were rarely studied. Another research paper, (Bahrani and Roelleke; 2022) has used a hybrid model using DCM frameworks like TD-IDF and BM25 to develop a semantic-aware retrieval and recommendation system. The framework is applied to multiple data sets like MovieLens, BookCrossing, and LastFM to understand the semantic relationship in data using users' reviews. The use of KNN similarity combined with the user-based TD-IDF model showed an exceptional result over the baseline model (i.e., only user-based TD-IDF). Hence, applying similarity algorithms the with TD-IDF model can help in getting better results.
In the paper, (Chirasmayee et al.; 2022), a content-based filtering recommendation is developed using vectorization of the TD-IDF model. The model is proposed to understand the impact of similarity on vector space. The researchers identified that TF and IDF vectors from the model helped in finding the song's lyrics in and across the analogous data. This helped in calculating noticeable genres in each song and their dominance across songs determined the weight of the genre. Applying cosine similarity to the vector scores helped in extracting a list of songs sorted by genre. TD-IDF vector model with cosine similarity proved to be a notable model to compare performance in models. Identifying podcasts having similar vocabulary but different genres can be a useful parameter to help in better training the model.
Another model which uses Vectors for word representation is GloVe (Pennington et al.; 2014). The paper published talks about capturing fine-grained semantic and syntactic variables using vectors. GloVe model considers both variables but has the origin of these variables as opaque. The model consists of a global log-bilinear regression model with global matrix factorization and local context window methodology. The researcher suggests the at model efficiently leverages statistical data by training non-zero elements in a word and word event matrix. The vector space model using cosine-similarity showed a meaningful sub-structure of word embedding and recognized 75 percent of word equivalence tasks. The paper argued that in all variable-controlled environments, using more than 80 negative samples declined the performance of the Word2Vec model by 5 percent, compared with the GloVe model. They inferred this decrease due to a lack of target probability distribution. Although, (Baroni et al.; 2014), suggest that both models per-

form better across a different range of the task, a comparison analysis between GloVe and supervised Word2Vec model with Skip-Gram framework all controlled but named entity recognition variable, will help in determining the performance of the proposed model.
Lastly a non-word embedding model, a BERT-based model is suggested by (Lazarova; 2021) to create a named entity-based recommendation system. The latest state-of-the-art model is used in many NLP-based systems to produce high-value language models. BERT is used to train a vast amount of textual data and uses its pre-trained weights to provide significant results. It can hence be considered that BERT could provide the most remarkable results for the project framework. Although, the model works best in the case of a highly dense textual database with multi-sentence correlations. For this research, the focus on finding the similarity in the podcast is using the descriptions and titles provided by the producer of the podcast. These mostly consist of fewer sentences and the point description and need textual analysis on short sentences. The BERT model is unable to serve this purpose and hence is not considered in the analysis.
The research work has suggested that to create an effective recommendation system for the podcast, the framework should consist of a category-based classification model with prominence given to features like genre, title, and description. When considering these features, the best framework design is to have a vector-based textual analysis and find their cosine similarity to create an effective list of user-specific podcasts. A log-bilinear predictive model – Word2Vec in supervised training with cosine similarity is the proposed model for creating the system as shows distinctive quality to capture words for short sentences with consideration of its context to the sentences. The model with the Skip-gram algorithm is most suitable for small databases focused on word embedding. The proposed model is also being compared with the globally recognized model like Count Vectorizer, TD-IDF, and GloVe with Cosine Similarity, suggested by the researcher as an efficient vector-based word embedding model.

# 3 Methodology

The methodology used in the project is created to provide a step-by-step procedure of data analysis and model training for constructing an optimal recommendation system for the podcast. It majorly consists of three processes divided into multiple tasks in each. The below section provides a detailed understanding of each of these processes.

## 3.1 DATA MINING

As the project is developed over analysis of various genres of the podcast, the author has considered collecting data from one of the prominent podcast streaming platforms for audio content consumption – "Spotify". This platform contains more than 3.6 million[1] podcasts in various genres, across different countries. The streaming service has country-wise charts for the popular podcast in every genre and for the reference of this project, the region considered is the United States of America, which has the second largest[2] audio-listening audience on the platform (after Europe). Although there are multiple ways to extract the metadata from the platform, Spotify itself provides a sophisticated licensing process to connect with its server to extract the data. Using the REST principles

[1]https:https://www.businessofapps.com/data/spotify-statistics/
[2]https://developer.spotify.com/documentation/web-api/

(Rodríguez et al.; 2016), they can provide a Web API – 'Spotipy'[3] endpoint which returns JSON metadata by connecting the client's authorized application with their server. The Web API functions as an authorized gateway to auto-access the developer's user account and download the required data available in their library. These accesses require permission granted by the user to download data. Spotify implements an OAuth 2.0 authentication framework (Darwish and Ouda; 2015), in which the end user grants permission to specific scopes requested by the client (i.e., author) to get access. To collect the data set, the author created a developer account on Spotify and got relevant client credentials with reference tokens. This is all the owed author to receive authorized GET Read requests and Edit genres scopes.
A query was performed to get popular podcasts from various genres like Comedy, News, Education, and Business. The detailed list of genres is available in the ICT Code Artifacts – Images/Ratings and Reviews. The popularity of the podcast was considered based on its ranking on the chart for the last 2 weeks. It was identified that the platform contains a limitation of 50 GET requests per loop. Therefore, a maximum offset limit was kept at 250 for each genre and the query was repeated till the maximum offset was reached. The whole process provided a dataset of around 4750 podcasts spread across 19 genres. For each of the genres, the data frame consists of a list of features like podcast name, description, genre, number of episodes, rating, and total no. of reviews. As the region chosen for this data collection was the USA, the raw data set consists of 90 percent of English-speaking podcasts. The non-native language podcast was removed from the database during data processing as the textual analysis using word embedding was done for the proposed recommendation system focusing on English as the preferred language.

## 3.2 DATA EXLORATION AND PRE-PROCESSING

As the data set was in raw form, a lot of pre-processing was required to make it available for further analysis. The data set collected using Spotify Web API was loaded into a data frame with discrete column names. Any row with NA values was removed from the data. For the title column, any unnecessary space between the sentences was removed and all the words were converted into the Capitalized format. As the textual analysis was being done for English-speaking podcasts only, all the non-English podcasts were identified and dropped from the data frame. This subsequently reduced the data set from 4750 to 4303 final podcasts with English as their language. For easy access to further analysis, the data set was saved in a pickle file for managing data storage.
To create a model for textual analysis, it was necessary to understand the different features of the data set. During the exploration of the data set, it was identified that all the 19 genres used in the data set have an almost equal number of popular podcasts ranging between 213 to 237. A graph showcasing this result is available in ICT code artifacts. Through this exploration, it can be inferred that although a user can have one genre of his/her liking, they are still bombarded with 200+ choices to select their next podcast. This can be a huge no-no for platforms that focus on capturing listeners' attention through visualization. Also, the overall rating for each genre is more than 4.5, which says that finding unique choices for podcasts using solely rating can be difficult as the threshold is minuscule. This can also be verified in the figure available in code Artefacts. The median chart of review per genre provides some substance in data exploration, as it suggests that Comedy, Society and Culture, and News are 3 highly reviewed

[3]https://podnews.net/article/how-many-podcasts

podcast genres with 3000, 1700, and 1500 being their respective median review counts. Hence, it can be considered that these three genres are among the most popular genres as they bring more engagement from listeners. In creating a recommendation system, these genres can be used to critically identify the performance of the models.
To further enhance the textual analysis, the podcast descriptions are processed to create a new column 'text' which will be used in word embedding for model development. A new list of stop words is created which will drop from the description. These stop words consist of words describing days or months, and non-relevant words like 'name, podcast, weekly, stories, host, join, etc.'. Once all stop words are dropped from the text column, a tokenizer is created using RegexpTokenizer class, to divide the sentences into separate words using regular expressions. Similarly, PorterStemmer and WordNeLemmatizer techniques available in Natural Language Processing are used to normalize the text for each podcast. These techniques provided a group of the most relevant words in the podcast description text. Finally, any mixed alphanumeric words or attached URL was also dropped from the text column, providing a clear data frame to build the recommendation model.

## 3.3 MODEL BUILDING

To create a model for recommending podcasts, the process is divided into two parts, the first is to create a model framework and the second is to create a process to find cosine similarity for vectorized text which has transformed after applying the model framework. The concept of this process is to transform vectors using a model algorithm, identify frequency-based embedding for transformed vectors and apply cosine similarity on the embedding to get the desired result of recommendation. To perform each task, multiple helper classes are created and called on demand. The proposed model used in the project is developed on the (Mikolov et al.; 2013) Skip-Gram Word2Vec framework Figure 1 but uses customized word embedding vectorization to enhance the performance of the model. The model is derived from the 'Gensim. models' library available in python and uses Skip-gram model design to identify the context of given words in the model. To understand the context of the word, the model creates '2 one-hot encoded target variables and 2 corresponding outputs.
Two distinct errors are calculated for two targeted variables creating two error vectors which are then added together to get the final error vector with updated weights. The input and hidden layers are given weights as vectors after training the model. The word embedding is created using a tokenizer for weighted values appending them to transform text documents into vectors. Each time a new token is created for each text in a row, it is converted into a vector and appended in an array of the transformed document. The mean embedding vectors are passed through the Word2Vec model to return a new array of the textual document.
As discussed in 2, there are three other models suggested by various research used to compare performance of the proposed model. Count vectorizer model is considered as the baseline model of the analysis. This model also called as the Bag of Words method downloaded from 'sklearn. feature.extraction.text' library, simply considers each word to create vectors. Theses vectors are created using fit and transformed generic class in which tokens are created for each text and appended into an array to return vectors. These transformed vectors are saved in a matrix and applied to cosine similarity class to get the desired results. Cosine similarity is performed on each of the model used in the project

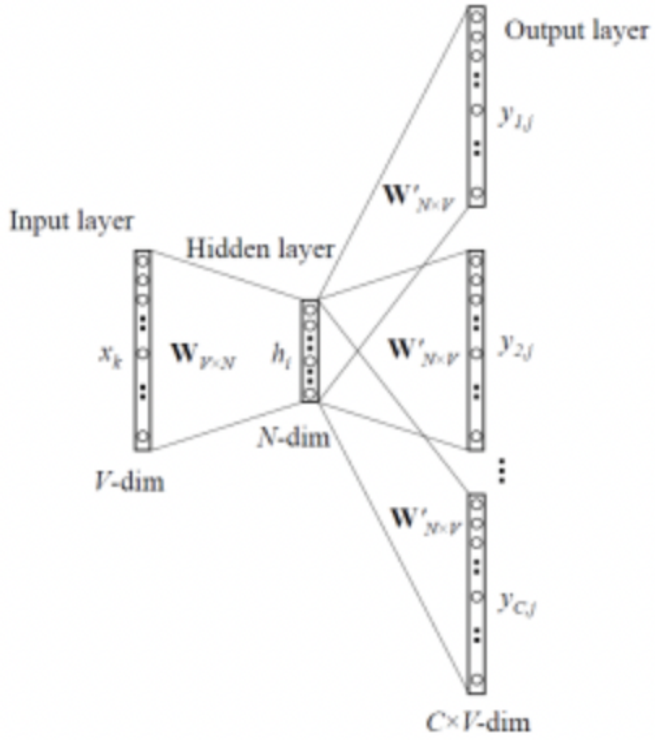


Figure 1: Miklovo's Skip-Gram Word2Vec Model Architecture

for comparison. It is derived by measuring the distance between the cosine values of the vectors converted from the text columns.

The second model used for comparison analysis in recommendation system, TF-IDF. This method also uses frequency-based word embedding for entire corpus applied on the transformed vectors. This model is also imported from 'sklearn. feature.extraction.text' library. The method of assigning vector values for TF-IDF is dependent on occurrence of word in the text. If a common word is identified, it receives a lower value and unique words are given higher importance. The underlying aspect of TF-IDF model is to identify unique words with least amount of occurrence in the document to receive highest vector. This system helps in fitting and transforming text into meaningful vectors in sorted with highest to lowest values, creating a vector matrix. When this vector matrix is applied to cosine similarity, the resultant array consists of podcast title having similarity derived for unique words.

Lastly GloVe model is loaded using KeyedVectors class from 'Gensim.models' library available in python. The model is trained on its unique word embedding where word-to-word co-occurrence matrix for given corpus is created for pair of words being appeared together in the context window. Each of the line in text column is split into multiple words and appended to a new array called as embedding. These embedding are mean vectorized similar to proposed model to get final weighted values of the vectors which is then passed through the model which identifies the co-occurrence probabilities of target words with numerous probe words in the vocabulary. Once the vectors are transformed using model, they are applied with cosine similarity to identify the final output. The belowFigure 2 summarizes the complete flow of process used in the proposed project.

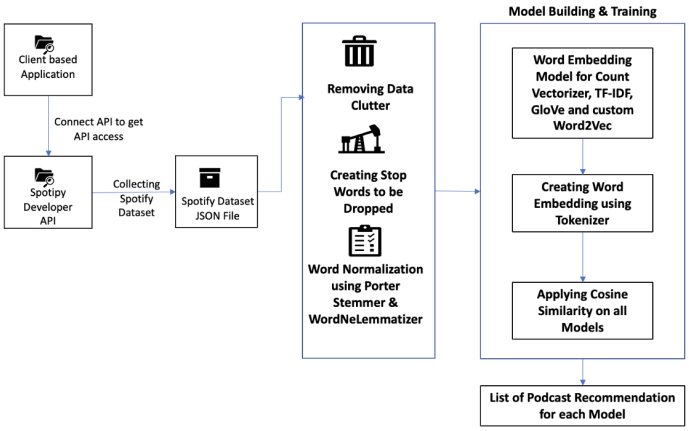


Figure 2: Process Flow

# 4 Implementation

The implementation of the process flow required input to the model to kick-start the recommendation. Hence a list of podcasts was created where one most popular podcasts from the different genres were considered. This was done with the assumption that when the user has to be recommended a list of podcasts as per his/her interest, the underlying consideration is that the user has already listened to at least one of the podcasts. Although this may not be in the case of cold start recommendations explained in relevant work, the concept of the framework is to engage the user into listening more and more of the podcast of their scope of interest. As discussed methodology section 3 genres like Comedy, Sports, News Society, and Culture are the prominent genres being considered for training the models. The below table Table 1 shows the list of podcasts being considered in various genres as input to the model.

Table 1: Input Podcast

| **Genres** | **Podcast Titles** |
|---|---|
| News | The Daily |
| News | Up First |
| Comedy | VIEWS with David Dobrik and Jason Nash |
| Comedy | Impulsive with Logan Paul |
| Sports | The Bill Simmons Podcast |
| True Crime | My Favorite Murder with Karen Kilgariff and Georgia Hardstark |
| Society and Culture | This American Life |
| Religious and Spirituality | Joel Osteen podcast |
| Technology | TED Radio Hour |
| Comedy | Call Her Daddy |
| Sports | Skip and Shannon: Undisputed |

It can be seen through the input values, the podcasts considered consist of a variety of textual features. Some have the producer's name in the titles, and few have the most common words used in English vocabulary. During the training of the model, the framework identified podcast descriptions using these titles as the unique ID and performed all the actionable on the description column. The implementation plan for the framework is to feed the selected podcast into all the suggested models and output the top 9 most similar podcasts for each of the genres with their cosine similarity with selected podcasts. The models are trained using the vectorized weights of the textual document and applied with cosine similarity. A generic helper function is created to get a list of recommendations of the podcast titles using the framework. These podcasts are captured in a list and sorted concerning their similarity values. The iteration for getting podcast titles is set to 11 iterations, appended in a list, and sorted concerning similarity scores. For each of the models including the proposed model, the helper function is called along with cosine similarity scores to get the recommended podcasts.

# 5 Evaluation

Once the dataset was trained and passed through all the word embedding and cosine similarity, it presented a list of the top 9 podcast title as per the input variables. Through manual investigation, it was found that the proposed model was able to successfully predict 93.2 percent of the relevant topics across different genres. So, the model was able to not only identify similar topics is given but also other genres. This served a major purpose of listener engagement as through the proposed model listeners were not restricted to a single genre and had various recommendations given across different genres. This ensures a massive engagement of the listeners with the podcast domain as through continuous streaming and cross-topic recommendation results, they will be able to listen to much more podcasts of their choice, rather than being thrown to topics running on high popularity.

The Figure 3 consists of the final list of the podcast containing the top 3 podcast titles suggested by the model with their genres. As per the final list of podcasts, the only podcast that is out of scope is The Archers and The MeatEater Podcast. They do not follow the input variables "VIEWS" and "Impulsive" podcast descriptions. This can be understood as the producers have not distinctly specified the overall theme of their podcast in the description and talk on a variety of random topics. A 2-dimensional PCA model from sklearn.decomposition library is applied to the vectors of the proposed model and a scatter plot Figure 4 is created on the transformed vector for performance analysis of word embedding. The plot shows the concentrated cluster of words embedded with high similarity. Hence validating the high performance of the model. The same scatter plot is used to identify clusters of similar words creating a concentrated 2D plot in PCA. The Figure 5 shows a bag of words with high similarity clustered together on the plot.

The significant performance of the proposed model is also compared with the other models proposed by different researchers. In the Subjective study shown in below Table 2 of model performance, it was identified that the baseline model – "Count Vectorizer" performed the lowest with 42 percent similarity and the GloVe model was able to reach closest to the proposed model with 85 percent of relevant topics to the input variables. This can be inferred as the GloVe model is derived from the baseline Word2Vec model and contains similar layers for creating word embedding.

| The Daily (News) | Impeachment: A Daily Podcast (News) | Up First (News) | The Takeaway (News) |
|---|---|---|---|
| Up First (News) | Impeachment: A Daily Podcast (News) | The Daily (News) | The Takeaway (News) |
| VIEWS with David Dobrik and Jason Nash (Comedy) | Amy Schumer Presents: 3 Girls, 1 Keith (Comedy) | Not Skinny But Not Fat (TV & Film) | The Archers (Arts) |
| Impulsive with Logan Paul (Comedy) | Comments by Celeb (Society & Culture) | Drew and Mike Show (Comedy) | Heartland Radio 2.0 (Comedy) |
| The Bill Simmons Podcast (Sports) | The Ryen Russillo Podcast (Sports) | ESPN Podcasts (Sports) | The Peter King Podcast (Sports) |
| My Favorite Murder with Karen Kilgariff and Georgia Hardstark (True Crime) | Fresh Hell Podcast (True Crime) | Wine & Crime (Comedy) | Murderous Minors: killer kids (True Crime) |
| This American Life (Society & Culture) | Radio Diaries (Documentary) | The Incomparable Radio Theater (Fiction) | Undiscovered (Science) |
| Joel Osteen Podcast (Religion & Spirituality) | Daily Grace (Religion & Spirituality) | Saddleback Church Weekend Messages (Religion & Spirituality) | The Porch (Religion & Spirituality) |
| TED Radio Hour (Technology) | TED Talks Society and Culture (Society & Culture) | WGRL NYC (Kids & Family) | Deepak Chopra's Infinite Potential (Society & Culture) |
| Call Her Daddy (Comedy) | Stiff Socks (Comedy) | Zane and Heath: Unfiltered (Comedy) | Bully and the Beast (Comedy) |
| Skip and Shannon: Undisputed (Sports) | First Things First (Sports) | High Noon (Sports) | Speak For Yourself with Whitlock & Wiley (Sports) |

Figure 3: List of Podcast Recommended

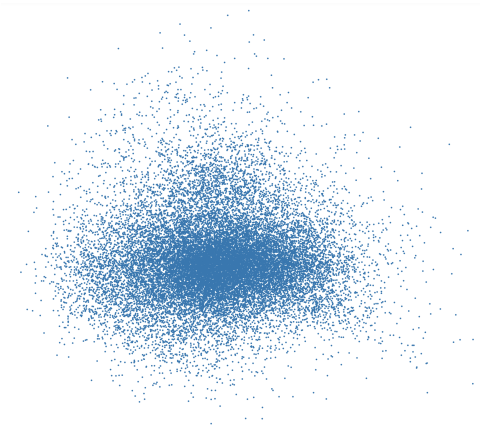

Figure 4: PCA 2D scatter plot

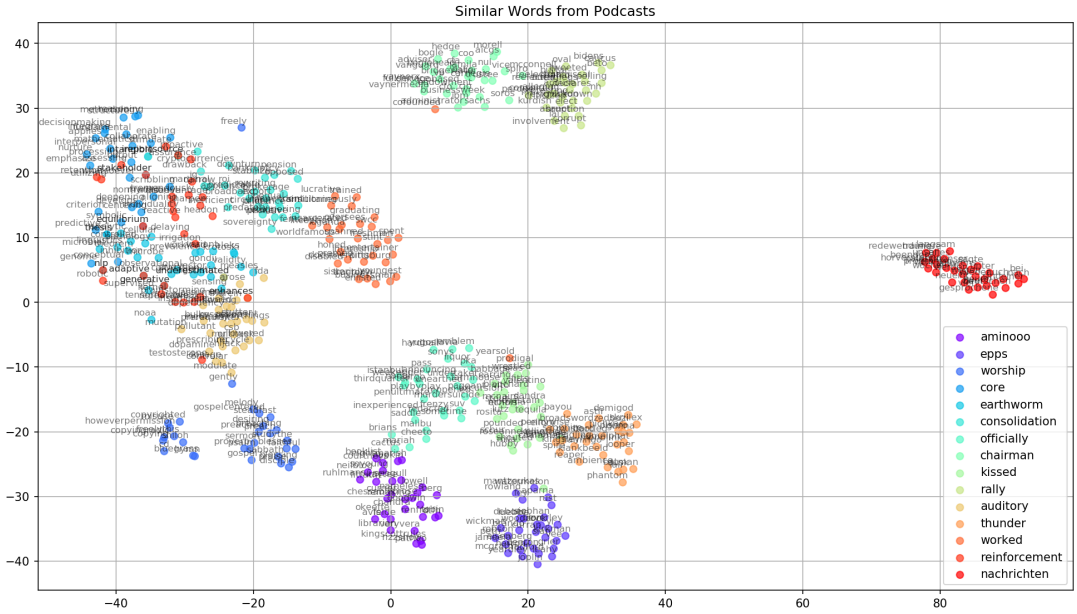


Figure 5: Similar word scatter plot

Table 2: Model Performance Matrix

| **Model Name** | **Cosine Similarity Achieved** |
|---|---|
| Count Vectorizer | 41.8 percent |
| TF-IDF | 67.38 percent |
| GloVe | 84.97 percent |
| Customized Word2Vec | 93.20 percent |

# 6 Conclusion and Future Work

The proposed model – the customized Word2Vec model was able to successfully train using a meaningful word embedding from the data set provided for the podcast recommendation system. And the use of cosine similarity on the embedding helped in generating the most relevant podcast as per the input podcast in various genres. The model showed a significant performance over the existing models, achieving 93.20 percent in successful similar topics. Hence it can be said that the approach of using a Natural Language Processing based word embedding on a semantic feature like podcast description can develop a significant framework for designing a recommendation model for the podcast.

The framework has shown that by only keeping one semantic feature common, Language, a wide variety of podcasts can be brought into the scope of textual analysis and used to recommend relevant topics to the user. Even though the input given to the system had assigned genres, the framework was able to find similar topics cross-genre and hence making the system sensible and consistent for the user's choice of interest. It can help in creating a long-term engagement of the user with the podcast environment as the user is not dragged along an assigned path or restricted to one type of genre. Therefore, the model has a significantly low deterioration and longer sustainability.

The research done was a limited database considered in one region only. For future work,

this can be expanded to other regions as well to help in identifying if the framework can suggest podcasts from various locations. A re-ranking algorithm based on user rating and review can be used with the framework to identify if the system can suggest podcasts with lower ranking but higher similarity. This will make the framework most robust and highly reliable to the domain knowledge. Expansion of the data set with rising demand in podcast consumption can help in improving the performance of the framework.

# References


Bahrani, M. and Roelleke, T. (2022). Semantic-aware retrieval and recommendation based on the dirichlet compound language model.

Baroni, M., Dinu, G. and Kruszewski, G. (2014). Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors, *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 238–247.

Chirasmayee, B. V. S., Sharmila, G., Sahithi, D. and Prabhakar, V. (2022). Song recommendation system using tf-idf vectorization and sentimental analysis.

Darwish, M. and Ouda, A. (2015). Evaluation of an oauth 2.0 protocol implementation for web server applications, *2015 International Conference and Workshop on Computing and Communication (IEMCON)*, pp. 1–4.

Dobkin, B. and Satheesh, B. (2020). To compliment a complement? modeling new avenues for podcast content discovery.

Galety, M. G., Thiagarajan, R., Sangeetha, R., Vignesh, L. K. B., Arun, S. and Krishnamoorthy, R. (2022). Personalized music recommendation model based on machine learning, *2022 8th International Conference on Smart Structures and Systems (ICSSS)*, pp. 1–6.

Jones, R., Zamani, H., Schedl, M., Chen, C.-W., Reddy, S., Clifton, A., Karlgren, J., Hashemi, H., Pappu, A., Nazari, Z., Yang, L., Semerci, O., Bouchard, H. and Carterette, B. (2021). Current challenges and future directions in podcast information access, Association for Computing Machinery, New York, NY, USA.

Korenius, T., Laurikkala, J. and Juhola, M. (2007). On principal component analysis, cosine and euclidean measures in information retrieval, *Information Sciences* **177**(22): 4893–4905.

Lazarova, M. (2021). Text content features for hybrid recommendations: Pre-trained language models for better recommendations.

Liu, A. (2019). Descriptive music search with domain-specific word embeddings.

Liu, F. and Guo, W.-W. (2019). Research on house recommendation model based on cosine similarity in deep learning mode in grid environment, *2019 International Conference on Virtual Reality and Intelligent Systems (ICVRIS)*, pp. 121–124.

Marappan, R. and Bhaskaran, S. (2022). Movie recommender model using machine learning approaches, *The Educational Review, USA* **6**(7): 317–319.

Mikolov, T., Chen, K., Corrado, G. and Dean, J. (2013). Efficient estimation of word representations in vector space, *arXiv preprint arXiv:1301.3781* .

Nazari, Z., Charbuillet, C., Pages, J., Laurent, M., Charrier, D., Vecchione, B. and Carterette, B. (2020). Recommending podcasts for cold-start users based on music listening and taste, *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1041–1050.

Pennington, J., Socher, R. and Manning, C. D. (2014). Glove: Global vectors for word representation, *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543.

Reddy, S., Lazarova, M., Yu, Y. and Jones, R. (2021). Modeling language usage and listener engagement in podcasts, *arXiv preprint arXiv:2106.06605* .

Rezapour, R., Reddy, S., Clifton, A. and Jones, R. (2021). Spotify at trec 2020: Genre-aware abstractive podcast summarization, *arXiv preprint arXiv:2104.03343* .

Rodríguez, C., Baez, M., Daniel, F., Casati, F., Trabucco, J. C., Canali, L. and Percannella, G. (2016). Rest apis: a large-scale analysis of compliance with principles and best practices, *International conference on web engineering*, Springer, pp. 21–39.

Saravanan, N., Olivares-Aguila, J. and Vital-Soto, A. (2022). Bibliometric and text analytics approaches to review covid-19 impacts on supply chains, *Sustainability* **14**(23): 15943.

Wijewickrema, M., Petras, V. and Dias, N. (2019). Selecting a text similarity measure for a content-based recommender system: A comparison in two corpora, *The Electronic Library* .