ISSN 2063-5346

TEXT GENRE CLASSIFICATION: A CLASSIFIED STUDY

B.Lavanya and R.Sowmiya***

Article History: Received: 01.02.2023 Revised: 07.03.2023 Accepted: 10.04.2023

Abstract

Classifying unstructured data is quite challenging as useful as it is. The overwhelming amount of textual data are collected in web, which when properly processed and categorized could open potential business opportunities. It has become significant to classify varieties of educative (books, poems, etc.,) and entertaining (movies, etc.,) information on web for recommendation systems to serve users better as well. With increasing innovations and breakthroughs in this domain, studying '(automatic) genre classification' stands non-trivial for its invaluable applications in improving web search results, information retrieval, etc., The aim of this paper is to bring to light the notable trends in this domain and its several stages. The distinctive features of different methods, along with the datasets and evaluation metrics used are compared. The crucial role of classifying text according to their genres in wide applications has also been pointed out in addition to the challenges, to draw research attention in these nascent areas.

Keywords: Text Genre, Data.

Department of Computer Science, University of Madras, Gunidy, Chennai, Tamilnadu, India.

*Email: lavanmu@gmail.com**, sowmiyar99r@gmail.com**

DOI:10.31838/ecb/2023.12.s1-B.383

I. Introduction

THE genre of a text document can be determined from different perspectives by different people. Simply put, the definition of the genre of a document change over time or according to the use case. In particular, two definitions of genre are pointed out: According to the authors of [1] “Genre is necessarily a heterogenous classificatory principle, which is based among other things on the way a text was created, the way it is distributed, the register of language it uses, and the kind of audience it is addressed to.”; Besides similar opinion in [2] the authors also add that the genre could be defined by the type of data structures used in it and the prescription of a process. And when authors of [3] classify the text categorization broadly, there are two branches, namely: Topic-based text categorization [4] where the texts are classified under one (or many) of the topics and the other being Genre classification where the texts are labelled to belong to one (or many) of the pre-defined genres. Further, the supervised learning task of text classification may concern with

1. Assigning one of the two labels (Binary classification) or one of the many labels (i.e., >2 multiclass classification) to a document, in which case, it can be collectively known as Hard Categorization [3] or

2. Assigning one or many labels to each document, in which case, it can be called multi-label classification or Ranking Classification [3].

Widely Genre classification is viewed as multiclass or multi-label classification since placing documents under one out of two genres (in many cases) might narrow down the options and not cater to the interest of many users - Especially in recommendation systems where the genre of the items (article/movie/book/songs, etc.,) provide an important source of information to decide about the next recommendation.

During this study, the papers contributed towards genre classification were largely found to also use multimodal features: audio features (for music genre classification) [5] or image/video features (Book/Movie genre classification) [6] along with Textual features as inputs. Since the concern is only about the textual features, only relevant works are filtered out for discussion.

The objective of this work is to collectively bring to the readers, the wide applications of Genre classification of text data, several methodologies followed (and the tools used), results of significance found and the challenges faced in this research area. This paper aims to present an insight into the problem of genre classification through this study.

The remainder of the paper is sectioned as follows: Section 2 discusses the steps in the Text Classification framework. Section 3 points out the common learning algorithms used for text classification. Section 4 is about Genre Classification in a variety of use cases: songs, poems, books, movies, and web pages. Section 5 explains the common metrics and Section 6 followed by the comparison of methodologies concludes as it hints the weight of the problem and improvements that can be done.

II. Text Genre Classification Framework

A. *Text Preprocessing*

As it goes for any dataset, the text dataset chosen for performing genre classification must be pre-processed and the widely practiced steps of text cleaning are as follows:

1) Stop words removal: The words that do not add much meaning to the context of the text (for instance, in English text, words such as the, a, is, etc.,) are removed.

2) Contraction mapping: If there are words written in their contracted form (such as didn’t, isn’t, etc.,) they are expanded (in other words, words are mapped to their expanded forms).

3) Lower casing: In order to maintain uniformity in cases, usually the words are lower cased. But there may arise loss of context in some cases (for example when US - United States of America is different from us – a pronoun). In that case other methods are resorted to (in US example contraction mapping can be used before lower casing).

4) Spelling correction: The text in dataset cannot be phonetically correct in several cases than not. There are many libraries providing help for correcting the spelling of the text which may serve to retain the context of the text.

5) Stemming and Lemmatization: The number of texts in a document can be reduced if different forms of the same word are reduced to a common form and thus keeping only one form of a word instead of many. Stemming deals with affixing (for example, ‘reading’ and ‘read’ to ‘read’) and lemmatization reduces the words to its lemmas (for instance, ‘bats’, ‘bat’ to ‘bat’).

6) Tokenization: Splitting every word/sentence/element of a text into smaller/meaningful tokens for easy investigation of frequency and other aspects of words in the text is called as Tokenization.

While preparing the dataset for classification task, the balance of instances in each class is to be taken care of.

B. *Text Representation*

Text must be represented in a way its salient features are extracted and for it to be fed into the model. There are many ways of representing text for learning classifier models to process the information (in a numeric form). A few of the notable and popular techniques are listed below:

1) Bag-of-words (BOW) approach: For each word in a document with vocabulary size n, there is a n-dimensional vector with value 1 at the index position of the word and 0 in the rest of the places: a one-hot encoded vector representation. BOW can also be interpreted as n-gram features of a document, specifically 1-gram. The words are selected, basically on some criteria, such as frequency to represent the content. tf-idf (Term Frequency – Inverse Term Frequency) which retains the information about the importance of the words as well can do better than BOW.

2)Word Embedding: While the former deals with capturing the syntactic representation of the word, Word embedding works on retrieving also the semantic relationship between the words e.g.: Word2Vec – This method strives to represent words as vectors in such a way that closely related words have lesser distance between them when represented in vector space. CBOW (Continuous Bag-of-Words) and Skip-gram are two basic models with simple architecture trained for predicting the word given the context and for predicting the context given the words respectively. Popular Word Embeddings are Word2Vec, GloVe (Global Vectors), FastText, Wiki2Vec, Wang2vec, Doc2vec, etc.,

3) Word Embedding and Position Embedding: In sequence-based models called transformers (e.g., BERT), it is also necessary to input the positional information of the words along with their word embeddings – both are considered for processing and the model takes care of it.

C. Dimensionality Reduction

Evading the curse of dimensionality problem is critical, particularly when dealing with documents that are lengthy and thus may have more features. Since typically text data contains more words (excluding short texts) thus more vectors (in general), feature selection (such as chi-square, information gain, document frequency, etc.,) or feature projection (or feature extraction) method is usually applied in order to bring the number of features to a manageable size, thereby retaining only the information-rich/relevant text. In the Literature, Context Indexing, PCA (Principal Component Analysis), LDA (Linear Discriminant Analysis) [7], LSA (Latent Semantic Analysis) [8], ICA (Independent Component Analysis), Autoencoder, t-SNE and other such feature extraction methods are commonly applied. ***The framework of this process is summarized in Figure 1.***

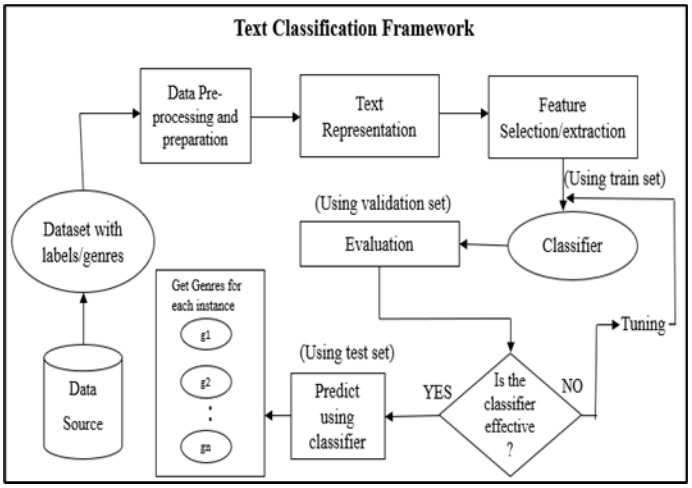
Text Classification Framework
Data Pre-processing and preparation
Text Representation
Feature Selection/extraction
(Using train set)
(Using validation set)
Evaluation
Classifier
Dataset with labels/genres
Get Genres for each instance
g1
g2
:
gn
Tuning
(Using test set)
Data Source
Predict using classifier
YES
Is the classifier effective ?
NO

Figure 1 Overview of Stages of Text Genre classification. Here g1,g2,..gn are the different genres/labels.

III. Learning Algorithms

A. Rule-based method

Simple rules are framed in such a way it helps to carry out classification. It is considered to be a more flexible and simple method.

B. Machine learning

Rocchio algorithm, ensemble methods, logistic regression, Naïve Bayes' classifier, SVM (Support Vector Machine), KNN (K-Nearest Neighbour), Decision Tree and Random Forest are a few of the widely employed text classification algorithms. There is also an unsupervised learning approach to this problem such as using K-means clustering, LDA (Latent Dirichlet Allocation) [9], etc., It is also possible to categorize text using Semi-supervised learning methods [10]

C. Graph-based method

Conditional Random Field (CRF), TextGNN (Text Graph Neural Networks) and its successors [11] – GNN-based models and Graph Attention Networks-based models (GAT) – which make the best use of graph concepts and attention mechanism can be found working well on this problem.

D. Deep learning

Deep Neural Networks, TextCNN and other CNN-based models and implementations [12] are great at learning data by simple to complex features, eventually aggregating the complexity of learning in each further layer. By comparing performance-wise, they are found to have an edge over most of the Machine learning algorithms.

E. Sequence-based models

RNN (Recurrent Neural Networks), LSTM (Long short-term memory), BLSTM (or Bi-LSTM or Bidirectional LSTM) [13], GRU (Gated recurrent units), Transformer–based models (such as BART, DistilBERT, etc.,) are recent breakthroughs which are efficient not only for the task of classification but also for many other NLP (Natural Language Processing) tasks paving way for remarkable improvement in NLU (Natural Language Understanding).

The following section discusses about the recent advances and progress made in the problem of genre classification. Though the concept of the genre could be applicable to a wide variety of documents such as to the style of writing in documents: if it is a narrative/descriptive/argumentative text or if a document is a short story/news article/essay and etc., [14]. Genre classification of poems, music, books, movies and web pages are selective interests of this study.

IV. Genre Classification

A. Music genre classification

Distinguishing music among a range of genres benefits the MIR (Music Information Retrieval systems) and music recommendation system. As critical as it is, it's also challenging. Mood classification of songs is another interesting topic of research. Moods are found to deal with musical features. Nevertheless, a part of lyrical features could also contribute to the mood [15]. [15] attempted to classify both the mood and genre of a collection of 600 songs into 10 moods (happy, sad, angry, relaxed, calm, gloomy, romantic, confident, disgusted, and aggressive) and 10 genres (pop, blues, country, folk, R & B, reggae, grunge, punk rock, soul, and metal) by making use of 10 types of POS-tags (nouns, verbs, relational pronouns, pronouns, prepositions, adverbs, articles, modals, interjections and adjectives). The steps followed in [15] are as follows: (i) pre-processing of lyric features (ii) identification of unique words (as featured selection process) and the grouping of synonyms with WordNet. (iii) WordNet-Affect to assign affective labels to concepts reflecting the feelings and moods of the songs followed by categorizing them into one of 10 categories (of moods) (iv) POS tagging using Lingpipe and employing classifiers provided by WEKA analyzer to classify music genres. [16] classified music into seven genres based on lyrics as well. The author experimented with two methods: After pre-processing the text, bag-of-words (BOW) method was opted and as for term weighting, tf-idf was employed as one of the methods whereas the other involved POS tagger for eliminating less important features. On training several WEKA's models, it was found that Naïve Bayes' classifier with BOW features performed well while POS features also yielded competitive accuracy in several classifiers, thus suggesting that POS features contribute a considerable amount of information to make a distinction between genres. The authors of [17] trained Logistic regression models, LSTM and BiLSTM with features of lyrics extracted from GloVe embedding and concluded the better performance of LSTM and logistic regression and diminished performance of BiLSTM after balancing the genres of the dataset. In [13], several vectorization techniques such as Word2Vec, Wang2Vec, FastText and skip-gram architecture models were implemented on the Brazil song lyrics dataset. Traditional models such as SVM and Random Forest, and a Deep learning model of BLSTM (Bi-directional LSTM) were trained with multiple combinations of the feature vectors and arrived at the conclusion that BLSTM model using Wang2Vec model obtained the best F1-score for classifying lyrics into 14 genres.

It is also to be noted that during the study, many approaches to the problem centered on multimodality (such as collecting audio features, tags, and other features in addition to lyric features) were found to be prodigal due to the idea that more features (in varied forms) could yield more information about the item and thus enhanced performance.

B. *Poem genre classification*

Challenging as much as interesting, poem classification accompanies style identification, author identification and emotion detection, etc., Organizing poems on basis of the genre could save the time of readers, thus attracting more readership. Recent work in Tamil, [18] has explored 'Tamil Pann' (melodic mode) and author classification on thirumurai dataset. Pann can also denote the moods of the poems. The tokenization step was carefully carried out by retaining the new line character between the lines of the poems and then three transformer-based models were trained, namely: LaBSE (Language-Agnostic BERT Sentence Embedding), mBERT (Multilingual BERT) and XLM-RoBERTa and a three-layered Bi-LSTM were also trained. It was reported that transformer-based models carried out the task without developing bias about the data despite the presence of class imbalance. Finally, with K-fold cross-validation (k=5), it was found that LaBSE had an edge over other models by marginal differences in both the author and Pann classification tasks. Authors of [19] apply two filter methods for feature selection of pre-processed (applied tokenization, stop word removal and stemming) English poems of 8 genres and employ rough set theory as a supervised learning algorithm to obtain a higher accuracy of 85%. In [20], a manually annotated Punjab poetry dataset (Kāvi) with Fleiss Kappa's index employed for inter-annotator agreement, was utilized to categorize the poems into 9 rasas/emotion states. Linguistic, statistical and poetic features were extracted. tf-idf was employed. SVM and Naïve Bayes' classifier were trained with the obtained features with 10-fold cross-validation. It concluded the superior performance of SVM and the usefulness of poetic features for improving the classification results. [21] constructs appropriate features to identify Ci pai (tunes) of the Song Ci Chinese poems by considering the number of sentences, Chinese characters, clauses with different character numbers, information of Level, Oblique Tones and author information respectively to train random forest, SVM and Naïve Bayes model. The best results (in terms of accuracy, recall and F1-Score) were arrived at by the Random Forest model when the first four features were used. Since the last feature (author information) was found to be scattered across classes, it only led to a decline in accuracy when introduced into the model.

Yet another poem classification [22] based on Punjabi poetry involved tokenizing the text followed by BOW features weighted by term frequency scheme, and experimenting with it on 10 classifiers of the WEKA tool. Among ten of the algorithms, four were found to outperform and they were (in the order of increasing accuracy): Hyperpipes (HP), K- nearest neighbour (KNN), Naive Bayes (NB) and Support Vector Machine (SVM).

The poem classification models could also extend to meter classification [23] and author identification [18],[24].

C. *Book genre classification*

In this section, the document genre classification, and style classification are also included besides the book genre classification. Largely benefitting the online catalogues and digital repositories, books arranged by their genres could also help target interested consumers in E-Commerce platforms. As discussed in preceding sections, the approach to the problem through multimodality, in this case, might consider features like the image on the book cover, the structure of the document, etc., The web document genre classification is partly introduced in this section and elaborated in Section E.

[8] suggested using movie reviews as the features representing the genres of the books. Gathering multilingual reviews on Portugal books, the authors started with translating the texts into English followed by pre-processing and tf-idf. For dimensionality reduction, LSA which uses truncated SVD (Singular Value Decomposition) was preferred. On trying several Machine learning algorithms, it obtained good accuracy when using Random Forest Classifier. Two notable implementations in [25] are: a bag of senses was constructed and unlabeled text was introduced for the classifier to better learn the features. With the help of WordNet, frequent senses of words were collected into a frequency table. As popularly used, tf-idf followed where the unlabeled text also improved the term weighting, and PCA was opted for dimensionality reduction. Lastly, Decision Tree Classifier enhanced with the

AdaBoost classifier (to reduce bias and variance) was used to achieve greater accuracy. In [26], a powerful representation of valuable features was constructed – the fanfictions were classified into multiple genres by exploiting the characters present in the text and determining their relations. Tokenization, lemmatization, POS tagging for knowing the nouns, NER (Named Entity Recognition) for discerning the proper nouns, character identification and unification were preliminary steps employed in this study to build a character graph which has each character as its vertex and the presence of interaction as the edge between them while the number of interactions between any two characters would constitute the weight of the edge between them. Thus, a character network was built ignoring very small interactions. Since no two graphs could be compared directly in this case, it was achieved using 'the difference in the sum of eigenvalues of the Laplacian matrix of the graphs as a measure of closeness'. K – closest graph for each book in the dataset was filtered out and arranged in the increasing order of their similarity score. Multiple classifiers were trained such as Multi-Layer Perceptron (MLP), Random Forest, Gaussian Naive Bayes' Classifier, Gaussian Process Classifier, SVM with linear & RBF kernel, Gradient boosting, Ada Boost classifier. Among them, SVM with Linear kernel came out to lead in accuracy, recall and F1-score.

[27] classified 200 text documents based on the style of writing: formal and informal for which it employed tf-idf for vectorizing and achieved a score of 94.97% accuracy with the Random Forest model. This task classified texts on writing style-based genres. A classification work on research papers, [28] employed tf-idf and LDA to decide on feature importance and used K-Means clustering to group documents of similar topics based on keywords extracted from the abstracts and topics of the paper. [29] exploited inter-relationships among research papers based on several criteria such as citations, common references and common authors eventually classifying research papers using a graph-based approach.

[30] implemented essay classification algorithm enhancing the distinction between narrative and argumentative construct. The features used include word frequency, special words, POS, clusters of similar words (in order to alleviate data sparsity) and the topics grouped by the Latent Dirichlet Allocation Model (LDA). Parallelly two studies were conducted experimenting with different classifiers, training on obtained features separately and then grouping them. A few other features such as temporal words were included. The remarkable performance of random forest was reported. [31] presented a simple method of web document classification with two datasets exploiting the BOW technique, feature selection then training the classifiers provided by the WEKA tool and concluded that Naïve Bayes' classifier outperformed other models when applied 10-fold stratified cross-validation.

D. Movie genre classification

As [32] hinted that the genre classification related to movies could be of two types: one predicting the genre of the movie (e.g., comedy/adventure/action) with text features or (/and) it could be predicted if the movie is suitable for an audience of specific age group with the help of 'movie rating' feature. Prevalently for the former, the features used could be movie reviews, movie summaries, titles, etc., In [33], the authors pre-processed the scraped data on movie reviews and lemmatized it. They applied tf-idf vectorization on the text and chose the top important words. Experimenting with MLP (Multilayer perceptron) and KNN with K=3 to classify movies into one or many of 27 genres (multi-label classification), they found KNN yielded the best results with lesser hamming loss. They emphasized that movie reviews carry latent information about movie genres. [34], though a work extending towards the Recommender system domain, it had to perform genre classification in its preliminary stage for which it pre-processed and applied tf-idf vectorizer on movie reviews and experimented baseline models: KNN, SVM with linear kernel, Random Forest classifier and Naïve Bayes' classifier, out of which it reported KNN had better accuracy. Later numericizing the genres of each movie into a continuous value, the authors of [34] experimented with regressor models and neural networks, opting to the deep neural network finally.

[35], the authors explored the effectiveness of 19 different feature representation approaches on the movie synopsis dataset (which was pre-processed and stemmed) and trained four models: MLP and three Tree-based models with the combinations of vectors. They concluded that the MLP classifier trained on tf-idf weighted features excelled comparatively. It was trained with a stochastic gradient descent optimizer, 100 neurons in its hidden layer activated by ReLU (rectilinear activation function) and iterated over 200 epochs. The superiority of tf-idf vectorizer over other feature representation techniques was reported.

A vectorizing technique incorporating character n-grams into the skip-gram model was employed in [36] and the vectorized movie summaries were used to train a Bi-LSTM model for classifying movies of 4 different genres. But instead of training with each review as each instance, the authors propose breaking it down into sentences, then vectorizing, followed by majority voting for the final decision aided by the posterior probabilities of genres assigned to the review sentences.

E. Web genre classification

Web page genre classification deals with assigning one of many labels to web pages for easy information retrieval based on users' interests. The authors of [37] quoted "it would be of much help if a search engine could deliver only documents of a desired — what is here called — 'genre'", Thus, web browsers are expected to serve the querying users with relevant materials for which the user's interest/genre of the content is crucial. Indeed, the web pages could be online shopping sites, news media, government sites, etc.,

[38] proposes web knowledge graph-based hierarchical genre classification of web pages. The features contribute to the construction of the graph. It has four-layered architecture namely: input, embedding, encoder and output layers. The input layer is responsible for determining the entity set of the website with the help of the website knowledge graph. The embedding layer employs region embedding, the embedded

features contribute to hidden variables and the entity representation averaged to generate text representation (a hidden variable, per se) is fed as an input into a linear classifier. The proposed algorithm carries out the task, optimizing the BCE With Logits Loss for recursive regularization and hierarchical classification. Semi-supervised multi-view learning (SML) and a graph convolutional neural network are combined in a method proposed by [39]. (GCN). It comprises of two modules, which are semi-supervised multi-view graph convolutional representation learning and multi-view graph construction, both of which are combined into a single framework. The latter module is believed to play the role of fusing multi-view representation through an inter-view attention strategy, whereas the former tends to acquire the best possible graph structure for each aspect. Both labelled and unlabeled data were used in this investigation to successfully classify the web pages.

In Figure 2, the distribution of metrics used to evaluate the best models in the works studied inclusive of those cited in Table.1 is represented as a pie-chart with 'other metrics' being hamming loss [33], no genre ratio [33], error rate [19],[27], the silhouette score [28] etc.,

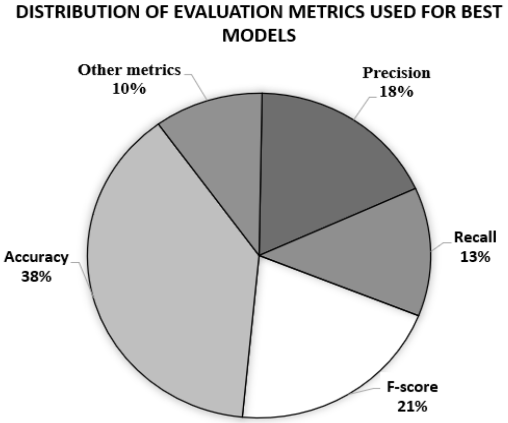

Figure 2: Distribution of metrics used to evaluate the best models of the research works cited in Table.1

V. Evaluation Metrics

Widely used classification metrics can be defined as answering the questions as follows:

1) *Precision:* Out of all the documents identified to be positive, how many are truly positive?

2) Recall: How many of actual true positive documents were predicted as true positive documents?

3) F1-Score: What would be the harmonic mean of both the precision and recall? It can be thus inferred to rely on both the above metrics. But all the above three metrics are found to be effective to evaluate the classifier well in the class-imbalanced scenarios.

4) Accuracy: How many predictions have the classifier made right?

5) Weighted precision, recall and F1-score: These weighted metrics depend on their per-class respective metrics and per-class weights. Especially F1-score can be macro F1-score or macro-averaged F1-score, etc.

6) Silhouette score: This is employed in evaluating clustering techniques. How to evaluate the performance of the clustering technique relating the inter-cluster (neighbouring clusters) and intra-cluster distances?

7) Hamming loss: How many genres are incorrectly predicted out of all the genres? It is effective in cases of multi-class classification problems. Lesser the loss, better the performance.

TABLE I: COMPARISON OF VARIOUS TYPES OF GENRE CLASSIFICATION, DATASETS, MODELS, TOOLS USED AND PERFORMANCE MEASURES USED.

Authors	Dataset	Dataset details	proposed classifier and tools used (if any)	Evaluation metric - score
[15]	http://lyrics.wikia.com/Lyrics_Wiki http://www.metrolyrics.com/ and http://amarok.kde.org/wiki/Scripts [http://www.allmusic.com/ & http://last.fm/] for genre selection.	#G: 10 Song genres and 10 moods #I: 600 Songs	c: SVM t: WordNet, WordNet-Affect, http://conceptnet5.media.mit.edu/, Lingpipe, WEKA	Accuracy – 0.39 (Genre) Accuracy – 0.59 (Mood)
[16]	musiXmatch and Last.fm MSD - Million Song Dataset	#G: 7 Song genres #I: Lyrics of 237,662 songs & 900,000 tags	c: Naïve Bayes t: WEKA	Accuracy – 0.79
[17]	https://www.kaggle.com/datasets/neisse/scrapped-lyrics-from-6-genres	#G: 3 Song genres #I: 7580 English songs	c: LSTM	Accuracy – 0.68
[13]	Vagalume site crawled with Scrapy https://www.vagalume.com.br/browse/style/ https://www.vagalume.com/br/roberto-carlos/como-e-grande-o-meu-amor-por-voce-letras.html2	#G: 14 Song genres #I: 138,368 Brazil songs	c: BLSTM with Wang2Vec	F1 score – 0.47
[18]	All the poems were scraped from the Tamil Virtual Academy http://www.tamilvu.org/	#G: 24 Panns (melodic modes) and 3 authors #I: 5548 poems	c: LaBSE	Precision – 0.91 (wt) Recall – 0.91 (wt) F1 score – 0.91
[19]	https://www.poetrysoup.com	#G: 7 English poem genres #I: 568 poems	c: Rough Set Theory	Precision – 0.9 Accuracy – 0.85 Error rate – 0.12
[20]	http://www.punjabi-Kāvita.com/ http://www.punjabizm.com/ http://punjabimaaboli.com/	#G: 9 Ras (emotional states) #I: 948 Punjabi poetries (Kāvi)	c: SVM	Precision – 0.69 Accuracy – 0.7
[21]	http://qsc.zww.cn/	#G: 73 Ci Pai #I: 3650 Song Ci	c: Random Forest	Precision – 0.93 Recall – 0.92 Accuracy – 0.92
[22]	http://www.punjabizm.com./ http://www.punjabi-kavita.com/ http://punjabimaaboli.com/	#G : 4 poem genres #I : 240 Punjabi poems	c: SVM	Accuracy – 0.58
[8]	PPORTAL, a dataset of public domain literature	#G: 24 Book genres #I: 3790 book reviews	c: Random Forest t: Translate tool, Goodreads API	Accuracy – 0.96
[25]	Scraped from Website: De Dillmont, T. (1987). Encyclopaedia of Needlework. Editions Th. de Dillmont, by using Beautiful Soup python library.	#G: ** #I: 3600 books	c: Decision tree classifier with AdaBoost t: WordNet	Accuracy – 0.92
[26]	Fanfiction.net	#G: 18 Book genres #I: 500 books	c: SVM with linear kernel	Recall – 1.00 F1 score – 0.80 Accuracy – 0.67
[27]	**	#G: 2 categories: (formal/ informal) #I: 200 text documents	c: Random Forest	Accuracy – 0.94
[28]	Future Generation Computer System (FGCS) journal	#G: NA #I: 500 research papers	c: LDA t:HDFS, Hadoop	F1 score >0.8
[29]	http://portal.acm.org	#G: ** #I: 255 research papers	c: Graph-based method t: Fetch Agent	Precision – 0.90 Recall – 0.85
[30]	**	#G: 2 styles: argumentative / declarative #I: 16584 Chinese documents in total	c: Random Forest	Accuracy – 0.8
[31]	LAUTECH website	#G: 2 classes #I: 100	c: Naïve Bayes t: WEKA	Accuracy – 0.77
[33]	Large Movie Review Dataset v1.0 (http://ai.stanford.edu/amaas/data/sentiment) and http://www.imdb.com/title/tt0211938/reviews	#G: 27 movie genres #I: 7000 movie reviews	c: KNN	Accuracy – 0.55 Hamming loss – 0.04
[34]	https://ieee-dataport.org/open-access/imdb-movie-reviews-dataset	#G: 17 Movie genres #I: 932464 reviews	c: Deep neural network	R2 Score – 0.82 RMSE – 0.06
[35]	TMDB (https://www.themoviedb.org/)	#G: 12 movie genres #I: 12094 synopses	c: MLP	Precision – 0.57 Recall – 0.53 F1 score – 0.54
[36]	https://grouplens.org/datasets/movielens/ http://www.omdbapi.com/	#G: 4 Movie Genres #I: 22278 Review sentences	c: Sentence-level BLSTM	(ma)Precision, (ma)Recall, (ma)F1 & (mi)F1scores =0.67

** - Information not disclosed in the original work or unavailable; NA – not applicable, #I – Number of instances, #G – Number of genres. t: tools, c: Classifier. (wt) – weighted metric, (ma) – macro metrics, (mi) – micro metrics

VI. Conclusion

This paper sheds light upon the significance and various research done on the roles of genre classification using text data. The innovative latest approaches towards the same have been listed out and analyzed based on their techniques and performance. We conclude with the brief discussion about the challenges in this classification problem: There are new terms being coined every day and brought into practice as are new genres created and thus needed to be learnt by the NLP models and maintained, so that the materials on the web would be consistent in serving relevant text to the user. One system of classifying genres could differ from another system, hence an item (article /song /book, etc.,) need to be mapped for resolving any ambiguity that may arise. Also, the cross-genre classification and the genre classification of low-resource languages are nascent areas of research. Web items to be classified such as books, sometimes need to be read entirely in order to decide on the genre in which it would fit in but that is not possible especially when there is a huge amount of data to be processed and several books exist in a digital repository. For such problems to be tackled more effective alternative ways are to be explored. Also, the ambiguity between the genres needs to be resolved in order to make the distinction clear and therefore the classification accurate. With the innovations made using transformer models in recent times, for solving the text genre classification problem, enhanced performance can be achieved. Yet undeniably there is more scope for improvement and exploration.

Acknowledgment

This work is supported by University Grants Commission (UGC) under JRF scheme.

References

[1] B. Kessler, G. Nunberg, and H. Schuetze, “Automatic Detection of Text Genre,” Jul. 1997 [Online]. Available: http://arxiv.org/abs/cmp-lg/9707002

[2] Y. Kim and S. Ross, ‘Formulating Representative Features with Respect to Genre Classification’, in *Genres on the Web: Computational Models and Empirical Studies*, A. Mehler, S. Sharoff, and M. Santini, Eds. Dordrecht: Springer Netherlands, 2011, pp. 129–147. doi: 10.1007/978-90-481-9178-9_6.

[3] M. Ikonomakis, S. Kotsiantis, and V. Tampakas, “Text Classification Using Machine Learning Techniques,” *WSEAS TRANSACTIONS on COMPUTERS*, vol. 4, no. 8, pp. 966–974, Aug. 2005, [Online]. Available: https://www.researchgate.net/profile/V-Tampakas/publication/228084521_Text_Classification_Using_Machine_Learning_Techniques/links/0c96051ee1dfda0e74000000/Text-Classification-Using-Machine-Learning-Techniques.pdf

[4] K. Nagarajaiah, M. H. Krishnappa, and A. K. Rukmini, ‘DOCUMENT CLASSIFICATION SYSTEM USING IMPROVISED RANDOM FOREST CLASSIFIER’, European Chemical Bulletin, vol. 12, no. 4, pp. 5751–5769, 2023, doi:10.48047/ecb/2023.12.si4.510

[5] M. P. V. N. Sai and S. Kalaiarasi, ‘Analyzing and Improving the Accuracy of Music Genre Classification using Novel Support Vector Clustering Algorithm compared with Logistic Regression Classifier’, European Chemical Bulletin, vol. 12, no. 1, pp. 3549–3558, 2023, doi:10.31838/ecb/2023.12.sa1.324

[6] C. S. Kundu, “Book Genre Classification By Its Cover Using A Multi-view Learning Approach,” Doctoral Dissertation, Western Kentucky University, 2020. [Online]. Available: https://digitalcommons.wku.edu/theses.

[7] Huang Ke and Ma Shaoping, “Text categorization based on. Concept indexing and principal component analysis,” in *2002 IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering. TENCOM ’02. Proceedings.,* Oct. 2002, vol. 1, pp. 51–56. doi: 10.1109/TENCON.2002.1181212.

[8] C. Scofield, M. O. Silva, L. de Melo-Gomes, and M. M. Moro, “Book Genre Classification Based on Reviews of Portuguese-Language Literature,” in *Lecture Notes in Computer Science* (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 13208 LNAI, Springer Science and Business Media Deutschland GmbH, 2022, pp. 188–197. doi: 10.1007/978-3-030-98305-5_18.

[9] S. Chauhan and P. Chauhan, “Music mood classification based on lyrical analysis of Hindi songs using Latent Dirichlet Allocation,” *2016 International Conference on Information Technology, InCITe* 2016 - The Next Generation IT Summit on the Theme - Internet of Things: Connect your Worlds, pp. 72–76, Feb. 2017, doi: 10.1109/INCITE.2016.7857593.

[10] S. C.Dharmadhikari, M. Ingle, and P. Kulkarni, “Analysis of Semi Supervised Learning Methods towards Multi Label Text Classification,” in *International Journal of Computer Applications*, vol. 42, pp. 15–20, 03 2012, doi: 10.5120/5775-8026.

[11] L. Yao, C. Mao, and Y. Luo, “Graph Convolutional Networks for Text Classification,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, pp. 7370–7377, Jul. 2019, doi: 10.1609/aaai.v33i01.33017370.

[12] N. Promrit and S. Waijanya, “Convolutional Neural Networks for Thai Poem Classification,” in *Lecture Notes in Computer Science* (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 10261 LNCS, Springer Verlag, 2017, pp. 449–456. doi: 10.1007/978-3-319-59072-1_53.

[13] R. de Araújo Lima, R. C. C. de Sousa, H. Lopes, and S. D. J. Barbosa, “Brazilian Lyrics-Based Music Genre Classification Using a BLSTM Network,” in *Lecture Notes in Computer Science* (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 12415 LNAI, pp. 525–534, 2020, doi: 10.1007/978-3-030-61401-0_49.

[14] A. Gonçalves, “A SUPERVISED TEXT MINING APPROACH FOR AUTOMATIC TEXT GENRE CLASSIFICATION,” *In 16th Doctoral Symposium in Informatics Engineering*, Apr. 2021, p. 168. [Online]. Available: https://scholar.archive.org/work/jr6knrg6krbzpakja7573hlpmy/access/wayback/https://books.fe.up.pt/index.php/feup/catalog/download/67/290/224#page=177

[15] T.C Ying, S. Doraisamy, and L.N Abdullah, “Genre and mood classification using lyric features,” in *2012 International Conference on Information Retrieval & Knowledge Management*, Mar. 2012, pp. 260–263. doi: 10.1109/InfRKM.2012.6204985.

[16] J. Yang, “Lyric-Based Music Genre Classification,” 2018. [Online]. Available: http://hdl.handle.net/1828/9378

[17] M. Leszczynski, A. Boonyanit, and A. Dahl, “Music Genre Classification using Song Lyrics,” 2022. [Online]. Available: https://web.stanford.edu/class/cs224n/reports/final_reports/report003.pdf.

[18] S. Mahadevan et al., “Thirumurai: A Large Dataset of Tamil Shaivite Poems and Classification of Tamil Pann,” in Proceedings of the Thirteenth Language Resources and Evaluation Conference, Jun. 2022, pp. 6556–6562. [Online]. Available: https://aclanthology.org/2022.lrec-1.704/

[19] S. Ali Alsaidi, A. T. Sadeq, and H. S. Abdullah, “English poems categorization using text mining and rough set theory,” Bulletin of Electrical Engineering and Informatics, vol. 9, no. 4, pp. 1701–1710, Aug. 2020, doi: 10.11591/eei.v9i4.1898.

[20] J. R. Saini and J. Kaur, “Kāvi: An Annotated Corpus of Punjabi Poetry with Emotion Detection Based on ‘Navrasa,’” Procedia Comput Sci, vol. 167, pp. 1220–1229, 2020, doi: 10.1016/j.procs.2020.03.436.

[21] B. Wang, J. Zheng, Y. Du, and L. Yang, “Automatic Recognition of Tune Names of Song Ci-Poetry,” in *2018 International Conference on Asian Language Processing (IALP)*, Nov. 2018, pp. 189–192. doi: 10.1109/IALP.2018.8629234.

[22] J. Kaur and J. R. Saini, “Punjabi Poetry Classification,” in *Proceedings of the 9th International Conference on Machine Learning and*

Computing, Feb. 2017, vol. Part F128357, pp. 1–5. doi: 10.1145/3055635.3056589.

[23] A. Mahmudi and H. Veisi, "Automatic Meter Classification of Kurdish Poems," Feb. 2021, [Online]. Available: http://arxiv.org/abs/2102.12109.

[24] C. Gallagher and Y. Li, 'Text Categorization for Authorship Attribution in English Poetry', in *Intelligent Computing*, 2019, pp. 249–261.

[25] S. Gupta, M. Agarwal, and S. Jain, "Automated Genre Classification of Books Using Machine Learning and Natural Language Processing," in *2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, Jan. 2019, pp. 269–272. doi: 10.1109/CONFLUENCE.2019.8776935.

[26] Rahul, Ayush, D. Agarwal, and D. Vijay, "Genre Classification using Character Networks," in *2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS)*, May 2021, pp. 216–222. doi: 10.1109/ICICCS51141.2021.9432303.

[27] K. M. G. S. Karunarathna, R. A. H. M. Rupasingha, and B. T. G. S. Kumara, "Classifying Documents based on Formal and Informal Writing Styles using Machine Learning Algorithms," in *2022 2nd International Conference on Advanced Research in Computing (ICARC)*, Feb. 2022, pp. 373–378. doi: 10.1109/ICARC54489.2022.9753774.

[28] S.-W. Kim and J.-M. Gil, "Research paper classification systems based on TF-IDF and LDA schemes," Human-centric Computing and Information Sciences, vol. 9, no. 1, p. 30, Dec. 2019, doi: 10.1186/s13673-019-0192-7.

[29] M. Taheriyan, "Subject classification of research papers based on interrelationships analysis," in *Proceedings of the 2011 workshop on Knowledge discovery, modeling and simulation*, Aug. 2011, pp. 39–44. doi: 10.1145/2023568.2023579.

[30] Z. Xu, L. Liu, W. Song, and C. Du, "Text genre classification research," in *2017 International Conference on Computer, Information and Telecommunication Systems (CITS)*, Jul. 2017, pp. 175–178. doi: 10.1109/CITS.2017.8035329.

[31] A. B. Adetunji, J. P. Oguntoye, O. D. Fenwa, and N. O. Akande, "Web Document Classification Using Naïve Bayes," Journal of Advances in Mathematics and Computer Science, vol. 29, no. 6, pp. 1–11, Dec. 2018, doi: 10.9734/jamcs/2018/34128.

[32] N. Fei and Y. Zhang, "Movie genre classification using TF-IDF and SVM," in *Proceedings of the 2019 7th International Conference on Information Technology: IoT and Smart City*, Dec. 2019, pp. 131–136. doi: 10.1145/3377170.3377234.

[33] A. Nyberg, "Classifying movie genres by analyzing text reviews," Feb. 2018, [Online]. Available: http://arxiv.org/abs/1802.05322.

[34] A. Pal, A. Barigidad, and A. Mustafi, "Identifying movie genre compositions using neural networks and introducing GenRec-a recommender system based on audience genre perception," in *2020 5th International Conference on Computing, Communication and Security (ICCCS)*, Oct. 2020, pp. 1–7. doi: 10.1109/ICCCS49678.2020.9276893.

[35] G. Portolese and V. D. Feltrim, "On the Use of Synopsis-based Features for Film Genre Classification," in *Anais do XV Encontro Nacional de Inteligência Artificial e Computacional*, Oct. 2018, pp. 892–902. [Online]. Available: https://sol.sbc.org.br/index.php/eniac/article/download/4476/4400/

[36] A. M. Ertugrul and P. Karagoz, "Movie Genre Classification from Plot Summaries Using Bidirectional LSTM," in *2018 IEEE 12th International Conference on Semantic Computing (ICSC)*, Jan. 2018, vol. 2018-January, pp. 248–251. doi: 10.1109/ICSC.2018.00043.

[37] S. Meyer Zu Eissen and B. Stein, "Genre Classification of Web Pages," in *Advances in Artificial Intelligence, 27th Annual German Conference on AI*, 2004. doi: https://doi.org/10.1007/978-3-540-30221-6_20.

[38] G. Dong, W. Zhang, R. Yadav, X. Mu, and Z. Zhou, "OWGC-HMC: An Online Web Genre Classification Model Based on Hierarchical Multilabel Classification," Security and Communication Networks, vol. 2022, pp. 1–9, Mar. 2022, doi: 10.1155/2022/7549880.

[39] F. Wu et al., "Semi-supervised multi-view graph convolutional networks with application to webpage classification," Information Sciences, vol. 591, pp. 142–154, 2022, doi: 10.1016/j.ins.2022.01.013.