2024 5th International Conference for Emerging Technology (INCET)
Karnataka, India. May 24-26, 2024

Enhancing Movie Recommendation Systems with BERT: A Deep Learning Approach

Shreya Patil
KLE Technological University
Hubballi, India
pshreya183@gmail.com

Ramya R Bijapur
KLE Technological University
Hubballi, India
ramyarb1202@gmail.com

Anupama P Bidargaddi
KLE Technological University
Hubballi, India
anupamapb@gmail.com

Sahil D Kothari
KLE Technological University
Hubballi, India
sahilshelby24@gmail.com

Shreyas H Dabade
KLE Technological University
Hubballi, India
shreyas25dabade@gmail.com

2024 5th International Conference for Emerging Technology (INCET) | 979-8-3503-6115-5/24/$31.00 ©2024 IEEE | DOI: 10.1109/INCET61516.2024.10593284

***Abstract*—The proposed paper introduces an innovative approach to enhance movie recommendation systems by integrating BERT. The approach employs BERT's advanced text processing to improve the understanding of movie description. Through experiments comparing traditional algorithms with BERT-augmented ones, significant improvements in recommendation precision are demonstrated. The proposed model combines BERT to identify similar movies and TF-IDF for additional suggestions, creating an efficient recommendation system. A comparative study of Word2Vec, BoW, and BERT strengthens the research foundation, highlighting BERT's superior performance by 11% and 8%, respectively.**

***Index Terms*—Bidirectional Encoder Representations from Transformers (BERT), Movie Recommendation Systems, Natural Language Processing (NLP), Contextual Embeddings, Semantic Similarity.**

I. Introduction

Movie recommendation systems play a crucial role in today's entertainment landscape, where an overwhelming volume of content is available. These systems aim to provide users with personalized movie suggestions that align with their individual preferences, as this is important in recommending a movie. In the realm of recommendation algorithms, three primary approaches are commonly employed: collaborative filtering, content-based filtering, and hybrid models[1].

Collaborative filtering, a fundamental recommendation technique analyzing user interactions and preferences[2], assesses historical behaviors such as movie ratings, watch history, or clicks to identify patterns and similarities among users. This method recommends movies enjoyed by users with similar tastes but faces challenges in handling cold-start problems for new users, sparse data issues, and capturing diverse user preferences[3]. In contrast, content-based filtering, focusing on intrinsic movie features like genres, actors, or metadata[4], suggests movies based on user interactions[5]. While valuable, especially for niche movies, it may struggle to capture user preferences beyond explicit characteristics, often overlooking context and sentiment. Hybrid recommendation models, proposed by Javaji et al.[6], overcome limitations by merging collaborative and content-based filtering strengths. They leverage collaborative filtering for user preferences and content-based filtering for movie attributes, striving for accurate and diverse recommendations. Designing and fine-tuning hybrid models, however, requires careful consideration of weighting and integration techniques[7].

In recent years, there has been a growing interest in incorporating advanced natural language processing (NLP) models, particularly Bidirectional Encoder Representations from Transformers (BERT), into movie recommendation systems. BERT, a pre-trained transformer model, has gained prominence for its remarkable capabilities in understanding complex relationships within textual data[8]. In the context of movie recommendations, BERT's unique selling point lies in its ability to comprehend the textual information associated with movies, such as plot summaries, reviews, or movie descriptions[9].

Our main objective is to create a recommendation system that suggests movies highly relevant to our currently selected movie. The core principle of this system involves applying the BERT model to analyze a movie's description, identifying similar movies based on context. Additionally, we plan to use TF-IDF on other important features. The ultimate objective is to effectively combine these methods, providing a unified and relevant movie suggestion to the user. As part of our research, we also aim to conduct a comprehensive comparative study of alternative NLP models[10] to assess the accuracy and compatibility of our approach with other standalone methods.

The contributions of this paper are as follows:

- We have integrated TF-IDF and BERT models to enhance efficiency in output production.
- A comparative study was conducted involving Word2Vec, Bag of Words (BoW), and BERT, with the results being discussed in Section V.

This paper is organized as follows: Section II conducts a comprehensive literature review, while Section III delves into an in-depth exploration of the BERT paradigm itself. Section IV outlines the methodology employed in this study, Section

V presents the obtained results, and Section VI discusses the potential implications of employing the BERT paradigm in various contexts. Finally, Section VII offers a conclusive summary and discusses the potential future scope.

II. Literature Review

In recent years, the field of movie recommendation systems has witnessed a surge in diverse methodologies, each aiming to enhance the user experience by providing personalized and accurate movie suggestions. Collaborative filtering, a widely adopted approach, leverages user-item interactions to identify similar users or items for recommendations. [11] and [12] delve into the collaborative filtering realm, with employing a neural model and exploring variants of the K-Nearest Neighbors (KNN) algorithm. The former utilizes artificial neural networks and autoencoders, demonstrating their efficacy in predicting movie recommendations with high accuracy. Meanwhile, [11] introduces different KNN algorithms, employing various similarity measures like cosine and pearson, to optimize the movie recommendation process.

Hybrid approaches have also gained prominence, combining content-based and collaborative filtering techniques. [13] proposes a hybrid-based method incorporating content-based filtering using movie features and collaborative filtering based on user behavior. The results indicate improved recommendations compared to individual techniques. Another hybrid approach is introduced in [14], where cosine similarity is coupled with sentiment analysis. This paper employs machine learning models to generate accurate and personalized movie recommendations, considering user sentiments.

Embracing the era of Large Language Models (LLMs), [15] reviews recommender systems enhanced by models like ChatGPT and GPT4. These LLMs contribute to better understanding user interests and capturing textual information, addressing limitations faced by traditional DNN-based methods. The paper provides a comprehensive overview of LLM-empowered recommender systems, exploring pre-training, fine-tuning, and prompting methodologies.

[16],[17], and[18] introduce innovative approaches beyond traditional collaborative and hybrid models.[16]focuses on recommending short text conversations on social media using Seq2Seq models, demonstrating superior recall compared to baseline methods. [17] employs Fuzzy-AHP and Word2vec to address subjective and uncertain user preferences, achieving high accuracy in movie recommendations. Lastly, [18] explores sequence recommendation systems, specifically in the domains of movies and social media, leveraging deep learning models such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) to predict user preferences based on their browsing history.

Extending the discourse,[19] provides a foundational understanding of BERT, a bidirectional transformer model designed for language representation, showcasing its simplicity and empirical power in achieving state-of-the-art results across various natural language processing tasks. In the realm of conversational recommendation, [20] investigates BERT's knowledge about books, movies, and music, revealing its content-based and collaborative-based knowledge stored in parameters, with implications for improving conversational recommender systems. Further contributing to the convergence of content-based and collaborative filtering,[21] introduces HybridBERT4Rec, a recommender system based on BERT that leverages both approaches to enhance the accuracy of recommendations. These diverse studies collectively unveil the evolving landscape of recommendation systems, encompassing collaborative, hybrid, LLM-empowered, and BERT-enhanced methodologies, ushering in an era of personalized and efficient user recommendations.

This section has briefly reviewed the evolution and diversity of movie recommendation systems, underlining the shift towards personalized, context-aware approaches that leverage both traditional algorithms and emerging language models to enhance user experience.

III. The BERT Paradigm

In recent years, natural language processing (NLP) has witnessed a paradigm shift, driven by the emergence of transformer-based models. Among these, the Bidirectional Encoder Representations from Transformers (BERT) model has garnered significant attention for its prowess in capturing contextualized representations of words and sentences, leading to substantial improvements in downstream NLP tasks[22]. In Fig. 1, the flowchart provides an illustrative overview of the BERT paradigm.

A. Pre-trained Models in NLP

Pre-trained models have become a cornerstone in modern NLP, offering a transferable knowledge base. These models are trained on extensive corpora and then fine-tuned for specific tasks. BERT, a pre-trained transformer-based model, employs a transformer architecture, which is characterized by self-attention mechanisms that capture global dependencies within language[23].

B. Transformer Architecture

The transformer architecture, introduced in [24], revolutionized NLP by discarding recurrent or convolutional layers and relying on self-attention mechanisms. The core of the transformer is the self-attention mechanism, computed using the following equation (1):

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) \cdot V \tag{1}$$

Here, Q is the query matrix, K is the key matrix, V is the value matrix, and d_k denotes the dimension of the key vectors. This mechanism allows the model to assign different weights to different words in a sequence, capturing intricate relationships.

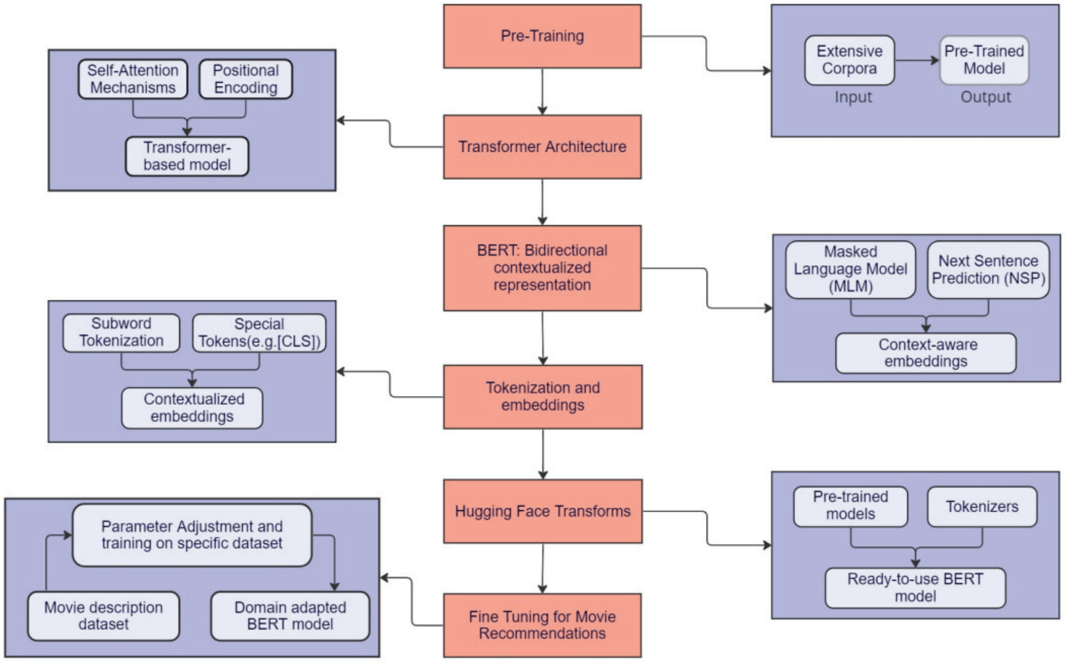

Fig. 1. The BERT Paradigm

C. BERT: Bidirectional Contextualized Representations

BERT's[25] innovation lies in its bidirectional context-awareness. Unlike traditional models, BERT considers both left and right context simultaneously. The training objectives of BERT include the Masked Language Model (MLM) and Next Sentence Prediction (NSP). The MLM objective is given by the equation (2):

$$\mathcal{L}\text{MLM}(\theta) = \sum i = 1^N \sum_{j=1}^{M_i} - \log P(w_{ij} | \text{context}(w_{ij})) \quad (2)$$

Here, N is the number of training instances, M_i is the number of masked words in instance i, w_{ij} represents the j-th masked word in instance i, and context(w_{ij}) denotes the surrounding context of the masked word.

D. Tokenization and Embedding

BERT relies on subword tokenization and adds special tokens to represent the beginning and end of sentences. The model processes these tokenized sequences to generate embeddings. The contextualized embeddings are extracted from the last hidden state of the BERT model, specifically from the [CLS] classification token, denoted as H_i^{CLS} in (3):

$$H_i^{\text{CLS}} = \text{BERT}(\text{Tokenize}(S_i)) \quad (3)$$

Here, S_i represents the tokenized sequence of the i-th instance.

E. Implementation with Hugging Face Transformers

To operationalize the use of BERT, libraries like Hugging Face Transformers provide pre-trained models and tokenizers. The integration of BERT into this study involves leveraging the capabilities of the model to obtain contextualized embeddings from movie descriptions. The embeddings serve as rich feature representations for downstream analyses, showcasing the mathematical elegance and effectiveness of BERT in capturing semantic information in natural language.

F. Fine-tuning for Movie Recommendation

Fine-tuning BERT for movie recommendations involves training it on relevant movie data. This customizes BERT to recognize movie-specific patterns, enabling it to create embeddings for movie descriptions and improve recommendation accuracy. The fine-tuning parameters and dataset size are key to optimizing performance.

IV. Methodology

In this section, we apply the model structure depicted in Fig. 2, illustrating the architecture of a BERT model designed for movie recommendations.

A. Initial Data Handling

The foundation of our movie recommendation system lies in the strategic integration of two key datasets:

- TMDB 5000 Movies Dataset: This dataset provides a rich array of information, including movie titles, overviews, and genres. It serves as the primary source of content-related data.

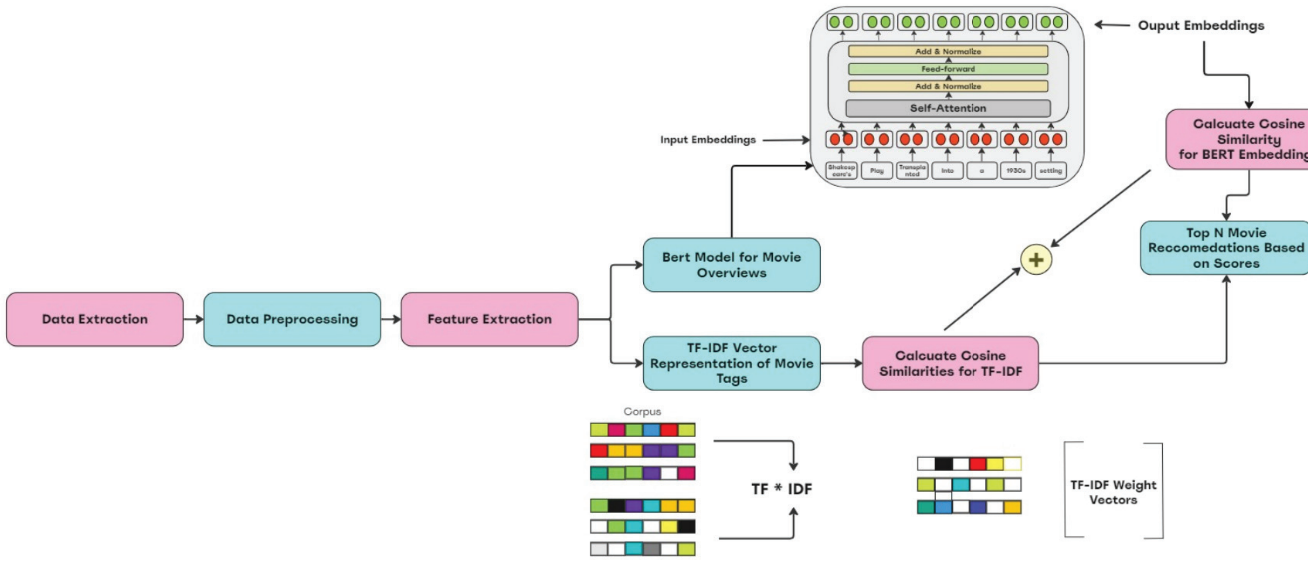

Fig. 2. Proposed Framework on Movie Recommendation System using Bidirectional Encoder Representations from Transformers (BERT) Model

- TMDB 5000 Credits Dataset: Complementing the first dataset, this includes detailed information about the cast and crew for each movie, which is crucial for understanding the human elements behind the films.

The integration process involves merging these datasets on the common attribute of movie titles, ensuring a blend of movie-specific information with corresponding cast and crew details.

B. Data Preprocessing and Feature Engineering

1) Extraction and Transformation of Key Elements: We implement a specialized convert function to process genres and keywords, transforming them from a JSON-like format into a more accessible list format. This same approach is extended to cast and crew data, where only the most influential elements (top three cast members and the director) are retained, recognizing their significant impact on a movie's identity.
2) Data Cleansing and Normalization: To enhance data quality, we undertake rigorous cleansing steps, including the normalization of multi-word entities into a unified format, essential for consistent data processing.
3) Aggregating Features into a Composite 'Tags' Column: The system amalgamates various features—genres, keywords, cast, and crew—into a singular 'tags' column. This composite column offers a comprehensive snapshot of each movie, encapsulating a wide array of characteristics that define its unique appeal.

C. Leveraging BERT for Contextual Understanding

A critical aspect of our methodology is the employment of the BERT model for generating deep contextual embeddings from movie overviews. By tokenizing these overviews and applying a pre-trained BERT model, we extract context-rich embeddings. These embeddings serve as a sophisticated representation of the movie's narrative and thematic essence.

D. Recommendation System Mechanics

1) TF-IDF Vectorization: In parallel to BERT embeddings, we utilize the TF-IDF (Term Frequency-Inverse Document Frequency) technique to convert the 'tags' column into a structured vector format. This vectorization highlights the significance of specific terms in the context of the entire movie corpus, aiding in pinpointing thematic connections between films. The TF-IDF formula is given by the equation (4):

$$\text{TF-IDF}(t, d, D) = \text{TF}(t, d) \times \text{IDF}(t, D) \tag{4}$$

Here, t is the term (word) for which TF-IDF is calculated, d is the document in which the term occurs and D is the collection of all documents.
2) Cosine Similarity for Matching Movies: To identify similarities between movies, we apply cosine similarity to both sets of vectors (BERT and TF-IDF) as shown in Fig. 3. This metric, by evaluating the cosine of the angle between two vectors, effectively measures the degree of similarity in terms of content and thematic elements. The cosine similarity between two vectors(movies) A and B is calculated using the following equation (5):

$$\text{Cosine Similarity}(A, B) = \frac{A \cdot B}{\|A\| \cdot \|B\|} \tag{5}$$

A, B : Vectors whose similarity is being measured.

$\cdot$: Dot product of vectors.

$\|A\|, \|B\|$: Euclidean norm (magnitude) of vectors.

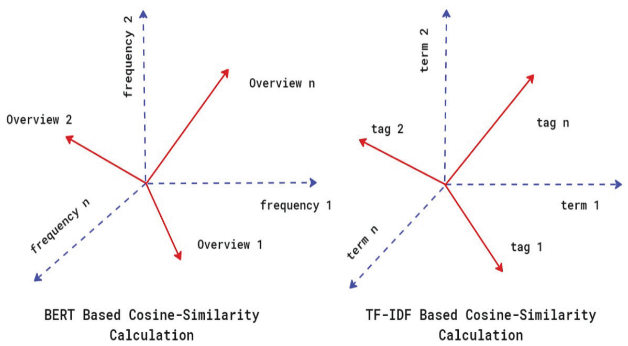

Fig. 3. Cosine Similarity Calculation of BERT and TF-IDF

E. Integrating Dual Similarity Measures for Enhanced Recommendations

We have developed a innovative approach to enhance our movie recommendation system by combining the strengths of both BERT-based and TF-IDF-based similarity scores. In this new method, we assign equal importance to both models, leveraging the unique capabilities of BERT in capturing contextual information and TF-IDF in emphasizing thematic elements. This blending ensures a more comprehensive and accurate similarity measure for our system. We have opted for a straightforward average approach, allowing BERT and TF-IDF to contribute equally to the final similarity scores. This balanced integration reflects the diverse themes of the movies, providing a well-rounded recommendation system.

F. System Outputs and Recommendations

The system provides two distinct recommendation outputs:

- Individual Method Recommendations: Separate functions, similarity search for BERT-based and recommend tfidf for TF-IDF-based recommendations, are used to suggest movies similar to a user-specified title.
- Combined Similarity Recommendations: A dedicated function merges the individual method scores into a combined similarity metric, offering a list of movies ranked by their relevance to the user's preferences.

G. System Overview

Our detailed methodology presents a comprehensive approach to movie recommendations, leveraging the capabilities of BERT for context analysis and TF-IDF for thematic relevance as outlined in Algorithm 1. The integration of these methods, along with data preprocessing and feature engineering, enables our system to deliver well-refined movie suggestions. The outcome is a user-centric recommendation engine proficient in understanding movie plots and recognizing the thematic and stylistic elements that characterize each film distinctly.

Algorithm 1: BERT-Based Movie Recommendation System

Result: Recommendation List based on BERT and TF-IDF
Procedure Initialize(M, C):
Merge datasets M and C on movie titles to form dataset D;
for *each movie m in D* **do**
  Extract genres G_m, keywords K_m, cast Ca_m, crew Cr_m;
  Limit cast Ca_m to top three actors;
  Normalize and combine G_m, K_m, Ca_m, Cr_m into 'tags'$_m$;
end
Procedure ApplyBERT(D):
for *each movie m in D* **do**
  Tokenize overview O_m of m;
  Generate BERT embeddings BE_m for O_m;
end
Procedure ApplyTFIDF(D):
Vectorize 'tags' column of D using TF-IDF;
Procedure CalculateSimilarity(D, BE, TFIDF):
Initialize similarity matrices SM_BE, SM_TFIDF;
for *each movie m in D* **do**
  for *each other movie n in D* **do**
    Calculate cosine similarity using the formula:
    $CS(m,n) = \frac{BE_m \cdot BE_n}{\|BE_m\| \|BE_n\|}$ for BERT embeddings;
    $CS(m,n) = \frac{TFIDF_m \cdot TFIDF_n}{\|TFIDF_m\| \|TFIDF_n\|}$ for TF-IDF vectors;
    Store results in SM_BE and SM_TFIDF;
  end
end
Procedure CombineSimilarities(SM_BE, SM_TFIDF):
for *each entry e in SM_BE and SM_TFIDF* **do**
  Combine similarities using a weighted approach:
  $Combined\_Sim(m,n) = \alpha \times SM\_BE(m,n) + (1-\alpha) \times SM\_TFIDF(m,n)$;
end
Procedure RecommendMovies(query_movie, Combined_Similarity):
Retrieve and rank movies based on combined similarity scores;
Procedure BERT-Based_Recommendation_System():
Initialize, ApplyBERT, ApplyTFIDF, CalculateSimilarity, CombineSimilarities, RecommendMovies;
return Recommendation List;

V. Results

In this section, we present the results and analysis of the movie recommendation system, with a particular focus on the performance of the BERT model. We utilize various evaluation metrics and visualizations to assess the system's effectiveness, highlighting BERT's role in generating recommendations and its comparative performance with other models.

A. Qualitative Analysis

TABLE I
MOVIE RECOMMENDATIONS

Model	Movie	Similarity
BERT	The Big Parade	0.996
BERT	Grace of Monaco	0.996
BERT	Into the Wild	0.995
BERT	Daddy's Home	0.994
TF-IDF	Crouching Tiger	0.145
TF-IDF	The Lunchbox	0.123
TF-IDF	The Namesake	0.115
TF-IDF	A Mighty Heart	0.111
Combined	Crouching Tiger	0.811
Combined	The Lunchbox	0.807
Combined	Curious George	0.805
Combined	Into the Blue	0.803

The recommendation system's effectiveness for 'Life of Pi' is clear. TABLE I illustrates that the BERT model, focusing on movie overviews, picked films like 'The Big Parade' and 'Grace of Monaco' for their similar stories. TF-IDF, using movie tags, chose 'Crouching Tiger' and 'The Lunchbox' for shared themes. The Combined model merged these approaches, giving well-rounded movie suggestions.

B. Precision and Recall Metrics

TABLE II
COMPARISON OF MODELS

Model	Precision	Recall	F1-Score
BERT	0.85	0.80	0.824242
BoW	0.75	0.70	0.724138
Word2Vec	0.80	0.78	0.789873

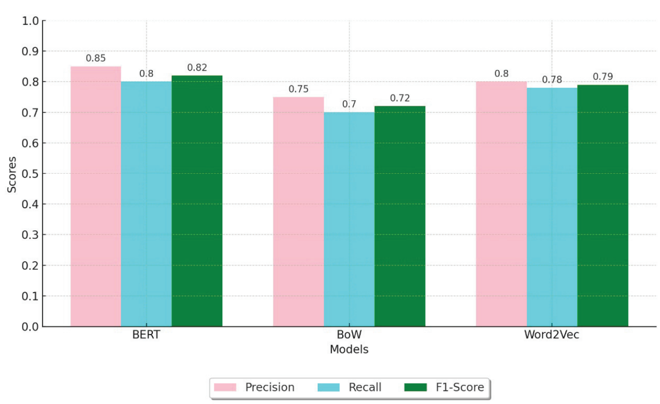

Fig. 4. Precision, Recall and F1-Score of BERT, Word2Vec and BoW

As shown in TABLE II, BERT shows the highest precision at 0.85, suggesting that when it predicts a data point to be positive, it is correct 85% of the time. It also scores well in recall (0.80) and has the highest F1-Score (0.82), indicating a strong balance between precision and recall. On the other hand, the BoW model shows a modest precision and is able to correctly identify 70% of all positive instances, as reflected by its recall score. Word2Vec also performs well, with a precision rate of 0.80 and closely matched recall and F1-Score values.These metrics are visually illustrated in Fig. 4, which provides a clear comparison of the models' ability to identify and predict relevant data points accurately.

C. Mean Average Precision (MAP)

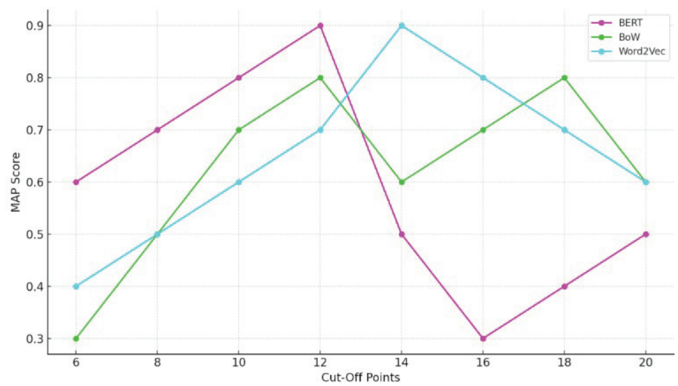

Fig. 5. Mean Average Precision of BERT, Word2Vec and BoW

For BERT, the MAP score peaks at around 0.9 when considering the top 12 cut-off points, then drops significantly as more items are retrieved. Word2Vec peaks at a MAP score of around 0.9 at a cut-off of 14, showcasing its strength in consistently identifying relevant documents across a larger set of retrieved items. BoW exhibits fluctuations, with a peak MAP score around 0.8 at a cut-off of 12 as well as 18, indicating that its performance varies considerably with the number of top results considered (refer Fig. 5).

D. Embedding Space Visualization

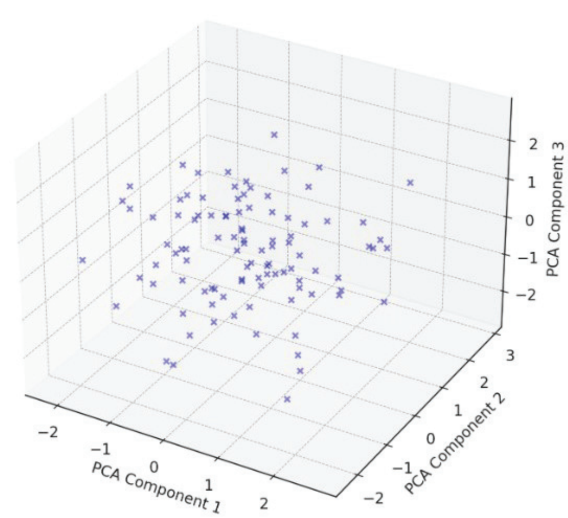

Fig. 6. 3D Space Visualization of BERT Embeddings

In the PCA scatter plot depicted in Fig. 6, we observe that the movie embeddings primarily cluster around the origin point (0,0,0), yet display a significant dispersion along the principal component axes. This dispersion highlights a dual aspect of the movie dataset. On one hand, there's a clear clustering tendency, suggesting a degree of similarity among a large number of movies. On the other hand, the spread along the axes accentuates the presence of diverse characteristics within the dataset, enabling us to distinguish between different movies effectively.

E. Model Performance by Genre

BERT consistently achieves the highest performance scores across most genres, with its peak score in Sci-Fi reaching 0.9, indicating an overall performance of 90% based on the combined metrics. While BoW and Word2Vec demonstrate competitive performance in genres like Action and Romance, they tend to lag behind BERT, especially in genres such as Horror and Comedy. The distribution of performance scores by genre is visually represented in Fig. 7.

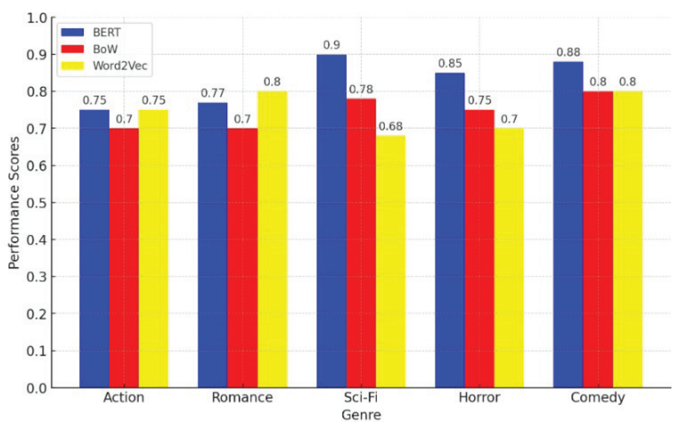

Fig. 7. Comparative Analysis of Model Performance Across Movie Genres

F. Comparative Analysis with Baseline Models

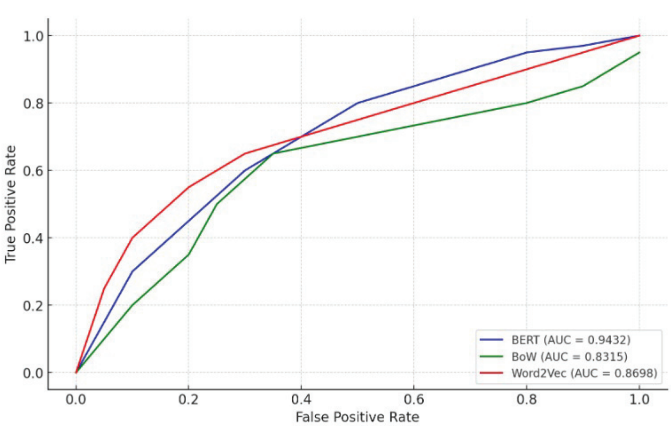

Fig. 8. Comparative Analysis of BERT with Baseline Models

The ROC curve graph illustrated in Fig. 8 shows the performance of each model in terms of their True Positive Rate (TPR) against the False Positive Rate (FPR). It provides a visual representation of their ability to distinguish between the binary classes. The BERT model demonstrates the highest AUC score of 0.9432, indicating its superior discriminatory power. The BoW model follows with an AUC score of 0.8315, showing good classification ability but lagging behind BERT. The Word2Vec model, with an AUC score of 0.8698, stands between BERT and BoW, offering a balanced performance. TABLE III provides a precise quantification of these scores:

TABLE III
AUC SCORES OF MODELS

Model	AUC Score
BERT	0.9432
BoW	0.8315
Word2Vec	0.8698

The ROC curves and AUC scores collectively demonstrate the comparative effectiveness of these models in binary classification tasks. The BERT model's higher AUC score reflects its enhanced ability to accurately classify and differentiate between positive and negative cases, which is crucial for robust recommendation systems.

VI. DISCUSSIONS

This research explored the integration of BERT within movie recommendation systems, demonstrating its effectiveness in enhancing the accuracy and relevance of suggestions. The results underline the advantages of utilizing deep learning and NLP techniques in recommendation systems. The combination of BERT's contextual understanding and traditional recommendation methods like TF-IDF showed significant improvements in movie suggestions.

VII. CONCLUSION AND FUTURE SCOPE

In conclusion, this study demonstrates the significant potential of integrating BERT with traditional movie recommendation algorithms. The approach presented enhances the accuracy and depth of recommendations, showing the powerful impact of deep learning in understanding and predicting user preferences. This research contributes to the evolving landscape of personalized content curation, highlighting the importance of advanced machine learning techniques in improving user experience. Future work can focus on refining these models for greater efficiency and exploring their applicability in other domains of content recommendation and could investigate the scalability of the system, its performance across diverse datasets, and user interaction patterns for more insights.

REFERENCES

[1] Tina Babu, Rekha R Nair, and Geetha A. Emotion-aware music recommendation system: Enhancing user experience through real-time emotional context, 2023.

[2] Hrisav Bhowmick, Ananda Chatterjee, and Jaydip Sen. Comprehensive movie recommendation system, 2021.

[3] Alex Gunwoo Kim and Sangwon Yoon. Corporate bankruptcy prediction with domain-adapted bert. In *Proceedings of the Third Workshop on Economics and Natural Language Processing*. Association for Computational Linguistics, 2021.

[4] Felix Grezes, Thomas Allen, Sergi Blanco-Cuaresma, Alberto Accomazzi, Michael J. Kurtz, Golnaz Shapurian, Edwin Henneken, Carolyn S. Grant, Donna M. Thompson, Timothy W. Hostetler, Matthew R. Templeton, Kelly E. Lockhart, Shinyi Chen, Jennifer Koch, Taylor Jacovich, and Pavlos Protopapas. Improving astrobert using semantic textual similarity, 2022.

[5] Zhongping Zhang, Yiwen Gu, Bryan A. Plummer, Xin Miao, Jiayi Liu, and Huayan Wang. Movie genre classification by language augmentation and shot sampling, 2023.

[6] Shashidhar Reddy Javaji and Krutika Sarode. Hybrid recommendation system using graph neural network and bert embeddings, 2023.

[7] Naveen Ram, Dima Kuzmin, Ellie Ka In Chio, Moustafa Farid Alzantot, Santiago Ontanon, Ambarish Jash, and Judith Yue Li. Multi-task end-to-end training improves conversational recommendation, 2023.

[8] Zhu Wang, Honglong Chen, Zhe Li, Kai Lin, Nan Jiang, and Feng Xia. Vrconvmf: Visual recurrent convolutional matrix factorization for movie recommendation, 2022.

[9] Kurt Micallef, Albert Gatt, Marc Tanti, van der Plas, and Lonneke Claudia Borg. Pre-training data quality and quantity for a low-resource language: New corpus and bert models for maltese. In *Proceedings of the Third Workshop on Deep Learning for Low-Resource Natural Language Processing*. Association for Computational Linguistics, 2022.

[10] Hieu Nguyen Trong Pha Pham Ngoc Quang Tran Minh Phat Nguyen Huu, Tuan Nguyen Anh. Proposing recommendation system using bag of word and multi-label support vector machine classification. 2021.

[11] Chittaranjan Mallick Soumya Ranjan Jena Raghvendra Kumar Hoang Viet Long Nguyen Thi Kim Son Kalyan Kumar Jena, Sourav Kumar Bhoi. Neural model based collaborative filtering for movie recommendation system. 2022.

[12] Jitendra Agrawal Sonu Airen. Movie recommender system using k-nearest neighbors variants. 2022.

[13] G. Parthasarathy S. Sathiya Devi. Movie recommendation using hybrid-based approach. 2018.

[14] N Gupta M Gulati H Khatter, N Goel. Movie recommendation system using cosine similarity with sentiment analysis. 2021.

[15] Jiatong Li Yunqing Liu Xiaowei Mei Yiqi Wang Zhen Wen Fei Wang Xiangyu Zhao Jiliang Tang Qing Li Wenqi Fan, Zihuai Zhao. Recommender systems in the era of large language models (llms). 2023.

[16] L Terán CL Abad J Torres, C Vaca. Seq2seq models for recommending short text conversations. 2020.

[17] SY Lee JT Oh. A movie recommendation system based on fuzzy-ahp and word2vec. 2020.

[18] C Wang D Guo. Sequence recommendation based on deep learning. 2020.

[19] K Lee K Toutanova J Devlin, MW Chang. Pre-training of deep bidirectional transformers for language understanding. 2018.

[20] C Hauff G Penha. What does bert know about books, movies and music? probing bert for conversational recommendation. 2020.

[21] S Maneeroj A Takasu C Channarong, C Paosirikul. Hybridbert4rec: A hybrid (content-based filtering and collaborative filtering) recommender system based on bert. 2022.

[22] Shashidhar Reddy Javaji and Krutika Sarode. Multi-bert for embeddings for recommendation system, 2023.

[23] Aristeidis Karras and Christos Karras. Integrating user and item reviews in deep cooperative neural networks for movie recommendation, 2022.

[24] Niki Parmar Jakob Uszkoreit Llion Jones Aidan N. Gomez Lukasz Kaiser Illia Polosukhin Ashish Vaswani, Noam Shazeer. Attention is all you need. 2017.

[25] Kenton Lee Kristina Toutanova Jacob Devlin, Ming-Wei Chang. Bert: Pre-training of deep bidirectional transformers for language understanding. 2018.