



OPEN Improving personalized recommendations system using graph attention networks driven by perceived complexity and innovation

Shoukat Ullah¹, Aurangzeb Khan¹, Khair Ullah Khan¹, Amanullah², Rehan Tariq Chohan³✉ & Muhammad Nawaz Khan⁴✉

This paper presents a product recommendation system (GAT-RS) based on perceived complexity and perceived innovation in product reviews. Perceived complexity refers to the usability of a product, while perceived innovation refers to the extent to which something looks novel. Such aspects significantly impact consumer dynamics and business performance indicators, including user engagement and sales results. In this study, product reviews are manually annotated, and then Explainable AI (XAI) is used to improve the decision-making process of the proposed GAT-RS model. The proposed model used pre-trained SimCSE embeddings to find high-quality textual representations of product reviews. It also used Graph Attention Networks (GAT) to discover the associations between the attributes of products and the perceptions of customers about complexity and innovation. The SMOTE oversampling on classes and loss class weights functions are used to handle the imbalance between reviews during training. The evaluation of the proposed GAT-RS is done on accuracy, precision, recall, F1 score, ROC AUC, and the system was found to have a higher accuracy of 94.61% and a ROC AUC of 98.94% compared to the baseline approaches. A combination of complexity and innovation will enhance user satisfaction by aligning recommendations with preferred styles of cognition and novelty. The offered solution would also strengthen the accuracy of personalized recommendations based on customer interests.

Keywords GAT-RS, XAI, SimCSE, SMOTE, Product complexity, Product innovation

Digital era has completely changed the way businesses interact with customers, using informed communication to build trust and provide reliable services. Customers expect personalized, reliable, and timely information about products and services. To meet these requirements, they focus on effective advertising with some automated tools to provide personalized services^{1,2}. These tools analyze customer behavior by noting customer preferences and purchasing orders. Based on these patterns in selecting and purchasing, the system attractively provides personalized products. This system also helps businesses make rational and viable decisions about new product launches based on analyzing customer behavior^{3,4}. Product recommendation systems are increasingly being used as e-Commerce sites implement product suggestions based on their individual attributes, or history of browsing, among other similar preferences^{5,6}. The developments provide the customer with the knowledge of what best suits them, and thus, the purchasing process becomes more personalized. They provide their help with order, payment, and customer services, which are regularly used on online shopping sites. Machine learning (ML) algorithms suggest products that satisfy the interests of the customer. Some of them are Spotify, Netflix, and YouTube; they are all affiliated with music, movies, and videos, respectively^{7,8}. Product recommendation systems that utilize Artificial Intelligence (AI) can play a crucial role in contemporary e-commerce, enhancing

¹University of Science and Technology, Bannu 28100, Pakistan. ²Department of Commerce Education and Management Sciences, Higher Education, Archives and Libraries Department, Peshawar, Khyber Pakhtunkhwa 25130, Pakistan. ³Faculty of Information Science and Technology, The National University of Malaysia, Qatar (UKM-Qatar Campus), 28000 Energy City, Qatar. ⁴Department of Smart Security, Gachon University, Seongnam 13120, Republic of Korea. ✉email: rehan.chohan@ukm.edu.qa; muhammadnawaz@gachon.ac.kr

customer experiences and driving sales. These systems can analyze consumer data to provide personal suggestions through the application of advanced machine learning algorithms⁹. Recommender systems help users make decisions by making recommendations for items or products using multiple types of active acquired data. The processing of data depends on the type of recommender system and the way it processes the data. The influence of social networks is enormous, as it provides valuable information to make recommendations for products, items, or other objects. The cloud platforms offer recommendations to users based on their interests, where Users share their views and perceptions of products. Such systems work admirably on websites such as Twitter and Facebook. The system relies on user and item-based recommendations, providing personalized recommendations based on sentiment data extracted from reviews^{10–12}.

Sentiment Analysis (SA) is an ML processing technique that aids in sentiment analysis, allowing business owners to gain insight into their customers' opinions through surveys, social media, and e-commerce website reviews, among other online media^{13,14}. SA has the potential to improve recommendations where the available rating data are sparse. Conventional recommendation algorithms often rely on minimal user ratings, which can be inadequate. In contrast, opinion-based ratings or ratings based on reviews, blogs, social media, and recommender systems can be more useful. The inclusion of sentiment can help combat data sparsity in recommender systems, thus improving the quality of recommendations^{15,16}. An Amazon product system based primarily on SA and ML algorithms has been used to classify and recommend products¹⁷. By analyzing product reviews in terms of quality and rating, the system was able to make custom product recommendations by analyzing the user's previous choices. The sentiments of the reviews were determined using Natural Language Processing (NLP) and divided into positive and negative. Incorporating reviews and ratings enhances the reliability of the system, and the Collaborative Algorithm further improves the recommendations for products. To recommend items to other users, the system calculated cosine similarity among the products using user ratings to form a cosine similarity-based matrix. In addition, customers who live in a digital environment use reviews to judge products, which can be challenging, as they often have many reviews but lack a helpful one. The machine learning-based filtering of reviews takes less time and suggests the corresponding products. The benefits of predicting user ratings based on text reviews include enhancing the consumer experience and helping businesses improve sales and products through a deeper understanding of customer needs. To develop a customer-centric recommender system, an SA and Opinion Mining (OM) are inevitable approaches used to analyze customer data¹⁰. SA resorts to text mining and NLP to achieve this, analyzing and retrieving positive, negative, or neutral sentiments from textual data. This method can aid in decision-making across various fields of work by categorizing the polarity of sentiments at multiple levels. In a recommender system, SA has been implemented as a decision support system, where the system makes more accurate recommendations based on the sentiment of user queries. Some adaptive websites are also used to address the domain of adaptive websites that employ an automated recommendation system¹⁸. This system delivers recommendations to users based on numerous robust methods, namely collaborative filtering, content-based filtering, and knowledge-based recommendation.

All of these methods are associated with several advantages and some inherent limitations. To provide a better recommender system based on analyzing customer data, we have integrated these techniques to form hybrid recommender systems.

Motivation

There is always a need for a recommendation system that provides customized recommendations in various areas, such as purchasing decisions, career guidance, social network connections, movie choices, school attendance, or news reading. Essentially, the proposed system is an individualized counseling system that provides personalized advice based on user interaction, reviews, ratings, and preferences. Improve the engagement between online customers and businesses. This paper presents an intelligent product recommendation system that involves blending two main factors: perceived complexity, which denotes ease of use or understanding of a product, and perceived innovation, which determines novelty. Such factors have a significant impact on consumer behavior, directly affecting user engagement and sales performance. The proposed system will employ pre-trained SimCSE embeddings to produce high-quality text representations of product reviews, utilizing Graph Attention Networks (GAT) to analyze the relationships between product attributes and customer perceptions of complexity and innovation. This combination of methods can personalize the recommendation, which eventually proves to be a more effective overall experience on the part of the customer and overall growth on the part of the business.

Main objectives

The key outputs of this research are as follows:

- We proposed a recommendation mechanism taking into account two key parameters, perceived complexity (usability of the product) and perceived innovation (novelty of the product). These are important parameters that help in shaping the performance of business and consumer behavior.
- The product reviews dataset is manually annotated so that the quality and correspondence of the data to be used to train the system is of excellent quality and consistency. This manual tagging enhances the accuracy of both sentiment analysis and aspect-based recommendations, significantly increasing the probability that product suggestions will coincide with product attractiveness and customer preferences. The way to succeed in training the system to work with trusted and executable data and get more accurate and individual recommendations lies in a careful labeling of the reviews.
- This research study used Explainable AI (XAI) approaches to help the ML model in decision-making processes. With XAI, the system can explain its recommendations in a transparent and comprehensible way. This allows users and companies to trust the suggestions as it reflects how various product attributes, customer moods, and perceived complexity or inventiveness will impact the final verdict.

- Pre-trained SimCSE embeddings are used in the system to provide a better representation of product reviews, and thus make better recommendations. It combines GAT to examine the connections between feature attributes and the customers' notions of simplicity and innovation. Moreover, SMOTE oversampling and class weight are employed to address class imbalance and ensure fairness and accuracy of training.
- To the best of our knowledge, the proposed model is the first to work on perceived complexity and innovation of products using a graph transformer model with an attention mechanism.

Key contributions

The main contributions of this study are as follows:

- The manual annotation of product reviews with respect to perceived complexity and perceived innovation.
- The proposed GAT-RS model combines the perceived complexity and perceived innovation from annotated reviews to better mirror the consumer behavior as personalized recommendations.
- XAI helps to increase the transparency of the decision-making process by making validation of manually annotated reviews.
- Pretrained SimCSE embeddings are used for quality textual representations and more good features of the product.
- GAT is applied for detecting complex structures between product attributes and consumer perception for enhancing personalized recommendations.
- SMOTE oversampling and class weight handling the issue of class imbalance during training can make the model robust and generalized.
- Evaluation and Performance: GAT-RS model shows better performance compared to the baseline models with an accuracy of 94.61%, proving its ability to deliver accurate and personalized recommendations.
- The combination of complexity and innovation has created better user satisfaction by matching the recommendations to the cognitive and novelty users prefer.

This paper is organized in sections: “[Related work](#)” is Related Work, “[Material and methods](#)” consists of methodology, dataset, and all other details, “[Experimental overview](#)” is about experimental setup, “[Results and discussions](#)” comprises results and discussions part of the paper, and finally, “[Conclusion and future work](#)” contains the conclusion and future work.

Related work

Social media's unstructured content makes typical data mining techniques labor-intensive and inefficient, and their inherent scale, noise, and bias further reduce their usefulness. In recent research papers, much attention has been paid to studying how customers express their feelings and opinions through online reviews in various areas, such as healthcare¹⁹, business²⁰, hotels²¹, academia²², etc. People can now easily express their ideas, comments, feedback, and blogs on various Internet platforms. Users can submit their opinions on the services or goods used on these open, free, and accessible sites. The businesses offering the services and products, as well as potential customers, profit from these reviews written by consumers.

*CREQ*²³ is a model that investigated the decline in Tokopedia online customers, which suggests customers are dissatisfied with their offerings. To evaluate the impact of online customer reviews and e-services on purchasing decisions, they performed a study with 278 participants. The study found that customer reviews and the caliber of e-services strongly influence consumers' decisions. Social media generates massive amounts of data, and ML enables us to manage and learn from it efficiently, bypassing manual analysis²⁴. *DL-SAA*²⁵ utilized SA to determine the match between Amazon reviews and their corresponding ratings. DL was used to detect the mismatches of sentiment and rating. Paragraph vectors were used to convert reviews into vectors, and these vectors were then used to train a gated recurrent unit of a Recurrent Neural Network (RNN). The model took into consideration the semantic relationships of review text and product information. A web service was also created to predict rating scores and provide feedback in cases where there is a mismatch between the predicted and submitted ratings. *MDL-TA*²⁶ analyzed Amazon reviews to categorize sentiment using NLP, ML, and DL methods. They tested a wide range of algorithms, including Naive Bayes (NB), Random Forest (RF), and BERT, by achieving an accuracy of 89% with BERT. The research guides consumers and businesses, enabling them to make informed choices that benefit them.

*SARS*²⁷ responded to the concerns of the traditional farmers; they proposed a product recommendation system of rice crop disease products. Agricultural website data was used for clustering, and the output applied K-means clustering to classify the products. The labeled data was analyzed using the Support Vector Machine (SVM) Kernel classifier. The top 5 products suggested by the system are based on queries made by a user. The technology assists farmers in choosing the best products for rice crop diseases, hence improving farming productivity. *SA-DML*²⁸ has examined SA in e-commerce reviews, attempting to predict positive and negative sentiment in product reviews using ML methods, such as RF, with an accuracy rate of 91.90%. They also used a DL model, RNN-LSTM, which achieved the best accuracy of 97.52%. *RNN-SA*²⁹ has discussed the SA of the Amazon product reviews, which is essential to businesses and customers. He categorized reviews based on three Amazon datasets by means of an RNN model that had an accuracy of 85%, 70%, and 70% as well. To identify aspects and conduct SA in customer reviews, *AI-SA*³⁰ created a prediction pipeline based on pretrained BERT and T5 models. Their attention was focused on eco-friendly goods, and they achieved 92% accuracy using BERT, which outperformed T5. The aspect detection was identified as the best in the BERT model. Their research was helpful to product designers and developers in understanding how to match products with customer preferences. The authors proposed that the use of data analytics can help analyze online product reviews, which can be transformed into interesting insights for designers as well. This structure allows the designer to make

informed choices regarding customer demands, taking into consideration online reviews, design theory, and data analysis. A case study on Amazon product reviews proved the effectiveness of the framework. It brings together quantitative analysis and the creative process of design, highlighting the value of the computational approach in design³¹.

*DLML-SA*³² examined the SA of reviews on Amazon products, with a focus on the transformation from the traditional transaction environment to the online one. ML, DL, and transformer-based approaches such as SVM, CNN, and BERT were compared to determine whether a specific review was positive or negative. Results indicated that BERT was more accurate (95.34%) than CNN (88.41%) and SVM (87.28%). The authors extended the idea of using graphs. Another *RGCNs*³³ as a method for performing SA, enabling the capture of complex relationships between data points. They combined the RGCN architecture with pretrained models BERT and RoBERTa and used it on Amazon and Digikala product reviews. Their experiments demonstrated that RGCNs are effective in capturing relations and can enhance sentiment analysis. *SAPR*³⁴ analyzed customer reviews through SA and OM. They used ML to train on Amazon product reviews, and the accuracy rate of NB was 98.17% on Camera reviews. The findings bring out the excellence of ML methods in the classification of product reviews. *SA-RA*³⁵ utilized the NLP method, RoBERTa, to perform SA of Amazon product reviews. He tested how well the models performed in creating the sentiment scores, as well as investigated trends that were conforming to the principles in behavioural economics, such as eWOM and the typically experienced confirmation bias. He analyzed and visualized data, by providing a picture of consumer emotional reaction. The results emphasized the successful application of NLP models in consumer behaviour research, and his findings can be further utilized in marketing and decision making. The authors performed a sentence-based sentiment analysis of Amazon product reviews using a Bi-LSTM model and an attention mechanism. They evaluated the interpretability of the model considering attention weights on sentences and key aspect terms. The model predicted an accuracy of 96%, indicating that aspect terms were frequently discussed as highly as, or even more so than, sentimental words³⁶.

*ML-CSA*³⁷ suggested the use of ML algorithms to process and categorize the product and shop details in accordance with the customer reviews. The data sample has been gathered by the Unified Computing System (UCS), which gives the data about hardware and software administration. The findings demonstrate that the suggested system performs better than other methods since it has a MAPE of 96% and almost 98% accuracy. The mean absolute error (0.6) of the system reveals that it is quite effective in forecasting the product information and experiences of the customers. *DL-RsX*³⁸ reviewed the current studies on the DL approach to recommender systems and its ability to address issues of information overload and enhance web applications. They provided a discussion of the influence of DL in areas such as computer vision and NLP, emphasizing the quality of learning feature representations. *DL-MA*³⁹ has utilized SA and matrix factorization on Amazon Product reviews to address the data sparsity problem and credibility issues associated with collaborative filtering. Based on LDA, the system retrieved the user and item topic distributions of the reviews and incorporated BERT to measure the sentiment data. This sentiment information is added as an enhancement to the user item rating matrix, thus enhancing the accuracy of the recommendation system. To study the impact of customer opinions on their purchasing behaviors, the author investigated sentiment classification of Amazon reviews⁴⁰. They used the ML algorithms, including Logistic Regression (LR), RF, and BERT, along with Bag-of-Words, TF-IDF, and GloVe vectorization methods.

*ASIF*⁴¹ proposed a new framework of product recommendation on the web, which can study the sentiment of the customer successfully in the multi-modal data, i.e., utilizing text review and image review. The model contains the processing of data on its ingestion, normalization, extraction of features, and recommendations based on transfer learning. The baselines were compared with ASIF which was tested on five datasets and performed better with an accuracy of 95.95% and F-score of 94.95%. *ECFRS*⁴² is a new recommendation system that integrates SA and collaborative filtering to provide personalized recommendations. To enhance the classification, they employed unsupervised GloVe vectorization and a Bi-LSTM sentiment model. The system was more accurate than other models by achieving an impressive accuracy of 93%. In *PBRT*⁴³, the authors proposed a segregation method to extract each aspect and refine the aspects based on the concepts of similarity towards the enhancement of product quality. The current tools that address requirement management concern only syntactical integration, which restricts quality assurance. The observation of the experiment proved the efficiency of the suggested approach in enhancing precision and quality.

*HC-LSTM*⁴⁴ suggested a recommendation system based on sentiment analysis to enhance product recommendations. The system integrated a hybrid DL model (CNN-LSTM) with sentiment scores from customer reviews to integrate these insights and combine them with ratings of similar items from similar users. The method increased the accuracy of the recommendations. The Amazon food review dataset was then used to validate the system's performance, demonstrating the utility of sentiment analysis for personalized recommendations. *PR-ECNN*⁴⁵ has introduced a recommendation system based on DL that enhances product recommendations. Text preprocessing methods, including stemming, lemmatization, and stopword removal, were used to improve data interpretation. To do sentiment analysis, feature values were obtained through TF-IDF and modified under an improved CNN. As compared to traditional CNN models, the model, which was trained with GloVe and consisted of a skip-gram embedding layer, offered precision (93.64%), accuracy (96.92%) and a recall of 94.80%. *ELM-SA*⁴⁶ proposed a hybrid recommendation framework that combines sentiment analysis, item-based recommendation, and user-based recommendations to enhance the quality and robustness of suggestions. They employed feature extraction methods and handled class imbalance using SMOTE based on a proprietary dataset comprising Amazon, Tmdb, and Google reviews. The ensemble model, which incorporates the machine learning algorithm XGBoost, has shown an accuracy of 96%.

According to related research, it is observed that the present AI systems used in e-commerce platforms for product recommendations depend on fundamental SA while neglecting vital elements, including product

complexity and innovation perceptions. The huge amount of instant social media reviews raises difficulties in extracting valuable insights to create intelligent recommendations. Machine learning approaches that use conventional algorithms do not identify how user preferences interact with the scale of perceived complexity and innovation of products, thus restricting their recommendation precision. The proposed research solution uses a graph transformer with an attention mechanism and XAI functionality to improve the accuracy of addressing these knowledge gaps.

Problem statement

Despite advances in recommendation systems, existing methodologies do not take into account the integration of perceived complexity and innovation of products, which are the other determinants of user preferences. In contrast to traditional models, which focus on customer interactions and product characteristics, they overlook the cognitive load of complex products and the novelty of innovative ones. The lack of this in the literature inhibits the provision of really personal recommendations. Furthermore, no prior studies have analyzed how transformer-based models can be applied to describe some of these elements within a recommendation framework.

Material and methods

This section comprises data acquisition, data manual annotation, XAI annotation, data augmentation, text preprocessing, description of the proposed system, GAT-RS algorithm, schematic representation of data annotation using XAI and its algorithm XAI, SimCSE, SMOTE, and GAT.

Data acquisition

In this study, the proposed model GAT-RS is implemented in Google Colab and the Amazon product data set. The Kaggle data set⁴⁷ has 28332 reviews and 24 attributes, which are suitable for checking and analysis of models.

Data manual annotation

Human annotators labeled the dataset for SA purposes as per established guidelines and obtained 5473 reviews. The guidelines need to be clear so that high-quality labels can be achieved regarding perceived complexity and innovation, as annotated data functions as the main standard for training and evaluating models. In the proposed study, the product reviews were manually annotated, using Cohen's Kappa, to ensure consistency and reliability within the labeling process. Two annotators were involved in the manual annotation, Cohen's Kappa was used to determine the level of agreement between the two annotators. This metric was specifically selected, because it is widely used when measuring the inter-annotator reliability, particularly for categorical data. We have ensured that the annotation process was both reliable and consistent by computing Cohen's Kappa. This reliability analysis provides robustness to the annotated data which ensures that the labels incorporated for proposed model training were of high quality. Perceived complexity was annotated on some terms such as "complex", "difficult", "challenging", "intricate" and "advanced" whilst perceived innovation was explained using a set of keywords such as "innovative", "novel", "cutting-edge" and "unique". These keyword lists were further expanded using WordNet to include both synonym and semantic variants in an effort to capture as much of the linguistic expression that could mean complexity or innovation. In order to minimise subjectivity and maximise inter-annotator reliability we used a lexicon-based tool called SentiWordNet in order to quantify the sentiment polarity (positive, negative, neutral) of the keywords within the context of each review, thus ensuring that the sentiment is included in the complexity and innovation was suitably considered. Moreover, formalized decision boundaries were also put in place for decisioning the recommendation labels. For instance, reviews that were marked with suggested innovation and considered simple were marked 'Medium', and those that were considered complex were marked 'No', and reviews that were considered non-innovative were marked 'No', unless innovation was considered necessary for a recommendation. This approach was taken in order to minimize ambiguity and ensure reproducibility in the phase of annotation. The consistency of the labeling was also checked using Cohen's Kappa, which is a statistical measure of inter-annotator agreement and was calculated to make sure the annotations were consistent across annotators. These formally defined annotation rules, and the use of semantic similarity metrics to compare the review embeddings, like cosine-similarity, and manually inspecting a convention of the labelled reviews ensured the reliability and validity of the dataset.

XAI annotation

XAI techniques are mostly developed to be used to explain the prediction of models rather than used to validate the labels. However, in the present study, data labeling consideration plus annotation validation by XAI were observed to ensure the accuracy of manually annotated data (i.e., product reviews). XAI was used to interpret the relevance and quality of labels applied to each review, showing the most important features that are involved in the classification process. This helped us to outline any possible inconsistencies in the annotation process. While XAI helps in increasing the interpretability by giving insights about the decision-making process of the model, its role was never pretentious as a validation mechanism. Instead, it was made use of as a complimentary device to ensure that the labels that were utilized for the training were pertinent and precise so as to allow for post-hoc inspection of the data. We generated a base annotated dataset in a CSV file from a manually annotated dataset by using the XAI annotation, as shown in Table 1.

For instance, "This keyboard combines simple design with innovative features, making it easy to use and efficient for work. It has a clean interface that reduces complexity while still offering an advanced user experience." Tokenized words are 26, while Simple words count is 4, and Complex words count is 0. Similarly, Innovative words, and non-innovative words count are 3 and 0 respectively.

$$\text{Complexity Score} = (4-0) / 26 = 0.154$$

Review type	Complexity	Innovation	Suggested innovation	Recommendation
Neutral review	N/A	N/A	N/A	No
Complex review	Complex	Other	Suggested innovation	No
Simple review, innovative	Simple	Innovative	No suggested innovation	Yes
Simple review, non-innovative	Simple	Other	No suggested innovation	Medium
Suggested innovation, simple	Simple	Other	Suggested innovation	Medium
Suggested innovation, complex	Complex	Other	Suggested innovation	No
Complex, non-innovative	Complex	Other	No suggested innovation	No

Table 1. Review type classification and recommendations.

$$\text{Innovation Score} = (3-0) / 26 = 0.115$$

Let threshold $\theta = 0$

- Score > $\theta \rightarrow$ Positive class
- Score $\leq \theta \rightarrow$ Negative class

Hence,
Complexity: $0.154 > 0 \Rightarrow \text{Simple}$
Innovation: $0.115 > 0 \Rightarrow \text{Innovative}$
The review is mathematically classified as a simple and innovative tablet using word-based scoring, normalization, and thresholding.

Data augmenting product reviews using back-translation for perceived complexity and innovation

In this research study we used back-translation strategically for creating syntactic diversity by translating product reviews into French (then back into English) using the googletrans library. This kind of procedure generated 7782 synthetic reviews improving the dataset from 5473 to 13,255 reviews to combat the class imbalance and improve data diversity. To ensure the effectiveness of such augmentation, we performed manual inspection of a subset of augmented reviews to check for the semantic consistency. We also used cosine similarity metrics based on word embeddings to measure similarity between the original and back-translated reviews. An ablation study was conducted as well to prove the effectiveness of the augmented data in improving the model performance by showing that the syntactic diversity acquired by back-translation resulted in improved recommendations without significant loss of the semantic information. This way enhanced the generalization of the proposed model because it improved diversity within the data and solved class imbalance.

Text preprocessing

Text preprocessing is included and uses the procedure applied in *E-BERT*⁴⁸. The process of cleaning and normalizing textual data by removing stopwords, punctuation, special characters and applying tokenization, stemming, and lemmatization to enhance text quality for analysis. It performs these functions in the following sub-procedures.

Tokenization
NLP relies heavily on tokenization because the technique transforms free-form text into structured data suitable for analysis. Tokenization is a process to make text data manipulation more accessible through the practical breakdown of texts into tokens, which supports sentiment analysis and machine learning functions⁴⁹. We have used the same tokenization method in our approach.

Conversion to lower case
Lowercase text data allows the analysis to detect and normalize words that may appear in different cases (“Good” and “good”)⁵⁰. The normalization step makes text more suitable for sentiment classification alongside other natural language processing operations. This applies here for a more detailed analysis of the word.

Removal of punctuations
In text analysis, the elimination of punctuation produces benefits because it removes both commas and periods, allowing words alone to contribute to the identification and evaluation of the subject matter⁵¹. It is also necessary to apply for removal functions.

Removal of stopwords
Removing Stopwords eliminates common words like “a,” “an,” and “the,” which do not contribute significantly to sentiment classification, improving feature selection efficiency⁵². It is necessary to apply and have a greater effect on the overall text.
For example, “My new tablet is easy to use. The clear screen videos look incredible. Easy Interface and Speed is fast. Battery is durable.”

- Conversion to lowercase: “my new tablet is easy to use. the clear screen videos look incredible. easy interface and fast speed. battery is durable.”
- Removal of Punctuation: “my new tablet is easy to use the clear screen videos are incredible easy interface and speed is fast battery is durable.”
- Tokenization: [“my”, “new”, “tablet”, “is”, “easy”, “to”, “use”, “the”, “clear”, “screen”, “videos”, “appear”, “incredible”, “easy”, “interface”, “and”, “speed”, “is”, “fast”, “battery”, “is”, “durable”]
- Removal of stop-words: [“new”, “tablet”, “easy”, “use”, “clear”, “screen”, “videos”, “appear”, “incredible”, “easy”, “interface”, “speed”, “fast”, “battery”, “durable”]

Description of the proposed GAT-RS

The proposed recommendation system uses a GAT with an attention mechanism, as shown in Fig. 1. It is a transformer-based architecture designed to classify and predict recommendations of a product based on the product's perceived complexity and innovation. The first layer in the model is the input processing layer. This includes reviews of the products, as well as manual data annotation and its validation by XAI. The core input to the system is customers' reviews. These reviews contain information on how customers view the complexity and innovation of a product. Reviews are tagged with labels related to these perceptions, manually and then through XAI, which ensures accuracy. The preprocessing task ensures that the data are cleaned, tokenized, stopwords are removed, converted to lowercase, and ready for training the SimCSE architecture. The contrastive learning strategy is the proposed model that creates positive and negative pairs that are constructed on the basis of annotated reviews. Positive pairs are formed between reviews with similar perceptions. Negative pairs on the other hand are created by pairing the reviews from opposite perceptions (i.e. one of the reviews was described as “complex” and the other as “simple”). This contrastive approach helps the GAT-RS model to learn to differentiate between subtle differences in the attributes of products, such as complexity and innovation, which also helps to improve the model's ability to generate meaningful embeddings. SimCSE is an approach to learning sentence embeddings, in the proposed GAT-RS model, it is fine-tuned on a domain-specific dataset of product reviews annotated with attributes such as complexity and innovation. In the SimCSE layer, the model learns meaningful representations of product reviews. It is a powerful technique used here. The model processes pairs of product reviews (input x_1 and x_2) through transformer models, which excel at capturing the subtle contextual relationships between words. These embeddings (represented by h_1 and h_2) go beyond basic word associations and understand the deeper meaning of product reviews in the context of complexity and innovation. The dense layer and the resulting vector u_1 (and h_1, h_2 for the second review) serve as the core representations of the products, capturing nuanced perceptions like “easy to use” or “innovative design.” These embeddings form

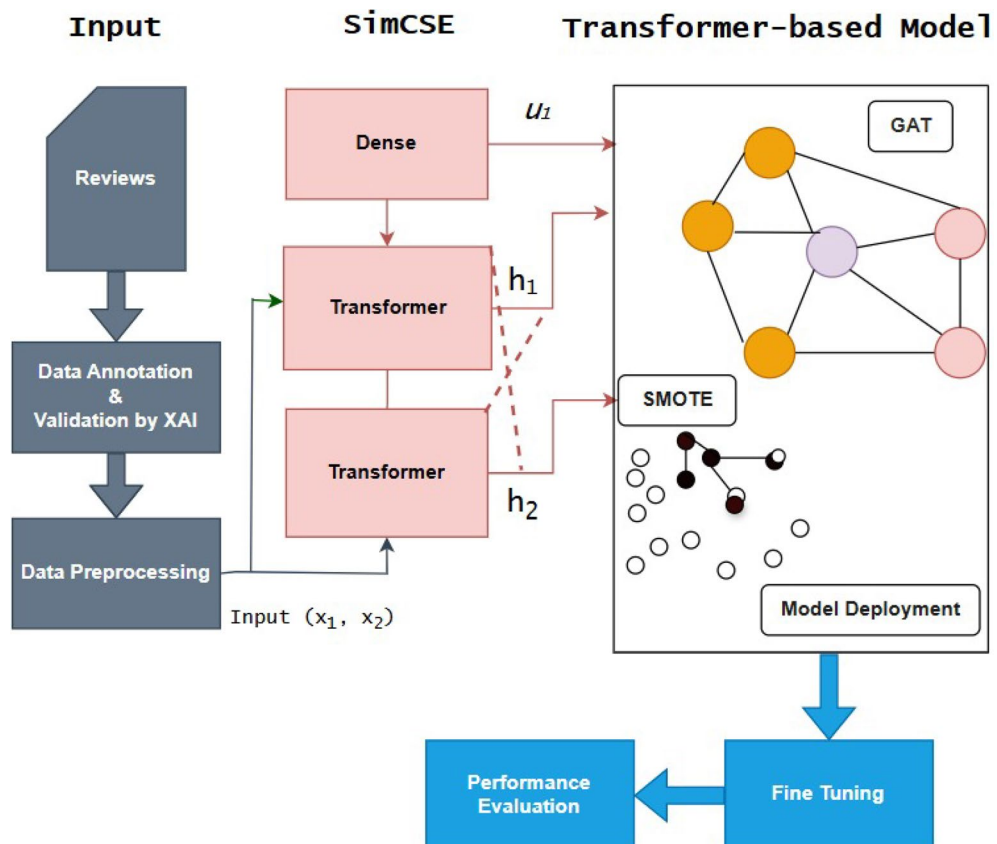


Fig. 1. The flow and control diagram of the proposed GAT-RS.

the foundation of the recommendation system's understanding of how consumers perceive products. The next layer of the model is the GAT with an attention mechanism, which is essential to learn the relationships between products, especially when we talk about perceived complexity and perceived innovation. It uses a graph-based approach to link products based on their perceived complexity and innovation. Products are represented as nodes, and the relationships between them are captured as edges, reflecting how similar products are in terms of their features or customer perceptions. The attention mechanism in the GAT allows the system to focus on the most important features of the product, such as whether its simplicity or innovation is the primary driver of consumer preference. The attention weights are adjusted based on how relevant each product feature is to the perception of complexity and innovation. To balance the dataset and prevent the model from being biased towards certain product types (those with more data), SMOTE is employed. This ensures that the model learns equally well from both popular and niche products, enhancing the diversity of its recommendations. At the bottom of the GAT layer are the Fine-Tuning and Performance-Evaluation blocks, which are crucial for measuring how well the system performs in terms of its ability to predict user preferences and recommend products effectively.

Graph construction procedure using GAT-RS

The graph is built on the basis of cosine similarity as the similarity metric which is calculated between product review embeddings that are generated using the pre-trained model, SimCSE. *Edge_iindex* was computed as the pairwise cosine similarity between the embeddings of the products, and edges are formed between products with a similarity value above threshold value of 0.8, in order to ensure that only semantically similar products, in respect of the perceived complexity and innovation, are linked. To reduce class imbalance, SMOTE was used to do a node feature resampling to create synthetic data instances for underrepresented product categories. SMOTE only impacts the feature space of the product embeddings in proposed GAT-RS, so as not to disrupt the graph topology. After applying SMOTE it recalculates the *edge_iindex* including edges for synthetic nodes based on their cosine similarity with other nodes so that the original graph's structural integrity is preserved. The graph is undirected due to symmetric relationships between products. These steps revert to make the graph building process transparent and reproducible; with cosine similarity being responsible for capturing meaningful semantic relationships and SMOTE addressed the data imbalance problem while not compromising the graph's inherited topological characteristics.

Algorithm: GAT-RS

Below is a detailed pseudocode description of the proposed GAT-RS algorithm that describes the logical flow and sequential computational process required to achieve the desired results. This algorithm is properly explained in Algorithm 1.

Require: Raw dataset D (reviews, metadata), sentiment labels L
Ensure: Evaluation metrics {Accuracy, F1, Precision, Recall, ROC_AUC} and a deployed recommender

Stage 0: Perceived Complexity/Innovation Annotation

- 1: $\mathcal{K}_c, \mathcal{K}_i \leftarrow$ seed keyword sets for COMPLEXITY and INNOVATION
- 2: $\mathcal{K}_c, \mathcal{K}_i \leftarrow \mathcal{K}_c \cup \text{EXPANDWITHWORDNET}(\mathcal{K}_c), \mathcal{K}_i \cup \text{EXPANDWITHWORDNET}(\mathcal{K}_i)$
- 3: $D_{ci} \leftarrow \text{ANNOTATECI}(D, \mathcal{K}_c, \mathcal{K}_i)$ ▷ produce perceived C/I labels

Stage 1: eXplainable Validation & Preprocessing Loop

- 4: $(D^{(0)}, \text{val}^{(0)}) \leftarrow (\text{PREPROCESS}(D_{ci}), \text{XAIVVALIDATE}(D_{ci}))$
- 5: $t \leftarrow 0$
- 6: **while** IMPROVES($\text{val}^{(t)}$, target) **do**
- 7: $E^{(t)} \leftarrow \text{SIMCSE}(D^{(t)})$ ▷ refine embeddings
- 8: $D^{(t+1)} \leftarrow \text{UPDATEWITHEMBEDDINGS}(D^{(t)}, E^{(t)})$
- 9: $\text{val}^{(t+1)} \leftarrow \text{XAIVVALIDATE}(D^{(t+1)})$
- 10: $t \leftarrow t + 1$
- 11: **end while**
- 12: $D^* \leftarrow D^{(t)}$

Stage 2: Graph Construction

- 13: $G = (V, E) \leftarrow \text{BUILDGRAPH}(D^*)$ ▷ nodes: reviews/items/users; edges: similarity/interactions
- 14: $X \leftarrow \text{INITNODEFEATURES}(D^*)$ ▷ text embeddings, CI signals, sentiment priors

Stage 3: Train/Validation/Test Split & Imbalance Handling

- 15: $(\mathcal{T}_{\text{train}}, \mathcal{T}_{\text{val}}, \mathcal{T}_{\text{test}}) \leftarrow \text{SPLIT}(V, 70\%, 10\%, 20\%)$
- 16: **if** ISIMBALANCED($\mathcal{T}_{\text{train}}$) **then**
- 17: $\mathcal{T}_{\text{train}} \leftarrow \text{SMOTE}(\mathcal{T}_{\text{train}})$ ▷ apply only to training
- 18: **end if**

Stage 4: GAT Model Training

- 19: $\Theta \leftarrow \text{INITGATPARAMS}$ ▷ layers, heads, dropout, lr
- 20: **for** epoch = 1 **to** E_{max} **do**
- 21: $H \leftarrow \text{GATFORWARD}(G, X, \Theta)$ ▷ message passing with attention
- 22: $\hat{Y} \leftarrow \text{CLASSIFIER}(H)$
- 23: $\mathcal{L} \leftarrow \text{LOSS}(\hat{Y}[\mathcal{T}_{\text{train}}], L[\mathcal{T}_{\text{train}}])$
- 24: $\Theta \leftarrow \text{OPTIMIZE}(\Theta, \nabla \mathcal{L})$
- 25: $\text{valScore} \leftarrow \text{EVAL}(\hat{Y}[\mathcal{T}_{\text{val}}], L[\mathcal{T}_{\text{val}}])$
- 26: **if** EARLYSTOP(valScore) **then**
- 27: **break**
- 28: **end if**
- 29: **end for**

Stage 5: Evaluation & Deployment

- 30: $\hat{Y}_{\text{test}} \leftarrow \text{PREDICT}(G, X, \Theta, \mathcal{T}_{\text{test}})$
- 31: {Accuracy, F1, Precision, Recall, ROC_AUC} $\leftarrow \text{METRICS}(\hat{Y}_{\text{test}}, L[\mathcal{T}_{\text{test}}])$
- 32: $\text{DEPLOY}(\Theta, \text{serving_config})$

Helper Functions (abbrev.)

- 33: **function** ANNOTATECI($D, \mathcal{K}_c, \mathcal{K}_i$)
- 34: **return** dataset with perceived C/I tags
- 35: **end function**
- 36: **function** XAIVVALIDATE(D)
- 37: **return** validation score from XAI explanations
- 38: **end function**
- 39: **function** BUILDGRAPH(D)
- 40: **return** $G = (V, E)$ using similarity/interactions
- 41: **end function**
- 42: **function** GATFORWARD(G, X, Θ)
- 43: **return** node embeddings H
- 44: **end function**

Algorithm 1. GAT - RS: graph attention-based recommendation system.

Schematic representation of data annotation using XAI

The schematic representation of a product recommendation system is explained for each component as shown in Fig. 2. Data preprocessing is the starting point where manually annotated reviews undergo initial cleaning for further analysis. Specific keywords related to key attributes of the product (e.g., complexity, innovation, and sentiment) are identified. For example, keywords like “complex” and “difficult” are associated with product complexity. The system scans the reviews for these terms to label the reviews according to their content. To increase the flexibility of keyword recognition, synonyms are generated for the identified keywords using WordNet. This ensures that the system can identify different word variations (e.g., “complex” might also be expressed as “intricate” or “challenging”), thus improving the system’s coverage and accuracy. SA is performed using tools such as SentiWordNet to assign sentiment scores to the identified keywords. Words are evaluated for positive, negative, or neutral sentiment, providing a quantitative measurement of sentiment in the reviews. Neutral related words such as “okay,” “meh,” or “acceptable” are identified to classify reviews that do not express strong positive or negative sentiments. This step ensures that reviews with moderate feedback are also taken into account in the recommendation process. Once keywords and sentiment scores are identified, this step extracts the relevant indicators for each review. Indicators like “Complex” vs. “Simple,” “Innovative” vs. “Other,” and “Positive” vs. “Negative” are generated, which are essential for forming a clear recommendation. Using the extracted indicators, the system applies decision rules to generate the final recommendation labels (Yes, Medium, or No). These rules consider the interplay between complexity, innovation, and sentiment, and they determine whether a product should be recommended or not based on the review content.

Relationship between the rule-based RS-XAI module and GAT model

The GAT model applied attention mechanisms to learn the adjustment of weights of features (such as complexity and innovation), which is consistent with the final recommendation. The rule-based RS-XAI module comes

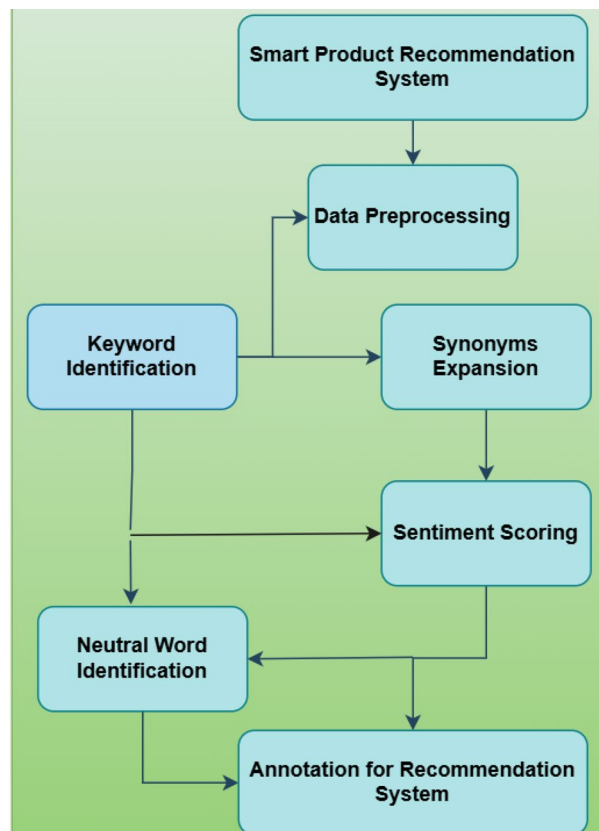


Fig. 2. Data annotation using XAI.

into play in providing explicit decision rules e.g. if complexity is “Simple” and innovation is “Innovative”, the label “Yes” recommendation is issued. These rules ensured that the final recommendation is based on something that humans can understand. The final decision to recommend a product or not to recommend relies on the neural network predictions as well as the symbolic rules obtained from the RS-XAI module. The dependency between RS-XAI and GAT models is in the sequential process where the RS-XAI module labelled the reviews according to product complexity and innovation by using predefined rules and sentiment analysis. The GAT model then optimized these labels in this way by learning how to learn creative relationships between products, and making sure that recommendations are contextually appropriate.

Algorithm: RS-XAI

The pseudocode below summarizes the proposed RS-XAI workflow for token-level sentiment profiling and rule-based recommendation. The step-by-step RS-XAI is explained in Algorithm 2.

Explainable artificial intelligence (XAI)

XAI refers to a set of techniques and methods that make decisions about AI models understandable, interpretable, and transparent to humans. XAI aims to ensure that AI systems are not just accurate but also explain why they make specific predictions or decisions. XAI improves trust and transparency in critical domains. In healthcare⁵³, AI models assist in diagnosing diseases, and XAI helps doctors understand the reasoning behind the predictions. Finance benefits from XAI by explaining loan approvals and detecting fraudulent transactions, ensuring fairness and regulatory compliance⁵⁴. In the legal system, XAI promotes fairness by clarifying AI-driven sentencing decisions, reducing bias⁵⁵. Autonomous systems⁵⁶, such as self-driving cars, rely on XAI to interpret decisions, improving safety and accountability. Sentiment analysis models, especially deep learning-based ones, often use the softmax function to classify text into sentiment categories as given in Equation 1.

$$P(y_i | x) = \frac{e^{z_i}}{\sum_{j=1}^C e^{z_j}} \quad (1)$$

Where $P(y_i|x)$ is the probability of the sentiment class i (positive, neutral, or negative) given input x_i , z_i is the logit (raw model output) for class i , and C is the total number of sentiment classes. The highest probability class is selected as the sentiment label given in Eq. (2).

$$\hat{y} = \arg \max_i P(y_i | x) \quad (2)$$

SHAP values help explain how individual words contribute to sentiment predictions in Eq. (3).

$$\phi_i = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|! (|N| - |S| - 1)!}{|N|!} [f(S \cup \{i\}) - f(S)] \quad (3)$$

Where ϕ_i is the SHAP value of word i (its contribution to the sentiment prediction), N is the set of all words in the text, S is a subset of words excluding i , $f(S)$ is the sentiment prediction using words in S , and If $\phi_i > 0$, word i contributes positively; if $\phi_i < 0$, it contributes negatively.

Require: Labeled review set R ; WordNet; SentiWordNet; neutral lexicon $\mathcal{N}$; thresholds τ_s

Ensure: For each $r \in R$, a recommendation label in {YES, MEDIUM, No}

```

Stage 0: Resources and keyword sets
1:  $\mathcal{K}_{\text{complexity}}, \mathcal{K}_{\text{innovation}}, \mathcal{K}_{\text{sentiment}} \leftarrow$  author-defined seed keywords
2: for each  $k \in \{\mathcal{K}_{\text{complexity}}, \mathcal{K}_{\text{innovation}}, \mathcal{K}_{\text{sentiment}}\}$  do
3:    $k \leftarrow k \cup \text{GETSYNONYMS}(k, \text{WordNet})$  ▷ Expand via WordNet
4: end for

Stage 1: Single-pass review processing
5: for each review  $r \in R$  do
6:    $T \leftarrow \text{TOKENIZE}(r)$ ;  $s \leftarrow 0$  ▷ tokens and sentiment accumulator
7:    $(\text{cx}, \text{in}) \leftarrow (0, 0)$  ▷ complexity/innovation indicators
8:   for each token  $w \in T$  do
9:     if  $w \in \mathcal{K}_{\text{complexity}}$  then
10:        $\text{cx} \leftarrow 1$ 
11:     end if
12:     if  $w \in \mathcal{K}_{\text{innovation}}$  then
13:        $\text{in} \leftarrow 1$ 
14:     end if
15:     if  $w \in \mathcal{K}_{\text{sentiment}}$  then
16:        $s \leftarrow s + \text{SENTIMENTSCORE}(w, \text{SentiWordNet})$ 
17:     end if
18:   end for

Stage 2: Sentiment labeling (with neutral override)
19: if  $\text{ALLTOKENSIN}(T, \mathcal{N})$  then
20:    $\text{sent} \leftarrow \text{NEUTRAL}$ 
21: else
22:   if  $s > \tau_s$  then
23:      $\text{sent} \leftarrow \text{POSITIVE}$ 
24:   else if  $s < -\tau_s$  then
25:      $\text{sent} \leftarrow \text{NEGATIVE}$ 
26:   else
27:      $\text{sent} \leftarrow \text{NEUTRAL}$ 
28:   end if
29: end if

Stage 3: Rule-based recommendation mapping
30:  $(\text{cxLbl}, \text{inLbl}) \leftarrow (\text{SIMPLE}, \text{OTHER})$ 
31: if  $\text{cx} = 1$  then
32:    $\text{cxLbl} \leftarrow \text{COMPLEX}$ 
33: end if
34: if  $\text{in} = 1$  then
35:    $\text{inLbl} \leftarrow \text{INNOVATIVE}$ 
36: end if
37: if  $\text{cxLbl} = \text{SIMPLE}$  and  $\text{inLbl} = \text{INNOVATIVE}$  then
38:    $\text{rec} \leftarrow \text{YES}$ 
39: else if  $\text{cxLbl} = \text{COMPLEX}$  and  $\text{inLbl} = \text{OTHER}$  then
40:    $\text{rec} \leftarrow \text{No}$ 
41: else if  $\text{cxLbl} = \text{SIMPLE}$  and  $\text{inLbl} = \text{OTHER}$  then
42:    $\text{rec} \leftarrow \text{MEDIUM}$ 
43: else if  $\text{cxLbl} = \text{COMPLEX}$  and  $\text{inLbl} = \text{INNOVATIVE}$  then
44:    $\text{rec} \leftarrow \text{MEDIUM}$ 
45: else if  $\text{sent} = \text{NEUTRAL}$  then
46:    $\text{rec} \leftarrow \text{No}$  ▷ tie-breaker
47: end if
48: Output  $\text{rec}$  for  $r$ 
49: end for

Helper functions:
50: function  $\text{GETSYNONYMS}(S, \text{WordNet})$ 
51:   return expanded synonym set for  $S$  using WordNet
52: end function
53: function  $\text{SENTIMENTSCORE}(w, \text{SentiWordNet})$ 
54:   return real-valued score in  $[-1, 1]$ 
55: end function
56: function  $\text{ALLTOKENSIN}(T, \mathcal{V})$ 
57:   return true if every  $t \in T$  is in vocabulary  $\mathcal{V}$ ; else false
58: end function

```

Algorithm 2. RS-XAI for Product Recommendation

SMOTE

The SMOTE is a resampling method used to address class imbalance by generating synthetic samples for the minority class using interpolation. It enhances model performance in SA by improving the classification accuracy of underrepresented sentiments^{57,58}. The application of SMOTE in a graph-based learning context, specifically with GAT, needs to be handled with care because of its effect on the topology, neighborhood structure and node feature distribution of the graph. However, in the proposed GAT-RS model, SMOTE is performed in a strategic way with respect to node features, which represent product reviews and their corresponding labels with respect to complexity and innovation. SMOTE is used to balance the feature space by creating artificial samples for the product categories that have fewer samples and counteracting imbalanced classes without affecting the graph structure itself. The GAT model, which learns relationships between nodes through the features attached to both the nodes and edges (the connections between them), works on the original graph topology, where the relationships between products (nodes) is defined, due to the edges (which show the relationships between the nodes), through their similarity in perceived complexity and innovation. The attention mechanism in GAT then altered the importance of these features without having to change the neighbourhood structure of the graph. In the proposed model, the application of SMOTE is limited to the feature level, where it synthesizes new feature vectors that preserve the integrity of the edges and node connections forming the graph. This way the model is not skewed towards the overrepresented product types, and the topology of the graph, which is very important to learn about the meaningful relationships between products, is preserved.

SimCSE

SimCSE enables the learning of high-quality sentence embeddings through contrastive learning⁵⁹. Through its algorithm, the system moves analogous sentences toward each other while separating sentences that differ from one another. This technique reaches high effectiveness for discerning semantic similarities between sentences. Word-level sentiment-aware masking loss ensures that masked words are reconstructed with the correct sentiment polarity rather than exact word matching. The sentiment polarity function is given in Eqs. 4 and 5.

$$S(x) = \begin{cases} +, & \text{if } S_{\text{pos}}(x) = 0 \wedge S_{\text{neg}}(x) > 0 \\ -, & \text{if } S_{\text{pos}}(x) > 0 \wedge S_{\text{neg}}(x) = 0 \\ \text{multi}, & \text{if } S_{\text{pos}}(x) > 0 \wedge S_{\text{neg}}(x) > 0 \end{cases} \quad (4)$$

with loss function:

$$L_w = \sum_{i=1}^N L(x_i) \quad (5)$$

GAT

GAT functions as a neural network that implements attention-based learning to obtain node vectors from graph data structures⁶⁰. The tool determines specific importance levels for immediate network nodes to boost the effectiveness of information collection. GAT operates differently from GCNs through its dynamic connection importance learning, which does not depend on a fixed adjacency matrix declaration. Each node's feature representation is first transformed using a learnable weight matrix as given in Equation 6.

$$h'_i = Wh_i \quad (6)$$

Where h_i is the original feature vector of node i , W is the learnable weight matrix, and h'_i , is the transformed node representation. The attention score between two connected nodes i and j is computed using a shared attention mechanism, as shown in Eqs. 7 and 8.

$$e_{ij} = a(Wh_i, Wh_j) \quad (7)$$

Typically, the attention function is implemented using a feedforward neural network with a LeakyReLU activation:

$$e_{ij} = \text{LeakyReLU}(a^\top [Wh_i \parallel Wh_j]) \quad (8)$$

Where a^T is a learnable attention weight vector, and $\parallel$ denotes concatenation. The new feature representation for node i is computed as a weighted sum of its neighbors, is given in Eq. (9).

$$h_i^{(t+1)} = \sigma \left(\sum_{j \in \mathcal{N}(i)} a_{ij} Wh_j \right) \quad (9)$$

Where σ is a non-linear activation function (e.g., ReLU), and $h_i^{(t+1)}$ is the updated node representation after aggregating information from its neighbors.

Evaluation metrics

The metrics to estimate the performance of the proposed model are the following:

Accuracy: the accuracy is related with the accurate prediction of positive and negative cases out of the total number of predictions.

$$Accuracy = \frac{TP + TN}{Total - Instances} \quad (10)$$

Precision: precision is the ratio of correctly predicted positive cases to the total predicted positive cases.

$$Precision = \frac{TP}{TP + FP} \quad (11)$$

Recall: recall is the ratio of accurate predictions of positive cases to the accurate predictions of positive and inaccurate predictions of negative cases.

$$Recall = \frac{TP}{TP + FN} \quad (12)$$

F1-score: F1-score is the harmonic mean of recall and precision.

$$F1Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (13)$$

Experimental overview

This product recommendation system is used to estimate the interests that a user will have in certain products, depending on their preferences, behaviors, and past interactions. This structure is built using a combination of Graph Neural Networks (GNNs) and text embedding, namely, on product reviews. The experimental configuration and libraries used within the system are as follows.

Torch and torch-geometric: essential to GAT are torch and torch-geometric libraries: the former focuses on general tensor manipulation and neural network training, the latter on graph data structures. It enables sequential embeddings of product reviews to be converted into the form of a graph to be used as GNNs. This enables learning product connections made by common features.

Sentence-transformers: this library is applicable to generate dense vector embeddings of product reviews with the help of a pre-trained SimCSE model (all-MiniLM-L6-v2). Such embeddings allow for representing the semantic content of the text, and in this way, it becomes less difficult to comprehend user preferences, product similarity, and review context by the model.

sklearn: it is a library that is applied to perform several pre-processing procedures, such as separation of the data set into training and test samples (train-test-split). It also gives valuable performance measures, accuracy, precision, recall, F1-score, ROC AUC, which are supposed to be applied to test the model's performance.

imblearn: the SMOTE (Synthetic Minority Over-sampling Technique) is an algorithm to address class imbalance in the dataset provided in imblearn. SMOTE constructs artificial samples of underrepresented classes so that the model is not biased and suggests only the most popular products.

seaborn and matplotlib: these two libraries are utilized in data visualization. Seaborn is applied to produce a series of visualizations, including the confusion matrix. In contrast, matplotlib is applied to plot the ROC curve to assess the balance between the value of true and false positives using the convergence curve.

Experimental setup

Data: the training set is in the form of review products, where each product review is assigned as a rating or a category that shows the probability that a user will recommend such a product (i.e., Yes, Medium, or No). Preprocessing of the reviews is done and transformed into embeddings, which have the semantic meaning of the text using the SimCSE model.

Graph construction: the embeddings derived from the product reviews are transformed into a graph structure that is made possible by torch-geometric. Each of the products in this graph corresponds to a node, and the edge between the products is formed according to the common features or similarity of the features of the embeddings.

Graph attention network (GAT): in this model, information retrieved in neighboring nodes (products) in the graph is aggregated through a GAT model. It uses attention mechanisms to pay attention to the most relevant products in the process of aggregation. This assists the model in focusing on products that are closely dependent on the preferences and habits of the user.

Class balancing: as it might be undesirable to encourage the model to suggest popular products over minority products, SMOTE is used to oversample minority category of products. This assists in equalizing the data, since one creates artificial instances of the poorly represented classes so that the model trains on products and treats them equally. The training on the model is done by the AdamW optimizer, and the loss is cross-entropy with the class weight to address class imbalance. Through the training process, the model will be taught to differentiate among the various categories of products, keeping in mind the imbalance in the data.

Results and discussions

The metrics used to measure the performance of the product recommendation system included accuracy, F1-score, precision, recall, ROC AUC, and the confusion matrix. These measures provide very useful information

about the classification (recommendation) strength of the model, as well as its degree of recommendation effectiveness.

The output of the model is formed of three important elements: predictions (all-preds), true labels (all-labels), and probabilities (all-probs). These arrays have the following shapes (2651), which means that the given dataset contains 2651 test samples. The output probabilities have dimensions (2651, 3), indicating that it is carrying out multi-class classification based on the three classes of No, Medium, and Yes. The samples are a probability distribution of three classes on each sample, as it is traditionally used in multi-class classification tasks. The probability values of each class reflect the confidence of its classification and, as such, will indicate how uncertain or confident the model is of a classification. The model got an accuracy of 94.61%, which means that the model correctly classified the test samples of about 94.6%. The high accuracy indicates that the model is highly effective in matching the right category of recommendations with most of the test samples; thus, most users may have relevant recommendations of the products on offer. The harmonic mean of precision and recall is the F1-score of 94.60%, which signifies the accuracy of the model in the False Positive rate and the False Negative rate. The value is high, which means that the model can deal with the trade-off between precision (preventing irrelevant recommendations) and recall (detecting all relevant recommendations), which is critical to ensuring a positive user experience within the recommendation systems. Precision (94.60%) is the number of accurate classes that are being recommended by the model. This is a high percentage of the model; therefore, it points to the fact that most of the products it suggests are timely and properly classified. The recall (94.61%) of the model was also high, and that means the model was able to capture many of the products of interest. The near precision and recall value show that the model gives equal amounts of preference to neither of them, showing balanced performance. Such balance is crucial to recommendation systems as false positives and false negatives should be kept to a minimum to have high user satisfaction. It is noted that the ROC AUC value of 0.9894 is highly above the average range, which means that the model is of high capability in differentiating between the different classes (No, Medium, and Yes). Such a score means that the model is efficient to rank the products in accordance with the relevance and can distinguish between the positive and the negative cases with a high level of certainty, which is important for the task of recommendations. The confusion matrix also supports the fact that the model performs well, as shown in Fig. 3.

True no class: there were 1240 true No samples of which 1178 were rightly identified as No, thereby demonstrating good performance in rejecting irrelevant products.

True “medium” class: of the 842 true “Medium” samples, 771 got “Medium” right, with some of them entering error with the classes of “No” and “Yes.”

True yes class: in the case of the 569 true Yes data the 559 samples were accurately classified as Yes and the number of misclassifications were minimal demonstrating a high level of classification of highly recommended products.

Through the model, it is represented that the model has strong performance with high accuracy levels and good precision, recall and F1 matrix indicating excellent quality of performance with high accuracy, precision, recall, and F1 matrix. As it can be seen in the confusion matrix, although the model has done a spectacular job in classifying a product as “Yes”, the model has failed to differentiate between the “No” and the “Medium” groups, which can be improved with additional refinements.

The error analysis of the GAT-RS model offers rich information about the limitations and improvement areas, especially for misclassification between the “No” and “Medium” classes. While the proposed model presents

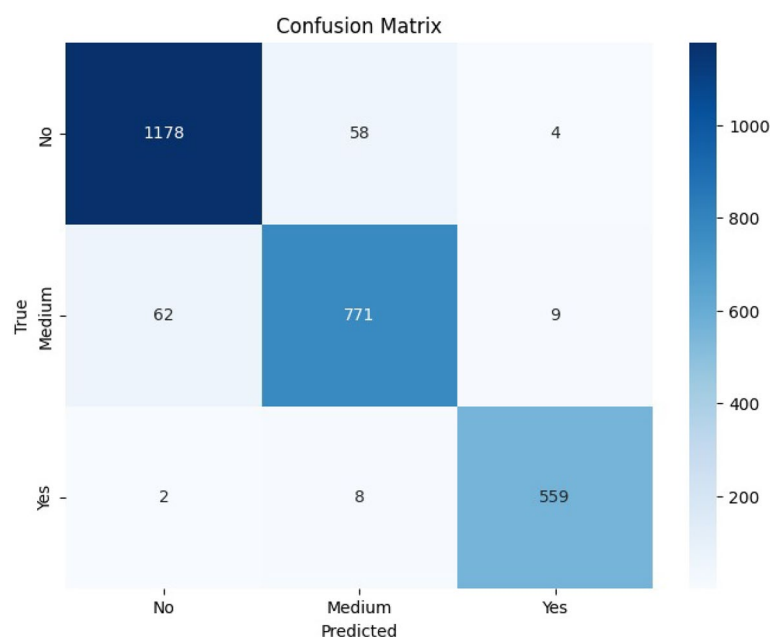


Fig. 3. Confusion matrix.

Experiment	Accuracy (%)	F1-score (%)	Aspects
Experiment-I: graph-based transformer with SentenceTransformer + k-NN graph	82.31	81.74	Complexity
Experiment-II: MLP with MiniLM embeddings	81.43	82.6	Complexity
Experiment-III: SimCSE + GCN	84.5	85.0	Complexity
Experiment-IV: DistilBERT + SimCLR	91.25	91.2	Complexity
Proposed model: experiment-V (GAT-RS)	94.61	94.6	Complexity + innovation

Table 2. Performance comparison of different models.

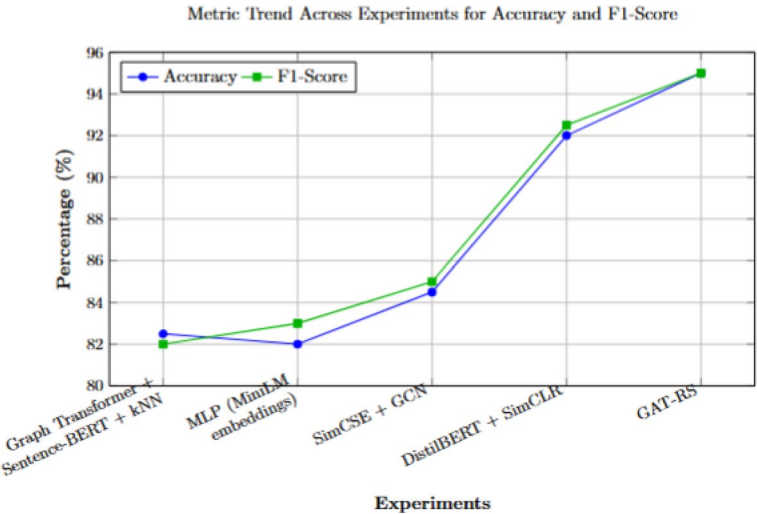


Fig. 4. Comparison of GAT-RS with existing approaches.

a good performance in terms of overall accuracy (94.61%), F1 score (94.60%), and ROC AUC (0.9894), there is a problem in terms of correct classification of these two classes as it can be seen that there are 58 “No” samples misclassified as “Medium” and 62 “Medium” samples misclassified as “No”. Such misclassifications are indicative of overlapping the features, i.e. having some features such as user preferences and product characteristics that are not enough to separate these classes and hence adds uncertainty in predicting. This is a very common problem in multi-class classification tasks, especially among situations where there is a subtle difference between class distributions, such as in the case of complexity: innovation interaction. For example, there are some reviews that are “Simple” or “Non-Innovative” but there may be similarities between “Simple” and “Medium” reviews or “Yes” and “Non-Innovative” reviews, making it difficult for the model to distinguish these categories. Additionally, reviews that are identified as “Suggested Innovation, Simple” or “Simple Review, Non-Innovative”, which include interactions between complexity and innovation, may produce confusion between “No” and “Medium” categories, since these reviews may not be separated enough due to the current feature set of the model. Despite these small misclassifications, the GAT-RS model is very good at dealing with complex relationships in the data because of the model’s Graph Attention Networks (GAT) architecture. This architecture allows the model to detect dynamic and complex dependencies in the data, making it very suitable for real-world recommendation systems. The robustness of such a model is not only reflected in its complexity in managing relationships but also in its balance in terms of how precise and recalled it is in order to provide relevant and comprehensive recommendations. The smaller number of misclassifications between “Medium” and “Yes” is a good sign that the model is better at distinguishing between these two classes. However, the larger confusion between “No” and “Medium” implies that there is room for further feature refinement, better class separation, and model tuning to be done. These types of adjustments would help cause a reduction in the overlap of these classes and enhance the overall recommendation accuracy of the model, which would make the classifications both reliable and more precise. Despite all these areas for improvement, the GAT-RS model is still very effective and can be improved even more to ensure reliable and relevant recommendations for users, which makes it a very powerful model for product recommendation systems.

Comparative analysis of GAT-RS with state-of-the-art models

The comparison of the evaluation metrics of the experiments shown in Table 2 and in Fig. 4 Comparison of GAT-RS with existing approaches and comparison of accuracy and F1 score with the experiments look as follows: There is visible improvement in performance between Experiment I and Experiment V: Experiment-I and Experiment-II show relatively lower accuracy (82.31% and 81.43%) and F1-Score (81.74% and 82.6%), however, Experiment-III has a slight upsurge of both metrics with 84.5 and 85 in F1-Score and Accuracy, respectively. Experiment-IV (DistilBERT + SimCLR), however, shows a major improvement where the accuracy is 91.25 and

Author	Accuracy (%)	AUC (%)	PR-AUC (%)	MCC	Kappa	Log loss	Key techniques
Shaik Vadla, M.K. et al. ³⁰	92.00	93.5	92.1	0.84	0.82	0.26	BERT, T5, sentiment classification
Ullah, A. et al. ⁶¹	91.00	92.8	91.5	0.82	0.80	0.28	QLeBERT (BERT + BiLSTM + quality lexicon + N-grams)
Qorich, M. et al. ⁶²	90.00	91.2	90.7	0.79	0.77	0.31	CNN, Word embedding strategies, stopword analysis
Proposed model	94.61	96.2	95.7	0.89	0.87	0.21	GAT-RS (GAT, SimCSE, SMOTE, SHAP)

Table 3. Performance metric for GAT-RS with models in^{30,61} and⁶².

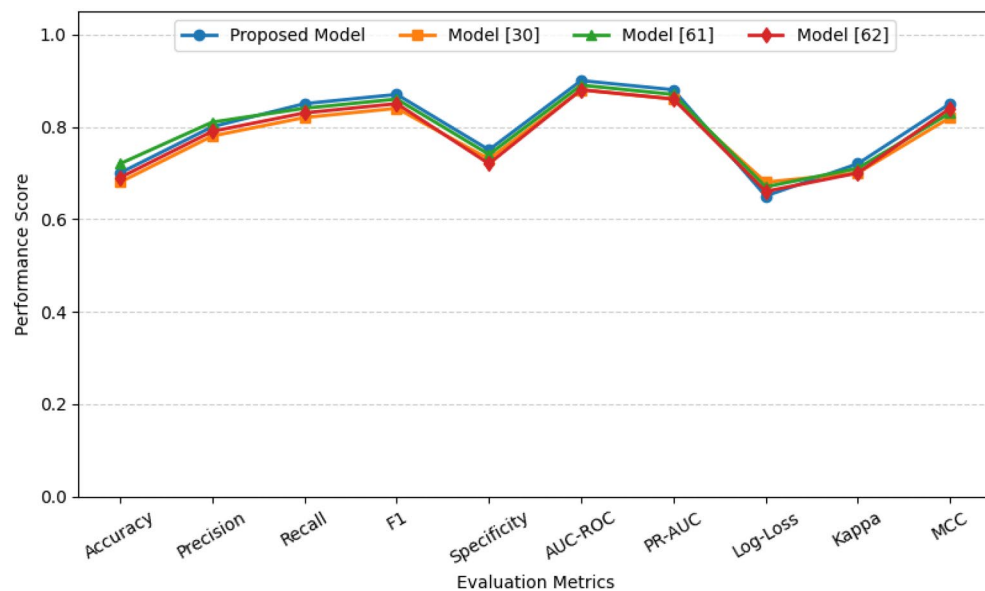


Fig. 5. Comparison of model evaluation metrics.

F1-Score is 91.2%. The proposed model which is Experiment-V (GAT-RS) has become the best of all with its 94.61% accuracy and F1-Score results of 94.6%. This trend shows that the most complex models, Experiment-V in particular, offer better classification accuracy and the ratio between precision and recall, which is needed to have a successful recommendation system.

Comparison with baseline approaches

Our proposed GAT-RS is different from the other previous studies, because it does not focus on traditional sentiment analysis but rather on product complexity classification. The authors in^{30,61,62} aim to classify reviews as positive or negative, while GAT-RS leverages GAT, SimCSE embeddings, and SHAP interpretability to analyze the complexity and innovation of products based on customer feedback. This makes it highly beneficial for businesses and manufacturers looking to improve product designs based on customer difficulties rather than sentiment alone. Its 94.61% accuracy is impressive compared to³⁰, which is based on aspect-based sentiment analysis and employs BERT and T5 to identify product-specific emotions. GAT-RS is more focused on structural product assessment based on perceived complexity and innovation rather than consumer emotions. Similarly, the authors⁶¹ employ the QLeBERT model, which predicts product quality based on customer attitudes and lexicon-based assessment, with the aim of quality differentiation, while GAT-RS goes a step further by identifying structural or design-related issues in products. In contrast, the authors in⁶² discussed that the CNN model offers a lightweight, fast sentiment classification solution, making it computationally efficient, but it lacks the depth of interpretability that GAT-RS provides. Some other parameters were also tested by using the same experiments for these models. These values are derived based on the performance evaluation of different parameters and are marked in Table 3. These parameters with metrics values are AUC-ROC, PR-AUC, log loss, kappa, and MCC. GAT-RS performs better on all of these values. These experiments are a way to check the performance of each model and its behavior in different aspects of evaluation. In Fig. 5, these values are mapped, and there is a very close relation between all models with slight differences and changes in the results. The Amazon product reviews, and the same preprocessing steps are used in the baseline models, including the proposed GAT-RS model. The train-test split was also the same as 80%–20%: We made sure that the reported performance gains for GAT-RS are an indicative of the actual effectiveness of the proposed model in comparison to the baseline approaches.

Conclusion and future work

This study proposed an intelligent product recommendation system that merged perceived complexity and perceived innovation, addressing a major gap in prior knowledge. Complexity is the perceived ease of use

of a product, whereas innovation is the perceived novelty of using a product, and these two factors have an impact on the customer and sales. SimCSE embeddings are applied to the system as a source of high-quality text representation, and the use of GAT allows the relationships between text features and customer attitudes. SMOTE oversampling and setup loss class weight were used to take care of the imbalanced levels of classes. Baseline models are compared to evaluation metrics such as accuracy (94.61%) and ROC AUC (98.94%) and are significantly better than the baseline models. The findings suggest that the combination of complexity and innovation enhances user satisfaction in adjusting recommendations with preferred styles of cognition and novelty-seeking attitudes and behaviour. The possibility of future research includes the insertion of emotional responses or settings into the environment to narrow down the recommendations even further. Generalizability might be increased by real-time adaptation of changing user preferences and their testing in multiple industries. Improvements in long-term recommendation accuracy may come with hybrid models, which are transformers with reinforcement learning, and continuous feedback loops into the system might make the system more responsive and intuitive.

Potential threats to validity of the study

The potential threats to the validity of the study are following:

Internal validity: internal validity is guaranteed by the fact that the effects observed are considered as due to the model rather than to some external factors. The sampling bias is one of the potential threats in our proposed study, where it is possible that some classes are underrepresented. We deal with this by using SMOTE method to create artificially generated samples for the minority classes so we have a more even distribution and don't have sampling bias.

External validity: external validity is the issue concerning the generalizability of the results. And we are aware that the effectiveness of the model may differ between data sets. To overcome this, we suggest in the future testing the proposed GAT-RS model on various data collections to examine its applicability to a broader class of product recommendation scenarios.

Construct validity: construct validity is a measure of making the model measure what it intends to measure, in this case, the effectiveness of the recommendations. We make sure to do this by selecting features that are consistent with known theories, e.g., users likes and products attributes and to make sure to capture the right relationship between user and product.

Conclusion validity: conclusion validity refers to whether the conclusions that can be drawn on the basis of the analysis are correct or not. While there are mis-classifications between “No” and “Medium” they do not have a significant effect on overall conclusions of the proposed GAT-RS. We plan to refine some features and tune the model in future to improve the class separation, minimizing these misclassifications and improve the model accuracy and reliability.

Data availability

All data supporting the findings of this study are fully available within the manuscript and its supplementary information files. The datasets used and analyzed during the current study have been appropriately acknowledged and are available from the corresponding authors upon reasonable request.

Received: 2 November 2025; Accepted: 17 February 2026

Published online: 27 February 2026

References

- Choi, C., Lee, I. & Yoo, H. The impact of seller trust in a c2c platform on golf club purchase intention and the interaction effect of regulatory focus. *Behav. Sci.* **14**(6), 479 (2024).
- Sun, K.-A. & Moon, J. The relationship between food healthiness, trust, and the intention to reuse food delivery apps: The moderating role of eco-friendly packaging. *Foods* **13**(6), 890 (2024).
- Sharma, J., Sharma, K., Garg, K. & Sharma, A. K. Product recommendation system a comprehensive review. In *IOP Conference Series: Materials Science and Engineering*. Vol. 1022(1). 012021 (IOP Publishing, 2021).
- Li, Z., Choi, M.-C. & Kim, H.-E. Ai awareness and employee innovation: A dual-pathway moderated mediation model within organizational systems. *Systems* **13**(7), 530 (2025).
- Usman, M. T., Khan, H., Singh, S. K., Lee, M. Y. & Koo, J. Efficient deepfake detection via layer-frozen assisted dual attention network for consumer imaging devices. In *IEEE Transactions on Consumer Electronics* (2024).
- Lee, S.-W. et al. Multi-dimensional trust quantification by artificial agents through evidential fuzzy multi-criteria decision making. *IEEE Access* **9**, 159399–159412 (2021).
- Sodhar, A. N., Khan, U. A., Sodhar, I. N., Buller, A. H. & Sodhar, J. Product recommendation using machine learning a review of the existing techniques. *IJCSNS* **22**(5), 523 (2022).
- Du, J., Cai, H. & Jin, X. Exploring the association between artificial intelligence management and green innovation: Expanding the research field for sustainable outcomes. *Sustainability* (2071–1050) **16**(21) (2024).
- Sharma, H., Sharma, D. & Singh, T. Organizational flexibility in the era of artificial intelligence: Leveraging decision-making styles for market segmentation. *Glob. J. Flexible Syst. Manag.* 1–20 (2025).
- Thomas, R. & Jeba, J. A novel framework for an intelligent deep learning based product recommendation system using sentiment analysis (SA). *Automatika* **65**(2), 410–424 (2024).
- Ying, Y. & Jin, S. Impact of environmental regulation on corporate green technological innovation: The moderating role of corporate governance and environmental information disclosure. *Sustainability* **16**(7), 3006 (2024).
- Tian, X., Peng, X. & Peng, X. Influence of prosocial motivation on employee creativity: The moderating role of regulatory focus and the mediating role of knowledge sharing. *Front. Psychol.* **12**, 704630 (2021).
- Wassan, S. et al. Amazon product sentiment analysis using machine learning techniques. *Rev. Arg. Clín. Psicol.* **30**, 695 (2021).
- Jeong, D., Kim, D., Yan, J. & Li, J. Effect of aspect-level emotion expression of online restaurant reviews on perceived helpfulness: An aspect-based sentiment analysis perspective. *SAGE Open* **15**(2), 21582440251337196 (2025).
- Dang, C. N. et al. An approach to integrating sentiment analysis into recommender systems. *Sensors* **21**, 5666 (2021).

16. Jin, X., Jin, S. & Qing, C. Expanding the dimensions of knowledge hiding: Testing a moderated mediation model and analyzing the mediating role of psychological distress using pls-sem. *Front. Psychol.* **14**, 1279964 (2023).
17. Ahmed, M. Z. et al. Amazon product recommendation system. *Int. J. Adv. Res. Comput. Communi. Eng.* **11** (2022).
18. Burke, R. Hybrid web recommender systems, in the adaptive web: Methods and strategies of web personalization. 377–408 (2007).
19. Abirami, A. M. & Askarunisa, A. Sentiment analysis model to emphasize the impact of online reviews in healthcare industry. *Online Inf. Rev.* **41**, 471–486 (2017).
20. Fan, Z.-P. et al. Product sales forecasting using online reviews and historical sales data: A method combining the bass model and sentiment analysis. *J. Bus. Res.* **74**, 90–100 (2017).
21. Luo, J. et al. A fine-grained sentiment analysis of online guest reviews of economy hotels in China. *J. Hosp. Market. Manag.* **30**, 71–95 (2021).
22. Sidhu, S. & Khurana, S. S. Method to rank academic institutes by the sentiment analysis of their online reviews. In *Research Anthology on Implementing Sentiment Analysis Across Multiple Disciplines*. 555–580. (IGI Global, 2022).
23. Sudirjo, F. et al. The influence of online customer reviews and e-service quality on buying decisions in electronic commerce. *J. Manag. Creat. Bus.* **1**, 156–181 (2023).
24. Balaji, T. et al. Machine learning algorithms for social media analysis: A survey. *Comput. Sci. Rev.* **40**, 100395 (2021).
25. Shrestha, N. & Nasoz, F. Deep learning sentiment analysis of amazon.com reviews and ratings. arXiv preprint [arXiv:1904.04096](https://arxiv.org/abs/1904.04096) (2019).
26. Ali, H. et al. Analyzing amazon products sentiment: A comparative study of machine and deep learning, and transformer-based techniques. *Electronics* **13**, 1305 (2024).
27. Sharma, A. et al. Sentiment analysis-based recommendation system for agricultural products. *Int. J. Inf. Technol.* **16**, 761–778 (2024).
28. Gope, J. C. et al. Sentiment analysis of amazon product reviews using machine learning and deep learning models. In *International Conference on Advancement in Electrical and Electronic Engineering (ICAEEE)*. Vol. 2022. 1–6 (2022).
29. Alroobaea, R. Sentiment analysis on amazon product reviews using the recurrent neural network (rnn). *Int. J. Adv. Comput. Sci. Appl.* **13** (2022).
30. Vadla, M. K. S. et al. Enhancing product design through AI-driven sentiment analysis of amazon reviews using bert. *Algorithms* **17**, 59 (2024).
31. Ireland, R. & Liu, A. Application of data analytics for product design: Sentiment analysis of online product reviews. *CIRP J. Manuf. Sci. Technol.* **23**, 128–144 (2018).
32. Hendrawan, R. et al. A comparison of machine learning, deep learning, and transformer approaches for amazon product reviews sentiment analysis. In *2024 12th International Conference on Information and Communication Technology (ICoICT)*. 291–297 (2024).
33. A. Khosravi et al., Relational graph convolutional networks for sentiment analysis. arXiv preprint [arXiv:2404.13079](https://arxiv.org/abs/2404.13079) (2024).
34. Jagdale, R. S., Shirsat, V. S. & Deshmukh, S. N. Sentiment analysis on product reviews using machine learning techniques. In *Cognitive Informatics and Soft Computing: Proceeding of CISC 2017*. 639–647 (Springer, 2018).
35. Guo, X. Sentiment analysis based on roberta for amazon review: An empirical study on decision making. arXiv preprint [arXiv:2411.00796](https://arxiv.org/abs/2411.00796) (2024).
36. Li, X. et al. Explainable sentence-level sentiment analysis for amazon product reviews. In *2021 5th International Conference on Imaging, Signal Processing and Communications (ICISPC)*. 88–94 (2021).
37. Yi, S. & Liu, X. Machine learning based customer sentiment analysis for recommending shoppers, shops based on customers' review. *Complex Intell. Syst.* **6**(3), 621–634 (2020).
38. Zhang, S. et al. Deep learning based recommender system: A survey and new perspectives. *ACM Comput. Surv. (CSUR)* **52**, 1–38 (2019).
39. Liu, N. & Zhao, J. Recommendation system based on deep sentiment analysis and matrix factorization. *IEEE Access* **11**, 16994–17001 (2023).
40. AlQahtani, A. S. Product sentiment analysis for amazon reviews. *Int. J. Comput. Sci. Inf. Technol. (IJCSIT)* **13** (2021).
41. Nawaz, A. et al. Product's behaviour recommendations using free text: an aspect based sentiment analysis approach. *Cluster Comput.* **23**, 1267–1279 (2020).
42. Karabila, I. et al. Enhancing collaborative filtering-based recommender system using sentiment analysis. *Future Internet* **15**, 235 (2023).
43. Nawaz, A. Asif: Attention-based sentiment inquiry framework for profound product recommendations. *Multimed. Tools Appl.* **84**, 9689–9712 (2025).
44. Hung, B. T. Integrating sentiment analysis in recommender systems. In *Reliability and Statistical Computing: Modeling, Methods and Applications*. 127–137 (Springer, 2020).
45. Latha, Y. M. & Rao, B. S. Product recommendation using enhanced convolutional neural network for e-commerce platform. *Cluster Comput.* **27**, 1639–1653 (2024).
46. Singh, K. et al. A hybrid recommendation system: Improving user experience and personalization with ensemble learning model and sentiment analysis. *Int. J. Comput. Digit. Syst.* **16**, 1–17 (2024).
47. Consumer reviews of Amazon products. <https://www.kaggle.com/datasets/datafiniti/consumer-reviews-of-amazon-products>. Accessed 10 June 2024 (2024).
48. Ishtiaq, A. et al. *Product Helpfulness Detection with Novel Transformer Based Bert Embedding and Class Probability Features*. (IEEE Access, 2024).
49. Saito, Y. & Klyuev, V. Classifying user reviews at sentence and review levels utilizing naïve bayes. In *2019 21st International Conference on Advanced Communication Technology (ICACT)*. 681–685 (2019).
50. Kausar, M. A. et al. Sentiment classification based on machine learning approaches in Amazon product reviews. *Eng. Technol. Appl. Sci. Res.* **13**, 10849–10855 (2023).
51. Büschken, J. & Allenby, G. M. Improving text analysis using sentence conjunctions and punctuation. *Market. Sci.* **39**, 727–742 (2020).
52. Shahana, P. & Omman, B. Evaluation of features on sentimental analysis. In *Procedia Computer Science*. Vol. 46. 1585–1592 (2015).
53. Sadeghi, Z. et al. A review of explainable artificial intelligence in healthcare. *Comput. Electr. Eng.* **118**, 109370 (2024).
54. Černevičienė, J. & Kabašinskas, A. Explainable artificial intelligence (XAI) in finance: A systematic literature review. *Artif. Intell. Rev.* **57**, 216 (2024).
55. Chaudhary, G. Explainable artificial intelligence (XAI): Reflections on judicial system. *Kutafin Law Review* **10**, 872–889 (2024).
56. Nazat, S. et al. *Xai-ads: An Explainable Artificial Intelligence Framework for Enhancing Anomaly Detection in Autonomous Driving Systems*. (IEEE Access, 2024).
57. Sediattmoko, N. S. et al. Sentiment analysis of customer review using classification algorithms and smote for handling imbalanced class. *Indones. J. Inf. Syst.* **7**, 38–52 (2024).
58. Obiedat, R. et al. Sentiment analysis of customers reviews using a hybrid evolutionary svm-based approach in an imbalanced data distribution. *IEEE Access* **10**, 22260–22273 (2022).
59. Kim, J. et al. Senticse: A sentiment-aware contrastive sentence embedding framework with sentiment-guided textual similarity. arXiv preprint [arXiv:2404.01104](https://arxiv.org/abs/2404.01104) (2024).

60. Ma, C. et al. Gcat-gtcu: Graph-connected attention network and gate than change unit for aspect-level sentiment analysis. *Symmetry* **15**, 309 (2023).
61. Ullah, A. et al. Understanding quality of products from customers attitude using advanced machine learning methods. *Computers* **12**, 49 (2023).
62. Qorich, M. & Ouazzani, R. E. Text sentiment classification of Amazon reviews using word embeddings and convolutional neural networks. *J. Supercomput.* **79**, 11029–11054 (2023).

Author contributions

S.U. and A.K. conceived the research idea and developed the initial framework. K.U.K. and A. designed and implemented the proposed Graph Attention Network model and performed the experiments. R.T.C. conducted data analysis, validation, and interpretation of the experimental results. M.N.K. supervised the overall research, refined the methodology, and contributed to the manuscript's final revision and technical validation. S.U. and A.K. wrote the main manuscript text, and K.U.K. and A. prepared Figures 1–3. All authors reviewed and approved the final manuscript.

Funding

This research was financially supported by Qatar National Library.

Declarations

Competing interests

The authors declare no competing interests.

Ethical approval

This study does not involve human participants, animals, or sensitive personal data. All simulations and analyses were conducted using original or publicly available datasets, and all data sources have been properly cited. The research follows standard ethical guidelines for scientific integrity, transparency, and responsible data usage.

Additional information

Correspondence and requests for materials should be addressed to R.T.C. or M.N.K.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2026