Aspect Sentiment Feature Extraction using Large Language Models for Cross Domain Recommendation

Zeeshan Zohaib*
zeeshan2@uwindsor.ca
School of Computer Science,
University of Windsor
Windsor, Ontario, Canada

C.I. Ezeife*
cezeife@uwindsor.ca
School of Computer Science,
University of Windsor
Windsor, Ontario, Canada

Ritu Chaturvedi
School of Computer Science,
University of Guelph
Guelph, Canada
chaturvr@uoguelph.ca

Abstract

Traditional Cross Domain Recommendation Systems (CDRS) use various sentiment analysis techniques on user reviews to extract and map user preferences across multiple domains like music and movie domains. This approach reduces the problem caused by insufficient information (data sparsity and cold start) about a user or product in a target domain (like music) by using information about the user/product from a source domain (like movies). One existing system, CRAS2024 only works with positive sentences within a review, discarding the negative sentences and this can cause information loss. It uses Biterm Topic Model (BTM), a probabilistic topic modeling technique for extracting topics from short texts to analyze how words co-occur and extract aspects from only positive sentence parts of reviews. It proceeds to use GloVe (Global Vectors for Word Representation), a statistical word embedding method to generate embeddings of the extracted aspects. CRAS2024 tools lack contextual understanding, making it unable to differentiate words like "bank" in different contexts. This paper proposes a Cross Domain Recommendation system called CRAS-LLM that employs Large Language Models for the task of Aspect Sentiment Feature Extraction (ASTE) from reviews of source domain to fill the needed information in a target domain for higher quality recommendation. The proposed CRAS-LLM system enhances CRAS2024 by considering user reviews irrespective of their polarity (positive, negative or neutral). Unlike CRAS2024, aspects from user reviews are extracted using DeBERTa (Decoding-enhanced BERT with disentangled attention) by retaining both positive and negative aspects from user reviews based on their context. Then, SimCSE-BERT (Simple Contrastive Learning of Sentence Embeddings with BERT) generates contextual embeddings of the resultant aspects, which are optimized using contrastive loss, bringing embeddings of similar sentiments closer and pushing those with dissimilar sentiments farther apart. Cross-Domain mapping is then performed on these embeddings using CycleGAN (Cycle Generative Adversarial Network), which maps these embeddings into target domain embeddings. Finally, similarity scores between embeddings generated from product reviews of the target domain and mapped aspect embeddings from the source domain are calculated. These similarity scores are then used to predict user ratings and generate personalized recommendations for products in the target domain.

CCS Concepts

• **Information systems → Recommender systems**;

Keywords

Aspect Opinion Mining, Sentiment Analysis, Cross-Domain Recommendations, Large Language Models, Data Sparsity, Cold Start

ACM Reference Format:
Zeeshan Zohaib, C.I. Ezeife, and Ritu Chaturvedi. 2025. Aspect Sentiment Feature Extraction using Large Language Models for Cross Domain Recommendation. In *Proceedings of LLM4ECommerce Workshop at the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 (LLM4ECommerce Workshop at KDD '25).* ACM, New York, NY, USA, 8 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

A "domain" refers to a specific category or type of content where recommendations are made. For example, Netflix's movie domain includes genres like action, comedy, and drama, while Amazon's book domain includes fiction, non-fiction, and mystery. Generally, recommendation systems are used to personalize user experiences by suggesting relevant content, products, or services and the systems are typically domain-specific because they are focussed on a single category, such as movies on Netflix, books on Amazon, or music on Spotify. Traditional association rule mining in market basket analysis, would find items frequently purchased together are used to recommend related products. For instance, if a user buys Bread, Milk, and Eggs, a simple association rule based system might recommend Beer next based on patterns of previously observed purchases from other customers and is in a single domain of grocery products.

The most well-known techniques in Recommender Systems include collaborative filtering and content-based methods. Collaborative filtering algorithms use ratings from multiple users, represented as a user-item rating matrix, to predict missing ratings and derive top recommendations. This approach predicts user's interests based on preferences of similar users. For example, if User A and User B both read technology articles like "Latest Smartphone Innovations" and "5G Network Expansion", and User A later reads "Future of Artificial Intelligence", the system may recommend this AI article to User B.

*All authors contributed equally to this research.

In content-based recommendation systems, the content (attributes or features, e.g., price) plays a primary role in recommendations. Both user ratings and product attributes are used to make predictions. Content-based systems assume that users' interests can be modeled using attributes of items they have rated or accessed in the past. This approach recommends items by analyzing product features, comparing them to items a user has liked previously. For example, if a user reads an article on machine learning, the system analyzes its content and recommends other articles on related topics, like deep learning or artificial intelligence, because they share similar features.

A hybrid approach combining both the content-based and collaborative filtering approach is also used in some systems. In advanced models, contextual data such as temporal, location, social or network information may be incorporated. Considering that there may be thousands of products or product features (representing columns of a user item rating matrix) that a user or item in the matrix row needs to interact with in the form of purchase, like or other, single-domain systems are faced with limitations of not having sufficient interactions in the input user item matrix. These limitations include cold-start (e.g., when a user is new to the domain and has made no purchases) and data sparsity (e.g., when a user has purchased only a few products out of the thousands available in a place like Amazon). These systems often overlook user preferences that span multiple interests, and this limits the depth of recommendations and hinder their recommendation performance. Cross-Domain Recommendation Systems (CDRS) [32] aim to overcome these limitations by transferring knowledge between different domains, thereby improving recommendation efficiency and accuracy. For instance, music is one domain, and movies is another domain. CDRS systems solve problems of single domain recommendation system with data from one domain (e.g., music) for the purposes of improving recommendations in another domain (e.g., movies).

While CDRS provides significant benefits, it also presents challenges. Transferring user preferences (e.g., customer reviews as discussed in Section 1.2) between domains requires careful alignment to avoid negative transfer, where irrelevant information lowers recommendation quality. For example, if a user enjoys action movies but dislikes thrillers, directly transferring this preference could result in poor book recommendations. Additionally, aligning distinct features (Aspects) across domains, such as genres in movies and themes in books, demands sophisticated mapping techniques. Researchers address these issues by distinguishing between general interests shared across domains and domain-specific interests to enable meaningful connections.

1.1 Important Concepts

This section describes important models and techniques used in solving the problem of aspect sentiment feature extraction using Large Language Models for Cross Domain Recomendation.

(1) Customer Reviews. Reviews often mention particular features or qualities of a product that users like. These mentions are key indicators of their preferences. For example, if a user frequently mentions "battery life" and "performance" in their laptop reviews, it suggests these features are particularly important to them. Understanding such preferences allows a recommendation system to suggest products that closely align with their interests. Some example customer reviews from three customers (Alice, Bob, Charlie) on products (Laptop, Headphones, Laptop) are listed as ($customer_i$, $product_i$, $review_i$): ("Great battery life and performance. Perfect for work.", "Excellent sound quality but slightly uncomfortable.", "Fast and reliable, but a bit heavy for travel."). Each review highlights specific product aspects. By identifying these aspects and analyzing sentiments, the system can infer which features are most important to each customer. Alice's review focuses on good battery life and performance. Bob highlights sound quality and comfort, prompting the system to recommend more comfortable headphones or accessories like ear cushions. Charlie values speed and reliability but is concerned about portability. To enhance satisfaction, the system learns these preferences to suggest lightweight laptops that still offer speed and reliability, or accessories to ease transport.

(2) Aspect Sentiment Extraction from Customer Reviews. Aspect mining is a technique used to provide more personalized recommendations by identifying specific product features mentioned in customer reviews. These features, or aspects, are often expressed as nouns. Example aspects are battery life, camera quality, screen size, or price in a smartphone review. Each aspect reflects a part of the product users may have opinions about. For example, the following two sentences S1 and S2 are part of a collection of customer review documents about a laptop:
S1: "The battery life on this laptop is amazing. The design is sleek and stylish."
S2: "The keyboard is uncomfortable, and the screen is too dim."
In these sentences, the words (battery, design, keyboard, and screen) represent the aspects (nouns) of the laptop. Thus, aspects and sentiments (positive or negative) on those aspects are mined from sentence reviews. An example result of aspect sentiment extraction on the above two sentences S1 and S2 is shown in table 1. The three main steps in aspect mining are: (a) Gathering Documents: Collect a set of documents that contains customer opinions about the product. These documents might come from online reviews on websites like BestBuy.com or social media platforms such as X (Twitter.com). (b) Identifying Aspects: Extract all the nouns from the documents since product aspects are often represented as nouns (e.g., "battery," "screen," "performance").

Table 1: Aspect Sentiment Extraction Result for Sentences S1 and S2

Sentence	Aspect	Sentiment
S1	battery	positive
	design	positive
S2	keyboard	negative
	screen	positive

(c) Pruning Non-Relevant Nouns: Filter out the nouns that do not represent significant aspects of the product. This step refines the list to include only the most relevant features of the product.

(3) Sentiment Analysis in Customer Reviews. While aspect mining identifies the sentiment of individual aspects in a sentence, sentiment analysis typically determines the overall polarity of the entire sentence. The process involves four main steps:
a. Text Preprocessing to clean the review by removing punctuation, stop words, and special characters.
b. Tokenization to split the text into individual words or phrases.
c. Polarity Detection to assign sentiment scores (positive, negative, or neutral) using lexicons (e.g., VADER [15]) or machine learning models.
d. Aggregation to combine scores and infer overall user sentiment toward the product.
A key difference between aspect-level and sentence-level sentiment is that a sentence may include both positive and negative opinions on different features, making the overall rating less meaningful to the user. Instead, users often care more about specific aspects like long battery life or a high-resolution display.

(4) Role of Aspect Mining and Sentiment Analysis in CDRS. Aspect Mining and Sentiment Analysis play a crucial role in enhancing Cross-Domain Recommendation Systems (CDRS) by providing fine-grained insights into user preferences and opinions. This aspect level mining approach [6], allows CDRS to capture both the aspects that users care about and their feelings about those aspects, enabling more accurate recommendations across domains. An example of Aspect Mining and Sentiment Analysis from Laptop Reviews is given in table 2.

Table 2: Aspect sentiment phrases from laptop reviews

Aspect	Sentiment	Supporting Phrase
Battery Life	Positive	"Lasts all day on a single charge"
Screen Resolution	Positive	"Delivers crisp, clear visuals"

The approach is useful in CDRS as user preferences expressed in one domain (e.g., movies with great visuals and soundtracks) can be transferred to another domain (e.g., recommending music albums with orchestral compositions). By linking aspect-level preferences and corresponding sentiments across domains, CDRS can better align recommendations with user interests, even for cold-start users who have not directly interacted with the target domain. Embeddings [31] are dense vector representations of words, phrases, or entire documents, created to capture their meanings, relationships, and context within a continuous numerical space between 0 and 1. Embeddings are widely used in Natural Language Processing (NLP) as they allow algorithms to understand semantic similarity between words, enabling recommendation systems and language models to process text data effectively. A traditional method for encoding products is one-hot encoding, which represents a finite set of products; for a four-word list of purchased items (e.g., pen, pencil, book, butter), there is a binary digit encoding indicating if the item is in that position within the vector space consisting of these items as attributes and columns. For example, a one-hot encoding for the 4-word vocabulary in a purchase sequence of these items is shown in Table 3. The one-hot encoding of "pen" is 1000 and "butter" is 0001. Embeddings encode words based on contextual usage; words with similar meanings tend to have closer vector representations in the embedding space.

Table 3: One-Hot Encoding represenation of a purchase sequence

	pen	pencil	book	butter
pen	1	0	0	0
pencil	0	1	0	0
book	0	0	1	0
butter	0	0	0	1

For example, word embeddings, showing how words like "cat," "dog," and "kitten" are close in space, representing their semantic similarity. Similarly, "king," "queen," "man," and "woman" demonstrate embeddings capturing relationships like gender and royalty. The attributes in the columns of the vector space are: (living being, feline, human, gender, royalty, verb, plural). The items whose embeddings are being learned as row labels are: cat, dog, kitten, dog, houses. The encoding for cat is (0.6, 0.9, 0.1, 0.4, -0.7, -0.3, -0.2) to show that cat is 0.6 true a living being, -0.2 true a plural. The embeddings for these three words cat, dog, kitten are shown in Table 4.

Table 4: Embedding representation of words

word	living-being	feline	human	gender	royalty	verb	plural
cat	0.6	0.9	0.1	0.4	-0.7	-0.3	-0.2
dog	0.7	-0.1	0.4	0.3	-0.4	-0.1	-0.3
kitten	0.5	0.8	-0.1	0.2	-0.6	-0.5	-0.1

Popular techniques commonly used to generate embeddings include: techniques are commonly used to generate embeddings: i. Word2Vec [21] which Learns embeddings by predicting neighboring words in a sentence (Continuous Bag of Words) or using context to predict a target word (Skip-gram).
ii. Doc2Vec [3] which extends Word2Vec to represent entire documents instead of individual words.
iii. GloVe (Global Vectors for Word Representation) [22] which captures co-occurrence statistics of words across a large corpus to generate embeddings, emphasizing word relationships.

User reviews can be converted into embeddings, allowing systems to understand preferences beyond simple ratings. Products, movies, or songs can also be represented as embeddings, enabling recommendations based on similarity in the embedding space. This capability is particularly valuable in Cross-Domain Recommendation Systems (CDRS), where user preferences in one domain (e.g., movies) can be mapped to another domain (e.g., music) using embedding-based techniques like CycleGAN.

(5) Deep Learning Models in Aspect Mining. Various types of neural networks, such as Feed-Forward Neural Networks (FFNN) [2], Long Short-Term Memory (LSTM) networks [14], Gated Recurrent Units (GRU) [5], Convolutional Neural Networks (CNN) [19], Attention mechanisms [25], Memory Networks [29], Neural Turing Machines [12], and Transformers [7], are used for aspect extraction and ABOM tasks; some are detailed below. Feed-Forward Neural Networks (FFNN) [2] consist of layers of neurons where each neuron processes inputs from the previous layer and passes the output forward. These networks are trained via backpropagation to minimize prediction errors. Recurrent Neural Networks (RNN) [27] are designed for sequential data, capturing dependencies across time steps—making them suitable for tasks like time-series forecasting and natural language processing. Long Short-Term Memory (LSTM) networks [14], a type of RNN, address the vanishing gradient problem, enabling learning of long-term dependencies in sequences. Bidirectional Encoder Representations from Transformers (BERT) [7] improves contextual understanding by processing text in both directions, using surrounding words to determine meaning. For example, BERT distinguishes between "river bank" and "savings bank" by leveraging context. Pre-trained on large corpora, BERT is fine-tuned for tasks like sentiment analysis, translation, and question answering, significantly boosting NLP performance. Decoding-enhanced BERT with Disentangled Attention (DeBERTa) [25] further improves BERT by separately modeling word meaning and position, enhancing language understanding. In BERT, word meaning and position are combined in the attention mechanism, sometimes blurring their individual importance. DeBERTa, however, handles them separately, allowing it to focus on each word's meaning independently of its position. For instance, "cat" means the same no matter where it is, but its relationships with other words depend on its position. BERT uses each word's fixed position (e.g., "first," "second"), while DeBERTa adds relative position embeddings, helping it understand words' relative positions, such as "near" or "far." This enhances DeBERTa's ability to capture word relationships, especially in longer sentences. Like BERT, DeBERTa is trained by masking certain words in a sentence and predicting them from context. DeBERTa improves this by placing special emphasis on masked words, allowing more accurate predictions. These enhancements enable DeBERTa to better capture the structure and meaning of sentences, making it particularly effective for tasks like sentiment analysis and question answering.

(6) Specialized models used for Cross-Domain Recommendation include SimCSE-BERT and CycleGAN. SimCSE-BERT [11] extends BERT (Bidirectional Encoder Representation from Transformers) to generate high-quality sentence embeddings (vector representations of sentences) through contrastive learning. While BERT's default embeddings often miss fine-grained semantics, SimCSE-BERT improves alignment by bringing semantically similar sentences closer in vector space and pushing dissimilar ones apart [16]. This approach enhances performance on tasks like sentence similarity, paraphrase detection, and semantic search. We focus on the unsupervised version, which fine-tunes BERT without labeled data. Another specialized model for cross domain recommendation is CycleGAN (Cycle Generative Adversarial Networks) [13], which uses a Generator–Discriminator setup to learn mappings between domains. The Generator produces data in the target domain, while the Discriminator distinguishes it from real samples, improving both over time. Traditional GANs struggle with cross-domain tasks, such as mapping abstract preferences (e.g., "vibrant colors" in movies to "rich soundscapes" in music). CycleGAN addresses this through cycle consistency, ensuring that data mapped to a target domain and back remains consistent with its original form.

1.2 Paper Contributions and Outline

This paper contributes the following original solutions to unsolved issues in cross domain recommendation systems as in its unpublished thesis copy [33]:

(1) Providing more accurate aspect-opinion-polarity extraction by fine-tuning a pre-trained transformer model to perform aspect-opinion-polarity (AOP) triplet extraction, replacing the topic-based aspect extraction approach used in the original CRAS paper. This enables the model to identify aspects not just as isolated words but based on their surrounding context, so it can capture the true sentiment behind user reviews more accurately.
(2) Incorporation of Both Positive and Negative Aspects with Contrastive Loss. both positive and negative aspects extracted through Aspect-Opinion-Polarity (AOP) triplet extraction are incorporated. Unlike CRAS2024, which relies solely on positive reviews, negative aspects offer additional insight into user preferences by highlighting what users wish to avoid. This allows the recommendation system to not only suggest preferred products but also reduce the likelihood of recommending undesirable ones (e.g., movies or music genres the user dislikes).
(3) Context-aware embeddings for aspect representation Traditional embedding techniques often fail to distinguish between multiple meanings of the same aspect based on context. For instance, the aspect "bank" may refer to a "financial institution" or a "river bank." Existing systems generate identical embeddings regardless of context, reducing recommendation accuracy.

(4) Preserving information by eliminating traditional preprocessing steps such as stemming (e.g., converting "organization" to "organ") in traditional systems often result in information loss, affecting the accuracy of sentiment analysis and recommendations. Existing research [4] highlights these drawbacks. In this paper, the proposed approach (CRAS-LLM) avoids stemming and other similar preprocessing steps to retain critical information and sentiment details.

To accomplish the above feature contributions, paper proposes Cross-Domain Recommendations Via Aspect Sentiment Feature Extraction using Large Language Models (CRAS-LLM) system, which implements the following steps:

(1) Extract aspect-opinion-polarity (A-O-P) triplets from raw user reviews by fine-tuning the DeBERTa model, identifying fine-grained sentiment expressions for each aspect and their polarity without traditional preprocessing steps such as stemming or stopword removal.
(2) Generate context-aware embeddings for the extracted aspects using SimCSE-BERT model, ensuring rich semantic representation of user preferences.
(3) Apply contrastive loss on SimCSE-BERT embeddings to refine aspect embeddings by ensuring that similar aspects remain close while dissimilar aspects are pushed apart.
(4) Perform cross-domain mapping using CycleGAN, transforming source domain (e.g., movie) embeddings into target domain (e.g., music) embeddings while preserving semantic consistency through cycle-consistency loss.
(5) Extract the AOP triplets and Generate embeddings for target domain items as well, using Fine-Tuned DeBERTa and SimCSE-BERT and compute cosine similarity scores between mapped user preferences embeddings and target domain item embeddings.
(6) Predict user ratings for target domain items using a regression model trained on cosine similarity scores and optimize the model using mean squared error (MSE) loss.
(7) Rank items in the target domain based on predicted ratings, selecting the top-k recommendations.

The rest of the paper is outlined as follows. Section 2 discusses related work, section 3 presents the proposed system for enriching recommendation using Cross-Domain Recommendations Via Aspect Sentiment Feature Extraction using Large Language Models (CRAS-LLM) , section 4 discusses experimental and performance analysis. Section 5 finally presents conclusions and future work.

2 Related Work

Some of the key techniques used to address the challenges of cross-domain recommendation are transfer learning, matrix factorization, and collaborative filtering. These approaches aim to use user-item interactions from one domain to enhance recommendations in another domain with the goal of addressing issues of data sparsity and cold start. Deep learning approaches from Neural networks have become increasingly popular for cross-domain recommendation, where embedding techniques are used to map users and items from different domains into a shared latent space. This summarizes some key models that address cross-domain recommendation challenges, cross-domain mapping architectures such as CycleGAN [30] and transfer learning models. The systems reviewed are:

1. CRAS: Cross-Domain Recommendation via Aspect-Level Sentiment Extraction Approach [34] is the closest existing system to the model being proposed in this paper. CRAS2024 addresses cold-start problem and improves cross-domain recommendation accuracy by extracting aspect-level sentiment from reviews. CRAS2024 enhances recommendation accuracy by extracting user preferences from review texts using the Biterm Topic Model (BTM) [30], mapping (converting) these preferences across domains (i.e., movie to music) with CycleGAN [13]. For example, if a user frequently reviews books with detailed story lines and complex character development, CRAS2024 can extract aspects like "story depth" and "character complexity" to recommend TV series or movies with similar qualities, addressing the cold-start problem to provide recommendations matching user's interests, even if the user has not interacted with movies or shows before.

Cras2024 goes through the six steps of (i) Sentiment Analysis and Preprocessing using VADER [15]; (ii) Aspect Extraction using the Biterm Topic Model (BTM) [30]; (iii) Embedding Layer which transforms each structured aspect into a vector using pre-trained embeddings like GloVe's [24] and generate aspects as dense vectors capturing semantic similarity between words. The rest of the steps are (iv) Cross-Domain Mapping with CycleGAN model to map preferences between two different domains without requiring paired examples by training on a common user set that has interactions in both domains; (v) Similarity Computation between the mapped movie domain aspect embeddings and the target domain music aspect embeddings is calculated using cosine similarity; (vi) Cross-Domain Rating Prediction showing how likely a user can enjoy an item in the target domain (e.g., music) based on their preferences from the source domain (e.g., movies) using similarity scores that applies a softMax-based transformation.

2. Graphical and Attention, GA (Graphical and Attentional) [35] addresses enhances recommendation accuracy in a single target domain using a richer source domain. While the conventional methods are effective in leveraging data from a richer domain to improve the recommendations in a sparser domain, they often fail to consider the potential for mutual improvement across domains. The GA framework introduces three innovative scenarios—Dual-Target CDR (DTCDR), Multi-Target CDR (MTCDR), and CDR+CSR [18], aiming to simultaneously improve recommendations across multiple datasets, allowing both richer and sparser domains to benefit from shared knowledge.

RC-DFM: Review and Content-based Deep Fusion Model for Cold Start Users in Cross-Domain Recommendation Systems [10].

3. The RC-DFM framework addresses the problem of cold-start users in cross-domain recommendation systems by deeply fusing review texts and item content with the rating matrix. It improves recommendations for users who have minimal or no feedback in the target domain by leveraging information from the auxiliary domain.

4. Other cross domain recommendation systems reviewed include: CDR-SAFM [28] (Cross-Domain Recommendation with Sentiment Analysis and Latent Feature Mapping), CDSAWE [20] a cross-domain sentiment-aware word embedding model, Hybrid Fake Review Detection System, a CNN and LSTM hybrid approach [1], MASM, The Microblog Aspect Sequence Miner system [9], Aspect-Based

Opinion Mining (ABOM) and Deep Learning Approach [8], and Deep Convolutional Neural Network (CNN) [23].

3 The Proposed CRAS-LLM System

The existing systems CRAS2024 [34] and RC-DFM2019 [10], have shortcomings that limit recommendation quality. CRAS2024 suffers from information loss due to traditional prep-rocessing steps, like stemming and stop word removal, and discards negative sentiment reviews, resulting in incomplete user preferences and loss of critical context. The system RC-DFM2019 is not very effective in sparse-data conditions, negatively impacting cold-start user scenarios. These limitations can be addressed by the proposed system, Cross-Domain Recommendations with Aspect Sentiment Feature Extraction using Large Language Models (CRAS-LLM) framework. The proposed approach is structured into three core modules: (a) Aspect-Level Sentiment Embedding Generation Module (ASEM), (b) Cross-Domain Preference Mapping Module (CDPM) and (c) Recommendation Generation Module (RGM). ASEM extracts and finds the contextual embeddings of sentiment rich aspects from user review, while CDPM maps these embeddings from one domain (i.e., movie) to another domain (i.e., music) embeddings, preserving sentiment context for accurate cross-domain knowledge transfer. Finally, RGM generates personalized recommendations for new users by predicting item ratings and ranking them according to their similarity. These modules enable CRAS-LLM preserve contextual semantics, capture both positive and negative user sentiments, and deliver more accurate, personalized cross-domain recommendations. The system architecture for the proposed system for cross domain recommendation using LLM (CRAS-LLM) is shown in Figure 1. The main algorithm for the proposed CRAS-LLM consists of 4 major steps which will be explained in greater detail is shown next.

```
Algorithm: CRAS-LLM-To recommend top-k items in a target
domain (e.g., music) based on user preferences extracted
from a source domain (e.g., movies).
Input:
1. R: Set of user reviews in the source domain.
2. I: Set of items in the target domain.

Variables:
1. E = Collection of aspect-opinion embeddings from
the source domain
2. E = Source-domain embeddings mapped into
target-domain style
3. P = Predicted ratings for items in the target domain
4. K = Number of top recommendations to select

Output:
Recommended-List: Top-k recommended items in the
target domain
START
1.E ← ASEM(R): {CRAS-LLM calls ASEM to extract aspect le-
vel embeddings from source domain review}
(summarized in section 3.1).
2. E ← CDPM(E): {CRAS-LLM calls CDPM to Map source-domai-
n embeddings (E) to target-domain embeddings (E)}
(summarized in section 3.1).
3. P ← RGM (E, I): {CRAS-LLM calls RGM to Generate pred-
icted ratings (P) for target items using (E) and sort (I)
by predicted ratings (P) in descending order
(summarized in section 3.1)
4.Recommended-List ← I[1...k]: {CRAS-LLM returns the t-
op-k items with highest predicted ratings in the target
domain.}
STOP
```

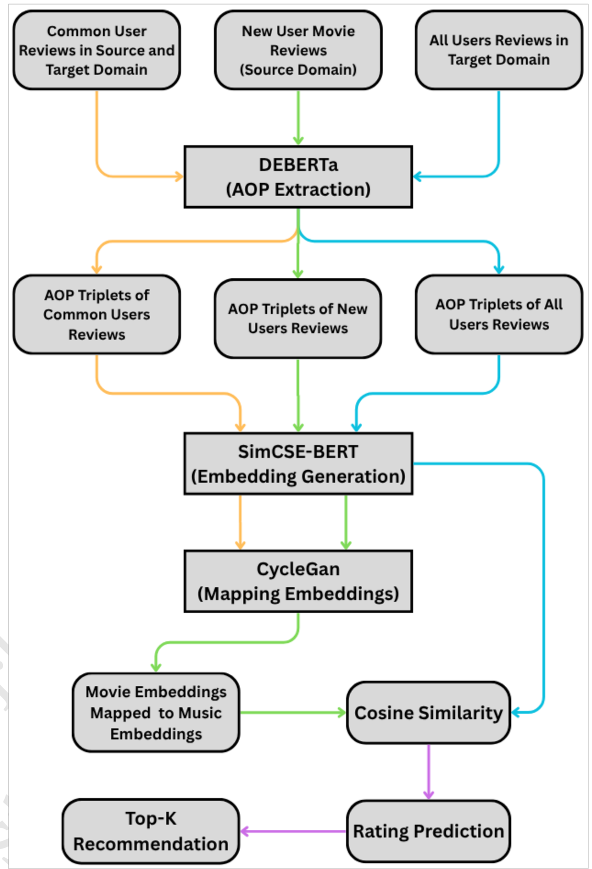

Figure 1: CRAS-LLM Architecture

3.1 Details of the CRAS-LLM Steps

1. Aspect-Level Sentiment Embedding Generation Module (ASEM) : This Module has two key Steps: (i) Aspect-Opinion-Polarity (AOP) Extraction using Fine Tuned DeBERTa Model so that each review is processed to identify distinct aspects (e.g., "shipping speed") and the corresponding user opinion (e.g., "too slow") along with its polarity (positive or negative) and (ii) Aspect Embedding Generation Using SimCSE-BERT with Contrastive Learning where these AOP triplets are converted into a dense embedding while maintaining the contextual meaning.

2. **Cross-Domain Preference Mapping Module (CDPM)**: The sentiment-labeled embeddings generated from each review in the previous step are mapped to a target domain (e.g., from movies to

Table 5: Comparison Results with Baseline Methods in Terms of MSE

Scenario	EMCDR	R-DFM	CATN	PTUP CDR	CRAS	CRAS LLM
Movie to Music	2.232	1.746	1.121	1.017	0.959	**0.749**
Music to Movie	2.385	1.853	1.162	1.117	1.029	**0.716**
Movie to Book	2.310	1.504	0.974	0.962	0.800	0.812
Book to Movie	1.866	1.610	0.975	0.965	0.965	**0.793**
Book to Music	2.586	1.293	1.007	1.136	0.717	**0.693**
Music to Book	2.804	1.483	1.096	1.155	0.850	**0.711**

music) using the CycleGAN model [13]. CycleGAN enables translation between domains by learning the relationships and similarities between aspects across domains without needing paired data. This allows taking user preferences in one domain (like movies) and matching them with similar items in a different domain (like music). Common users who have reviews in both domains serve as the basis for training CycleGAN.

3. **Recommendation Generation Module (RGM)** : The goal here is to predict ratings for a new user who has never interacted with the music domain. The user provides a review in the movie domain, which is then processed to generate aspect embeddings. These embeddings are mapped to the music domain, where they are compared with pre-existing music aspect embeddings to generate personalized recommendations.

4 Experimental and Performance Analysis

Experiments were performed in three pairs of cross-domains: Movie-Music, Movie-Book, and Book-Music using the Amazon review data set. After filtering out users with fewer than 10 interactions and items with fewer than 20 ratings, we simulate cold-start conditions by removing target-domain interactions for half of the shared users. Post-processing, the Movie domain includes up to 1,332 users and 80,460 ratings, the Book domain up to 1,776 users and 35,508 ratings, and the Music domain up to 1,199 items and 31,798 ratings. These datasets support our cross-domain and cold-start experiments. We employ Bayesian optimization [26] to tune hyperparameters such as learning rate, batch size, and embedding dimensions for both embedding generation and CycleGAN mapping. Each domain pair is trained independently for 50 epochs using the Adam optimizer[17] with an initial learning rate of 0.001.

In these experiments, evaluation of the quality of rating predictions using Mean Square Error (MSE) and top-K recommendations using Precision, Recall and F1-score was done. The proposed CRAS-LLM was also compared against several baselines systems to demonstrate the contributions of each key module. Table 5 compares the MSE values achieved by each method across six cross-domain directions: Movie→Music Music→Movie, Movie→Book, Book→Movie, Book→Music, and Music→Book. These results show that CRAS-LLM achieves the lowest MSE in all six scenarios. The differences are most prominent in the historically challenging "Music → Movie" transfer, where CRAS-LLM reduces MSE to approximately 0.716 compared to CRAS's 1.0296. This performance improvement can be attributed to CRAS-LLM retaining both positive and negative sentiments and leveraging contrastive embeddings that reflect complex user preferences. CRAS-LLM achieves a Precision of 0.753, Recall of 0.712 and F1-Score of 0.723, exceeding the best baseline (CRAS) by a comfortable margin. CRAS-LLM proposed, is evaluated using four standard recommendation metrics:

- Precision: which measures the proportion of recommended items that are truly relevant. Precision is (True Positive (TP))/(True Positive + False Positive(FP)).
- Recall measures the proportion of all truly relevant items that were successfully recommended. Recall = TP/(TP + FN (false negative)).
- The F1-Score is the harmonic mean of precision and recall. F1-Score = 2 x 2×(Precision × Recall)/(Precision + Recall)).
- Mean Square Error (MSE) quantifies the extent to which the model's predictions deviate from the true user ratings, assigning higher penalties to larger prediction errors.

$$MSE = frac1, m \sum_{i=1}^{m} (r_i - r_{ip})^2$$

where m is the total number of test (user-item) pairs (primarily from cold-start users in the target domain), r_i represents the actual rating provided by the user for the i-th item and r_{ip} denotes the rating predicted by the model for that same item. A lower MSE value indicates more accurate predictions.

To evaluate the contribution of each component in CRAS-LLM, ablation experiments were conducted by systematically modifying the model. The full model achieves the lowest mean squared error (MSE) of 0.716, demonstrating the effectiveness of the complete CRAS-LLM pipeline. When negative sentiment handling is removed (CRAS-LLM−NS), the MSE increases to 0.742, suggesting that negative opinions provide valuable contextual information in the full CRAS-LLM. Replacing the SimCSE-BERT embeddings with standard Word2Vec (CRAS-LLM−W2V) results in the highest MSE of 0.774, highlighting the importance of contrastive, sentiment-aware representations. Substituting the CycleGAN mapping with a simpler MLP (CRAS-LLM−MLP) yields an MSE of 0.736, indicating that while simpler mappings can function, cyclic consistency enhances predictive accuracy. These results confirm that each module—sentiment inclusion, advanced embeddings, and cross-domain mapping—significantly contributes to the model's overall performance.

5 Conclusions and Future Work

This work proposes CRAS-LLM, a cross-domain recommendation system that enhances aspect-sentiment feature extraction and cross-domain mapping to overcome the limitations of existing models as CRAS2024. By fine-tuning DeBERTa for aspect-opinion-polarity (A-O-P) extraction, SimCSE-BERT with contrastive loss for embedding generation, and CycleGAN for cross-domain adaptation, the system provides more accurate and personalized recommendations compared to traditional approaches. Unlike prior models, CRAS-LLM retains both positive and negative aspects, ensuring a more comprehensive understanding of user preferences. The proposed system eliminates traditional NLP pre-processing steps, enabling more effective sentiment-aware aspect extraction and improving embedding quality for cross-domain recommendations. Contrastive loss enhances the semantic alignment of aspect embeddings, leading to a significant reduction in Mean Squared Error (MSE) across various domains. Future work can explore multi-step CycleGAN training to improve fine-grained semantic mapping between domains. One key area is enhancing the extraction of nuanced sentiments from user reviews by exploring advanced language models and fine-tuning methods that capture subtler emotional cues. In addition, further investigation into the domain mapping process using innovative architectures like enhanced CycleGANs or emerging generative models could improve the precision of cross-domain embeddings. Expanding the framework to integrate multi-modal data—such as visual or contextual user behavior, may lead to even more personalized and robust recommendations. Finally, applying the proposed methodology to larger and more diverse datasets will be essential for validating its scalability and real-world applicability, ultimately paving the way for more adaptive and intelligent recommendation systems.

6 Acknowledgments

This research was supported by the Natural Science and Engineering Research Council (NSERC) of Canada under an Operating grant (OGP-0194134) and a University of Windsor grant.

References

[1] Gourav Bathla, Pardeep Singh, Rahul Kumar Singh, Erik Cambria, and Rajeev Tiwari. 2022. Intelligent fake reviews detection based on aspect extraction and analysis using deep learning. *Neural Computing and Applications* 34, 22 (2022), 20213–20229.

[2] Yoshua Bengio, Yann LeCun, and Geoffrey Hinton. 2021. Deep Learning for AI How can neural networks learn the rich internal representations required for difficult tasks such as recognizing objects or understanding language? *Commun. ACM* 64, 7 (2021), 58–65.

[3] Shraey Bhatia, Jey Han Lau, and Timothy Baldwin. 2016. Automatic Labelling of Topics with Neural Embeddings. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics*. The COLING 2016, Osaka, Japan, 953–963.

[4] Christine P Chai. 2023. Comparison of text preprocessing methods. *Natural Language Engineering* 29, 3 (2023), 509–553.

[5] Junyoung Chung, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS 2014 Workshop on Deep Learning, December 2014*.

[6] Alfredo Daza, Néstor Daniel González Rueda, Mirelly Sonia Aguilar Sánchez, Wilmer Filomeno Robles Espíritu, and María Elena Chauca Quiñones. 2024. Sentiment Analysis on E-Commerce Product Reviews Using Machine Learning and Deep Learning Algorithms. *International Journal of Information Management Data Insights* 4, 2 (2024), 100267.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 4171–4186.

[8] Ruihai Dong, Michael P O'Mahony, Markus Schaal, Kevin McCarthy, and Barry Smyth. 2016. Combining similarity and sentiment in opinion mining for product recommendation. *Journal of Intelligent Information Systems* 46, 2 (2016), 285–312.

[9] C. I. Ezeife, Ritu Chaturvedi, Mahreen Nasir, and Vinay Manjunath. 2022. Mining Twitter Multi-word Product Opinions with Most Frequent Sequences of Aspect Terms. In *Information Integration and Web Intelligence*. Springer Nature Switzerland, Cham, 126–136.

[10] Wenjing Fu, Zhaohui Peng, Senzhang Wang, Yang Xu, and Jin Li. 2019. Deeply fusing reviews and contents for cold start users in cross-domain recommendation systems. In *Proceedings of the AAAI conference on artificial intelligence*. 94–101.

[11] Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 6894–6910.

[12] Alex Graves, Greg Wayne, and Ivo Danihelka. 2014. Neural turing machines. *arXiv preprint arXiv:1410.5401v2* (2014).

[13] Rajeev Kumar Gupta, Santosh Bharti, Nilesh Kunhare, Yatendra Sahu, and Nikhlesh Pathik. 2022. Brain tumor detection and classification using cycle generative adversarial networks. *Interdisciplinary Sciences: Computational Life Sciences* 14, 2 (2022), 485–502.

[14] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[15] Clayton Hutto and Eric Gilbert. 2014. Vader: A parsimonious rule-based model for sentiment analysis of social media text. *Proceedings of the international AAAI conference on web and social media* 8, 1 (2014), 216–225.

[16] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. *Advances in neural information processing systems* 33 (2020), 18661–18673.

[17] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[18] Longfei Li and Guanfeng Liu. 2021. A Unified Framework for Cross-Domain and Cross-System Recommendations. *CDR* 7 (2021), 8.

[19] Zewen Li, Fan Liu, Wenjie Yang, Shouheng Peng, and Jun Zhou. 2021. A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems* 33, 12 (2021), 6999–7019.

[20] Jun Liu, Shuang Zheng, Guangxia Xu, and Mingwei Lin. 2021. Cross-domain sentiment aware word embeddings for review sentiment analysis. *International Journal of Machine Learning and Cybernetics* 12 (2021), 343–354.

[21] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems* 26 (2013).

[22] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 1532–1543.

[23] Soujanya Poria, Iti Chaturvedi, Erik Cambria, and Federica Bisio. 2016. Sentic LDA: Improving on LDA with semantic similarity for aspect-based sentiment analysis. In *2016 international joint conference on neural networks (IJCNN)*. IEEE, 4465–4473.

[24] Flora Sakketou and Nicholas Ampazis. 2020. A constrained optimization algorithm for learning GloVe embeddings with semantic lexicons. *Knowledge-Based Systems* 195 (2020), 105628.

[25] Marco Siino. 2024. Deberta at semeval-2024 task 9: Using deberta for defying common sense. In *Proceedings Of The 18th International Workshop On Semantic Evaluation (SemEval-2024)*. 291–297.

[26] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. 2012. Practical bayesian optimization of machine learning algorithms. *Advances in neural information processing systems* 25 (2012).

[27] Dana Utebayeva, Lyazzat Ilipbayeva, and Eric T Matson. 2022. Practical study of recurrent neural networks for efficient real-time drone sound detection: A review. *Drones* 7, 1 (2022), 26.

[28] Yongpeng Wang, Hong Yu, Guoyin Wang, and Yongfang Xie. 2020. Cross-domain recommendation based on sentiment analysis and latent feature mapping. *Entropy* 22, 4 (2020), 473.

[29] Jason Weston, Sumit Chopra, and Antoine Bordes. 2014. Memory networks. *arXiv preprint arXiv:1410.3916* (2014).

[30] Xiaohui Yan, Jiafeng Guo, Yanyan Lan, and Xueqi Cheng. 2013. A biterm topic model for short texts. In *Proceedings of the 22nd international conference on World Wide Web*. 1445–1456.

[31] Kevin K Yang, Zachary Wu, Claire N Bedbrook, and Frances H Arnold. 2018. Learned protein embeddings for machine learning. *Bioinformatics* 34, 15 (2018), 2642–2648.

[32] Tianzi Zang, Yanmin Zhu, Haobing Liu, Ruohan Zhang, and Jiadi Yu. 2022. A survey on cross-domain recommendation: taxonomies, methods, and future directions. *ACM Transactions on Information Systems* 41, 2 (2022), 1–39.

[33] Zohaib Zeeshan. 2025. *Cross-Domain Recommendations Via Aspect Sentiment Feature Extraction using Large Language Models (CRAS-LLM)*. Master's thesis. University of Windsor (Canada).

[34] Fan Zhang, Yaoyao Zhou, Pengfei Sun, Yi Xu, Wanjiang Han, Hongben Huang, and Jinpeng Chen. 2024. CRAS: Cross-domain recommendation via aspect-level sentiment extraction. *Knowledge and Information Systems* 66, 9 (2024), 5459–5477.

[35] Feng Zhu, Yan Wang, Jun Zhou, Chaochao Chen, Longfei Li, and Guanfeng Liu. 2021. A unified framework for cross-domain and cross-system recommendations. *IEEE Transactions on Knowledge and Data Engineering* (2021).