

A Conceptual and Computational Framework for Aspect-Based Collaborative Filtering Recommender Systems

Samin Poudel, Marwan Bikdash

Department of Computational Data Science and Engineering, North Carolina A & T University, Greensboro, North Carolina, USA

Email: spoudel@aggies.ncat.edu

How to cite this paper: Poudel, S. and Bikdash, M. (2023) A Conceptual and Computational Framework for Aspect-Based Collaborative Filtering Recommender Systems. *Journal of Computer and Communications*, 11, 110-130.

<https://doi.org/10.4236/jcc.2023.113009>

Received: February 24, 2023

Accepted: March 28, 2023

Published: March 31, 2023

Copyright © 2023 by author(s) and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Many datasets in E-commerce have rich information about items and users who purchase or rate them. This information can enable advanced machine learning algorithms to extract and assign user sentiments to various aspects of the items thus leading to more sophisticated and justifiable recommendations. However, most Collaborative Filtering (CF) techniques rely mainly on the overall preferences of users toward items only. And there is lack of conceptual and computational framework that enables an understandable aspect-based AI approach to recommending items to users. In this paper, we propose concepts and computational tools that can sharpen the logic of recommendations and that rely on users' sentiments along various aspects of items. These concepts include: The sentiment of a user towards a specific aspect of a specific item, the emphasis that a given user places on a specific aspect in general, the popularity and controversy of an aspect among groups of users, clusters of users emphasizing a given aspect, clusters of items that are popular among a group of users and so forth. The framework introduced in this study is developed in terms of user emphasis, aspect popularity, aspect controversy, and users and items similarity. Towards this end, we introduce the Aspect-Based Collaborative Filtering Toolbox (ABCFT), where the tools are all developed based on the three-index sentiment tensor with the indices being the user, item, and aspect. The toolbox computes solutions to the questions alluded to above. We illustrate the methodology using a hotel review dataset having around 6000 users, 400 hotels and 6 aspects.

Keywords

Recommender System, Collaborative Filtering, Aspect based recommendation, Recommendation System Framework, Aspect Sentiments

1. Introduction

The ever-expanding growth of the e-commerce websites and applications have enriched the recommender dataset with the aspect-level data [1] [2] [3] [4]. Many recommender datasets with ample information about the preferences of many users towards various aspects of many items are available [5] [6]. The aspect related information is either explicitly mentioned or implicit, mainly hidden in the texts such as reviews [7] [8] [9]. NLP algorithms are well advanced to extract the users' aspect-sentiments from the texts [10] [11]. Different Aspect Based Sentiment Analysis (ABSA) algorithms are regularly developed to mine the opinions of users towards different aspects of items [10] [12] [13] [14].

Despite the availability of detailed sentiment data, most of the CF Recommender Systems (RS) are still based on overall ratings [15] [16] [17]. Collaborative Filtering (CF) techniques typically recommend relevant items to a user based upon the overall preferences of other users towards the items [18] [19], in part because a theoretical framework for using user sentiments towards aspects has not been developed.

Aspect-sentiment based studies generally mine the opinion of users towards different aspects of items and present the mined results [20] [21]. The extracted aspects and the related sentiments are left without further analysis as in the two studies in [21] and [22]. In [23], the authors perform sentiment analysis of reviews to identify the nearest-neighbor items in terms of aspect sentiments but no discussion of emphasis on aspects or of the popularity of aspects was included, and further analysis of aspect-sentiments was not conducted. In [24], the authors extract aspect-level preferences of a user from the reviews, then compare the users' aspect-level preferences with aspect-level details of a review to score the helpfulness of the review and subsequently recommend the reviews based on their helpfulness score. The impact on the item-level aspect sentiments, popularity of aspects, or on item recommendation was not included.

Aspect-based information was also discussed in [25] [26] [27], but their implications to recommendation systems were not fully explored. In [28], the authors consider the popularity of each aspect of an item during recommendation. But the study does not discuss the impact of user's emphasis on aspects and does not consider the aspect-sentiment of a user towards specific aspects of an item. In [25] and [29], the authors introduce approaches to compute the weighted aspect ratings which are then used to infer a user's overall rating toward an item, but does not involve the analysis of popularity of an aspect of an item. The authors in [30] consider the users' emphasis on aspect and average sentiment of all users towards product of an item for recommendation but do not consider individual users' sentiment towards an aspect of an item. The authors in [8] propose a CF RS relying on the user's experience with the aspects of particular items but not with the overall emphasis of a user on an aspect and it is also missing the involvement of overall popularity of an aspect of an item.

The impact of aspect sentiment on recommendation was explored in several

studies, but incompletely [31] [32]. In [33], a method was proposed to include the sentiment-based explanation of features of items, so that users can make better choices, but the emphasis of a user towards an aspect and the sentiment of a user toward a particular aspect of an item were not used, which we consider in this paper as a crucial ingredient in making recommendation based on the available aspect sentiment. Similarly for [34], where user-level and item-level importance of aspects were discussed but the sentiments themselves were not used. [35] combines the aspect-level popularity of items with the importance of an aspect to a user to make recommendations, but does not use the user sentiments. The study in [36] determines whether a user is influenced more by positive or negative opinions, then combines the influence score with the item-level aspect importance to rank items. Authors in [7] consider aspect emphasis and item-aspect availability but do not make use of aspect-sentiments. The study in [37] uses the aspect-utility and the aspect-importance values to predict an overall rating towards an item from a review, but the users and items are not related via user sentiments.

The above discussion suggests that researchers in this field are exploring the relationships and uses of several key concepts in the recommendation problem, but no clear framework that unifies these key concepts has been developed. In this paper, we propose and develop such a framework, mainly by showing that the key concepts can be all computed and related once the 3-index Aspect-Sentiment Tensor (AST) $s(u, i, a)$ is defined or sampled, where s , u , i , and a denote, sentiment, user, item, and aspect, respectively. Subsequently, we will define, in terms of the AST, concepts like popularity, emphasis, controversy, and similarity of users, items, and aspects. The relationships between the concepts will be made clearer as they are all derived in a consistent manner from an underlying sentiment tensor.

Finally, we end up with an Aspect-based Collaborative Filtering Toolbox (ABCFT) that simplifies the process of developing aspect-based CF approaches and has an extensive potential of developing an explanatory aspect-based CF recommender system. We encourage to extend ABCFT with additional tools so that we together can speed up involving aspect-based information to make more justifiable recommendations.

2. Notation and Concepts

Common notations used in this study are as listed below:

- 1) Capital letters represent sets, matrices, or tensors. And the notation $E(u, a) = e_{ua}$ to indicate their elements.
- 2) Small letters represent elements in sets or matrices. When sub-scripted or super-scripted, they represent the elements in the matrices or tensors. For example, e_{ua} represents the emphasis score of aspect a according to user u and is the element in the corresponding matrix E .
- 3) $|S|$ represents the number of elements in the set S .

4) R is the Rating Matrix of size m by n , where m is the total number of users and n is the total number of items. It is further discussed in Section 2.1.

5) S is the Sentiment Tensor of size m by n by k , where m is the total number of users, n is the total number of items and k is the total number of aspects. It is further discussed in Section 2.1.

6) $U = \{u_1, u_2, \dots, u_m\}$ = the set of all the users in S .

7) $I = \{i_1, i_2, \dots, i_n\}$ = the set of all the items in S .

8) $A = \{a_1, a_2, \dots, a_k\}$ = the set of all the aspects in S .

9) U' is a sub-set of users of U , I' is a sub-set of items of I , and A' is a sub-set of aspects of S .

10) U_i = the set of all users that have reviewed an item i .

11) I_u = the set of all the items reviewed by a user u .

12) A_i = the set of all aspects of an item i . In our study, usually $A_i = A$.

13) $L^+(a, i) = l_{ai}^+$ = the popularity of an aspect a of the item i . It is discussed in Section 2.2.1.

14) $E(u, a) = e_{ua}$ represents the emphasis score of a user u on an aspect a . It is discussed in Section 2.4.

15) $d(u, v | i)$ is the distance between users u and v based on item i .

16) $d(u, v | a)$ is the distance between users u and v based on aspect a .

17) $C^{U'}$ is a set of clusters of users in U' .

2.1. Rating Matrix and Aspect Sentiment Tensor

A rating matrix R is the matrix (aspect-free) of m users and n items as in the classical recommender systems and $R(u, i) = r_{ui}$ represents the rating of user u on item i [16]. Generally, the ratings in a rating matrix are discrete numerical values.

A sentiment tensor S is a three-index tensor of m users, n items and k aspects. Here, $S(u, i, a) = s_{ui}^a$ denotes the sentiment of user u about an item i along an aspect a . A value in S is either +1 or 0 or -1. 1 represents positive sentiment, -1 represents negative sentiment and 0 represents no sentiment.

2.2. Popularity and Controversy

There may be different ways to define the popularity and controversy of an aspect of an item. Simple definitions of popularity and controversy of an aspect of an item, computable using the Sentiment Tensor S are as explained in Sections 2.2.1 and 2.2.2.

The popularity of aspects of a specific item can have a significant role in building an aspect-driven recommender system but are rarely used [38]. The aspect-level popularity of an item can be combined with aspect-level preference of users to build a CF based RS [35]. For an item i recommend to a user u , the popularity score of the aspects of that item can be used as the criteria to recommend the top aspects of that item to the user u .

2.2.1. Popularity of Aspects and Items

The popularity $l_{ai}^+ \in [0, 1]$ of an aspect a of an item i is defined as the proportion

of users reviewing aspect a of item i positively. It can be interpreted as the probability of assigning a positive sentiment to that aspect by a randomly selected user, given that the selected user reviewed the aspect. $\Pr(s^+ | a, i)$ denotes this probability and can be estimated as follows:

U_i = set of all users that have reviewed item i .

$n_{ai}^+ = \left| \sum_{u \in U_i} S_{ui}^a = 1 \right|$ represents the number of users who have rated aspect a of item i positively.

$n_{ai}^- = \left| \sum_{u \in U_i} S_{ui}^a = -1 \right|$ represents the number of users who have rated aspect a of item i negatively.

$n_{ai}^0 = \left| \sum_{u \in U_i} S_{ui}^a = 0 \right|$ represents the number of users who have not rated aspect a of item i .

Then, the popularity l_{ai}^+ and its complement l_{ai}^- for an aspect a of an item i can be computed as:

$$l_{ai}^+ = \frac{n_{ai}^+}{n_{ai}^+ + n_{ai}^-} \quad (1)$$

$$l_{ai}^- = \frac{n_{ai}^-}{n_{ai}^+ + n_{ai}^-} \quad (2)$$

Note that $l_{ai}^+ + l_{ai}^- = 1$. However, l_{ai}^+ does not represent the probability of assigning a positive sentiment to that aspect by a randomly selected user, because such a user may not rate the aspect of the item. To estimate the probability, we can correct with the probability that a random user rates the aspect. This probability is:

$$\Pr(s \neq 0 | a, i) = \frac{n_{ai}^+ + n_{ai}^-}{n_{ai}^+ + n_{ai}^- + n_{ai}^0} \quad (3)$$

For simplicity, we will use l_{ai}^+ computed on U_i as an estimate of $\Pr(s^+ | a, i)$ and $l_{ai}^- = 1 - l_{ai}^+$ as estimate of its complement.

2.2.2. Controversy of Aspects and Items

The Controversy measure of an aspect a of an item i , κ_{ai} is a measure of disagreement in sentiment between the users regarding an aspect a of an item i . Moreover, κ_{ai} lies in $[0, 1]$. Mathematically,

$$\kappa_{ai} = 1 - |l_{ai}^+ - l_{ai}^-| \quad (4)$$

Notice that if $l_{ai}^+ = l_{ai}^-$, meaning that an equal number of users liked and disliked that aspect, then κ_{ai} will be one, indicating maximum controversy. On the other hand, a complete agreement among users regarding the aspect a of item i gives $l_{ai}^+ = 0$ or $l_{ai}^- = 0$, then κ_{ai} will be zero, indicating no controversy, and hence consensus.

The most controversial aspect of an item i noted as $\text{ConAsp}(i)$ is the aspect a of item i having the highest controversy κ_{ai} among $a \in A_i$. The most controversial item based on an aspect a noted as $\text{ConItem}(a)$ is the item i having high-

est controversy score κ_{ai} among all $i \in I$.

2.3. Relationship of Users Based on Aspects

The relationship between two or more users can be assessed based on how they rate the aspects of items. For instance, users can be related based on sentiments towards all aspects of an item or based on their sentiments towards one aspect but considering all the items.

In general, positively biased users are the users tending to review every item or aspect under consideration positively. The Most Positive Users (MPU) about all aspects of an item i , noted as $MPU(i)$ are users who review most or all the aspects of item i positively.

Two users are said to be the most disagreeing users if they tend to review every item or aspect under consideration with extreme opposite values in the reviewing scale. The Most Disagreeing Users on a specific aspect a considering all the items, noted as $MDU(a)$ are the users who have reviewed aspect a with opposite sentiment to each other for most, if not all of the items.

The Nearest Neighbors to a user u based on an aspect a considering all the items is a set denoted as $NN(a)$ are the users who think most like the user u toward aspect a considering all the items.

In general, clustering of objects is the process of grouping objects in a way that objects belonging to the same group are more similar to each other based on certain criteria than to the objects in other groups [39]. In our study, a cluster of users expressing similar sentiments to all aspects of item i , denoted as $ClSent(U_i)$ is a group of users who have most similar sentiments to all aspects of item i . Similarly, a cluster of users emphasizing similar aspects of item i , denoted as $ClEmp(U_i)$ consists of users who emphasize similar aspects of item i .

2.4. Emphasis Score

The preference level of users toward different aspects is an important part of aspect-based CF approaches [40] [41] [42]. Usually, the importance of an aspect to a user is inferred from reviews and are involved in aspect-based CF as one of the latent factors [18] [43] [44] [45]. Here, we present a simple approach to compute emphasis of a user towards an aspect based on the information stored in Sentiment tensor S .

The emphasis score of a user u towards an aspect a , e_{ua} can be defined as the ratio of times the aspect a is reviewed by user u over the total number of items reviewed by u . The value of e_{ua} lies in $[0,1]$. Mathematically, the emphasis score of a user u toward an aspect a can be computed as:

$$E(u, a) = e_{ua} = \frac{1}{u} \sum_{i \in I_U} |s_{ui}^a| \quad (5)$$

2.5. Similarity and Dissimilarity between Users

Item-Based User Disagreement, $IBUD(u, v | i)$ between two users u and v is

the dissimilarity score between them based on their aspect-sentiments towards all aspects of item i . Similarly, Item-Based User Agreement, $IBUA(u, v | i)$ between two users u and v is the similarity score between them based on their aspect-sentiments towards all aspects of item i .

Aspect-Based User Disagreement, $ABUD(u, v | a)$ between two users u and v is the dissimilarity score between them based on their sentiments toward aspect a considering all the items. Similarly, Aspect-Based User Agreement, $ABUA(u, v | a)$ between two users u and v is the similarity score between them based on their sentiments toward an aspect a considering all the items.

In general, similarity between two data objects is a numerical measure to determine how alike they are [46]. And the value of similarity in general is in $[0, 1]$. And dissimilarity is a numerical measure to find different two data objects are. Dissimilarity or distance between two objects, not necessarily lie between $[0, 1]$ until normalized.

The distance or similarity measure between objects is a key step in data mining tasks like classification and clustering [47]. The distances may be computed in different ways based on the type of data we are dealing with. Distance between two numerical or ordinal vectors x and y can generally be defined by any mathematical norm for the difference vector $d(x, y) \in R^n$ [48]. The Minkowski distance of different orders can be used to compute the distance between vectors formed from the numerical and ordinal data [49]. Minkowski distance between the ordinal vectors x and y of order p can be computed as:

$$d(x, y) = \left(\sum_{i=1}^n |x_i - y_i|^p \right)^{\frac{1}{p}} \quad (6)$$

Minkowski distance of order 1 ($p = 1$) is Manhattan distance or 1-norm. And Minkowski distance of order 2 ($p = 2$) is the euclidean distance or 2-norm. K-means clustering which is one of the widely used unsupervised machine learning algorithms also uses the Minkowski distance of different order during clustering.

In this study, the distance between two users u and v based on item i , $d(u, v | i)$ is termed as Item-Based User Disagreement (IBUD). $IBUD(u, v | i)$ is computed based on Euclidean distance of aspect-sentiments of u and v based on all aspects in A of item i . Mathematically,

$$IBUD(u, v | i) = d(u, v | i) = \sqrt{\sum_{a \in A} (s_{ui}^a - s_{vi}^a)^2} \quad (7)$$

IBUD may be normalized as required by the problems. For a normalized IBUD denoted as $IBUD^*$ which lies in $[0, 1]$, we define Item-based User Agreement (IBUA) as $IBUA = 1 - IBUD^*$. In this work, the similarity between two users u and v based on an item i , $sim(u, v | i) = IBUA(u, v | i)$ is computed based on the aspect sentiments of users u and v towards all aspect in A' of item i . And the weight between two users u and v is defined as

$$w_{uv} = \sum_{i \in I_u \cap I_v} (u, v | i) \quad (8)$$

The distance between two users u and v based on aspect a , $d(u, v | a)$ is termed as Aspect-Based User Disagreement (ABUD). $ABUD(u, v | a)$ can be computed considering all the items in I as:

$$ABUD(u, v | a) = d(u, v | a) = \sqrt{\sum_{i \in I} (s_{ui}^a - s_{vi}^a)^2} \quad (9)$$

ABUD may be normalized as required by the problems. For a normalized ABUD denoted as $ABUD^*$ which lies in $[0, 1]$, we define Aspect-based User Agreement (ABUA) as $ABUA = 1 - ABUD^*$.

3. Methodology

In this Section, Aspect-Based Tools, their tasks, and process to solve the tasks are discussed. The algorithm of each tool is discussed in Section 4. Each tool presented here is a tool in the proposed Aspect-based Collaborative Filtering Toolbox (ABCFT). ABCFT can be used to build a complete aspect-based explanatory recommender system.

The list of eight Aspect-Based CF Tools is as below:

1) Determine the most controversial aspect a of an item i denoted as $ConAsp(i)$

The tool $ConAsp(i)$ determines the most controversial aspect a of an item i . This can be achieved by finding the aspect a of the item i with the highest controversial measure κ_{ai} or the lowest uncontroversial measure κ'_{ai} . The proposed algorithm is presented in Section 4.1.

2) Find the most controversial item i based on aspect a denoted as $ConItem(a)$

The tool $ConItem(a)$ finds the most controversial item i based on an aspect a . This is achieved by finding the item i with the highest controversial measure κ_{ai} or the lowest uncontroversial measure κ'_{ai} for specific aspect a . The proposed algorithm is presented in 4.2.

3) Determine users who are most positive about all aspects of an item i denoted as $MPU(A|i)$

The tool $MPU(A|i)$ finds the users who are most positive about all aspects in A of an item i . This is achieved by computing the dissimilarity of every user u with an assumed user u' who has got positive sentiments for all the aspects in A of the item i . Here, u belongs to the set of users reviewing item i i.e. $u \in U_i$. The users in U_i with least value of the defined measure of proximity with u' are most positive. The algorithm is as presented in Section 4.3.

4) Determine users who feel most like (agree with) specified user u' based on an aspect a denoted as $NN(u|a)$

The tool $NN(u|a)$ determines the users who feel most like (agree with) specified user u' based on an aspect a . This is achieved by computing the dissimilarity between user u and every other user based on their aspect-sentiments toward aspect a of all the items. The users with least dissimilarity with u' mostly agree with user u' based on aspect a . The algorithm is as presented in Section 4.4.

5) Determine pairs of users disagreeing most on a specific aspect a consider-

ing all the items denoted as $MDU(a|I)$

The tool $MDU(a|I)$ determine pairs of users disagreeing most on a specific aspect a considering all the items in I . This is achieved by computing dissimilarity between every unique pair of users u and u' , meaning $u \neq u'$. The dissimilarity is based on sentiments of u and u' towards aspect a considering all the items in I . The pairs of users with the highest value of dissimilarity are the pairs of users disagreeing most on a specific aspect a considering all the items in I . The algorithm is as presented in Section 4.5.

6) Find groups of users mostly agreeing on all aspects of an item i or find Aspect-Sentiment based User Clusters of a given item i , $ASBUC(U_i)$

The tool, $ASBUC(U_i)$ finds the groups of users mostly agreeing on all aspects of an item i . This is achieved by clustering the users reviewing item i based on the sentiment values users provide to all aspects of the item i . The algorithm is as presented in Section 4.6.

7) Find groups of users who emphasize the same aspects of an item i or Aspect-Emphasis based User Clusters of a given item i , $AEBUC(U_i)$

The tool $AEBUC(U_i)$ finds groups of users who emphasize the same aspects of an item i . This is achieved by clustering the users reviewing item i based on the sentiment values users provide to all aspect of the item i , but by treating the positive and negative sentiment as same. The algorithm is as presented in Section 4.7.

8) Rank the aspects based on the emphasis given by a user u to them or Emphasis based Ranking of Aspects in A for a given user u , $EBRA(A|u)$

The tool $EBRA(A|u)$ ranks all aspects in A based on the emphasis given by a user u to them. This is achieved by computing the emphasis score of a user u towards every aspect in A . Then, aspects are sorted descending based on their emphasis scores. The one with the highest value of emphasis score gets the rank one and so on. The algorithm is as presented in Section 4.8.

The tools in ABCFT, their tasks and the concepts used in each tool are summarized in **Table 1**.

4. Algorithms and Illustrations

In this section, the algorithms for the aspect-based CF tools proposed in Section 3 are presented. And example solutions of the implementation of the tools to a Hotel dataset are provided. The Hotel dataset [44] [45] involves around 6000 users and 400 hotels from Tripadvisor. Hotel dataset was reformatted to an aspect-sentiment tensor made up of six aspect-sentiment matrices. The sentiment values in hotel data sentiment tensor are +1 for positive sentiment, -1 for negative sentiment and 0 for no sentiment. In the hotel dataset downloaded from [50], aspects were rated in the discrete values from 1 to 5. Aspect-ratings were converted to aspect-sentiments based on the condition, if aspect-rating > 3.0 then aspect-sentiment = positive (1.0) and if aspect rating ≤ 3.0 then aspect-sentiment = negative (-1.0). The aspects involved are Location, Service,

Table 1. Summary of tools in aspect-based CF toolbox.

Tool	Task	Concepts and Terminologies Used
ConAsp(i)	To determine the most controversial aspect a of an item i	Popularity and Controversy, AST
ConItem(a)	To find the most controversial item i based on an aspect a	Popularity and Controversy, AST
MPU($A i$)	To find the users who are most positive about all aspects in A of an item i	IBUD, AST
NN($u a$)	To determine the users who feel most like specified user u based on an aspect a	ABUD, ABUA, AST
MDU($a i$)	To determine pairs of users disagreeing most on a specific aspect a considering all the items in i	ABUD, AST
ASBUC(U)	To find the groups of users mostly agreeing on all aspects of an item i	ABUD, ABUA, AST
AEBUC(U)	To find the groups of users who emphasize the same aspects of an item i	ABUD, ABUA, AST
EBRA($A u$)	To rank all aspects in A based on the emphasis given by a user u to them	Emphasis Score, AST

Cleanliness, Value, Sleep Quality, and Rooms.

Proposed Algorithms for the tools in ABCFT are to follow. All the tools assume the availability of the sentiment tensor S where $S(u, i, a)$ represents the sentiment of user u about an aspect a of an item i .

4.1. Determine the Most Controversial Aspect a of an Item i

The algorithm for finding the most controversial aspect a of an item i noted as $\text{ConAsp}(i)$ is as below:

- 1) For each aspect a of item i ,
 - a) Compute popularity l_{ai}^+ using Equation (1) and its compliment l_{ai}^- using Equation (2).
 - b) Compute controversy κ_{ai} of the aspect a of the item i using Equation (4).
- 2) The most controversial aspect a of item: $\text{ConAsp}(i) = \text{aspect with the maximum } \kappa_{ai}$.

The idea of finding the most controversial aspect may look like a simpler problem compared to the big machine learning problems in the Recommendation Systems. But the solution of this problem can play a vital role in making meaningful recommendations to the users when combined with other solutions.

The example in **Table 2** gives the controversy κ_{ai} of aspects of the item 0. This example is based on the Hotel dataset used in this study. The aspect Value is the most controversial aspect of hotel 0, because it has the highest controversial measure κ_{ai} among the six aspects of the hotels under consideration.

4.2. Find the Most Controversial Item i Based on Aspect a

The algorithm for finding the most controversial item i based on an aspect a noted as $\text{ConItem}(a)$ is as below:

- 1) For each item i ,
 - a) Compute popularity l_{ai}^+ using Equation (1) and its compliment l_{ai}^- using Equation (2) for specific aspect a .
 - b) Compute controversy κ_{ai} of the specific aspect a of item i using Equation (4) and store it.

2) The most controversial item i based on specific aspect a : $\text{ConItem}(a)$ = item with the maximum κ_{ai} for specific aspect a .

The solution to the problem of finding the most controversial item based on an aspect a can also be a solution to the challenge of recommending items to new users. The most controversial items can be avoided while recommending to new users with insufficient rating or sentiment data.

For the hotel data used in this study, item 300 is the most controversial item based on aspect Location. The result is as disclosed in **Table 3**.

4.3. Determine the Top N Users Who Are Most Positive about All Aspects of an Item i

Let U_i be the set of all users that have reviewed the item i . The proposed algorithm

Table 2. Example solution of the approach in Section 4.1 for determining the most controversial aspect a of an item 0 using Hotel Dataset.

Aspect a	Controversy for item 0, κ_{a0}	Number of Users with positive sentiment towards aspect a of item 0, n_{ai}^+	Number of Users with negative sentiment towards aspect a of item 0, n_{ai}^-
Value	0.69	59	31
Service	0.58	62	25
Rooms	0.54	63	23
Sleep Quality	0.46	34	10
Cleanliness	0.42	72	19
Location	0	87	0

Table 3. Example solution of the approach in Section 4.2 for finding the most controversial item i based on aspect a using Hotel Dataset.

Item i	Controversy of aspect Location of item i $\kappa_{\text{Location } i}$	Number of Users with positive sentiment towards aspect Location of item i	Number of Users with negative sentiment towards aspect Location of item i
300	0.98	22	21
162	0.96	25	23
393	0.95	43	39
76	0.94	40	36
325	0.94	70	63

to solve the problem of finding the top N users who are most positive about all aspects of an item i as below:

- 1) Assume a reference user u' as a user who has got positive sentiments for all the aspects of the item i .
- 2) Compute the dissimilarity, Item-Based User Disagreement $IBUD(u, v | i)$ of each user in U_i with the assumed user u' using Equation (7).
- 3) Sort the users in U_i based on the dissimilarity values $IBUD(u', u | i)$ ascendingly.
- 4) $MPU(A | i) = \text{top } N \text{ sorted users of } U_i$. Here, $MPU(A | i)$ are the N users that are the N most positive about all aspects of the item i .

Here, in **Table 4**, we give an example solution of top 5 most positive users about all aspects of item 7. This example is based on Hotel Dataset.

Based on the proposed solution for finding the top N users who are most positive about all aspect of item i , a hypothesis can be proposed as:

Hypothesis1: The top N users who are most positive about $k - 1$ aspects of an item are likely to be positive about the k^{th} aspect, which has not been used for finding the top N most positive users.

We evaluated Hypothesis1 by introducing an approach called Leave One Aspect Out. The steps involved during this evaluation are as follows:

$I = \text{set of all items,}$

$A = \text{set of all aspects.}$

1) For an item i in I ,

a) $U_i = \text{set of all users that have reviewed item } i$

b) For an aspect a in A ,

i) Split the data into training and test data,

Let, $A' = A - a$,

Trn = sub-tensor $S(u, i, a')$ for all $u \in U_i, a' \in A'$ is training data.

Tst = sub-tensor $S(u, i, a)$ for all $u \in U_i$ is testing data.

ii) Let MPU = set of N most positive users about all aspects in A' found using approach in Section 4.3.

iii) Find $n_{ai}^+ | \text{MPU}$ = number of users with positive sentiment towards aspect a of item i for all $u \in \text{MPU}$.

iv) Find $n_{ai}^- | \text{MPU}$ = number of users with negative sentiment towards

Table 4. Example solution of the approach in Section 4.3 for determining the top N users who are most positive about all aspects of an item i based on Hotel Dataset.

User	Distance from perfect positivity based on item 7, $IBUD(u', u 7)$
436	0
117	0
432	0
442	0
448	0

aspect a of item i for all $u \in \text{MPU}$.

v) Accuracy [51] = $\frac{n_{ai}^+ | \text{MPU}}{n_{ai}^+ | \text{MPU} + n_{ai}^- | \text{MPU}}$, accuracy obtained is store in a list acCL.

c) Step 1b is performed for each aspect in A .

2) Step 1 is performed for each item in I

3) Overall accuracy measure = arithmetic means of accuracy values in acCL of all aspects and items

A sentiment s_{ui}^a of a user $u \in \text{MPU}$ for aspect a in the test set is not considered during evaluation if sentiment $s_{ui}^a = 0$.

The results of Evaluation of Hypotheis 1 using the Hotel Dataset are tabulated in **Table 5**.

4.4. Determine the Top N Users Who Feel Most Like a Specified User u' Based on an Aspect a

The algorithm to find top N users who feel most like a specified user u' based on an aspect a is as follows:

1) Find $I_{u'}$ = the set of all items reviewed by user u'

2) Find U' = the set all users reviewing at least one item in $I_{u'}$.

3) For all $u \in U'$, compute Aspect-Based User Disagreement $\text{ABUD}(u', u | a)$ between users u' and u based on sentiment toward aspect a considering items rated both and normalize by common number of items reviewed by users u' and u . Normalized $\text{ABUD}(u', u | a)$ can be computed as:

$$\text{ABUD}^*(u', u | a) = \frac{1}{|I_u \cap I_{u'}|} \sqrt{\sum_{i \in I_u \cap I_{u'}} (s_{u'i}^a - s_{ui}^a)^2} \quad (10)$$

where I_u is the set of items reviewed by u ,

$I_{u'}$ is the set of items reviewed by u' and,

$|I_u \cap I_{u'}|$ is the cardinal number of set of items reviewed by both I_u and $I_{u'}$.

4) Sort $\text{ABUD}^*(u', u | a)$ ascendingly and the user u associated with $\text{ABUD}^*(u', u | a)$.

5) N nearest neighbors to u' based on aspect a , $\text{NN}(u' | a) =$ top N ascendingly sorted users based on $\text{ABUD}^*(u', u | a)$. Hence the top N sorted users

Table 5. Evaluation of Hypothesis1 using Hotel Dataset.

	Top N most positive users considered = 10	Top N most positive users considered= 20
Number of users in MPU having positive sentiments toward the aspect in test set during evaluation process	1225	2859
Number of users in MPU having negative sentiments toward the aspect in test set during evaluation process	19,565	38,716
Overall Accuracy measure	0.94	0.93

based on the values of $ABUD^*(u', u | a)$ are the users who think most like u' based on an aspect a .

Table 6 gives an example solution for finding the top 5 users who feel most like the user 10 based on the aspect Rooms. This example is based on Hotel Dataset.

4.5. Determine the Top N Pair of Users Disagreeing Most on a Specific Aspect a Considering All the Items

Let $U =$ set of all the users and $I =$ set of all the items. Then, the top N pair of users disagreeing most on a specific aspect a considering all the items noted as $MDU(a | I)$ can be found using the following steps:

1) Compute Aspect-Based User Disagreement $ABUD(u', u | a)$ between users u' and u for $u \in U, u' \in U, u \neq u'$ considering the sentiments of a specific aspect a of all the items in I using Equation (9). In other words, compute distance between each pair of users in U considering sentiments of the specific aspect a of all items in I .

2) Sort pair of users (u, u') descendingly based on distances $ABUD(u', u | a)$. $MDU(a | I) =$ top N descendingly sorted (u, u') based on $ABUD(u', u | a)$. Hence, $MDU(a | I)$ are the pair of users who disagree most on the considered specific aspect a considering all the items in I .

Table 7 gives an example solution for finding the top 5 pairs of users who

Table 6. Example solution of the approach in Section 4.4 for finding the top N users who feel most like a specified user u' based on an aspect a based on Hotel Dataset.

User u	Dissimilarity between user 10 and user u , based on aspect Rooms, $ABUD^*(10, u a)$	Number of common items reviewed by the user 10 and user u , $ I_u \cap I_{u'} $
4252	0	5
1023	0	5
2376	0	4
2710	0	4
986	0	4

Table 7. Example solution of the approach in Section 4.5 for determining the top N pair of users disagreeing most on a specific aspect a considering all the items based on Hotel Dataset.

User u	User u'	$ABUD(u', u a)$
1201	1246	7.62
3170	3698	7.62
1006	2032	7.48
498	1006	7.48
1409	3170	7.42

disagree most on aspect Location considering all the items based on the Hotel dataset.

4.6. Find the Groups of Users Who Are Most Similar in All Aspects of an Item i

To find the groups of users who are most similar in all aspects of an item i , we can cluster the users based on the sentiment values users provided to all aspects of item i . The K-means clustering algorithm for finding the groups of users who are most similar in all aspects of an item i is used as follows.

- 1) Find U_i = the set of users that have reviewed item i .
- 2) Cluster the users in U_i based on their sentiments toward all aspects of item i using K-means clustering.

This algorithm uses Item-Based User Disagreement $IBUD(u, u' | i)$ computed using Equation (7) during K-means clustering.

Clusters obtained are the groups of users who are most similar in all the aspects of an item i .

Figure 1 provides an example solution of finding the groups of users who are most similar in all aspects of an item i . This example is based on Hotel Dataset. One can see 5 clusters or groups of users who are most similar in all aspects of item 99.

4.7. Find the Groups of Users Who Emphasize the Same Aspects of Item i

The group of users who emphasize the same aspects of an item i can be found by clustering the users reviewing the same aspects of item i . In our approach, we cluster the users by treating positive and negative sentiment as same. We use K-means clustering algorithm to find users who emphasize same aspect of item i as follows:

- 1) Find U_i = the set of the users that have reviewed the item i . Then,
- 2) Cluster the users in U_i based on their sentiments toward all aspects of the item i , but by treating positive and negative sentiment as same. Here, the K-means clustering is performed using distance $IBUD(u, u' | i)$ based on the absolute value of the aspect sentiments. Equation (7) is modified as below to compute modified $IBUD(u, u' | i)$.

$$IBUD(u, u' | i) = \sqrt{\sum_{a \in A} (|s_{ui}^a| - |s_{u'i}^a|)^2} \quad (11)$$

Cluster of most similar users based on all aspects of item 99 :

Cluster 0 : [4230, 2213, 2653, 3734, 4247, 4250, 3873, 759, 4258, 1050, 4259, 4110, 261, 4046, 2629, 4268, 179, 4270, 3514, 4273, 1340, 1229, 2836, 1762, 2843, 4280, 4284, 2864]
 Cluster 1 : [4229, 2523, 4252, 229, 4255, 3837, 2526, 4256, 663, 4235, 4260, 4262, 658, 3829, 4265, 4266, 4269, 1787, 4271, 4272, 719, 4274, 4275, 2595, 4277, 568, 3657, 1452, 4281, 2428, 671, 4286, 4287, 10]
 Cluster 2 : [4225, 3732, 3161, 243, 4174, 2585, 4144, 996, 7, 4248, 4238, 3045, 2813, 3874, 247, 4267, 588, 3492, 3100, 724, 888, 4278, 3373, 4279, 2869, 3880, 2311]
 Cluster 3 : [151, 2698, 4227, 4228, 496, 3550, 4231, 3456, 4140, 4206, 517, 3141, 4233, 4236, 4237, 4239, 4240, 4241, 4242, 311, 3753, 4245, 4246, 1019, 1022, 2371, 4098, 4131, 4251, 991, 3155, 3340, 2338]
 Cluster 4 : [4226, 4232, 4243, 4244, 3960, 4234, 2586, 1357, 2535, 224, 4249, 39, 2174, 3487, 886, 4253, 4254, 4257, 1129, 2574, 4261, 4263, 4264, 3560, 2335, 2640, 3508, 222, 4276, 2598, 4282, 4283, 4285, 2026]

Figure 1. Example solution of finding the groups of users who are most similar in all aspects of an item i based on Hotel Dataset.

Figure 2 provides an example solution of finding the groups of users who emphasize the same aspects of item 99 in the Hotel Dataset.

4.8. Rank the Aspects Based on the Emphasis Given by User u to Them

The aspects of an item can be ranked based on the emphasis given by a user u to them using following steps:

- 1) Find $I_u =$ set of items reviewed by user u and $A =$ set of all aspects.
- 2) Compute the emphasis score of the user u toward each aspect a using Equation (5).
- 3) Sort $E(u, a)$ by aspects descendingly and rank. The aspect with the highest value of $E(u, a)$ will be the most emphasized aspect.

Table 8 presents an example solution for ranking the aspects based on emphasis given by user 10 to them. This example is based on the Hotel dataset. **Table 7** shows the emphasis score of user 10 towards each aspect in A . and indicates that user 10 gives strong emphasis to the aspects Service and Cleanliness whereas aspect Sleep Quality is of least emphasis to user 10.

Here, we presented eight aspect-based CF tools in ABCFT as a start of compiling the tools that can be extracted from Aspect-Sentiment Tensor. And, we would like to encourage exploring and adding the new tools to ABCFT, so the area of recommendation techniques using aspect-based information can grow rapidly.

5. Conclusions and Future Work

In this work, a general framework applicable to the future studies of aspect-based

Cluster of most similar users based on emphasis
given to aspects of item 99 :

Cluster 0 : [243, 4174, 2585, 4144, 7, 4247, 4248, 3873, 4253, 759, 1050, 4261, 247, 4266, 4267, 4268, 4269, 179, 4270, 4271, 3514, 4273, 4274, 3508, 4276, 4278, 2598, 3373, 4279, 4282, 4285, 2311, 4286]
 Cluster 1 : [4229, 4244, 2523, 224, 4254, 229, 4255, 3837, 2526, 4256, 4257, 663, 3874, 1129, 4259, 2574, 4235, 4260, 261, 4262, 658, 3829, 4263, 4264, 4265, 1787, 4272, 3560, 4275, 2595, 2335, 222, 4277, 568, 2836, 1452, 2843, 2428, 671, 4287, 10]
 Cluster 2 : [4225, 4230, 2213, 3734, 3161, 3960, 2535, 4131, 4250, 4238, 39, 2813, 4258, 4110, 4046, 2629, 588, 3492, 3100, 724, 719, 888, 1340, 2640, 1229, 3657, 1762, 4280, 4283, 4284, 2864, 2026]
 Cluster 3 : [3732, 311, 3753, 996, 4245, 4234, 4246, 2586, 4098, 991, 2338, 2869, 4281, 3880]
 Cluster 4 : [151, 2698, 4226, 4227, 4228, 496, 3550, 2653, 4231, 3456, 4140, 4232, 4206, 517, 3141, 4233, 4236, 4237, 4239, 4240, 4241, 4242, 4243, 1019, 1022, 1357, 2371, 4249, 4251, 3155, 3340, 2174, 3487, 3045, 4252, 886]

Figure 2. Example solution of finding the groups of users who emphasize the same aspects of an item i based on Hotel Dataset.

Table 8. Example solution of the approach in Section 4.8 for ranking the aspects based on emphasis given by a user u to them for Hotel Dataset.

Aspect a	Emphasis score, $e_{ua} = E(u, a)$	Rank
Service	1.0	1
Cleanliness	1.0	2
Rooms	0.9	3
Value	0.8	4
Location	0.6	5
Sleep Quality	0.5	6

Collaborative Filtering (CF) approaches is presented. We present an Aspect-Based Collaborative Filtering Toolbox (ABCFT) consisting of eight tools which can be developed based on Aspect-Sentiment Tensor (AST) only. Eight tools in ABCFT are the partial aspect-based CF problems that can be utilized to develop sophisticated aspect-based recommendation approaches. One goal of developing ABCFT is to ease the process of involving aspect-based information into the recommendation approaches, which can enhance the possibility of making rational recommendations to the users. ABCFT promotes the extensive use of aspect-sentiments extracted from well-advanced Aspect Sentiments Based Analysis (ABSA) techniques, which in general are just used surfacely and left after the extraction.

The use of ABCFT to develop new simple to complex aspect-based recommender systems is encouraged. And the use of ABCFT to improve the performance of current recommender systems can be explored. We initiated the work with 8 simple tools in ABCFT and the work of extension of ABCFT with additional tools can be persuaded to expedite the development of aspect-based recommender approaches.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] Pero, Š. and Horváth, T. (2013) Opinion-Driven Matrix Factorization for Rating Prediction. In: Carberry, S., Weibelzahl, S., Micarelli, A. and Semeraro, G., Eds., *User Modeling, Adaptation, and Personalization. UMAP 2013. Lecture Notes in Computer Science*, Vol. 7899, Springer, Berlin, 1-13.
https://doi.org/10.1007/978-3-642-38844-6_1
- [2] Diaz, G.O. and Ng, V. (2018) Modeling and Prediction of Online Product Review Helpfulness: A Survey. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Melbourne, 15-20 July 2018, 698-708.
- [3] Ray, B., Garain, A. and Sarkar, R. (2021) An Ensemble-Based Hotel Recommender System Using Sentiment Analysis and Aspect Categorization of Hotel Reviews. *Applied Soft Computing*, **98**, Article ID: 106935.
<https://doi.org/10.1016/j.asoc.2020.106935>
- [4] Poudel, S. and Bikdash, M. (2022) Collaborative Filtering System Based on Multi-Level User Clustering and Aspect Sentiment. *Data and Information Management*, **6**, Article ID: 100021. <https://doi.org/10.1016/j.dim.2022.100021>
- [5] Krestel, R. and Dokoohaki, N. (2011) Diversifying Product Review Rankings: Getting the Full Picture. 2011 *IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*, Lyon, 22-27 August 2011, 138-145.
<https://doi.org/10.1109/WI-IAT.2011.33>
- [6] Chen, Y.Y. (2019) Aspect-Based Sentiment Analysis for Social Recommender Systems. Robert Gordon University, Aberdeen.
<https://rgu-repository.worktribe.com/output/638015/aspect-based-sentiment-analysis>

[is-for-social-recommender-systems](#)

- [7] He, X., Chen, T., Kan, M.-Y. and Chen, X. (2015) TriRank: Review-Aware Explainable Recommendation by Modeling Aspects. *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, Melbourne, 18-23 October 2015, 1661-1670, <https://doi.org/10.1145/2806416.2806504>
- [8] Bauman, K., Liu, B. and Tuzhilin, A. (2017) Aspect Based Recommendations: Recommending Items with the Most Valuable Aspects Based on User Reviews. *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Halifax, 13-17 August 2017, 717-725, <https://doi.org/10.1145/3097983.3098170>
- [9] Diao, Q., Qiu, M., Wu, C.-Y., Smola, A.J., Jiang, J. and Wang, C. (2014) Jointly Modeling Aspects, Ratings and Sentiments for Movie Recommendation (JMARS). *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, 24-27 August 2014, 193-202. <https://doi.org/10.1145/2623330.2623758>
- [10] Yadav, K. (2020) A Comprehensive Survey on Aspect Based Sentiment Analysis. ArXiv: 2006.04611.
- [11] Hernández-Rubio, M., Cantador, I. and Bellogín, A. (2019) A Comparative Analysis of Recommender Systems Based on Item Aspect Opinions Extracted From User Reviews. *User Modeling and User-Adapted Interaction*, **29**, 381-441. <https://doi.org/10.1007/s11257-018-9214-9>
- [12] Meyer, B.B. (2017) Using Syntactic Patterns to Enhance Text Analytics. North Carolina Agricultural and Technical State University, Greensboro, 139.
- [13] Laskari, N.K. and Sanampudi, S.K. (2016) Aspect Based Sentiment Analysis Survey Deep Learning for NLP and IR View Project Extraction of Events, Times from Natural Language Text and Mapping of the Relations between Them View Project Aspect Based Sentiment Analysis Survey. *IOSR Journal of Computer Engineering*, **18**, 24-28.
- [14] Mowlaei, M.E., Saniee Abadeh, M. and Keshavarz, H. (2020) Aspect-Based Sentiment Analysis Using Adaptive Aspect-Based Lexicons. *Expert Systems with Applications*, **148**, Article ID: 113234. <https://doi.org/10.1016/j.eswa.2020.113234>
- [15] Lei, X., Qian, X. and Zhao, G. (2016) Rating Prediction Based on Social Sentiment from Textual Reviews. *IEEE Transactions on Multimedia*, **18**, 1910-1921. <https://doi.org/10.1109/TMM.2016.2575738>
- [16] Poudel, S. and Bikdash, M. (2022) Optimal Dependence of Performance and Efficiency of Collaborative Filtering on Random Stratified Subsampling. *Big Data Mining and Analytics*, **5**, 192-205. <https://doi.org/10.26599/BDMA.2021.9020032>
- [17] Poudel, S. and Bikdash, M. (2023) Closed-Form Models of Accuracy Loss due to Subsampling in SVD Collaborative Filtering. *Big Data Mining and Analytics*, **6**, 72-84. <https://doi.org/10.26599/BDMA.2022.9020024>
- [18] Nie, Y.P., Liu, Y. and Yu, X. (2014) Weighted Aspect-Based Collaborative Filtering. *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, Gold Coast, 6-11 July 2014, 1071-1074. <https://doi.org/10.1145/2600428.2609512>
- [19] Pawlicka, A., Pawlicki, M., Kozik, R. and Choraś, R.S. (2021) A Systematic Review of Recommender Systems and Their Applications in Cybersecurity. *Sensors*, **21**, Article No. 15. <https://doi.org/10.3390/s21155248>
- [20] Barrière, V. and Kembellec, G. (2018) Short Review of Sentiment-Based Recom-

- mender Systems. *Proceedings of the 1st International Conference on Digital Tools & Uses Congress*, Paris, 3-5 October 2018, 1-4.
<https://doi.org/10.1145/3240117.3240120>
- [21] Moghaddam, S. and Ester, M. (2010) Opinion Digger: An Unsupervised Opinion Miner from Unstructured Product Reviews. *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, Toronto, 26-30 October 2010, 1825-1828. <https://doi.org/10.1145/1871437.1871739>
- [22] Musto, C., Rossiello, G., De Gemmis, M., Lops, P. and Semeraro, G. (2019) Combining Text Summarization and Aspect-Based Sentiment Analysis of Users' Reviews to Justify Recommendations. *Proceedings of the 13th ACM Conference on Recommender Systems*, Copenhagen, 16-20 September 2019, 383-387.
<https://doi.org/10.1145/3298689.3347024>
- [23] D'Addio, R.M. and Manzato, M.G. (2015) A Sentiment-Based Item Description Approach for KNN Collaborative Filtering. *Proceedings of the 30th Annual ACM Symposium on Applied Computing*, Salamanca, 13-17 April 2015, 1060-1065.
<https://doi.org/10.1145/2695664.2695747>
- [24] Huang, C., Jiang, W., Wu, J. and Wang, G. (2020) Personalized Review Recommendation Based on Users' Aspect Sentiment. *ACM Transactions on Internet Technology*, **20**, Article No. 42. <https://doi.org/10.1145/3414841>
- [25] Wang, Y., Yang, C., Yu, X., Liu, Y. and Nie, Y. (2016) Collaborative Filtering With Weighted Opinion Aspects. *Neurocomputing*, **210**, 185-196.
<https://doi.org/10.1016/j.neucom.2015.12.136>
- [26] Bai, P., Xia, Y. and Xia, Y. (2020) Fusing Knowledge and Aspect Sentiment for Explainable Recommendation. *IEEE Access*, **8**, 137150-137160.
<https://doi.org/10.1109/ACCESS.2020.3012347>
- [27] McAuley, J., Leskovec, J. and Jurafsky, D. (2012) Learning Attitudes and Attributes from Multi-aspect Reviews. 2012 *IEEE 12th International Conference on Data Mining*, Brussels, 10-13 December 2012, 1020-1025.
<https://doi.org/10.1109/ICDM.2012.110>
- [28] Du, Q., Zhu, D. and Duan, W. (2021) Recommendation System with Aspect-Based Sentiment Analysis. Technology Report. <http://www.yelp.com/dataset>
- [29] Da'u, A., Salim, N., Rabi'u, I. and Osman, A. (2020) Weighted Aspect-Based Opinion Mining Using Deep Learning for Recommender System. *Expert Systems with Applications*, **140**, Article ID: 112871. <https://doi.org/10.1016/j.eswa.2019.112871>
- [30] Zhang, Y., Liu, R. and Li, A. (2015) A Novel Approach to Recommender System Based on Aspect-Level Sentiment Analysis. *Proceedings of the 2015 4th National Conference on Electrical, Electronics and Computer Engineering*, Xi'an, 12-13 December 2015, 1453-1458. <https://doi.org/10.2991/nceee-15.2016.259>
- [31] Zhang, J., Chen, D. and Lu, M. (2018) Combining Sentiment Analysis With a Fuzzy Kano Model for Product Aspect Preference Recommendation. *IEEE Access*, **6**, 59163-59172. <https://doi.org/10.1109/ACCESS.2018.2875026>
- [32] Li, W. and Xu, B. (2020) Aspect-Based Fashion Recommendation with Attention Mechanism. *IEEE Access*, **8**, 141814-141823.
<https://doi.org/10.1109/ACCESS.2020.3013639>
- [33] Chen, L., Yan, D. and Wang, F. (2019) User Evaluations on Sentiment-based Recommendation Explanations. *ACM Transactions on Interactive Intelligent Systems*, **9**, 1-38. <https://doi.org/10.1145/3282878>
- [34] Chin, J.Y., Joty, S., Zhao, K. and Cong, G. (2018) ANR: Aspect-Based Neural Recommender. *Proceedings of the 27th ACM International Conference on Informa-*

- tion and Knowledge Management*, Torino, 22-26 October 2018, 147-156.
<https://doi.org/10.1145/3269206.3271810>
- [35] Li, G., Chen, Q., Zheng, B., Hung, N.Q.V., Zhou, P. and Liu, G. (2020) Time-Aspect-Sentiment Recommendation Models Based on Novel Similarity Measure Methods. *ACM Transactions on the Web*, **14**, 1-26. <https://doi.org/10.1145/3375548>
- [36] Serrano-Guerrero, J., Olivas, J.A. and Romero, F.P. (2020) A T1OWA and Aspect-Based Model for Customizing Recommendations on Ecommerce. *Applied Soft Computing*, **97**, Article ID: 106768. <https://doi.org/10.1016/j.asoc.2020.106768>
- [37] Liu, P., Zhang, L. and Gulla, J.A. (2021) Multilingual Review-Aware Deep Recommender System via Aspect-Based Sentiment Analysis. *ACM Transactions on Information Systems*, **39**, 1-33. <https://doi.org/10.1145/3432049>
- [38] Ahn, H.J. (2006) Utilizing Popularity Characteristics for Product Recommendation. *International Journal of Electronic Commerce*, **11**, 59-80.
<https://doi.org/10.2753/JEC1086-4415110203>
- [39] Wikipedia (2019) Cluster Analysis.
https://en.wikipedia.org/wiki/Cluster_analysis
- [40] Jing, N., Jiang, T., Du, J. and Sugumaran, V. (2018) Personalized Recommendation Based on Customer Preference Mining and Sentiment Assessment from a Chinese E-Commerce Website. *Electronic Commerce Research*, **18**, 159-179.
<https://doi.org/10.1007/s10660-017-9275-6>
- [41] Li, H., Cui, J., Shen, B. and Ma, J. (2016) An Intelligent Movie Recommendation System Through Group-Level Sentiment Analysis in Microblogs. *Neurocomputing*, **210**, 164-173. <https://doi.org/10.1016/j.neucom.2015.09.134>
- [42] Sun, L., Guo, J. and Zhu, Y. (2020) A Multi-Aspect User-Interest Model Based on Sentiment Analysis and Uncertainty Theory for Recommender Systems. *Electronic Commerce Research*, **20**, 857-882. <https://doi.org/10.1007/s10660-018-9319-6>
- [43] Abbasi-Moud, Z., Vahdat-Nejad, H. and Sadri, J. (2021) Tourism Recommendation System Based on Semantic Clustering and Sentiment Analysis. *Expert Systems with Applications*, **167**, Article ID: 114324. <https://doi.org/10.1016/j.eswa.2020.114324>
- [44] Wang, H., Lu, Y. and Zhai, C.X. (2011) Latent Aspect Rating Analysis without Aspect Keyword Supervision. *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Diego, 21-24 August 2011, 618-626. <https://doi.org/10.1145/2020408.2020505>
- [45] Wang, H., Lu, Y. and Zhai, C.X. (2010) Latent Aspect Rating Analysis on Review Text Data: A Rating Regression Approach. *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Washington DC, 25-28 July 2010, 783-792. <https://doi.org/10.1145/1835804.1835903>
- [46] Vangipuram, S.K. and Appusamy, R. (2021) A Survey on Similarity Measures and Machine Learning Algorithms for Classification and Prediction. *International Conference on Data Science, E-Learning and Information Systems 2021*, Ma'an, 5-7 April 2021, 198-204. <https://doi.org/10.1145/3460620.3460755>
- [47] Alamuri, M., Surampudi, B.R. and Negi, A. (2014) A Survey of Distance/Similarity Measures for Categorical Data. 2014 *International Joint Conference on Neural Networks (IJCNN)*, Beijing, 6-11 July 2014, 1907-1914.
<https://doi.org/10.1109/IJCNN.2014.6889941>
- [48] Boriah, S., Chandola, V. and Kumar, V. (2008) Similarity Measures for Categorical Data: A Comparative Evaluation. In *Proceedings of the 2008 SIAM International Conference on Data Mining (SDM)*, Society for Industrial and Applied Mathemat-

- ics, Philadelphia, 243-254. <https://doi.org/10.1137/1.9781611972788.22>
- [49] Teknomo, K. (2018) Similarity Measurement. <https://people.revoledu.com/kardi/tutorial/Similarity/>
- [50] Trip Advisor (2020) Data Set. https://www.tripadvisor.com/ShowTopic-g1-i12105-k10292711-Datasets_from_trip_advisor-Tripadvisor_Support.html
- [51] Poudel, S. (2022) A Study of Disease Diagnosis Using Machine Learning. *Medical Sciences Forum*, **10**, Article No. 8. <https://doi.org/10.3390/IECH2022-12311>