



The cognitive comparison enhanced hierarchical clustering

Chun Guan¹ · Kevin Kam Fung Yuen^{2,3}

Received: 15 June 2021 / Accepted: 2 September 2021 / Published online: 28 October 2021
© The Author(s) 2021

Abstract

The growth of online shopping is rapidly changing the buying behaviour of consumers. Today, there are challenges facing buyers in the selection of a preferred item from the numerous choices available in the market. To improve the consumer online shopping experience, recommender systems have been developed to reduce the information overload. In this paper, a cognitive comparison-enhanced hierarchical clustering (CCEHC) system is proposed to provide personalised product recommendations based on user preferences. A novel rating method, cognitive comparison rating (CCR), is applied to weigh the product attributes and measure the categorical scales of attributes according to expert knowledge and user preferences. Hierarchical clustering is used to cluster the products into different preference categories. The CCEHC model can be used to rank and cluster product data with the input of user preferences and produce reliable customised recommendations for the users. To demonstrate the advantages of the proposed model, the CCR method is compared with the rating approach of the analytic hierarchy process. Two recommendation cases are demonstrated in this paper with two datasets, one collected by this research for laptop recommendation and the other an open dataset for workstation recommendation. The simulation results demonstrate that the proposed system is feasible for providing personalised recommendations. The significance of this research is the provision of a recommendation solution that does not depend on historical purchase records; rather, one wherein the users' rating preferences and expert knowledge, both of which are measured by CCR, is considered. The proposed CCEHC model could be further applied to other types of similar recommendation cases such as music, books, and movies.

Keywords Decision making · Expert system · Pairwise comparisons · Recommender system · Clustering

1 Introduction

Online shopping has already influenced the purchasing behaviour of consumers. Today, buyers face an overload of information to select the most preferred goods. Recommender systems (RSs) are developed to recommend

appropriate products to consumers on the basis of their historical records. An effective RS service can boost sales by building and increasing customer loyalty (Aggarwal 2016). Reviews of RS technologies can be found in (Aggarwal 2016; Haruna, et al. 2017; Adomavicius and Kwon 2015; Kunaver and Požrl 2017; Kotkov et al. 2016; Zhang et al. 2017; Ma et al. 2018). RSs are typically categorised into three types: collaborative filtering, content-based, and hybrid (Aggarwal 2016). Since these types are based on user profiles including their historical ratings and purchase records (Lika et al. 2014), the RSs have insufficient information to learn the interests of new users. Lacking information for newly joined users is known as the cold-start problem, which is a critical challenge of RS (Kunaver and Požrl 2017; Lika et al. 2014; Volkovs et al. 2017; Viktoratos et al. 2018). A discussion and review of the cold-start problem can be found in Lika et al., (2014).

Cold start problems have significant influence on high-end consumer electronics such as smartphones, laptops,

✉ Kevin Kam Fung Yuen
kevinkf.yuen@gmail.com

Chun Guan
chun_guan@163.com

¹ The Engineering Research Center for Intelligent Robotics (ERCIR), Academy of Engineering and Technology, Fudan University, Shanghai, China

² Department of Computing, The Hong Kong Polytechnic University, Hong Kong SAR, China

³ The Research Institute of Big Data Analytics, Xi'an Jiaotong-Liverpool University, Suzhou, China

game consoles, and audio–video equipment. Since their electronic components and technologies are frequently updated, recommendations based on historical purchasing records could possibly not be applicable to new products. The motivation of this research is to propose an expert system for product recommendations that is based on the current individual users' preferences and expert knowledge elicited from cognitive comparison rating (CCR) method. The proposed model does not have such a cold-start problem, as historical information is not used for the recommendations.

The evaluation of expert judgments and user preferences for products is complicated as numerous products such as the aforementioned high-end consumer electronics consist of different attributes. Multi-criteria decision making (MCDM) methods, which can measure both user preferences and expert judgments for multiple product attributes, have been used in RSs (van Capelleveen et al. 2019; Song 2018; Zhang et al. 2018). The analytic hierarchy process (AHP), a classical MCDM, has been adopted to evaluate user preferences for different product attributes (Hinduja and Pandey 2018; Karthikeyan et al. 2017; Pamučar et al. 2018; Wang and Tseng 2013). CCR, an improved alternative to AHP, is introduced in this study for evaluating expert judgments and user preferences. As an approach to rectify the mathematical representation problem of the perception of the paired differences in AHP, CCR is an ideal method for weighing product attributes and defining numerical values of nominal scales based on user preferences (Yuen 2009, 2012, 2014a; b).

To provide product recommendation services, the hierarchical clustering (HC) method is used to group the products based on the evaluation results of CCR. Different clustering analysis methods have been applied to identify groups of products that have similar attributes with respect to consumer preferences (Nilashi 2017; Frémal and Lecron 2017; Katarya and Verma 2017; Selvi and Sivasankar 2019). HC (Murtagh 1983; Ward Jr 1963; Han et al. 2011) is a popular clustering method; for example, HC has been adopted in other RSs (Selvi and Sivasankar 2019; Gupta and Patil 2015; Zheng et al. 2013; de Aguiar Neto et al. 2020). A hierarchical decomposition of a dataset can be built by HC in the form of a tree graph (called a dendrogram). The major advantage of HC is that the dendrogram can be easily interpreted since the distances between the objects are directly presented. HC has limitations when applied to product-recommendation cases. Firstly, the attributes of products are equally considered; however, different consumers can have different preferences for each attribute. Secondly, the product attributes of nominal scales cannot be directly used in clustering processes. To address these limitations, CCR is used to weigh product attributes and define numerical values of nominal scales with respect

to user preferences. A novel system, cognitive comparison-enhanced hierarchical clustering (CCEHC), is proposed to provide product recommendations with respect to the current individual user's rating preferences. The new method provides a solution to the cold start problem in RSs by using the expert knowledge elicited from CCR instead of the users' historical data. In addition, non-specialized consumers can express their references to interact with the system.

This paper offers a significant extension of the previous initial work (Guan and Yuen 2015; Guan 2018), especially for the sections of methods, experiments, comparisons, and discussions. The remainder of this paper is organised as follows. Section 2 proposes the novel CCEHC system. Section 3 demonstrates the validity and feasibility of the proposed method using a laptop recommendation case, for which the dataset was collected in this study. Section 4 discusses the advantages and limitations of the proposed approach. Section 5 presents the application of CCEHC for workstation recommendations using an open dataset. Finally, Sect. 6 concludes the study.

2 Cognitive comparison enhanced hierarchical clustering

The procedures of the CCEHC model are presented in Fig. 1. In Steps 1 and 2, the attributes of the products are structured as an attribute tree. According to the attribute tree, a raw data table is collected from different sources. In

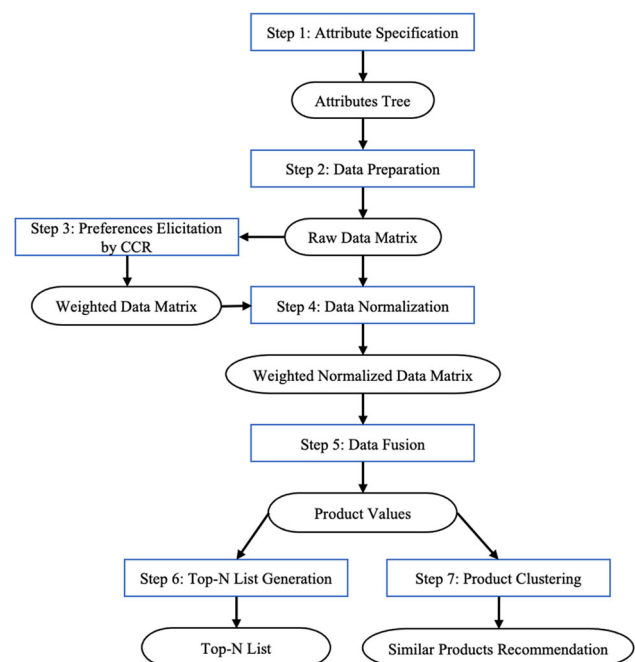


Fig. 1 CCEHC procedures

Step 3, CCR is applied to measure the nominal attribute values and attribute weights with user preferences. The resulting table is normalised in Step 4. In Step 5, the values of the products are produced by aggregating the normalised table and attribute weights. In Step 6, a personalised top- N recommendation is produced by ranking the product values. In the final step, the products are clustered by HC, and similar products can be recommended to the different users.

2.1 Specifying attributes

Detailed product information can be obtained from different sources including manufacturer websites, product engineers, and retailers. A product is represented as a group of attributes, $\{\delta_i\} = (\delta_1, \delta_2, \dots, \delta_i, \dots, \delta_n)$, where δ_i is the i th attribute of the product. Attributes can have sub-attributes. For example, an attribute δ_i is represented by n_i sub-attributes, $\{\delta_{i,j}\} = (\delta_{i,1}, \delta_{i,2}, \dots, \delta_{i,j}, \dots, \delta_{i,n_i})$, where $\delta_{i,j}$ is represented by the j th sub-attribute of δ_i ; the attribute $\delta_{i,j}$ is represented by $n_{i,j}$ sub-attributes, $\{\delta_{i,j,k}\} = (\delta_{i,j,1}, \delta_{i,j,2}, \dots, \delta_{i,j,k}, \dots, \delta_{i,j,n_{i,j}})$, where $\delta_{i,j,k}$ is the k th sub-attribute of $\delta_{i,j}$. The attributes of the different levels are structured as an attributes tree. A sample of the laptop attribute tree is presented in Fig. 2 in Sect. 3.

2.2 Preprocessing data

The leaf attributes, denoted as L , are attributes without sub-attributes. The measurable values of leaf attributes are collected from different sources, as mentioned in Sect. 2.1. Product dataset D consisting of m products and l leaf

attributes is denoted as $D = \{d_{\alpha\beta} | \forall \alpha \in (1, \dots, m), \forall \beta \in (1, \dots, l)\}$. An example of a laptop data matrix is presented in Sect. 3.2. D cannot be directly clustered since it could contain nominal scales that do not have a natural ordering. In the proposed CCEHC system, the nominal scales are substituted by the numerical values measured using the CCR approach presented in the next step.

2.3 Evaluating user preferences by CCR

The user preferences for different attributes and nominal scales are measured using the CCR method. A sample of the CCR interface is displayed in Fig. 3.

Table 1 is a typical measurement scale schema (\aleph, \bar{X}) applied to CCR (Yuen 2009, 2014a). The space of the linguistic labels \aleph of the paired interval scales is {Equally, Slightly, ..., Outstandingly, Absolutely}. The numerical representation of the paired interval scales \bar{X} is as follows:

$$\bar{X} = \left\{ \bar{x}_q = \frac{q\kappa}{\tau} | \forall q \in \{-\tau, \dots, -1, 0, 1, \dots, \tau\}, \quad \kappa > 0 \right\}. \quad (1)$$

The subjective perception of the difference between pairs is represented as the normal utility, κ . By default, κ is set to $\max(\bar{X})$. Denoting the number of linguistic labels as τ , the number of scales is $2\tau + 1$.

To measure the user preferences in paired interval scales, a pairwise opposite matrix (POM) is defined as follows.

Fig. 2 Attributes tree for laptops with weights for User A

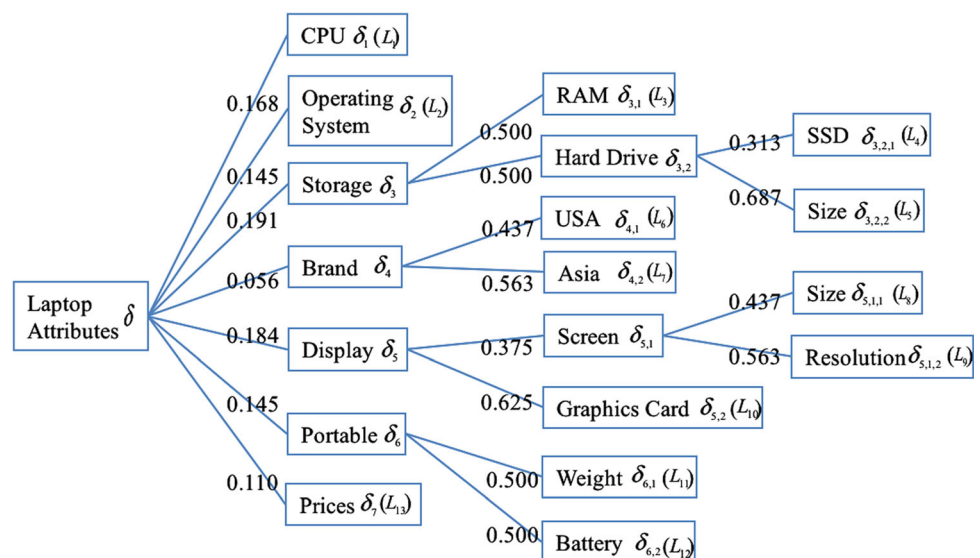


Fig. 3 Cognitive comparison Interface for evaluating laptop attributes

Regarding the attributes of Laptop :
Processor (δ_1), OS (δ_2), Storage (δ_3), Brand (δ_4), Display (δ_5), Prices (δ_6), Portable (δ_7).
Compare the relative preference for each pair, and circle the mark accordingly.

	Absolutely	Highly	Equally	Highly	Absolutely	
Processor	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	1	2 3 4 5 6 7 8		OS
Processor	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	1	0 1 2 3 4 5 6 7 8		Storage
Processor	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	1	0 1 2 3 4 5 6 7 8	7	Brand
Processor	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	1	0 1 2 3 4 5 6 7 8		Display
Processor	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	1	0 1 2 3 4 5 6 7 8		Prices
Processor	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	1	0 1 2 3 4 5 6 7 8	3	Portable
OS	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	3	2 1 0 1 2 3 4 5 6 7 8		Storage
OS	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	2	1 0 1 2 3 4 5 6 7 8	5	Brand
OS	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	2	1 0 1 2 3 4 5 6 7 8		Display
OS	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	0	1 2 3 4 5 6 7 8		Prices
OS	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	0	1 2 3 4 5 6 7 8		Portable
Storage	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	0	1 2 3 4 5 6 7 8	7	Brand
Storage	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	0	1 2 3 4 5 6 7 8		Display
Storage	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	0	1 2 3 4 5 6 7 8	3	Prices
Storage	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	0	1 2 3 4 5 6 7 8	5	Portable
Brand	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	7	6 5 4 3 2 1 0 1 2 3 4 5 6 7 8		Display
Brand	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	5	6 5 4 3 2 1 0 1 2 3 4 5 6 7 8		Prices
Brand	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	3	6 5 4 3 2 1 0 1 2 3 4 5 6 7 8		Portable
Display	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	2	6 5 4 3 4 5 6 7 8		Prices
Display	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	2	6 5 4 3 4 5 6 7 8	4	Portable
Prices	8 7 6 5 4 3 2 1 0	1 2 3 4 5 6 7 8	2	6 5 6 7 8		Portable

Table 1 Measurement scale schema for CCR

Label (κ)	Paired Interval Scale (\bar{X})
Equally	
Slightly	1 $\kappa/8$
Moderately	2 $2\kappa/8$
Fairly	3 $3\kappa/8$
Highly	4 $4\kappa/8$
Strongly	5 $5\kappa/8$
Significantly	6 $6\kappa/8$
Outstandingly	7 $7\kappa/8$
Absolutely	8 κ

$$\begin{aligned}
 B = [b_{ij}] &= \begin{bmatrix} 0 & v_1 - v_2 & \cdots & v_1 - v_n \\ v_2 - v_1 & 0 & \cdots & v_2 - v_n \\ \vdots & \vdots & \ddots & \vdots \\ v_n - v_1 & v_n - v_2 & \cdots & 0 \end{bmatrix} \\
 &\cong \begin{bmatrix} 0 & b_{12} & \cdots & b_{1n} \\ b_{21} & 0 & \cdots & b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & 0 \end{bmatrix} = [b_{ij}] = B, \quad (2)
 \end{aligned}$$

where B denotes a POM. v_i denotes the priority value, and $b_{ij} \triangleq [v_i - v_j]$ denotes the approximate comparison value between objects i and j . The values of b_{ij} are obtained from a questionnaire. For example, $b_{13} = 3$ means that the customer considers the first object to be fairly more important than the third.

To verify the validity of the POM, an Accordance Index (AI) is defined in Eq. (3). $AI = 0$ indicates that B is absolutely accordant. If $0 < AI \leq 0.1$, then B is recommended. If $AI > 0.1$, B is unacceptable, the survey should be rechecked.

$$AI = \frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n \sqrt{\frac{1}{n} \sum_{p=1}^n \left(\frac{b_{ip} + b_{pj} - b_{ij}}{\kappa} \right)^2}. \quad (3)$$

The priority values of objects are computed using the row average plus normal utility (RAU) as follows:

$$RAU(B, \kappa) = \left\{ v_i : v_i = \frac{1}{n} \sum_{j=1}^n b_{ij} + \kappa, \forall i \in \{1, \dots, n\} \right\}. \quad (4)$$

The RAU values are subsequently normalised as a vector W as follows:

$$W = \left\{ w_i : w_i = \frac{v_i}{n\kappa}, \forall i \in \{1, \dots, n\} \right\}, \text{ where } \sum_{i \in \{1, \dots, n\}} v_i = n\kappa. \quad (5)$$

The vector W can represent a variety of items such as the priorities of options, item utilities, weights of features, and preferences for nominal values. In CCEHC, the weights of the product attributes and nominal scales in raw dataset D are substituted with their normalised RAU values.

2.4 Normalising dataset

Two equations are introduced to normalise the raw dataset D . If a higher value indicates a higher preference for a leaf attribute, the dividing maximal function Δ_{\max} defined in Eq. (6) is used to rescale the column of raw attribute values, i.e., $D_{\beta}^T = \{d_{1,\beta}, \dots, d_{\alpha,\beta}, \dots, d_{m,\beta}\}$. If a lower value reveals a higher preference, the minimal dividing function Δ_{\min} defined in Eq. (7) is applied. The normalised data matrix is denoted as $D' = \{x_{\alpha\beta} | \forall \alpha \in (1, \dots, m), \forall \beta \in (1, \dots, l)\}$.

$$x_{\alpha\beta} = \Delta_{\max}(d_{\alpha\beta}) = \frac{d_{\alpha\beta}}{\max(D_{\beta}^T)}, \forall \alpha \in (1, \dots, m), \forall \beta \in (1, \dots, l), \quad (6)$$

$$x_{\alpha\beta} = \Delta_{\min}(d_{\alpha\beta}) = \frac{\min(D_{\beta}^T)}{d_{\alpha\beta}}, \forall \alpha \in (1, \dots, m), \forall \beta \in (1, \dots, l). \quad (7)$$

2.5 Fusing data

The product values $\{\rho^{(\alpha)} | \forall \alpha \in (1, \dots, m)\}$ are the weighted summation of the product attribute values, where α is the index of the product. Attribute values are the weighted summation of their sub-attributes. The detailed calculations of product/attribute are presented in Eqs. (8)–(10), where r_i , $r_{i,j}$, and $r_{i,j,k}$ are the weights of δ_i , $\delta_{i,j}$, and $\delta_{i,j,k}$, respectively. The leaf attribute values are obtained from the normalised data matrix D' .

$$\delta_{ij}^{(\alpha)} = \sum_{k=1}^{n_{ij}} r_{i,j,k} \cdot \delta_{i,j,k}^{(\alpha)}, \forall i \in (1, \dots, n), \forall j \in (1, \dots, n_i), \forall \alpha \in (1, \dots, m), \quad (8)$$

$$\delta_i^{(\alpha)} = \sum_{j=1}^{n_i} r_{i,j} \cdot \delta_{i,j}^{(\alpha)}, \forall i \in (1, \dots, n), \forall \alpha \in (1, \dots, m), \quad (9)$$

$$\rho^{(\alpha)} = \sum_{i=1}^n r_i \cdot \delta_i^{(\alpha)}, \forall \alpha \in (1, \dots, m). \quad (10)$$

2.6 Generating top-N list

According to the product values, a personalised top- N list consisting of the N highest value products in descending order is provided to the user; the calculation details are described in Algorithm 1. For different users, the top- N lists are different since the product values are calculated with respect to personal preferences.

Algorithm 1: Top-N list

Input: $\{\rho^{(\alpha)}\}, N$
For $i=1$ *to* N ,
 $M = \text{Max}(\{\rho^{(\alpha)}\})$
 $\text{TopN}[i] = M$
 $\{\rho^{(\alpha)}\} = M / \{\rho^{(\alpha)}\}$, where $/$ is a complement operator.
Output: TopN

2.7 Clustering products

HC is used to group products according to their product values. The aim of HC is to iteratively combine the two nearest clusters into a larger cluster until all the objects are in one cluster or a preset termination condition is reached (Han et al. 2011). Murtagh (1983) briefly described the steps of hierarchical clustering methods. The steps of HC applied to the CCEHC are described below.

Step 1: One product is an atomic cluster, $C_{\sigma} = \{\rho^{(\alpha)}\}$. The distances between each pair of clusters are computed in the following form:

$$d_{\alpha,\alpha'} = |\rho^{(\alpha)} - \rho^{(\alpha')}|, \forall \alpha, \alpha' \in (1, \dots, m), \quad (11)$$

where $d_{\alpha,\alpha'}$ is the dissimilarity of product values for any two different products $\rho^{(\alpha)}$ and $\rho^{(\alpha')}$.

Step 2: The two closest clusters, C_s and C_t , where $(s, t) = \text{argmin}(\{d_{\alpha,\alpha'}\})$, are combined into a larger cluster, i.e., $C_s = C_s \cup C_t$, which means C_s is updated by merging C_t and C_s . The distances between the updated cluster C_s and other clusters C_{-s} are computed as the average distance (Han et al. 2011) in the following form:

$$d_{\text{avg}}(C_s, C_{-s}) = \frac{1}{\eta_s \eta_{-s}} \sum_{\rho^{(\alpha)} \in C_s, \rho^{(\alpha')} \in C_{-s}} d_{\rho^{(\alpha)}, \rho^{(\alpha')}}, \quad (12)$$

where η_s is the number of objects in cluster C_s . $d_{\text{avg}}(C_s, C_{-s})$ updates the distances between clusters C_s and C_{-s} . $d_{\rho^{(\alpha)}, \rho^{(\alpha')}}$ is the distance between products $\rho^{(\alpha)}$ and

$\rho^{(a')}$, where $\rho^{(a)} \in C_s$ and $\rho^{(a')} \in C_{-s}$. Step 2 is repeated until all products are in one cluster.

Step 3: A dendrogram indicating the arrangement of the merged clusters is produced. Two examples of dendrograms for similar laptop clusters are displayed in Fig. 4. The products are grouped into different clusters by cutting the branches at an appropriate height, which represents the distance between the clusters. Clustering results can be used for similar product recommendations. When a user searches for product $\rho^{(a)}$ such that $\rho^{(a)} \in C_\sigma$, the other products in Cluster C_σ , i.e., R , are recommended to the user. R is defined as follows:

$$R = C_\sigma / \left\{ \rho^{(a)} \right\}, \quad (13)$$

where $/$ is a complement operator.

3 Application of laptop recommendation

Laptops can be represented by a set of attributes. Consumers search for a set of preferred product attributes when searching for a laptop. To demonstrate the applicability and validity of the proposed CCEHC system, a laptop recommendation case for a consumer (denoted as User A) is illustrated. For the cases in Sects. 3–5, a dataset of 27 laptop configurations was manually collected from the websites of online retail shops and manufacturers in 2015.

3.1 Specifying attributes

A large number of laptop configurations can be found on websites for selling, introducing, and comparing electronic products. The majority of consumers are likely unfamiliar with specific technical properties such as the wireless type and video output details. Certain laptop components could be unimportant to other consumers such as USB ports,

DVD/CD burners, and speakers. These attributes are not considered in this recommendation case. The selected attributes for choosing an ideal laptop are structured as a 3-level attribute tree, as indicated in Fig. 2.

The attributes in the first level of the tree are CPU (δ_1), Operating System (δ_2), Storage (δ_3), Brand (δ_4), Display (δ_5), Portable (δ_6), and Price (δ_7). Five of these have sub-attributes. For example, Storage includes the Hard Drive and Random-Access Memory (RAM). The sub-attributes of the first-level attributes are structured in the second level, including {RAM ($\delta_{3,1}$), Hard Drive ($\delta_{3,2}$)}, {USA ($\delta_{4,1}$), Asia ($\delta_{4,2}$)}, {Screen ($\delta_{5,1}$), Graphics Card ($\delta_{5,2}$)}, {Weight ($\delta_{7,1}$), Battery ($\delta_{7,2}$)}. The sub-attributes of the second-level attributes are in the third level of the tree, which are {SSD ($\delta_{3,2,1}$), Size ($\delta_{3,2,2}$)}, {Size ($\delta_{5,1,1}$), Resolution ($\delta_{5,1,2}$)}.

3.2 Preprocessing data

From the attributes tree presented in Fig. 2, a laptop has 13 leaf attributes. A raw data matrix D of is obtained from the laptop configurations as indicated in Table 15 of the Appendix. The quantification approaches used to preprocess leaf attributes are summarised in Table 2. For example, the attribute values of the CPU and Graphics Card are quantified by their performance scores (3DMARK 2015). The SSD attribute has three nominal labels: *SSD*, which indicates that the laptop has an SSD, *No SSD* indicating that the laptop has no SSD and *Hybrid* indicating that the laptop has SSD and another type of hard disk. The three labels are respectively replaced by “2”, “0”, and “1”. The screen resolution attribute is represented by the production of the width and height pixels of the screen. The nominal scales of the attributes OS and Brand are measured by CCR in Sect. 3.3.

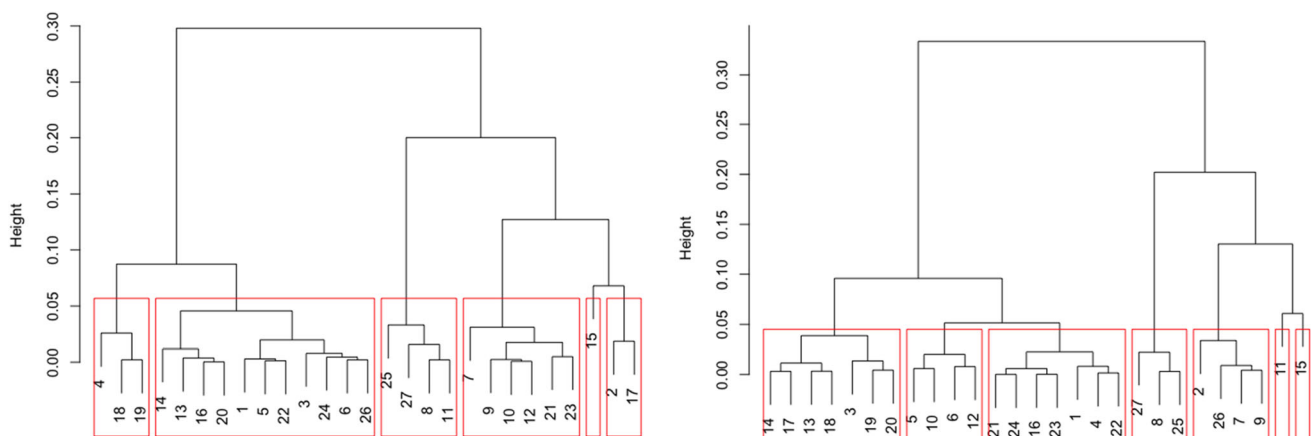


Fig. 4 Dendrogram for clusters of similar laptops produced for User A (left) and User B (right)

Table 2 Schema of laptop leaf attributes

	Leaf attribute name	Measurement scale	Quantification approach	Normalization function
L_1	CPU δ_1	Nominal: CPU model	3DMark06 Score	Δ_{\max}
L_2	OS δ_2	Nominal: Linux OS X Windows 7 Windows 8	CCR	Δ_{\max}
L_3	RAM $\delta_{3,1}$	GB	GB	Δ_{\max}
L_4	SSD $\delta_{3,2,1}$	Nominal: SSD Hybrid No SSD	SSD: 2 Hybrid: 1 No SSD: 0	Δ_{\max}
L_5	Hard Drive Size $\delta_{3,2,2}$	GB	GB	Δ_{\max}
L_6	Brand (USA) $\delta_{4,1}$	Nominal: Alienware Apple Dell Microsoft	CCR	Δ_{\max}
L_7	Brand (Asia) $\delta_{4,2}$	Nominal: Acer ASUS HP Lenovo Samsung	CCR	Δ_{\max}
L_8	Screen Size $\delta_{5,1,1}$	Inch	Inch	Δ_{\max}
L_9	Screen Resolution $\delta_{5,1,2}$	DPI	Width pixel \times height pixel	Δ_{\max}
L_{10}	Graphics Card $\delta_{5,2}$	Nominal: Graphics Card model	3DMark06 Scores	Δ_{\max}
L_{11}	Weight $\delta_{6,1}$	Kg	Kg	Δ_{\min}
L_{12}	Battery $\delta_{6,2}$	Hour	Hour	Δ_{\max}
L_{13}	Price δ_7	RMB	Thousand RMB	Δ_{\min}

3.3 Evaluating user preferences by CCR

The preferences of user were gathered from a CCR questionnaire. An example of a questionnaire using CCR is presented in Fig. 3. The measurement scale schema defined in Table 1 is used in this case, and κ is set to “8”. The POM for User A presented in Table 3 is obtained from the questionnaire results in Fig. 3 based on Eq. (2). The AI for the POM computed by Eq. (3) is less than 0.1, which

means that the POM is acceptable. Table 3 lists the weights of the 1st level laptop attributes computed by Eqs. (4) and (5) within the detailed calculations steps. The POMs, AIs, and weights of the remaining sub-attributes are provided in Table 4. All the attribute weights for User A are given, including the attribute tree, in Fig. 2. The nominal attribute labels for User A of Operating System (L_2), Asia Brand (L_6) and USA Brand (L_7) are also measured by CCR. The POMs and prioritisation results (called as preference

Table 3 Comparison matrices for 1st level laptop attributes (User A)

B_0	δ_1	δ_2	δ_3	δ_4	δ_5	δ_6	δ_7	$\sum_{j=1}^7 b_{ij}$	$\frac{1}{7} \sum_{j=1}^7 b_{ij}$	$v_i = \frac{1}{7} \sum_{j=1}^7 b_{ij} + 8$	$r_i = w_i = \frac{v_i}{7 \times 8}$
δ_1	0	1	-1	7	-1	1	3	10	1.429	9.429	0.168
δ_2	-1	0	-3	5	-2	0	2	1	0.143	8.143	0.145
δ_3	1	3	0	7	0	3	5	19	2.714	10.714	0.191
δ_4	-7	-5	-7	0	-7	-5	-3	-34	-4.857	3.143	0.056
δ_5	1	2	0	7	0	2	4	16	2.286	10.286	0.184
δ_6	-1	0	-3	5	-2	0	2	1	0.143	8.143	0.145
δ_7	-3	-2	-5	3	-4	-2	0	-13	-1.857	6.143	0.110
AI = 0.051											

Table 4 Comparison matrices for 2nd and 3rd levels laptop attributes (User A)

B_3	$\delta_{3,1}$	$\delta_{3,2}$	$r_{3,i}$	$B_{3,2}$	$\delta_{3,2,1}$	$\delta_{3,2,2}$	$r_{3,2,i}$	B_4	$\delta_{4,1}$	$\delta_{4,2}$	$r_{4,i}$
$\delta_{3,1}$	0	0	0.5	$\delta_{3,2,1}$	0	− 6	0.313	$\delta_{4,1}$	0	− 2	0.437
$\delta_{3,2}$	0	0	0.5	$\delta_{3,2,2}$	6	0	0.687	$\delta_{4,2}$	2	0	0.563
AI = 0				AI = 0				AI = 0			
B_5	$\delta_{5,1}$	$\delta_{5,2}$	$r_{5,i}$	$B_{5,1}$	$\delta_{5,1,1}$	$\delta_{5,1,2}$	$r_{5,1,i}$	B_6	$\delta_{6,1}$	$\delta_{6,2}$	$r_{6,i}$
$\delta_{5,1}$	0	− 4	0.375	$\delta_{5,1,1}$	0	− 2	0.437	$\delta_{6,1}$	0	0	0.5
$\delta_{5,2}$	4	0	0.625	$\delta_{5,1,2}$	2	0	0.563	$\delta_{6,2}$	0	0	0.5
AI = 0				AI = 0				AI = 0			

values) are displayed in Table 5. The nominal attribute values in raw dataset D can be substituted with their preference values.

3.4 Normalising dataset

The suitable normalisation equations for the leaf attributes are listed in Table 2. For example, a CPU (L_1) with a higher performance score is attractive. Δ_{\max} defined in Eq. (6) is therefore applied to normalise the CPU attribute values. Typically, consumers prefer a lower product price; therefore, Δ_{\min} defined in Eq. (7) is used to normalise Price (L_{13}). The normalised data matrix D' is provided in Table 16 in the Appendix. Two samples of the normalisation process for the attribute values of CPU and Price for laptop ID1 are given below.

$$x_{1,1} = \Delta_{\max}(d_{1,1}) = \frac{d_{1,1}}{\max(D_1^T)} = \frac{3367}{7060} = 0.447, \quad (14)$$

$$x_{1,13} = \Delta_{\min}(d_{1,13}) = \frac{\min(D_{13}^T)}{d_{1,13}} = \frac{2}{7} = 0.285. \quad (15)$$

3.5 Fusing data

For each laptop, the 2nd level attribute values are calculated using Eq. (8), the weights in Tables 3 and 4, and the normalised data matrix D' in Table 16 in the Appendix. An example of the calculation process for $\delta_{3,2}^{(1)}$ is presented below.

$$\begin{aligned} \delta_{3,2}^{(1)} &= \sum_{k=1}^2 r_{3,2,k} \bullet \delta_{3,2,k}^{(1)} = (r_{3,2,1} \bullet \delta_{3,2,1}^{(1)}) + (r_{3,2,2} \bullet \delta_{3,2,2}^{(1)}) \\ &= (0.313 \bullet x_{1,4}) + (0.687 \bullet x_{1,5}) \\ &= (0.313 \bullet 1.000) + (0.687 \bullet 0.169) = 0.429. \end{aligned} \quad (16)$$

The value of attribute $\delta_{5,2}^{(1)}$ is computed as 0.556. The 1st level attribute values are computed using Eq. (9). The

calculation process for $\delta_3^{(1)}$ is given in Eq. (17) as an example.

$$\begin{aligned} \delta_3^{(1)} &= \sum_{j=1}^2 r_{3,j} \bullet \delta_{3,j}^{(1)} = (r_{3,1} \bullet \delta_{3,1}^{(1)}) + (r_{3,2} \bullet \delta_{3,2}^{(1)}) \\ &= (r_{3,1} \bullet x_{1,5}) + (r_{3,2} \bullet \delta_{3,2}^{(1)}) \\ &= (0.500 \bullet 0.250) + (0.500 \bullet 0.429) = 0.340. \end{aligned} \quad (17)$$

The values of attributes $\delta_4^{(1)}$, $\delta_5^{(1)}$ and $\delta_6^{(1)}$ are 0.563, 0.327 and 0.550, respectively. The laptop product values are computed using Eq. (10). For example, the product value of the first laptop is 0.448; the detailed steps are presented in Eq. (18). All 27 laptop product values for User A are listed in Table 6.

$$\begin{aligned} \rho^{(1)} &= \sum_{i=1}^7 r_i \bullet \delta_i^{(1)} \\ &= (r_1 \bullet \delta_1^{(1)}) + (r_2 \bullet \delta_2^{(1)}) + (r_3 \bullet \delta_3^{(1)}) \\ &\quad + (r_4 \bullet \delta_4^{(1)}) + (r_5 \bullet \delta_5^{(1)}) + (r_6 \bullet \delta_6^{(1)}) \\ &\quad + (r_7 \bullet \delta_7^{(1)}) \\ &= (r_1 \bullet x_{1,1}) + (r_2 \bullet x_{1,2}) + (r_3 \bullet \delta_3^{(1)}) + (r_4 \bullet \delta_4^{(1)}) \\ &\quad + (r_5 \bullet \delta_5^{(1)}) + (r_6 \bullet \delta_6^{(1)}) + (r_7 \bullet x_{1,13}) \\ &= 0.448. \end{aligned} \quad (18)$$

3.6 Generating top- N list

The top- N list for laptops is produced using Algorithm 1. According to User A's preferences for laptop attributes, a top-10 list of laptops is provided in Table 7. The information and correspondingly web links of the laptops in the top-10 list can be recommended to User A in descending order after the user has completed the CCR survey.

Table 5 Comparison matrices for nominal attribute of L_2 , L_6 and L_7 (User A)

B_2	Linux	OS X	Windows 7	Windows 8	Preference value	
Linux	0	− 2	− 3	0	0.211	
OS X	2	0	− 1	2	0.273	
Windows 7	3	1	0	3	0.305	
Windows 8	0	− 2	− 3	0	0.211	
AI = 0						
B_6	Alienware	Apple	Dell	Microsoft	HP	Preference value
Alienware	0	1	3	4	4	0.260
Apple	− 1	0	2	3	3	0.235
Dell	− 3	− 2	0	1	2	0.190
Microsoft	− 4	− 3	− 1	0	0	0.160
HP	− 4	− 3	− 2	0	0	0.155
AI = 0.043						
B_7	Acer	ASUS	Lenovo	Samsung	Preference value	
Acer	0	− 1	− 3	2	0.234	
ASUS	1	0	− 2	3	0.266	
Lenovo	3	2	0	4	0.320	
Samsung	− 2	− 3	− 4	0	0.180	
AI = 0.042						

3.7 Clustering products

The details of the HC method are described in Sect. 2.7. The HC method is used to cluster the similar laptop products into different groups by measuring the dissimilarities between the product values calculated using Eq. (11). After merging the two closest clusters, the dissimilarities are updated using Eq. (12). The dendrogram produced by HC for User A is displayed in Fig. 4a. By cutting the dendrogram at height of 0.05, six clusters are generated: {4, 18, 19}, {14, 13, 16, 20, 1, 5, 22, 3, 24, 6, 26}, {25, 27, 8, 11}, {7, 9, 10, 12, 21, 23}, {15} and {2, 17}. The clustering results are used for product recommendations. For example, if User A browses the webpage of Laptop 4, Laptops 18 and 19, which are in the same cluster of Laptop 4, are recommended to the user. Similarly, if User A browses Laptop 2, Laptop 17 is recommended.

4 Discussions

Comparisons and discussions are presented in this section to demonstrate the advantages of the proposed RS. To demonstrate the advantage of providing personalization recommendations, the recommendations for User B are presented in Sect. 4.1. To demonstrate the differences between CCR and AHP, the results produced by AHP

enhanced method are presented in Sect. 4.2. The limitations of the proposed method are discussed in Sect. 4.3.

4.1 Personalization

User B completes the questionnaire. The rating scores are presented in Tables 8, 9, 10; the rating scores of User A are given in Tables 3, 4, 5. The product values for Users A and B are listed in Table 6. The system produces personalised top-10 laptop lists and laptop clusters with respect to the two users' preferences. Table 7 presents the top-10 laptops recommended for User A; Table 11 lists the top-10 laptops recommended for User B. The two dendrograms in Fig. 4 indicate the laptops clustering results for User A and B.

Comparing the preferences indicated in Tables 3 and 8, both users require a laptop with a high-speed CPU, large storage, and acceptable graphics. Three differences between the preferences of the two users can be summarised by comparing Tables 3, 4, 5 with Tables 8, 9, 10. Firstly, User A is not very price sensitive, whereas for User B, the price is considerably more important. Secondly, User A requires a portable laptop that is light and has a long battery life; User B hardly considers portability. Thirdly, User A has no strong preference for the brand, whereas User B strongly prefers laptops produced by US companies, especially Apple and Alienware.

From the laptop configurations presented in Table 15 and the two top-10 lists generated for Users A and B,

Table 6 Laptop product values for 2 users

ID (α)	Product value ($\rho^{(\alpha)}$) for User A	Product value ($\rho^{(\alpha)}$) for User B
1	0.448	0.392
2	0.553	0.479
3	0.465	0.347
4	0.403	0.398
5	0.447	0.443
6	0.459	0.431
7	0.507	0.503
8	0.660	0.661
9	0.478	0.507
10	0.476	0.437
11	0.662	0.609
12	0.475	0.423
13	0.427	0.382
14	0.419	0.374
15	0.603	0.548
16	0.431	0.408
17	0.535	0.377
18	0.380	0.386
19	0.378	0.356
20	0.431	0.361
21	0.488	0.414
22	0.445	0.400
23	0.493	0.409
24	0.457	0.414
25	0.643	0.658
26	0.462	0.512
27	0.676	0.680

Table 7 The top-10 laptops for User A

Rank	ID (α)	Product value ($\rho^{(\alpha)}$)
1	27	0.676
2	11	0.662
3	8	0.660
4	25	0.643
5	15	0.603
6	2	0.553
7	17	0.535
8	7	0.507
9	23	0.493
10	21	0.488

respectively, in Tables 7 and 11, it can be concluded that the laptops in the two top-10 lists meet the common requirements (high speed CPU, large storage and acceptable graphics) of the two users. There are three issues for the two users' preferences leading to the top-10 lists results. Firstly, the best laptop for Users A and B is the same: Laptop 27. This laptop has almost all the best configurations, yet is the most expensive. The product value of Laptop 27 for User B is less than for User A. The main reason is that User B is more price sensitive. Secondly, two portable laptops, Laptops 21 and 23, are recommended to User A, even though the other configurations of the two laptops are not attractive. The main reason is that User A prefers portable laptops. Thirdly, as User B is faithful to the brands Apple and Alienware, all the laptops of the two brands are recommended to User B in the top-10 list. The laptop recommendations provided for the two users match their requirements, and it can be concluded that the proposed CCEHC method can provide personalized recommendations with respect to user preferences.

4.2 Comparisons between CCR and AHP

CCR is based on the cognitive network process (CNP) (Yuen 2009, 2014a). The CNP is proposed as an ideal alternative to AHP to solve the rating scale problem in AHP. The numerical definition of the AHP's paired ratio scale inappropriately represents the human intuitive judgment of paired difference; CNP uses a paired interval scale instead of a paired ratio scale. Detailed comparisons between CNP and AHP can be found in Yuen (2009, 2014a).

This study uses the original version of AHP proposed in Saaty (1980) for comparisons. To produce the AHP results, the CCP rating scales are transformed to AHP scales. The method employed is called AHP Enhanced Hierarchical Clustering (AHPEHC). The transformation method of the rating scale between AHP and CCR is given in Yuen

Table 8 Comparison matrices for 1st level laptop attributes (User B)

B_0	δ_1	δ_2	δ_3	δ_4	δ_5	δ_6	δ_7	r_i
δ_1	0	4	2	0	2	8	0	0.184
δ_2	−4	0	−2	−3	−2	4	−4	0.115
δ_3	−2	2	0	−2	0	6	−2	0.148
δ_4	0	3	2	0	2	8	0	0.181
δ_5	−2	2	0	−2	0	6	−2	0.148
δ_6	−8	−4	−6	−8	−6	0	−8	0.041
δ_7	0	4	2	0	2	8	0	0.184

AI = 0.024 < 0.1

Table 9 Comparison matrices for 2nd and 3rd levels laptop attributes (User B)

B_3	$\delta_{3,1}$	$\delta_{3,2}$	$r_{3,i}$	$B_{3,2}$	$\delta_{3,2,1}$	$\delta_{3,2,2}$	$r_{3,2,i}$	B_4	$\delta_{4,1}$	$\delta_{4,2}$	$r_{4,i}$
$\delta_{3,1}$	0	2	0.5625	$\delta_{3,2,1}$	0	8	0.75	$\delta_{4,1}$	0	6	0.6875
$\delta_{3,2}$	− 2	0	0.4375	$\delta_{3,2,2}$	− 8	0	0.25	$\delta_{4,2}$	− 6	0	0.3125
AI = 0				AI = 0				AI = 0			
B_5	$\delta_{5,1}$	$\delta_{5,2}$	$r_{5,i}$	$B_{5,1}$	$\delta_{5,1,1}$	$\delta_{5,1,2}$	$r_{5,1,i}$	B_6	$\delta_{6,1}$	$\delta_{6,2}$	$r_{6,i}$
$\delta_{5,1}$	0	− 8	0.25	$\delta_{5,1,1}$	0	− 4	0.375	$\delta_{6,1}$	0	− 2	0.4375
$\delta_{5,2}$	8	0	0.75	$\delta_{5,1,2}$	4	0	0.625	$\delta_{6,2}$	2	0	0.5625
AI = 0				AI = 0				AI = 0			

Table 10 Comparison matrices for nominal attribute of L_2 , L_5 and L_7 (User B)

B_2	Linux	OS X	Windows 7	Windows 8		Preference value
Linux	0	− 8	− 1	− 5		0.141
OS X	8	0	7	3		0.391
Windows 7	1	− 7	0	− 4		0.172
Windows 8	5	− 3	4	0		0.297
AI = 0						

B_6	Alienware	Apple	Dell	Microsoft	HP	Preference value
Alienware	0	0	4	8	6	0.290
Apple	0	0	4	7	6	0.285
Dell	− 4	− 4	0	4	3	0.195
Microsoft	− 8	− 7	− 4	0	− 2	0.095
HP	− 6	− 6	− 3	2	0	0.135
AI = 0.059						

B_7	Acer	ASUS	Lenovo	Samsung		Preference value
Acer	0	− 6	− 1	0		0.195
ASUS	6	0	5	6		0.383
Lenovo	1	− 5	0	1		0.227
Samsung	0	− 6	− 1	0		0.195
AI = 0						

(2009, 2014a). Table 12 presents the transformed rating matrix and weights of the ratings listed in Table 3. The product values and clustering results of the laptops computed using AHP are shown in Table 13 and Fig. 5, respectively.

The 27 laptop product values are displayed in Fig. 6. A significant difference between the values computed by CCR and AHP is that the product values computed by CCR are considerably closer than those computed by AHP. The results of CCR reflect that the recommendations for the products are difficult to make, whereas AHP results reflect that the problem is trivial. The reason for this difference is that the paired ratio scales applied in AHP typically exaggerate the human perception of the paired difference in

ratio. It can be concluded that CCR outperforms AHP in reflecting the preferences of both expert and users.

4.3 Limitations

Regarding the limitations, as the proposed approach is typically designed for recommendations of the latest launch products, the datasets consider the latest products (assuming that a consumer is not likely to buy an obsolete product). As the obsolete products are not considered, the data set should not be excessively large. The proposed method is not designed for processing large-scale data; the processing capability for large datasets is limited; however, this is not typically a problem as it is rare that there are a

Table 11 The top-10 laptops for User B

Rank	ID (α)	Product Value ($\rho^{(\alpha)}$)
1	27	0.680
2	8	0.661
3	25	0.658
4	11	0.609
5	15	0.548
6	26	0.512
7	9	0.507
8	7	0.503
9	2	0.479
10	5	0.443

Table 12 AHP comparison matrices for 1st level laptop attributes (User A)

B_0	δ_1	δ_2	δ_3	δ_4	δ_5	δ_6	δ_7	weight
δ_1	1	2	1/2	8	1/2	2	4	0.168
δ_2	1/2	1	1/4	6	1/3	1	3	0.100
δ_3	2	4	1	8	1	4	6	0.298
δ_4	1/8	1/6	1/8	1	1/8	1/6	1/4	0.022
δ_5	2	3	1	8	1	3	5	0.264
δ_6	1/2	1	1/4	6	1/3	1	3	0.100
δ_7	1/4	1/3	1/6	4	1/5	1/3	1	0.048

large number of new products. The scope of the proposed RS is not to address the problems solved by content based and collaborative filtering RSs; in turn, the content-based and collaborative filtering RSs are not designed to address the research problem solved by the proposed approach. The clustering validity of the proposed method is not discussed as no ground truth class labels can be used to verify the results. Internal clustering criteria, such as Davies and Bouldin (1979) and Silhouette (Rousseeuw 1987) are normally used as references, although these do not necessarily reflect real validity.

5 Workstation recommendation with open dataset

The proposed CCEHC method can be applied to different kinds of RSs. A workstation RS is developed to demonstrate the usability of the CCEHC. As a special type of laptop, workstation is designed for technical, scientific, and other professional purposes. In general, the workstations are more expensive than the laptops. Thus, users typically spend more time selecting a suitable workstation. The RS

Table 13 Laptop product values by AHP (User A)

ID (α)	Product Value ($\rho^{(\alpha)}$)
1	0.328
2	0.511
3	0.373
4	0.268
5	0.378
6	0.391
7	0.379
8	0.597
9	0.340
10	0.412
11	0.734
12	0.336
13	0.342
14	0.265
15	0.605
16	0.376
17	0.517
18	0.240
19	0.226
20	0.282
21	0.366
22	0.280
23	0.384
24	0.296
25	0.682
26	0.398
27	0.758

built by CCEHC with expert opinions and user preferences could be helpful for workstation recommendation.

An open dataset for the characteristics and prices of laptop models (Kaggle 2018) is used for the workstation RS. The dataset contains 29 items related to workstation. The characteristics and prices of workstations obtained from the original datasets can be summarised as 13 attributes organised as a two-level attributes tree, as displayed in Fig. 7. The POMs are presented in Table 17 in the Appendix. The weights of the seven attributes (B_0) in the first level and three second-level attributes, Screen (B_2), Processor (B_3), and Memory (B_4), are evaluated by CCR. In addition, the attributes, Company (L_1), Screen Type (L_3), Hard Disk Type (L_8) and OS (L_{10}), are nominal and evaluated by CCR.

To demonstrate the usability of the workstation RS, two users, User C and D, use the CCEHC system to determine what workstation would fit their purpose. The comparison matrices of User C and D are presented in Table 17 in the Appendix. The weights of the users are indicated in the attribute trees presented in Fig. 7. To compare the results

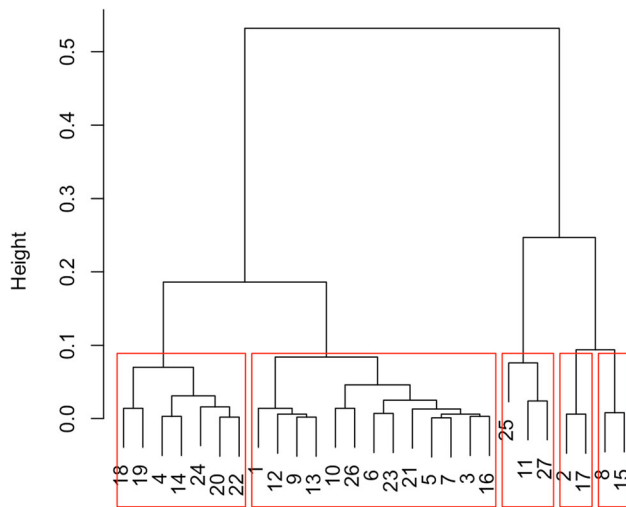


Fig. 5 Dendrogram for clusters of similar laptops produced by AHPEHC

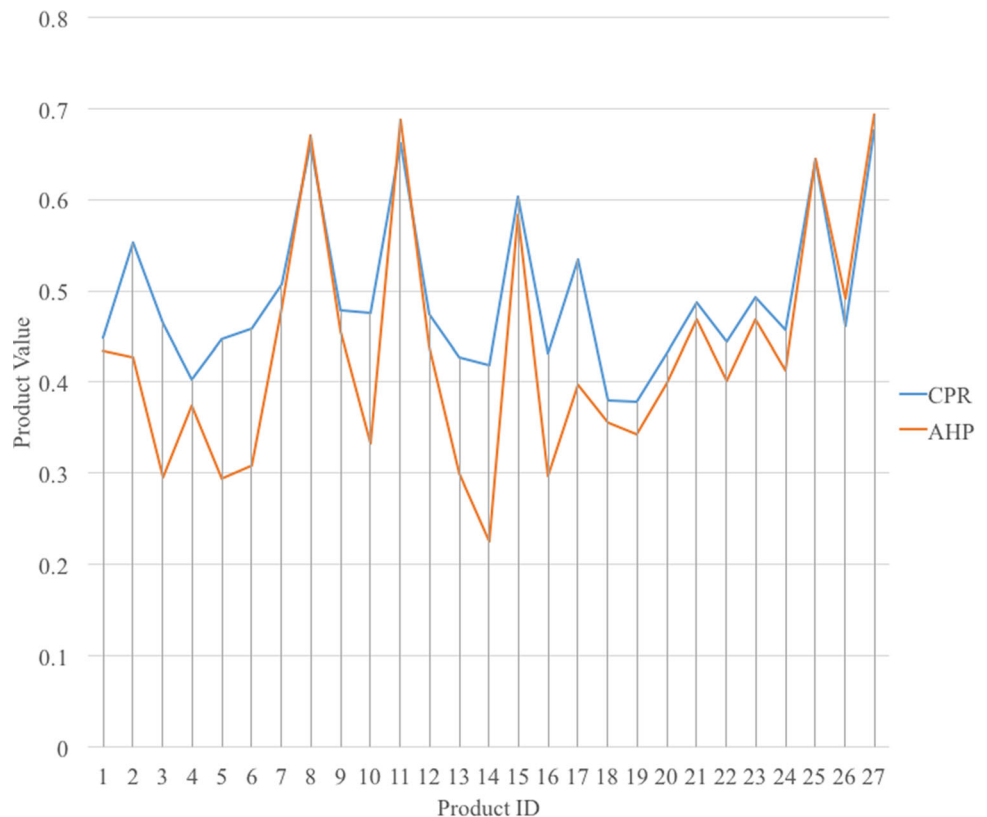
of CCR and AHP, both CCEHC and AHPEHC are used to build the workstation RSs. The recommendations produced by CCEHC and AHPEHC RSs for both users are displayed as dendrograms with clusters in Fig. 8 and the top-10 lists presented in Table 14.

By reading the preferences of the two users in Table 17 and Fig. 7, it can be determined that the preferences of Users C and D are not significantly different. For example,

the two users both feel the memory and OS are important, and the price and weight are less important. Their preferences for Memory in the second-level attributes are also similar. The larger RAM and better type of hard disk (such as solid-state disk) are more important; however, a large hard disk capacity is less essential. The preferences of the company and processor are different. User C has certain preferences for the company of workstation, and feels the CPU and GPU are equally important; User D is not overly concerned with the company and feels the CPU is more important than GPU. By comparing the four top-10 lists produced by the two RSs for the two users, it can be observed that the RS applying CCEHC produces similar recommenders for the two users, whereas the RS applying AHPEHC method does otherwise. The recommendations produced by the two RSs are different for each user.

For the two users with similar preferences for the workstation, the RS applying CCEHC provides similar recommendations, whereas the RS applying AHPEHC provides considerably different results. The results demonstrate that the CCEHC can better reflect the user preferences than AHPEHC. The reason for the different results of CCR and AHP is the different mathematical representation of human opinions. As mentioned in Sect. 4.2, the paired ratio scales applied in AHP typically exaggerate the human perception of the paired difference in times; hence, the marginal difference in user preferences

Fig. 6 Comparison between product values computed by CCR and AHP



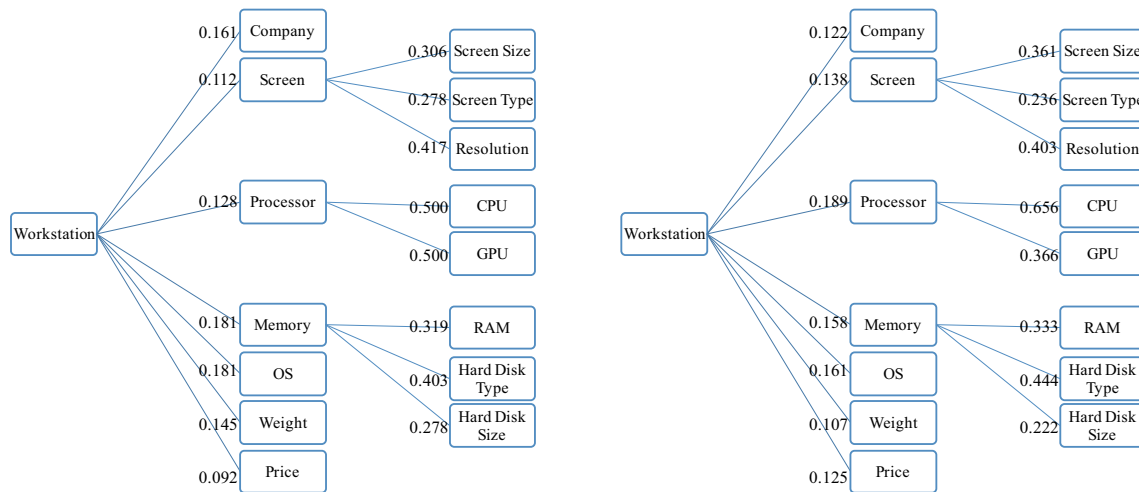


Fig. 7 Workstation attribute trees with the weights for User C (left) and User D (right)

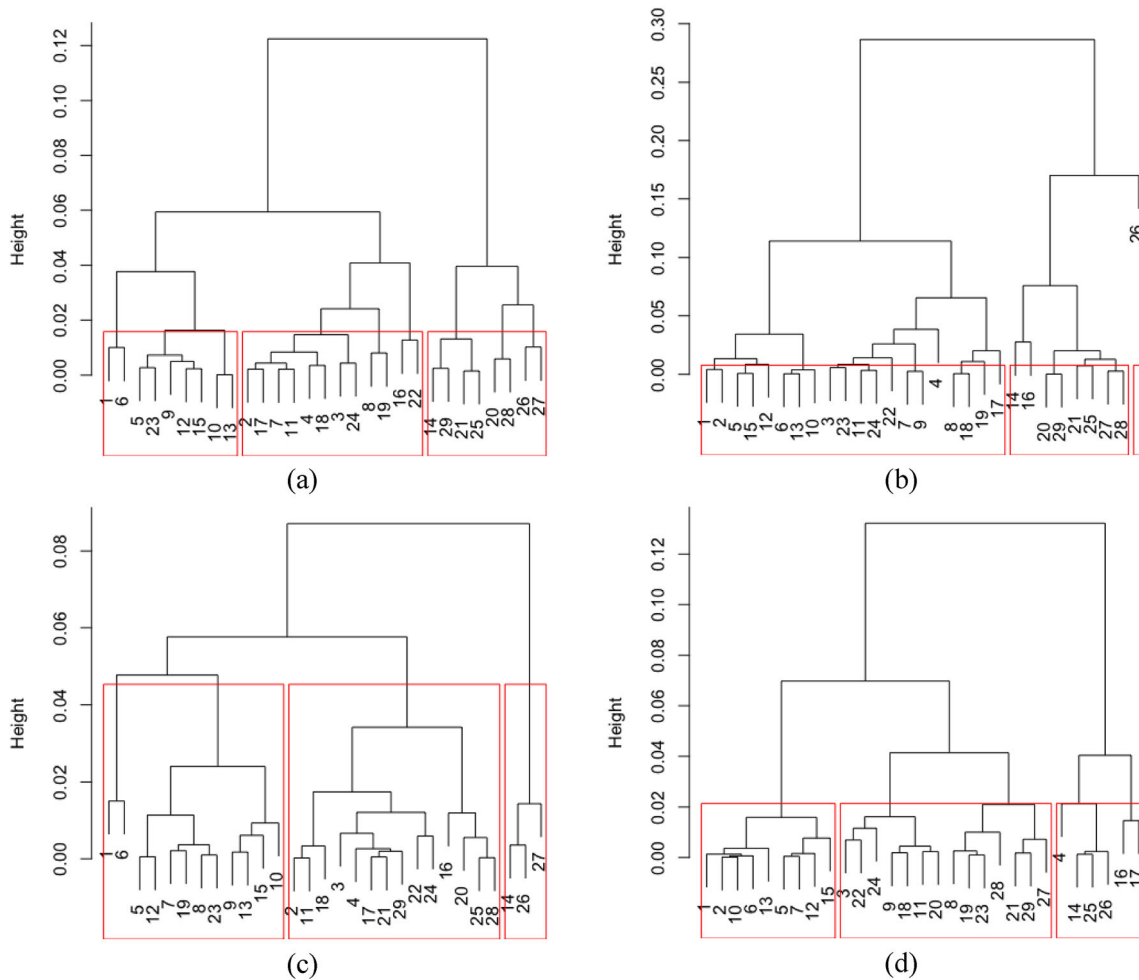


Fig. 8 Dendrogram for workstations for **a** User C applying CCEHC, **b** User C applying AHPEHC, **c** User D applying CCEHC, **d** User D applying AHPEHC

Table 14 The top-10 workstations for each user using CCEHC and AHPEHC

Rank	User C				User D			
	CCEHC		AHPEHC		CCEHC		AHPEHC	
	ID (α)	PV ($\rho^{(\alpha)}$)	ID (α)	PV ($\rho^{(\alpha)}$)	ID (α)	PV ($\rho^{(\alpha)}$)	ID (α)	PV ($\rho^{(\alpha)}$)
1	1	0.860	6	0.896	1	0.852	10	0.874
2	6	0.850	13	0.895	6	0.837	2	0.874
3	13	0.829	10	0.892	10	0.818	6	0.873
4	10	0.829	12	0.871	13	0.812	1	0.872
5	9	0.819	5	0.863	9	0.810	13	0.870
6	12	0.815	15	0.862	15	0.805	15	0.862
7	15	0.813	2	0.854	12	0.795	12	0.856
8	23	0.810	1	0.850	5	0.794	5	0.854
9	5	0.807	19	0.815	19	0.786	7	0.854
10	19	0.797	8	0.804	7	0.784	18	0.826

can lead to considerably different results. The application of workstation RS demonstrates that the CCEHC method can produce reasonable personalized recommendations to users.

6 Conclusions

RSs are helpful for consumers making choices among different products. To address the limitations of current AHC methods applied to RSs, this paper proposes a CCEHC approach for providing personalised product recommendations. CCEHC consists of two major parts: CCR and hierarchical clustering. CCR is used for user preferences elicitation. The user preferences elicited by CCR can be used to weigh the multi-level product attributes and quantify the nominal attribute values. The product values can be calculated by considering the attribute weights and normalised numerical attribute values. Hierarchical clustering is used to group similar products according to their product values. Recommendations can be produced according to the product values and clustering results. The applications of a laptop RS, where the dataset is collected by this research, and a workstation RS with an open dataset are demonstrated to confirm the validity and applicability of the proposed method. In RS applications, CCEHC can provide a top-10 list of products and similar products recommendations to customers based on their preferences provided.

The CCEHC method can be considered as an expert system that serves the recommendation function. As CCR can be used for expert judgments and user preferences, product data with human input can be processed by the clustering method and recommendations can be generated. The experimental results demonstrated that the proposed CCEHC method can provide personalised recommendations based on different user preferences. CCR outperformed AHP in reflecting the preferences of both expert(s) and users.

There are several possible paths for future work based on this research. Firstly, other clustering methods can be considered. Secondly, the interfaces for user input and recommendation output could be further improved for a better user experience. Thirdly, the approach to addressing missing values, such as user input data and the product data, could be further investigated. Fourthly, regarding the size of the data, the proposed method could be further improved to process large scale of data. Finally, to extend the application areas, the proposed CCEHC method could be further applied to numerous other product recommendation applications such as movie, music, book, cars, and smartphones.

Appendix

See Tables 15, 16 and 17.

Table 15 Raw matrix D of 27 laptops

ID	Laptop model	L_1	L_2	L_3	L_4	L_5	L_6	L_7	L_8	L_9	L_{10}	L_{11}	L_{12}	L_{13}
1	Lenovo Yoga3 14-IFI	3367	Win8	4	1	256	0	Lenovo	14	2,073,600	2385	1.6	6	7
2	Lenovo Y430p AT-ISE	6830	Win8	8	0	1000	0	Lenovo	14	1,049,088	4385	2.5	5	6
3	Lenovo ThinkPad E540 20C60019CD	3882	Linux	4	0	1000	0	Lenovo	15.6	1,049,088	1848	2.44	6	4
4	Dell XPS 11 (XPS11D-1508T)	2039	Win8	4	1	256	Dell	0	11.6	3,686,400	638	1.13	6	8
5	Dell Inspiron 15 (INS15UD-1748S)	3807	Win8	8	0	1000	Dell	0	15.6	1,049,088	1705	2.3	4	5
6	Dell Inspiron 15 7000 (Ins15BD-1748)	3807	Win8	8	0	1000	Dell	0	15.6	1,049,088	1857	2.11	7	7
7	MacBook 256 GB	2589	OS	8	1	256	Apple	0	12	3,317,760	658	0.92	9	9
8	MacBook Pro 15'	6990	OS	16	1	512	Apple	0	15.4	5,184,000	2543	2.02	9	17
9	MacBook Air (MJVE2CH/A)	3393	OS	4	1	128	Apple	0	13.3	1,296,000	1333	1.35	9	7
10	ASUS A550JK4200	4361	Win8	4	0	1000	0	ASUS	15.6	2,073,600	4385	2.35	4	5
11	ASUS GFX71JY4710	6980	Win8	16	0.5	1256	0	ASUS	17.3	2,073,600	12,632	4.8	3	19
12	ASUS U305FA5Y71 (8 GB/ 256 GB)	2503	Win8	8	4	256	0	ASUS	13.3	2,073,600	658	1.2	10	6
13	Acer VN7-591G-56BD	3367	Win8	4	0	500	0	Acer	15.6	2,073,600	4385	2.4	4	5
14	Acer E1-470G-33212G50Dnkk	2229	Linux	2	0	500	0	Acer	14	1,049,088	1213	2.1	4	2
15	Acer VN7-791G-78KL	7060	Win8	8	0.5	1064	0	Acer	17.3	2,073,600	9840	3	3	8
16	HP Envy 15-k222tx	3367	Win8	4	0	1000	HP	0	15.6	1,049,088	4385	2.34	4	5
17	HP ProBook 440 G2 (J7W06PA)	3420	Win7	8	0	1500	HP	0	14	1,440,000	1784	1.83	9	6
18	HP Pavilion 11-h112tu × 2 (G0A07PA)	2071	Win8	4	1	128	HP	0	11.6	1,049,088	638	1.49	6	5
19	Samsung 910S3G-K04	1375	Win8	4	1	128	0	Samsung	13.3	1,049,088	638	1.44	5	4
20	Samsung 930X2K-K01	2492	Win8	4	1	128	0	Samsung	12.2	4,096,000	658	0.95	7	8
21	Samsung 900X3K-K01	3807	Win8	8	1	256	0	Samsung	13.3	5,760,000	968	1.07	6	10
22	Surface Pro 3 (i3/64 GB)	1675	Win8	4	1	64	Microsoft	0	12	3,110,400	638	0.8	9	4
23	Surface Pro 3 (i7/512 GB/ Profession)	3249	Win8	8	1	512	Microsoft	0	12	3,110,400	1033	0.8	9	12
24	Surface 3 (4 GB/128 GB)	2320	Win8	4	1	128	Microsoft	0	10.8	2,457,600	638	0.887	10	4
25	Alienware 15 (ALW15ED-1718)	6980	Win8	16	0.5	1128	Alienware	0	15.6	2,073,600	9809	3.207	4	15
26	Alienware 13 (ALW13ED-2708)	3807	Win8	8	1	384	Alienware	0	13.3	2,073,600	5249	2.058	3	13
27	Alienware 17 (ALW17ED-2728)	7060	Win8	16	0.5	1512	Alienware	0	17.3	2,073,600	12,632	3.78	3	21

Table 16 Comparison matrices of User C and User D (for attribute weights and nominal attribute values)

User C												User D															
B_0	δ_1	δ_2	δ_3	δ_4	δ_5	δ_6	δ_7	B_2	$\delta_{2,1}$	$\delta_{2,2}$	$\delta_{2,3}$	B_0	δ_1	δ_2	δ_3	δ_4	δ_5	δ_6	δ_7	B_2	$\delta_{2,1}$	$\delta_{2,2}$	$\delta_{2,3}$				
δ_1	0	3	2	-1	-2	1	4	$\delta_{2,1}$	0	1	-3	δ_1	0	-1	-4	-2	-2	1	0	$\delta_{2,1}$	0	3	-1				
δ_2	-3	0	-1	-3	-4	-2	1	$\delta_{2,2}$	-1	0	-3	δ_2	1	0	-3	-1	-2	2	1	$\delta_{2,2}$	-3	0	-4				
δ_3	-2	1	0	-3	-3	-1	2	$\delta_{2,3}$	3	3	0	δ_3	4	3	0	2	2	4	3	$\delta_{2,3}$	1	4	0				
δ_4	1	3	3	0	1	2	5	AI=0.048				2	1	δ_4	0	0	3	2	AI=0								
δ_5	2	4	3	-1	0	2	5	B_4	$\delta_{4,1}$	$\delta_{4,2}$	$\delta_{4,3}$	δ_5	2	2	-2	0	0	3	2	B_4	$\delta_{4,1}$	$\delta_{4,2}$	$\delta_{4,3}$				
δ_6	-1	2	1	-2	-2	0	3	$\delta_{4,1}$	0	-2	1	δ_6	-1	-2	-4	-3	-3	0	-1	$\delta_{4,1}$	0	-3	3				
δ_7	-4	-1	-2	-5	-5	-3	0	$\delta_{4,1}$	2	0	3	δ_7	0	-1	-3	-2	-2	1	0	$\delta_{4,1}$	3	0	5				
AI = 0.057												AI = 0.050															
												$\delta_{4,3}$ -1 -3 0															
												AI = 0															
L_3	v_1	v_2	v_3	v_4					L_8	v_1	v_2	v_3	v_4	v_1	v_2	v_3	v_4	L_8	v_1	v_2	v_3	v_4					
v_1	0	2	1	-1					v_1	0	-1	2	1	v_1	0	3	2	-2	v_1	0	-1	4	1				
v_2	-2	0	-1	-2					v_2	1	0	3	2	v_2	2	0	-1	-5	v_2	1	0	5	2				
v_3	-1	1	0	-3					v_3	-2	-3	0	-1	v_3	1	-1	0	-4	v_3	-4	-5	0	-4				
v_4	1	2	3	0					v_4	-1	-2	1	0	v_4	-1	-3	-2	0	v_4	-1	-2	4	0				
AI = 0.077												AI = 0								AI = 0.042							
L_I	v_1	v_2	v_3					AI = 0				L_I	v_1	v_2	v_3					L_{I0}	v_1	v_2	v_3				
v_1	0	2	1					B_3	$\delta_{3,1}$	$\delta_{3,2}$	$\delta_{3,3}$	L_I	v_1	v_2	v_3	B_3	$\delta_{3,1}$	$\delta_{3,2}$	$\delta_{3,3}$	L_{I0}	v_1	v_2	v_3				
v_2	-2	0	1					$\delta_{3,1}$	0	0	4	v_1	0	0	2	$\delta_{3,1}$	0	5	0	v_1	0	0	0				
v_3	-1	1	0					$\delta_{3,2}$	0	0	0	v_2	0	0	2	$\delta_{3,2}$	-5	0	0	v_2	0	0	0				
AI = 0.096												AI = 0								AI = 0							
												v_3 -2 -2 0															
												AI = 0								AI = 0							

Table 17 Normalized data matrix D' of 27 laptops For User A

ID	L_1	L_2	L_3	L_4	L_5	L_6	L_7	L_8	L_9	L_{10}	L_{11}	L_{12}	L_{13}
1	0.477	0.692	0.250	1.000	0.169	0.000	1.000	0.809	0.360	0.189	0.500	0.600	0.286
2	0.967	0.692	0.500	0.000	0.661	0.000	1.000	0.809	0.182	0.347	0.320	0.500	0.333
3	0.550	0.692	0.250	0.000	0.661	0.000	1.000	0.902	0.182	0.146	0.328	0.600	0.500
4	0.289	0.692	0.250	1.000	0.169	0.672	0.000	0.671	0.640	0.051	0.708	0.600	0.250
5	0.539	0.692	0.500	0.000	0.661	0.672	0.000	0.902	0.182	0.135	0.348	0.400	0.400
6	0.539	0.692	0.500	0.000	0.661	0.672	0.000	0.902	0.182	0.147	0.379	0.700	0.286
7	0.367	0.897	0.500	1.000	0.169	0.983	0.000	0.694	0.576	0.052	0.870	0.900	0.222
8	0.990	0.897	1.000	1.000	0.339	0.983	0.000	0.890	0.900	0.201	0.396	0.900	0.118
9	0.481	0.897	0.250	1.000	0.085	0.983	0.000	0.769	0.225	0.106	0.593	0.900	0.286
10	0.618	0.692	0.250	0.000	0.661	0.000	0.829	0.902	0.360	0.347	0.340	0.400	0.400
11	0.989	0.692	1.000	0.500	0.831	0.000	0.829	1.000	0.360	1.000	0.167	0.300	0.105
12	0.355	0.692	0.500	1.000	0.169	0.000	0.829	0.769	0.360	0.052	0.667	1.000	0.333
13	0.477	0.692	0.250	0.000	0.331	0.000	0.732	0.902	0.360	0.347	0.333	0.400	0.400
14	0.316	0.692	0.125	0.000	0.331	0.000	0.732	0.809	0.182	0.096	0.381	0.400	1.000
15	1.000	0.692	0.500	0.500	0.704	0.000	0.732	1.000	0.360	0.779	0.267	0.300	0.250
16	0.477	0.692	0.250	0.000	0.661	0.466	0.000	0.902	0.182	0.347	0.342	0.400	0.400
17	0.484	1.000	0.500	0.000	0.992	0.466	0.000	0.809	0.250	0.141	0.437	0.900	0.333
18	0.293	0.692	0.250	1.000	0.085	0.466	0.000	0.671	0.182	0.051	0.537	0.600	0.400
19	0.195	0.692	0.250	1.000	0.085	0.000	0.561	0.769	0.182	0.051	0.556	0.500	0.500
20	0.353	0.692	0.250	1.000	0.085	0.000	0.561	0.705	0.711	0.052	0.842	0.700	0.250
21	0.539	0.692	0.500	1.000	0.169	0.000	0.561	0.769	1.000	0.077	0.748	0.600	0.200
22	0.237	0.692	0.250	1.000	0.042	0.328	0.000	0.694	0.540	0.051	1.000	0.900	0.500
23	0.460	0.692	0.500	1.000	0.339	0.328	0.000	0.694	0.540	0.082	1.000	0.900	0.167
24	0.329	0.692	0.250	1.000	0.085	0.328	0.000	0.624	0.427	0.051	0.902	1.000	0.500
25	0.989	0.692	1.000	0.500	0.746	1.000	0.000	0.902	0.360	0.777	0.249	0.400	0.133
26	0.539	0.692	0.500	1.000	0.254	1.000	0.000	0.769	0.360	0.416	0.389	0.300	0.154
27	1.000	0.692	1.000	0.500	1.000	1.000	0.000	1.000	0.360	1.000	0.212	0.300	0.095

Acknowledgements The research work reported in this paper is partially supported by Research Grants from Shanghai Municipal Science and Technology Major Project (Project Number 2021SHZDZX0103) and National Natural Science Foundation of China (Project Number 61503306).

Declarations

Conflict of interest The authors declare they have no conflicts of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright

holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- 3DMARK (2015) 3DMARK. <http://www.futuremark.com/benchmarks/legacy>. Retrieved 3 2015
- Adomavicius G, Manouselis N, Kwon Y (2011) Multi-Criteria Recommender Systems. In: Ricci F, Rokach L, Shapira B, Kantor P (eds) Recommender Systems Handbook. Springer, Boston, MA, pp.769–803
- Aggarwal CC (2016) Recommender systems. Springer International Publishing, Berlin
- Davies DL, Bouldin DW (1979) A cluster separation measure. IEEE Trans Pattern Anal Mach Intell 2:224–227
- de Aguiar Neto FS, da Costa AF, Manzato MG, Campello RJ (2020) Pre-processing approaches for collaborative filtering based on hierarchical clustering. Inf Sci 534:172–191
- Frémal S, Lecron F (2017) Weighting strategies for a recommender system using item clustering based on genres. Expert Syst Appl 77:105–113

- Guan C (2018) Evolutionary and swarm algorithm optimized density-based clustering and classification for data analytics. Ph.D. thesis, University of Liverpool
- Guan C, Yuen KK (2015) Towards a hybrid approach of primitive cognitive network process and agglomerative hierarchical clustering for music recommendation. In: Heterogeneous networking for quality, reliability, security and robustness (QSHINE), 2015 11th international conference on. IEEE, pp 206–209
- Guan C, Yuen KK, Coenen F (2018) Particle swarm optimized density-based clustering and classification: supervised and unsupervised learning approaches. *Swarm Evol Comput* 44:876–896
- Gupta U, Patil N (2015) Recommender system based on hierarchical clustering algorithm chameleon. In: 2015 IEEE international advance computing conference (IACC)
- Han J, Pei J, Kamber M (2011) Data mining: concepts and techniques. Elsevier, Amsterdam
- Haruna K, Akmar Ismail M, Suhendroyono S, Damiasih D, Pierewan AC, Chiroma H, Herawan T (2017) Context-aware recommender system: a review of recent developmental process and future research direction. *Appl Sci* 46:1211
- Hinduja A, Pandey M (2018) An intuitionistic fuzzy AHP based multi criteria recommender system for life insurance products. *Int J Adv Stud Comput Sci Eng* 38(5):1–8
- Kaggle (2018) Characteristics and price for 1300 laptop models. (Kaggle Inc.). <https://www.kaggle.com/ionaskel/laptop-prices>. Retrieved 11 2018
- Karthikeyan R, Michael G, Kumaravel A (2017) A housing selection method for design, implementation & evaluation for web based recommended systems. *Int J Pure Appl Math* 42(3):23–28
- Katarya R, Verma OP (2017) An effective web page recommender system with fuzzy c-mean clustering. *Multim Tools Appl* 34(2):21481–21496
- Kotkov D, Wang S, Veijalainen J (2016) A survey of serendipity in recommender systems. *Knowl Based Syst* 111:180–192
- Kunaver M, Požrl T (2017) Diversity in recommender systems—a survey. *Knowl-Based Syst* 12(4):154–162
- Lika B, Kolomvatsos K, Hadjiefthymiades S (2014) Facing the cold start problem in recommender systems. *Expert Syst Appl* 41(4):2065–2073
- Ma YY, Zhang HR, Xu YY, Gao L (2018) Three-way recommendation integrating global and local information. *J Eng* 16:1397–1401
- Murtagh F (1983) A survey of recent advances in hierarchical clustering algorithms. *Comput J* 26(4):354–359
- Nilashi MB (2017) Clustering items for collaborative A recommender system for tourism industry using cluster ensemble and prediction machine learning techniques. *Comput Ind Eng* 109:357–368
- Pamućar D, Stević Ž, Zavadskas EK (2018) Integration of interval rough AHP and interval rough MABAC methods for evaluating university web pages. *Appl Soft Comput* 42(3):141–163
- Rousseeuw PJ (1987) Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J Comput Appl Math* 20:53–65
- Saaty T (1980) Analytic hierarchy process: planning, priority, setting resource allocation. McGraw-Hill, New York
- Selvi C, Sivasankar E (2019) A novel optimization algorithm for recommender system using modified fuzzy c-means clustering approach. *Soft Comput* 23:1901–1916
- Song WS (2018) An environmentally conscious PSS recommendation method based on users' vague ratings: a rough multi-criteria approach. *J Clean Prod* 26(2):1592–1606
- van Capelleveen G, Amrit C, Yazan DM, Zijm H (2019) The recommender canvas: a model for developing and documenting recommender system design. *Expert Syst Appl* 22(3):97–117
- Viktoratos I, Tsadiras A, Bassiliades N (2018) Combining community-based knowledge with association rule mining to alleviate the cold start problem in context-aware recommender systems. *Expert Syst Appl* 108:78–90. <https://doi.org/10.1016/j.eswa.2018.01.044>
- Volkovs M, Yu G, Poutanen T (2017) Dropoutnet: addressing cold start in recommender systems. *Adv Neural Inf Process Syst* 40(3):4957–4966
- Wang Y, Tseng MM (2013) Customized products recommendation based on probabilistic relevance model. *J Intell Manuf* 24(5):951–960
- Ward JH Jr (1963) Hierarchical grouping to optimize an objective function. *J Am Stat Assoc* 58(301):236–244
- Yuen KK (2009) Cognitive network process with fuzzy soft computing technique in collective decision aiding. The Hong Kong Polytechnic University
- Yuen KK (2012) Pairwise opposite matrix and its cognitive prioritization operators: comparisons with pairwise reciprocal matrix and analytic prioritization operators. *J Oper Res Soc* 63(3):322–338
- Yuen KK (2014a) Fuzzy cognitive network process: comparisons with fuzzy analytic hierarchy process in new product development strategy. *IEEE Trans Fuzzy Syst* 22(3):597–610
- Yuen KK (2014b) The Primitive cognitive network process in healthcare and medical decision making: comparisons with the analytic hierarchy process. *Appl Soft Comput* 14:109–119
- Zhang HR, Min F, Shi B (2017) Regression-based three-way recommendation. *Inf Sci* 378:444–461. <https://doi.org/10.1016/j.ins.2016.03.019>
- Zhang C, Zhang H, Wang J (2018) Personalized restaurant recommendation method combining group correlations and customer preferences. *Inf Sci* 454–455:128–143. <https://doi.org/10.1016/j.ins.2018.04.061>
- Zheng L, Li L, Hong W, Li T (2013) PENETRATE: personalized news recommendation using ensemble hierarchical clustering. *Expert Syst Appl* 6(3):2127–2136

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