Information Sciences 686 (2025) 121336

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Leveraging ensemble clustering for privacy-preserving data fusion: Analysis of big social-media data in tourism

Natthakan Iam-On [a], Tossapon Boongoen [a,*], Nitin Naik [b], Longzhi Yang [c]

[a] *Advanced Reasoning Research Group, Department of Computer Science, Aberystwyth University, Aberystwyth, UK*
[b] *School of Informatics and Digital Engineering, Aston University, Birmingham, UK*
[c] *Department of Computer and Information Sciences, Northumbria University, Newcastle, UK*

ARTICLE INFO

Keywords:
Social media
Big data
Tourism
Ensemble clustering
Data fusion
Privacy preserving

ABSTRACT

Discovering knowledge from social media becomes a trend in many domains such as tourism, where users' feedback and rating are the basis of recommendation systems. In this context, cluster analysis has been a major tool to disclose user groups by which the process of collaborative filtering can better determine a personalised suggestion. Matching this to the curse of big data is a challenge with previous studies either implementing conventional techniques on a distributed system or making use of data sampling. Specific to ensemble clustering, only a few aim to obtain both scalability and privacy preserving that are significant to handling social data. This paper presents a new bi-level framework of ensemble clustering in which an instance-segment based analysis is adopted to ensure data privacy and reduce the complexity of clustering the whole dataset. Unlike existing studies, instead of drawing a single clustering from each segment, multiple clusterings are selected to better represent instances therein. Based on published tourism datasets and different experimental settings, the new approach usually outperforms its baselines whilst being competitive to related methods found in the literature. Additional case studies on simulated big datasets and noisy variations are reported and discussed in addition to the analysis of algorithmic parameters.

1. Introduction

Social media has played an important role in human interaction, group behaviour, and business operations. In the past years, more and more people catch up with this trend, thus raising over a billion new online users [1]. It is not surprised the amount of data generated among social platforms like Twitter, Facebook and YouTube is huge, with around 18 millions text messages being exchanged in a minute and 90% of data generated over the Internet being seen as images and videos [2]. Given this pool of big data, many studies investigate applications of machine learning to extract new knowledge for a variety of problems. These include, customer behavioural analysis, disaster management, and pandemics tracking. Specific to hospitality and tourism, social networks like TripAdvisor and Yelp happen to be the sources to harvest business insight using user review and sentimental analysis [3]. See [4] for further detail on implications of modern information, communication technologies and social media on the tourism industry.

Given the tendency that most travellers prefer to explore new destinations, a recommendation system has been introduced to deliver personalised responses. Traditional systems usually exploit collaborative filtering techniques to a pile of data available on

[*] Corresponding author.
E-mail address: tob45@aber.ac.uk (T. Boongoen).

https://doi.org/10.1016/j.ins.2024.121336
Received 15 February 2024; Received in revised form 8 August 2024; Accepted 9 August 2024

Available online 14 August 2024

social media. This aims to project possible options based on previous patterns similar to a given preference [5]. Cluster analysis is one of machine learning models to support this task, especially for scaling up a collaborative filtering engine to big online data [6]. By restricting a system to clusters of the data collection under question, the overall complexity can be substantially reduced. In fact, the resulting recommendation may be more accurate and in line with those traits seen with actual traveller groups, disclosed from online reviews, feedbacks and blogs [7]. Apart from scalability, some of modern big data algorithms are also designed to facilitate a privacy-preserving analytic, which becomes desirable in the context of online media [8]. Realising both through extending conventional methods has been a challenge, with only a few extending focusing on the methodology of ensemble clustering [9].

Background and assumption: A long list of extensions to classical clustering techniques for big data is seen. With the common goal of scalability, most are modelled around concepts of data sampling [10], dimensional reduction [11], multi-view analysis [12], parallel and distributed computing [13]. Only a handful of these such as [14–16] actually consider the conjunction between privacy preserving and scalability. Specific to ensemble clustering, a combination of multi-view clustering and tensor-based representation is put forward by [14]. The scalability is achieved by aggregating multiple clustering results, each of which is created using a subset of original features. An additional cost occurs from the need to map a given feature space to its intermediate form. Yet, it may not be straight forward to interpret or compare outputs across data segments or sources. In contrary, the multi-agent model of [15] and the multiple-batches processing [16] tackle the scalability problem through a division of data samples. It is assumed that non-overlapping subsets or segments of raw data can be handled and analysed separately, while their representatives (i.e., cluster centroids) are aggregated to derive a correct approximation of the whole data collection. This gives way to an efficient generation of a global clustering that is later propagated to those initial subsets for further inference at the instance level.

Problem and scope: In [15,16], one clustering with a specific number of clusters is produced from each segment using the k-means algorithm. Then, the corresponding cluster centroids are aligned to determine the consensus partition. By those segment-specific cluster centroids, the global clustering process preserves data privacy. Despite this, having each segment represented by a single clustering may limit the level of information extracted from all the samples therein. There might be some sub-regions not well presented by a small set of centroids. Yet, exploiting different data partitions of the same number of clusters is usually less effective to promote diversity within an ensemble than a random-selection alternative [17]. The current work extends this line of research with an improved framework, which maintains data privacy through k-means based centroids and improves the quality of ensemble clustering by making use of multiple and diverse clusterings per instance segment. Putting this into the test, it is applied to identify user groups in tourism related data [18] and simulated big datasets. Similar to the empirical study of [15] and [16], the proposed experiments are implemented in a single machine to initially demonstrate the potential of a new framework.

Contributions: These are identified as follows.

- As an extension to the approaches of [15,16], this work introduces a new bi-level, multiple-clustering framework for analysing big data. Firstly at the segment level, representative clusterings of each segment are drawn from a pool of data partitions, initially created by k-means with different numbers of clusters and other parameter settings. This selection procedure is designed as an iterative greedy search that aims to maximise the diversity within the target set of clusterings [17]. A desired size of this collection serves as the termination condition for the forward search, which repeatedly selects one from the pool to obtain the highest group-wise diversity. This is an efficient implementation, for which a more complex optimisation or swarm intelligence techniques [19] can be exploited to avoid a sub-optimal solution. Provided this, the proposed framework better represents segment-level information than the previous methods using the optimised multiple-clustering concept. Then, at the dataset-wide level or global clustering, data privacy is ensured through the use of those segment-specific centroids. Of course, there is a tradeoff between improved quality of centroids and a higher complexity brought about by the exploitation of multiple clusterings.
- At the dataset-wide level, those centroids are considered as inputs to an ensemble clustering, for which benchmark graph- and similarity-based techniques like CSPA, HBGF and EAC-AL [9] are investigated. With a specified number of clusters, the resulting partition is then used to map segment-specific instances to one of the so-called global clusters. Compared to the previous studies that limit only to a direct approach to ensemble clustering (i.e., searching for the optimal alignment of clusters from different segments), the proposed framework allows other benchmark ensemble clustering methods to be used to deliver a dataset-wide partition. Since the direct approach is often less effective than others, the new framework may boost the quality of ensemble clustering in this context.
- This work also provides a comparative study between the proposed approach, baseline models and related techniques found in the literature. It is based on published datasets in the domain of tourism [18] and simulated big datasets. In addition to parameter analysis, a further case study on datasets with noisy feature values is also conducted to assess the robustness of both new and baseline methods. This provides an insight of applying these in a real-world setting where data can often be imperfect.

The rest of this paper is organised as follows. Detail of related works is provided in Section 2. This is followed by the description of the proposed method in Section 3. Then, Section 4 presents experiment design, evaluation results, discussion on parameter analysis and implications of the new model. The paper is concluded in Section 5 with perspectives of future research.

2. Related works

Within the domain of hospitality and tourism, the recent review of [20] has pointed out an exponential growth of published works on social media and big data analysis since 2010. Moreover, a number of major topics widely investigated by scholars include demand prediction, tourist experience and satisfaction. Online platforms and social networks such as Twitter and travel reviews on

TripAdvisor appear to be the common sources of data. Before 2017, simple analytical methods like regression and statistical text analysis have been a typical choice to disclose useful knowledge. Later, more sophisticated algorithms are introduced for sentimental analysis and topic modelling. Of course, machine learning techniques have also been extensively explored in this field, especially to support the development of a recommendation system [21]. To finalise destinations, points of interest and perhaps hotel choices, most tourists generally rely on this innovative approach to retrieve relevant information from a huge volume of data generated and shared regularly online. Among others, data clustering is often adopted to provide descriptive analysis on customer ratings and online reviews, which can be extended to finding user groups or representative profiles [22]. As such, traits identified from past records can be exploited as references for future recommendations, which are in par with actual users' preference. Some systems make use of cluster analysis to divide users based on places they previously visited and present the same suggestion to members of each cluster [23].

Specific to collaborative filtering that is a branch of recommendation system, a prediction to user's preference is achieved through comparisons between the input ratings and those memorised or modelled from historical data. Despite the flexibility to view a suggestion along user- or item-based perspective, the underlying calculation suffers from two main drawbacks of sparsity and scalability. To alleviate the former, [24] introduces a clustering-based smoothing method to deal with unrated data. For the other, clustering techniques are included in a target system to reduce the space of user data such that calculations are constrained to small groups, not the entire population. For example, k-means has been an efficient tool to determine tourist segments with respect to ratings and demographic properties [25]. There are different modifications that aim to improve this baseline further using a domain-specific ontology [26] and the concept of fuzzy or soft clustering, respectively. In particular, the fuzzy c-means algorithm is picked up to provide user groups for the recommendation system called Personalised Sightseeing Information System or PSIS [27]. Even though the complexity of k-means and alike is usually linear to the number of data instances (N), i.e., $O(N)$, it becomes clear that a constant update of tourist segments with new records can still be expensive. Yet, a hierarchically structure of those segments may lead to a more accurate response, but its implementation is strongly constrained by the magnitude of N. Based on these, advance clustering methods developed within the umbrella of big data are candidates to deliver improvements on collaborative filtering [7]. In addition to scalability, the issue of privacy-preserving has also gained a great deal of attention, given that recommendations are normally distilled from social media data, owned partly by different parties [8]. With k-means and its variants being the common baseline, Table 1 summarises some of the scalable clustering methods that strive to address the issue of privacy preserving.

According to the review of [28], sampling-based clustering techniques like BIRCH, CURE, and CLARANS emerge as initial solutions to the curse of big data. The idea is to draw a subset of samples from the whole dataset for the actual clustering stage, with the resulting data partition being mapped to all instances later. Within this category, major factors motivating new investigations include size and quality of the sampling. In addition, the concept of subspace clustering has been introduced to determine a feature subset that is informative for each of the clusters [10]. Despite good results reported therein, these techniques do not pay attention to the protection of data privacy since large databases under investigations usually belong to a single institution. This issue has been handled partly by several projection-based algorithms included in Table 1, which aim to reduce the data dimension through a random projection [11,29] or a learning framework like AutoEncoder [30]. However, they seem appropriate for a high dimensional dataset, where the number of instances tends to be small or moderate. It is noteworthy that [29] is the pioneer of exploiting multiple projected feature sets to develop an accurate ensemble clustering. This practice is generalised by the approach of multi-view clustering, where multiple clusterings are produced from different feature subsets to present various possible results. As such, the process of cluster analysis becomes less expensive, with possible representation schemes of tensor [12] and graph [31] to partly preserve data privacy. Further this line of research, [32] implements a multi-view clustering framework on the cloud computing environment, in which tensor-based features are also encrypted.

Based on the famous Map-Reduce stack, several scalable clustering methods like Map-Reduce based CURE [34,36] have been put forward to both accommodate a big volume of data to be examined as well as protect its privacy. The latter is obtained through a native mechanism of distributed data management where a number of data blocks are stored and manipulated at local units called data nodes. This so-called distributed-computing approach is not only coupled with the classical k-means [13,37,38], but also extended to its soft clustering counterpart like the fuzzy c-means technique [35]. Apart from the families of clustering techniques presented thus far, another group of works consider the use of instance segments to resolve the difficulty with data volume. Table 2 compares techniques belonging to the family that are highly related to the current work on scaling up the ensemble clustering [17,9] to big and privacy-preserving data analysis.

As summarised in the previous table, initial models like that of [39] have managed to scale the determination of instance-to-instance similarity measurement up to a big data collection. With each segment containing only a subset of all instances (N), a pairwise similarity matrix $N \times N$ can be constructed, where measures between those not in the segment are simply regarded as zeros (i.e., unknown relations). A collection of partly overlapping segments are considered to allow the corresponding matrices to complement each other, especially for those unknown in some. Given this centralised aggregation framework, the method does not strongly support the privacy protection. A similar issue has also been observed with the study of [41]. In spite of facilitating a de-centralised system using non-overlapping segments, instance-related detail is made available for the final phase. To promote a more complete privacy preservation, both [15] and [16] similarly exploit k-means to create data partitions from non-overlapping segments, then only cluster-based representatives are forwarded or shared across segments to disclose the consensus clustering. However, a single clustering with a fixed number of clusters has been generated from each of the instance segments, thus limiting the solution space being covered. These motivate the present research that aims to implement multiple clusterings at each segment as a way to boost diversity among inputs in the stage of consensus cluster analysis. In addition, the resulting framework is generally applicable to different settings of virtually [16] and physically [15] distributed systems.

Table 1
Summarisation of scalable clustering methods and their concepts of handling the privacy preserving issue. Abbreviations exploited herein to denote different approaches to scalable clustering, SP: sampling-based, PJ: projection-based, FS: feature-selection based, MV: multi-view analysis, and DC: distributed computing.

Methods	Approach to scalability	Approach to privacy preserving	Brief description
BIRCH, CURE and CLARANS [28]	SP	n/a	A set of samples are initially drawn from the target dataset for a cluster analysis, then the result is mapped to all data instances in the original set
Scalable subspace clustering [10]	SP	n/a	Samples drawn from a dataset is used to find an appropriate feature subset (or subspace) for each cluster, then the result is propagated to all data instances
Projection-based ensemble clustering [29]	PJ	Partly, with projected features	Clustering the whole set of instances, i.e., with no sampling or segmentation. This is repeated multiple times to create inputs for the ensemble clustering
Random projections [11]	PJ	Partly, with projected features	Clustering the whole set of instances, i.e., with no sampling or segmentation
Structured AutoEncoder [30]	PJ	Partly, transformed features are created using AutoEncoder	Clustering the whole set of instances, i.e., with no sampling or segmentation. An optimal feature subspace is determined for each of the resulting clusters
Parallel ensemble clustering [33]	FS	n/a	A parallelism of base clustering generation and dimension reduction using unsupervised feature selection methods. Each clustering is produced from the whole dataset
Tensor constrained multi-view subspace clustering [12]	MV	Partly, using a tensor based representation	Different views (feature spaces) of the whole dataset are exploited to give different results that can be fused later. It is similar to [14] with a focus on subspace determination.
Auto-weighted multiple graph learning [31]	MV	Partly, using view-specific graphs	In the fusing stage, view-specific graphs are prioritised and assigned with distinct weights
Privacy-preserving tensor-based multiple clustering [32]	MV and DC	Fully, data is presented in a tensor-based space and encrypted	Use a cloud-based framework to obtain multiple view-specific results, with an application in the IoT domain
Map-Reduce based CURE [34]	DC	Fully, different instance segments are stored and analysed at nodes	Implemented on the Map-Reduce stack, allowing a clustering process to be distributed across data nodes
Weighted-kernel possibilistic c-means [35]	DC	Fully, different instance segments are stored and analysed at nodes	An implementation of soft clustering, with a new focus on identifying and dropping noisy instances
Knowledge-based document clustering [36]	DC	Fully, different instance segments are stored and analysed at nodes	Similar to Map-Reduce based CURE, with a focus on text document clustering using knowledge ontology
Differential privacy protecting k-means [13]	DC	Fully, with data nodes and noise injection	An implementation of distributed k-means, with an approach to share boundary information among data nodes
Cloud-based k-means [37]	DC	Fully, cluster centres are encrypted/shared	Another variation of distributed k-means, with a new focus on exchanges of encrypted information
Fixed-width clustering [38]	DC	Fully, instances are encrypted/shared	Similar to cloud-based k-means w.r.t. encrypted information

3. Proposed method

This section provides an explanation of the proposed framework, which can be referred to as Scalable and Privacy-Preserving Ensemble Clustering or SPP-EC hereafter. It is hypothesised that multiple clusterings of each instance segment may promote the diversity amongst segment-specific inputs within the process of ensemble clustering. In order to acquire this, the new method has been designed with three processing stages of (i) preparing a pool of multiple clusterings for each of the segments; (ii) selecting a subset of those to establish a representative collection of cluster centroids, using a greedy forward-search; and (iii) aggregating cluster representatives to obtain the consensus clustering to which all original instances will be later mapped. Fig. 1 provides an overview of this bi-level framework in which those three processing stages are depicted in a logical order. In particular, those parts highlighted in blue colour are new in this context of segment-based analysis. These include the generation of multiple data partitions and optimised selection of representative clusterings, which have not been exploited by the existing methods. Likewise, through the provision of multiple clusterings with different cluster numbers, other well-known ensemble clustering algorithms like CSPA can be reused to deliver an accurate global clustering. The new framework opens up this opportunity that has not been feasible with a single data partition and a fixed number of clusters employed in the previous works.

Table 2
A comparison of instance-segment based techniques and their approaches to handling data privacy.

Method	Approach to privacy preserving	Brief description
Spectral ensemble clustering [39]	Partly, at the stage of clustering data in each of the segments. But, instance-level similarities are shared to create the summarisation across segments.	Instance segments are randomly generated and separately clustered. Then, a segment-specific co-association matrix is produced using the concept of Incomplete Base Partition (IBP). These are later merged to provide an input to the consensus clustering.
Random sample partition-based ensemble clustering [41]	Partly, at the stage of clustering data in each segment. But, detail of instance-cluster associations obtained from these segments are later aggregated to obtain a median partition.	Non-overlapping instance segments are created using the method of [40]. Like other methods listed in this table, k-means or its variants is used to produce a single data partition from each segment (with the unified number of clusters).
Validated distributed ensemble clustering [15]	Fully, both segment-level clustering and exchanges of only cluster representatives in the later stage of consensus clustering.	Non-overlapping instance segments are randomly generated and separately clustered. Later, cluster representatives are used to align cluster labels across segments.
Multiple-batches k-means [16]	Fully, an approach taken is similar to that of [15]	Non-overlapping instance segments are randomly created and separately clustered. Later, cluster representatives are aggregated for the step of dataset-wide clustering. The focus is to develop a model to handle big data in a single machine.

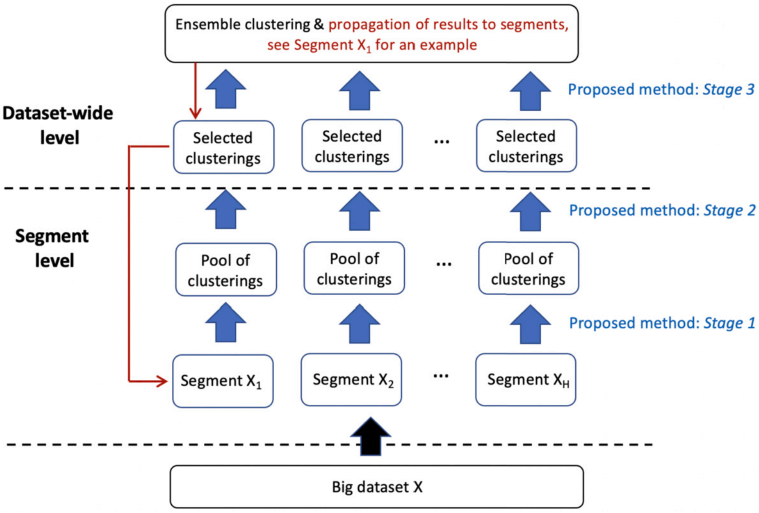

Fig. 1. Overview of the proposed bi-level framework. Note that the propagation of results shown for the first instance segments (see the red arrow line) similarly takes place for all other segments.

3.1. Generating a pool of multiple clusterings

With the goal of setting up segment-based representatives for the following consensus analysis stage, this first phase exploits the concept of ensemble clustering [9] to create a pool of diverse results for each of the instance segments. Let $X \in [0,1]^{N\times D}$ be a big dataset under examination, where each of the N instances or samples is represented by a vector of D normalised feature values $\in [0,1]$. There are H non-overlapping subsets or segments $X_h, h = 1 \dots H$ of X, each of which contains N_h instances, i.e., $N_1 + N_2 + \dots + N_H = N$. To achieve this, the random sampling approach used by [15,16] is adopted here with the ratio of each segment having less than or equal to 10% of the original number of instances. It is worth noting that segment-specific analysis can be more efficient with smaller segments but possibly adding up more work at the dataset-wide level if they are too small. This decision on the random approach is to reflect a common case of big data analysis, where the knowledge of dataset-wide distribution is rarely available. Please consult the report of [40] for another research direction that aims to discover the dataset property across blocks of instances, thus allowing a distribution-preserving sampling.

For a segment X_h, k-means is applied to generate a collection Π^h of B clusterings, i.e., $\Pi^h = \{\pi^h_1, \dots, \pi^h_B\}$. This pool is achieved using a generation function $\alpha(X_h, B)$, with the specific α employed in this work being a combination of the following strategies.

- Random-k strategy: this is used in addition to the random initialisation of k-means. For each clustering, a number of clusters is arbitrarily picked up from the range of $\{2, \dots \sqrt{N_h}\}$ or $\{2, \dots 50\}$ when $\sqrt{N_h} > 50$.

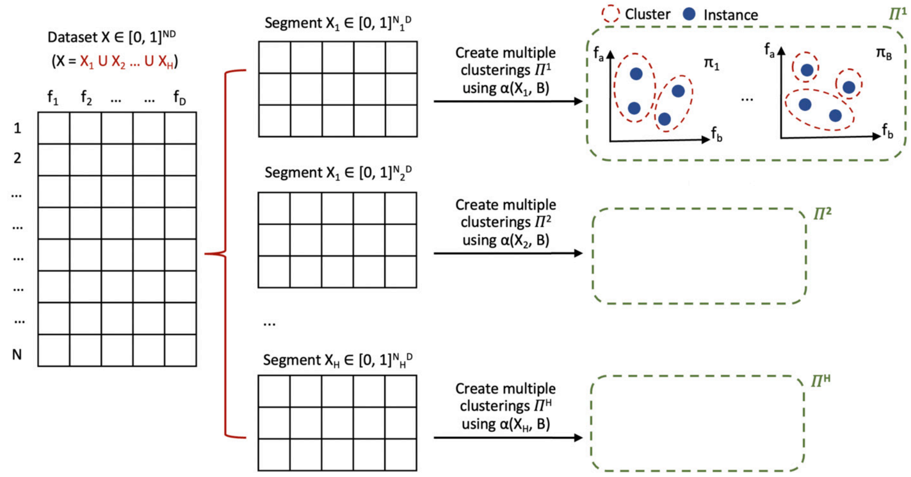

Fig. 2. Illustration of the first stage in SPP-EC framework: generation of a pool B clusterings Π^h for each data segment X_h of the whole dataset X, where $h = 1 \dots H$. To simplify the presentation, clustering results are shown against two dimensions of f_a and f_b, instead of D.

- Random-subspace strategy: a clustering $\pi^h_g, g = 1 \dots B$ is produced from a subset of X_h, or a feature subspace. This is perceived as a random subset $X^*_h \in [0,1]^{N_h \times D^*}$ of the original feature space X_h, whose number of chosen features is estimated by the following.

$$D^* = D^*_{min} + \lfloor \eta(D^*_{max} - D^*_{min}) \rfloor, \tag{1}$$

where $\eta \in [0,1]$ is a uniform random number, D^*_{max} and D^*_{min} stand for upper and lower limits of a subspace X^*_h. Based on the report of [42], these are set to $0.85D$ and $0.75D$, without duplicated features appearing in X^*_h. Fig. 2 illustrates the processing steps in this first stage of SPP-EC.

3.2. Selecting the reference set of multiple clusterings from each segment

This is the stage in which the diversity-driven greedy search for a desired set of clusterings is explained. To be precise, it is divided into multiple steps whose procedural details and supporting information are presented below.

- *Step 1:* For each segment X_h, one of the clusterings in pool Π^h is chosen as a seed of the target collection $\Pi^{h\prime} = \{\pi^{h\prime}_1, \dots, \pi^{h\prime}_M\}$, i.e., the first of its M members. It is simply a clustering $\pi^h_s \in \Pi^h$ with the largest sum of difference to the rest in this pool.

$$\pi^h_s = \underset{\forall \pi^h_g \in \Pi^h}{\arg\max} \sum_{\forall \pi^h_q \in \Pi^h, \pi^h_q \neq \pi^h_g} DA(\pi^h_g, \pi^h_q), \tag{2}$$

where $DA(\pi^h_a, \pi^h_{a'}) \in [0,1]$ denotes a measure of diversity or disagreement between a pair of partitions $\pi^h_a, \pi^h_{a'} \in Pi^h$. It can be calculated by the next question.

$$DA(\pi^h_a, \pi^h_{a'}) = 1 - MI(\pi^h_a, \pi^h_{a'}), \tag{3}$$

provided that $MI(\pi^h_a, \pi^h_{a'}) \in [0,1]$ denotes the metric of mutual information [43] justifying a degree of agreement between two data partitions π^h_a and $\pi^h_{a'}$ of the same set of instances X_h. At the end, π^h_s is excluded from the pool Π^h as a preparation for the next step, i.e., $\Pi^h \leftarrow \Pi^h - \pi^h_s$.
- *Step 2:* Having initialised the reference collection $\Pi^{h\prime}$, this step repeatedly adds a new member until the size of $\Pi^{h\prime}$ becomes M. This iterative process is explained below.
(2.1) In each iteration, all clusterings left in the pool Π^h are examined for their diversity measures against those current members in the reference collection $\Pi^{h\prime}$. Again, a clustering $\pi^h_x \in \Pi^h$ with the highest average of diversity is selected to join $\Pi^{h\prime}$, i.e., $\Pi^{h\prime} \leftarrow \Pi^{h\prime} \cup \pi^h_x$.

$$\pi^h_x = \underset{\forall \pi^h_v \in \Pi^h}{\arg\max} \frac{\sum_{\forall \pi^{h\prime}_w \in \Pi^{h\prime}} DA(\pi^h_v, \pi^{h\prime}_w)}{|\Pi^{h\prime}|} \tag{4}$$

(2.2) This step terminates if $|\Pi^{h\prime}| = M$. Otherwise, update the pool using $\Pi^h \leftarrow \Pi^h - \pi^h_x$ and repeat Step 2 for another new member of the target reference set $\Pi^{h\prime}$.

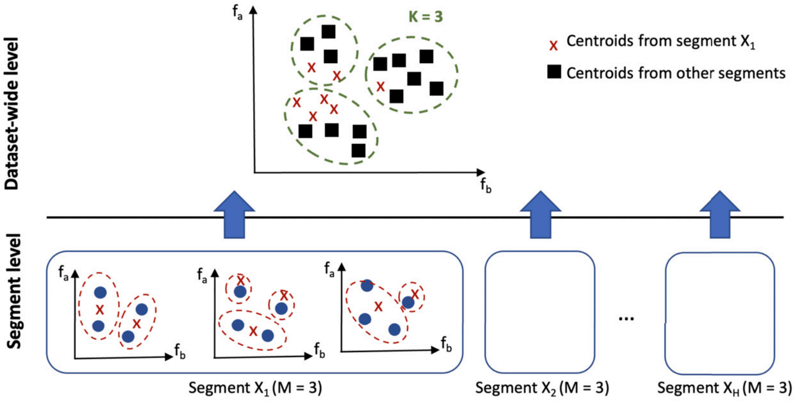

Fig. 3. Illustration of the consensus clustering stage in SPP-EC framework, where centroids disclosed in the segment level are inputs to the next layer of dataset-wide clustering.

3.3. Generating the consensus clustering from segment-specific representatives

Having acquired the reference set $\Pi^{h\prime}$ of M clusterings from each segment X_h, the corresponding set of cluster centroids C_h is estimated, where a centroid $c_i^h \in C_h$ is presented by D normalised feature values, i.e., $c_i^h \in [0,1]^{1\times D}$. By aggregating these across all H segments, the new dataset of representatives $C = C_1 \cup C_2 \ldots \cup C_H$ is obtained as in input to the consensus clustering stage. Given a target number of clusters K, it is possible to simply employ a conventional algorithm like k-means or hierarchical clustering methods to produce the consensus clustering $\pi^* = \{c_1^*, c_2^*, \ldots, c_K^*\}$. This can be regarded as the universal or global partition to which all data instances in different segments are mapped. For an instance $x \in X_h$, it is assigned to the cluster $c^* \in \pi^*$ if most of centroids in the reference collection $\Pi^{h\prime}$ that it belongs to have been members of c^*. When there is a tie, distances between x and centroids of those candidate clusters are calculated, with the minimum distance determine the final outcome. This stage can be depicted by Fig. 3 that includes both segment-level and dataset-wide processes.

Specific to this study, the process of consensus clustering mentioned above is carried out using one of the benchmark techniques found in the literature: CSPA, HBGF and EAC-AL, respectively. The former two belong to the graph-based category while the other is a similarity-based one. To generate an ensemble of base clusterings, k-means is chosen for its efficiency, with those two strategies highlighted in Section 3.1 being reused here, i.e., $\alpha(C, M^*)$. Note that the ensemble size of is another user-defined variable M^*, which is normally set to the range between 30 to 50 in many works. With this, an ensemble at the dataset-wide level can be specified as $\Pi^C = \{\pi_1^C, \ldots, \pi_{M^*}^C\}$. Specific to both CSPA and EAC-AL, a $P \times P$ pairwise similarity matrix S is firstly summarised from Π^C, where $P = |C|$ or the total number of centroids considered at this level. The similarity between any two centroids $y_i, y_j \in C$ can be initially estimated with respect to each clustering in the ensemble Π^C, i.e., $sim_m(y_i, y_j), m = 1 \ldots M^*$. This is 1 if both centroids are assigned to the same cluster, 0 otherwise. Following that, the final similarity $sim(y_i, y_j)$ that is based on all the clusterings in Π^C can be calculated by the next equation.

$$sim(y_i, y_j) = \frac{\sum_{\forall m=1\ldots M^*} sim_m(y_i, y_j)}{M^*} \tag{5}$$

Having acquired S, EAC-AL considers this as a data matrix representing P instances by P different features, row- and column-wise, respectively. It makes use of the agglomerative hierarchical clustering technique with average linkage metric to create the clustering π^* of K global clusters. In contrast, CSPA represent this similarity matrix as a graph, in which nodes and links correspond to all $y_i \in C$ and non-zero similarity measures between any pairs of $y_i, y_j \in C$. Note that the weight of a link between nodes representing $y_i, y_j \in C$ is the similarity $sim(y_i, y_j)$ specified in S. Then, CSPA exploits a graph cut technique to divide this graph into K parts, each of which becomes one of the global clusters. Without having to construct S, HBGF directly translates the instance-cluster relations in Π^C to a bipartite graph, with two node types being included to represent instances and clusters. A link occurs between two nodes of different types and only when that specific instance is assigned to the cluster. Again, HBGF obtains a set of global clusters by dividing this graph into K parts with a graph cut technique. Please consult the review of [9] for further details of those ensemble clustering methods.

4. Performance evaluation

Having explained the proposed framework, this section presents its performance evaluation against several compared techniques, based on different experimental settings and published datasets.

4.1. Experimental design

At first, this assessment is conducted on two datasets, which have been investigated in the comparative study of [18]. These sets that will be referred to as Data1 and Data2 hereafter, represent users' rating averages collected through social media platforms of TripAdvisor.com and Google reviews, respectively. In particular, Data1 contains 980 user records ($N = 980$), each of which is represented by 10 feedback attributes ($D = 10$) and summarised from feedbacks on destinations in East Asia. For the other, Data2 is a collection of 5,456 user records and 24 types of attractions across Europe ($N = 5,456$ and $D = 24$). From the original work, optimal numbers of clusters (K) for these two datasets are 8 and 3, which will be exploited in the stage of consensus clustering. Prior the evaluation, all features are normalised to the standard scale of $[0, 1]$, and Euclidean is set to be the default distance measurement. In order to justify quality of the proposed model for big data analysis, the following procedure to simulate additional instances is applied to Data2. Given a dataset X, randomly select an instance $x \in X$ and find its k nearest neighbours, where k is also arbitrarily chosen from $\{1, 2, 3, 4, 5\}$. A new instance x' is a vector of averages between those identified neighbours along each of the D features. It will be added to X only when $x \notin X$, x' is dropped otherwise. This process is repeated until the size of X reaches the target, which is 10,000,000. The resulting set is named Data2-simulated for a future reference.

With these three sets, another preparation stage is required to obtain instance segments. The random sampling approach is adopted here with the investigated ratios being 10%, 5%, 2% and 1% of the original number of instances. In this work, it is also assumed that each segment should not be smaller than 100 to provide useful and diverse data partitions to the global stage. In other words, the lowest ratio for Data1 is 10% and 2% for Data2, while those four ratios are applicable to the simulated dataset. The new framework of SPP-EC has been designed to generalise to a rich collection of alternative techniques employed to create the global clustering. Specific to this paper, these include the classical algorithm of k-means (KM) and other consensus clustering techniques mentioned earlier. As such, the resulting models to be assessed are SPP-EC(KM), SPP-EC(CSPA), SPP-EC(HBGF) and SPP-EC(EAC-AL). Note that k-means with a random initialisation is the common method to produce ensemble of clusterings at both instance-segment and dataset-wide levels.

To achieve a rigorous evaluation, a number of compared techniques are considered to cover baseline models and those recent works found in the literature. To start with, 4 basic choices of KM, CSPA, HBGF and EAC-AL are examined on the whole dataset X to set the targets for their counterparts implemented on SPP-EC. In addition, two of those previous works indicated in Table 2 are explored here: Multi-Batches [16] and Validated Distributed Ensemble Clustering or V-DEC [15]. Again, k-means with the fixed number of clusters K is used to create segment-specific results for these methods. Similar to SPP-EC, the corresponding set of cluster centroids is collected for the final clustering, for which k-means has also been employed by Multi-Batches. For V-DEC, those centroids are aligned across all segments using the segment-pairwise validation. Without a definite structure of distributed segment allocation, the underlying validation is implemented with a random pair of segments firstly. Then, one of these is further aligned with a new segmented arbitrarily chosen from the rest. To acquire the consensus clustering, this is repeated until all segments have been covered.

Other experimental settings are summarised below.

- For new models of SPP-EC(KM), SPP-EC(CSPA), SPP-EC(HBGF) and SPP-EC(EAC-AL), the size of initial pool B is configured to 50 and the size of target reference set of multiple clusterings M is 10. For those three ensemble clustering deployed at the dataset-wide level, the ensemble size M^* is 30, which is similarly applied to the three baselines of CSPA, HBGF and EAC-AL.
- Following the original work on these datasets [18], internal validity indices of SC: Silhouette coefficient [9] and CH: Calinski-Harabasz index [44] are employed to justify and compare the goodness of clustering results. Note that SC in the range of $[-1, 1]$ with a high positive value indicating a good clustering with compact and well-separated clusters, while a higher value of CH similarly suggests the better a result is.
- Adding to those previously clarified, each specific experiment setting is repeated for 30 runs to generalise the results. The proposed SPP-EC framework is implemented in a standard workstation: Intel(R) Core(TM) i7-4170HQ CPU@2.50 GHz, RAM 128 GB, while those baseline models on the whole datasets are achieved using the cloud-based service with GPU processing units and RAM 512 GB. Given these different settings, a direct comparison of run times would not be appropriate, thus a complexity analysis being elaborated after the result report to demonstrate their scalability.

4.2. Experimental results

For the first part of this report, the results obtained from the proposed and related segment-based models are presented for original datasets of Data1 and Data2. In order to compare with baseline counterparts that analyse the whole set of instances X, the sampling ratio of 10% has been exploited as the default. Based on the SC metric, Fig. 4 illustrates method-specific measures that are averages across the two datasets and 30 trials for each setting. These results suggest that the proposed SPP-EC framework can usually sustain the clustering performance of both basic k-means and those ensemble clustering methods included in this evaluation. To be precise, SC scores of KM and SPP-EC(KM) are marginally different, i.e., 0.1457 and 0.1428, respectively. Likewise, a similar observation is obtained considering the pair of CSPA and SPP-EC(CSPA), with the measures of 0.1646 and 0.1627. Among segment-based techniques, SPP-EC(EAC-AL) provides the best result, with all variants of SPP-EC frequently outperform the two compared algorithms of Multi-Batches and V-DEC. It is also noteworthy that the use of those ensemble clustering methods at the consensus clustering stage appear to be more effective than the more efficient choice of k-means. Moreover, Fig. 5 provides a similar comparison using the CH metric. Those trends previously pointed out appear again in this figure, where SPP-EC(CSPA) possesses the highest averaged score among segment-based alternatives. Besides these figures, detailed results are presented in Table 3.

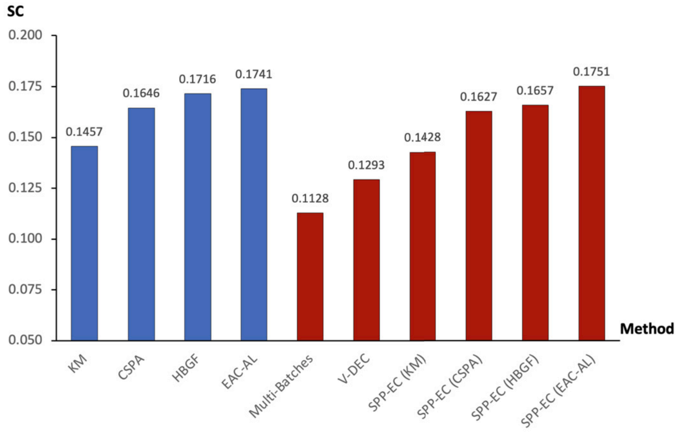

Fig. 4. Averaged SC scores obtained by investigated methods across two datasets and 30 trials, categorised into 2 groups of: baseline models that analyse a whole dataset (blue bars) and segment-based techniques using 10% sampling ratio (red bars).

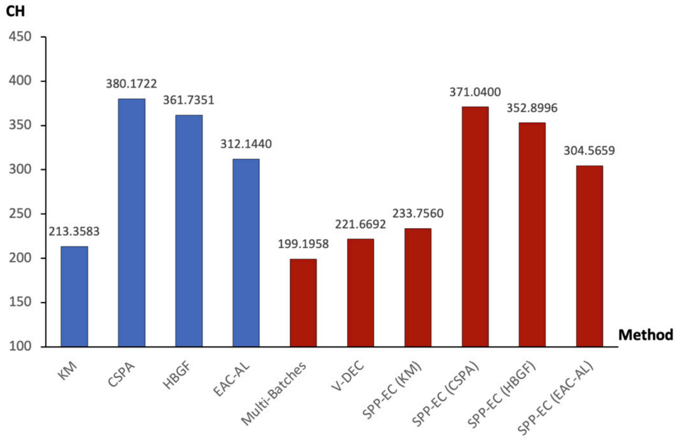

Fig. 5. Averaged CH scores obtained by investigated methods across two datasets and 30 trials, categorised into 2 groups of: baseline models that analyse a whole dataset (blue bars) and segment-based techniques using 10% sampling ratio (red bars).

Besides the improvement in clustering quality made by the proposed models, privacy of the actual instances are preserved through the exploitation of only segment-specific centroids in the global clustering phase. Despite the concept is in line with those of Multi-Batches and V-DEC, the number of centroids created per segment is undoubtedly larger than a fixed number employed by those compared techniques. Hence, it is interesting to see how many actual instances are highly similar to their nearest centroids, which have been generated and selected across segments. Ideally, the proportion of these cases in a dataset should be low to ensure the best possible data protection. Based on Data1 and Data2 with the 10% sampling rate, Fig. 6 presents the percentages of actual instances in each of these datasets that are similar to their closest centroids. There are two distance ratios illustrated therein, $< 2\%$ and $< 5\%$, which correspond to the distance between an actual instance to its centroid as a percentage of the maximum distance between any pair of instances in the dataset. For 'Single clustering' that is the case for Multi-Batches and V-DEC, only around 5.13% of actual instances in Data1 are similar to one of the centroids at the similarity level of $< 2\%$. It is slightly higher with SPP-EC at 6.04%. This trend is also observed in Data2 with the precise percentages of 3.51% and 4.66%, respectively. Bigger differences are reported at the level of $< 5\%$ for both datasets. With these results, the proposed framework is likely to be less effective for privacy preserving than Multi-Batches and V-DEC. However, all of them are far better than the use of actual instances at the global clustering stage, whose similar percentages are also provided in this figure for a reference.

As specified in the experimental design that the only sampling ratio of 10% is applicable to Data1, while two other smaller rates of 5% and 2% should be investigated on Data2. For this examination, Fig. 7 shows both averaged SC and CH scores from 30 trials of a specific setting of a segment-based model and the sampling rate. In general, all techniques perform worse as the size of segments becomes smaller, i.e., each contains around 545, 272 and 109 instances for the ratios mentioned above. Nonetheless, SPP-EC based methods remain better than those two compared methods that make use of a single clustering to represent each segment, thus suggesting the benefit of introducing multiple clusterings to this problem context. In fact, even at the 2% ratio, some models like SPP-EC(EAC-AL) remains more effective than KM (i.e., clustering the whole dataset), with SC and CH scores of 0.1864 and 444.1287.

In the second part of this section, the same set of experiments is carried out with Data2-simulated containing a big collection of 10,000,000 instances. This provides a testbed for the SPP-EC framework that has demonstrated a great potential given the results

Table 3
Dataset-specific results as averaged SC and CH scores obtained by investigated methods across 30 trials, with corresponding standard deviations being shown in parentheses. Note that the sampling ratio of 10% is employed to create segments for relevant methods.

Dataset/Method	SC scores	CH scores
Data1		
KM	0.1231 (0.0212)	121.6769 (30.1335)
CSPA	0.1377 (0.0098)	129.6375 (9.2198)
HBGF	0.1564 (0.0113)	143.6900 (5.3157)
EAC-AL	0.1405 (0.0105)	139.7827 (8.1691)
Multi-Batches	0.0953 (0.0283)	118.1778 (7.0318)
V-DEC	0.1042 (0.0251)	123.1983 (8.0121)
SPP-EC (KM)	0.1164 (0.0284)	127.3215 (10.0824)
SPP-EC (CSPA)	0.1326 (0.0116)	128.1716 (8.0601)
SPP-EC (HBGF)	0.1517 (0.0182)	145.7285 (4.9127)
SPP-EC (EAC-AL)	0.1489 (0.0159)	138.1154 (6.3109)
Data2		
KM	0.1683 (0.0141)	305.0397 (56.0359)
CSPA	0.1915 (0.0045)	630.7069 (30.0045)
HBGF	0.1867 (0.0134)	579.7803 (26.0021)
EAC-AL	0.2076 (0.0208)	484.5054 (34.4321)
Multi-Batches	0.1303 (0.0423)	280.2137 (29.7031)
V-DEC	0.1543 (0.0311)	320.1401 (22.3503)
SPP-EC (KM)	0.1691 (0.0218)	340.1905 (31.7012)
SPP-EC (CSPA)	0.1928 (0.0104)	613.9083 (26.1217)
SPP-EC (HBGF)	0.1797 (0.0197)	560.0706 (22.9073)
SPP-EC (EAC-AL)	0.2012 (0.0175)	471.0164 (25.1502)

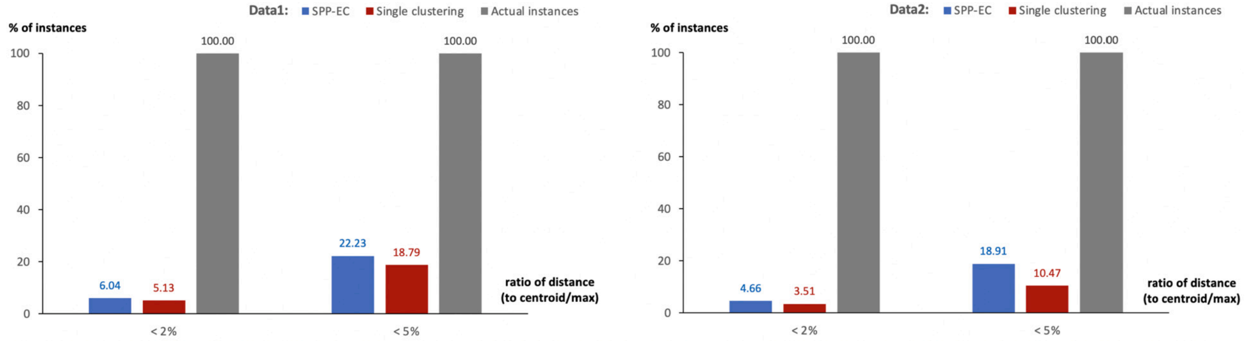

Fig. 6. Percentages of actual instances in Data1 (left) and Data2 (right) that are similar to their nearest centroids. Two distance ratios are included, < 2% and < 5%, which correspond to the distance between an actual instance to its centroid as a percentage of the maximum distance between any two instances in the dataset.

presented so far. Similar to the previous table, Table 4 shows the average SC and CH scores that each of the examined methods achieves on this dataset. With the sampling ratio of 10%, SPP-EC driven models are able to produce clusterings of comparable quality to their baselines that become extremely expensive to implement. In addition, they usually outperform the direct competitors, i.e., Multi-Batches and V-DEC, thus reinforcing the previous finding of an advantage brought about by multiple clusterings. Note that SPP-EC(EAC-AL) and SPP-EC(CSPA) manage to obtain high SC and CH scores of 0.1918 and 609.5232, which are significantly higher than those of KM (0.1411 and 288.1257). As such, it is worth applying SPP-EC to a dataset even when KM remains obtainable. An obvious question arises whether SPP-EC models can sustain good performance when segments get smaller. To answer that, Fig. 8 depicts average SC scores obtained by different segment-based techniques at decreasing ratios of instance sampling, from 10%, 5%, 2% and then 1%. Both Multi-Batches and V-DEC become gradually less effective as the ratio drops, while those SPP-EC based counterparts keep the quality of clustering results roughly at the same level. This observation is different from those results reported in Fig. 7, largely due to the size of a segment generated from Data2-simulated is much bigger than that of Data2. At the same sampling ratio of 2%, segments from the former contains 200,000 instances, with only 109 being members in the latter case. As a result, multiple clusterings created for each segment in Data2-simulated would be more informative and diverse, hence the goodness of the final clustering. A similar tendency has been revealed in Fig. 9 that presents additional assessment statistics based on the CH metric.

To consolidate the previous comparison based on average measures, the next part provides another comparative evaluation using the statistical test adopted the previous work [45]. The number of times that a method $b \in \Psi$ is 'significantly better' and 'significantly worse', at the 95% confidence level, than others $\forall c \in \Psi, c \neq b$ are examined, where Ψ denotes a set of seven methods (i.e., six segment-based models and KM). Let $\mu_b^t(d, ms)$ be the average of validation scores $t \in \{SC, CH\}$ obtained by the method $b \in \Psi$ across 30 trials, on the dataset $d \in$ {Data1, Data2, Data2-simulated} with the sampling ratio of $ms \in \{1\%, 2\%, 5\%, 10\%\}$. In the previous

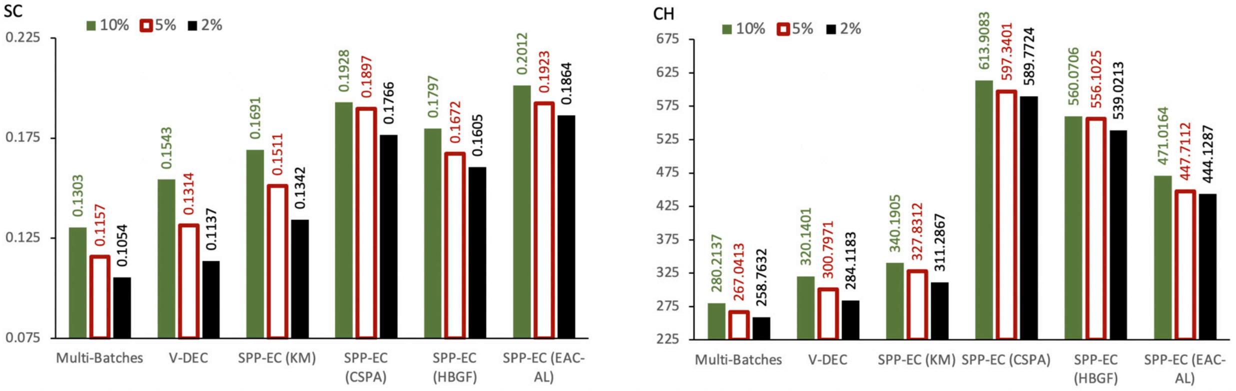

Fig. 7. Averaged SC and CH scores obtained by segment-based methods across 30 trials, using three sampling ratios of 10%, 5% and 2% on Data2.

Table 4
Averaged SC and CH scores obtained by investigated methods across 30 trials on Data2-simulated, with corresponding standard deviations being shown in parentheses. Top-3 scores of each metric are highlighted in **bold** font. Note that the sampling ratio of 10% is employed to create segments for relevant methods.

Method	SC scores	CH scores
KM	0.1411 (0.0382)	288.1257 (60.1204)
CSPA	0.1757 (0.0103)	**615.0978** (32.3200)
HBGF	0.1672 (0.0122)	552.7721 (28.0170)
EAC-AL	**0.1847** (0.0164)	481.6925 (31.6743)
Multi-Batches	0.1171 (0.0312)	272.1154 (27.1287)
V-DEC	0.1288 (0.0286)	309.3409 (23.8801)
SPP-EC (KM)	0.1502 (0.0201)	332.2218 (32.2110)
SPP-EC (CSPA)	**0.1774** (0.0131)	**609.5232** (27.3245)
SPP-EC (HBGF)	0.1626 (0.0202)	**558.9614** (24.1086)
SPP-EC (EAC-AL)	**0.1918** (0.0163)	468.0946 (27.7641)

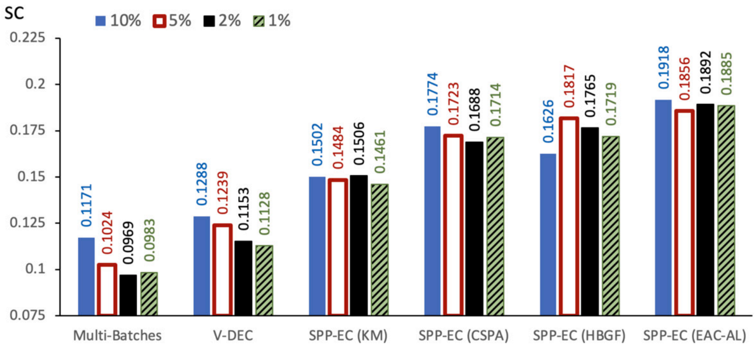

Fig. 8. Averaged SC scores obtained by segment-based methods across 30 trials, using three sampling ratios of 10%, 5% 2% and 1% on Data2-simulated.

parts, only the average values $\mu^t_{b_1}(d,ms)$ and $\mu^t_{b_2}(d,ms)$ are considered to compare the two methods $b_1, b_2 \in \Psi$ with respect to a specific experimental setting, i.e., $t = SC$, $d = Data1$ and $ms = 1\%$. Given the methods examined herein are non-deterministic, both average and corresponding standard deviation values should be taken into account to statistically justify the underlying comparison. This is accomplished using the concept of confidence interval, where the average $\mu^t_b(d,ms)$ can be presented by its lower and upper bounds, i.e., $[L(\mu^t_b(d,ms)), U(\mu^t_b(d,ms))]$. These can be defined by the following equations that are specific to the 95% confidence level. Note that $SD^t_b(d,ms)$ represents a standard deviation of $\mu^t_b(d,ms)$ summarised across 30 trials.

$$L(\mu^t_b(d,ms)) = \mu^t_b(d,ms) - 1.96\frac{SD^t_b(d,ms)}{\sqrt{30}} \tag{6}$$

$$U(\mu^t_b(d,ms)) = \mu^t_b(d,ms) + 1.96\frac{SD^t_b(d,ms)}{\sqrt{30}} \tag{7}$$

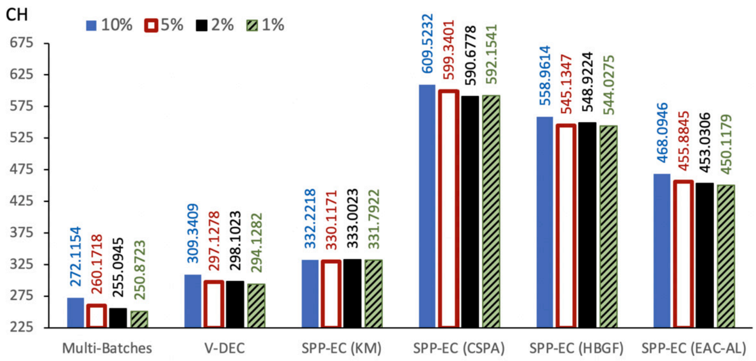

Fig. 9. Averaged CH scores obtained by segment-based methods across 30 trials, using three sampling ratios of 10%, 5% 2% and 1% on Data2-simulated.

With these and the interpretation of both validity indices employed in this work (the higher the better), it is possible to conclude that $\mu^t_{b_1}(d, ms)$ is higher than $\mu^t_{b_2}(d, ms)$, i.e., the method b_1 performs significantly better than b_2 w.r.t. the index t, on the dataset d and sampling ratio ms only when the following holds. Otherwise, the two methods are comparable in this specific experimental setting.

$$L(\mu^t_{b_1}(d, ms)) > U(\mu^t_{b_2}(d, ms)) \tag{8}$$

On the other hand, b_1 performs significantly worse than b_2 only when the next condition is true.

$$U(\mu^t_{b_1}(d, ms)) < L(\mu^t_{b_2}(d, ms)) \tag{9}$$

Formally, a function $better(b_1, b_2, d, ms, t)$ returns 1 if b_1 performs significantly better than b_2, and 0 otherwise. Another function $worse(b_1, b_2, d, ms, t) = 1$ only when b_1 performs significantly worse than the other, and 0 if this is not true. Then, the number of times $B(b_1)$ that a technique $b_1 \in \Psi$ is significantly better than others can be estimated as follows.

$$B(b_1) = \forall_{d,ms,t} \forall_{c \in \Psi, c \neq b_1} \quad better(b_1, c, d, ms, t) \tag{10}$$

Likewise, the frequency $W(b_1)$ that $b_1 \in \Psi$ is significantly worse than others can be defined by

$$W(b_1) = \forall_{d,ms,t} \forall_{c \in \Psi, c \neq b_1} \quad worse(b_1, c, d, ms, t) \tag{11}$$

Following that, Fig. 10 presents the results of this statistical test on those segment-based methods included in this paper, with KM being the baseline that projects a commonly expected level of quality without segments. On the left of this figure, the illustration shows method-specific frequencies of significantly better and worse that are summarised from both validity indices and the sampling rates between 2% to 10% (i.e., to cover more than one dataset). It is clear that the SPP-EC framework is able to support big data clustering, with its least effective model of SPP-EC(KM) performs better than the baseline KM and those two related techniques. This is also observed in the other graph on the right of Fig. 10, where better and worse counts are compiled from the lowest sampling rate of 1% (i.e., only Data2-simulated is included). In practice, SPP-EC(KM) is recommended to reduce the complexity of model implementation, whereas the other three SPP-EC variations are more appropriate if clustering quality is the major goal. This tradeoff will be further discussed in the next section, with respect to algorithmic variables and their relations to analytic performance. Despite the technique becoming more expensive, applying the concept of multiple clusterings to prepare segment-specific results should improve both Multi-Batches and V-DEC. However, it is expected to be more effective for the former where the number of clusters found in each of the segments is not fixed to K.

4.3. Discussion and parameter analysis

Having reported those results and comparison, this section continues with the discussion on parameter analysis of SPP-EC as well as its complexity. Based on the general intuition behind ensemble clustering [9], clustering results to be aggregated should be both diverse and accurate, with the latter referring to the case that each partition partly agree to others in the ensemble. To realise this within the proposed framework, each of them is expected to contain a sufficient number of instances to exhibit common patterns shared across segments (of the same or different sources). Also, disagreement between multiple clusterings produced from a small collection of instances would not be significant, hence only a marginal improvement gained from the consensus clustering can be foreseen. Given these, another experiment on Data2-simulated is conducted to evaluate the effect of segment sizes to the quality of final clustering obtained by SPP-EC based models. In particular, three lower order-of-magnitude ratios are introduced, i.e. 0.1%, 0.01% and 0.001%, where the resulting segment sizes are 10,000, 1,000 and 100, respectively. Fig. 11 shows the corresponding results specific to SPP-EC(KM) and SPP-EC(EAC-AL), where similar observations have been witnessed with the others. Specific to the former, the recommend ratio would be around 0.1% to 1%, i.e., each segment with about 10,000 instances from the dataset of more than a million. Of course, smaller segments would be more efficient with a tradeoff of being less accurate. On the other hand, SPP-EC(EAC-AL) remains competitive at the lower rate of 0.01% or a segment of 1,000 samples. However, this is achieved by paying

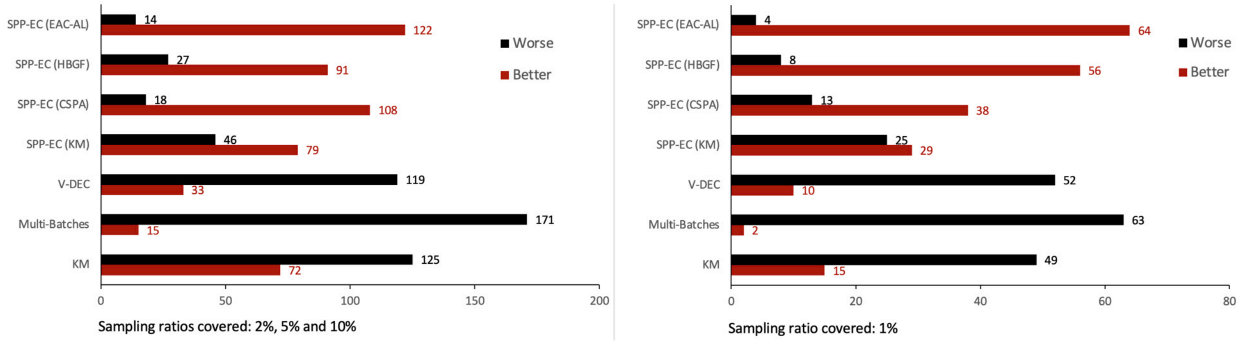

Fig. 10. Summarisation of better and worse performance statistics obtained by six segment-based models and KM as the baseline, categorised by two levels of sampling ratios: 2-10% (left, summarised from results on more than one datasets) and 1% (right, obtained from Data2-simulated only).

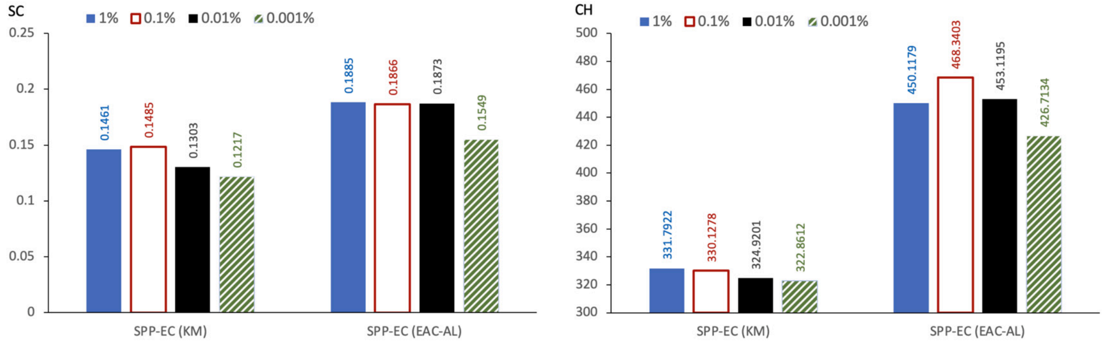

Fig. 11. Averaged SC and CH scores obtained by SPP-EC(KM) and SPP-EC(EAC-AL) across 30 trials on Data2-simulated, categorised by low sampling ratios of 1%, 0.1%, 0.01% and 0.001%.

an additional cost to ensemble clustering at the dataset-wide level. These provide an ideal setting that is sometimes not feasible for a real system, in which segments of uneven sizes hosted by different owners. Nonetheless, a median of segments under question can be logically matched to the guideline given here.

The previous investigation relates to two of algorithmic parameters of SPP-EC, H as the number of segments and N_H for the average number of instances in each segment. Note that H gets higher as N_H becomes small, given that $N_H = \frac{N}{H}$ and N denotes the number of all instances in a dataset. In general, one may prefer N_H to be minimal so that the stage of segment-specific clustering can be facilitated in a standard machine, with the complexity being $O(BN_H)$ for the generation stage and $O(MB)$ for the selection of representatives. Given the results in Fig. 11, the memory requirement posted by SPP-EC(EAC-AL) is much less demanding than the conventional EAC-AL, i.e., a minimal space to accommodate 1,000 instances against 10 millions at any time. Before moving on to discuss the complexity of ensemble clustering on segment-based centroids, both variables B and M mentioned above will be analysed next. Firstly, B denotes the size of segment-specific pool (the default value of 50 has been exploited for experiments reported thus far) from which its M representative clusterings are selected and fed to the next level of SPP-EC. To examine its relation to the quality of final clustering, the previous experiment on Data2-simulated with the sampling ratio of 1% and default values for other parameters ($M = 10$ and $M^* = 30$) is conducted again using a range of $B \in \{50, 100, 150\}$. Fig. 12 gives the corresponding results, which suggest that the quality of clusterings produced by SPP-EC(EAC-AL) and SPP-EC(KM) can be improved from the default setting if B is enlarged to 100. Slight increases are still seen when $B = 150$, but with further computational cost. Given these findings that are also witnessed with other SPP-EC models, the size of initial pool should not be too small since it might constrain the goodness of those M clusterings selected for the next analysis phase.

Another similar experiment has also been carried out to investigate M, using the same setting, the original B of 50 and $M \in \{10, 20, 30\}$. Fig. 13 reports averaged SC and CH scores acquired by SPP-EC(EAC-AL) and SPP-EC(KM), where the incline of both measures has been recorded as M grows from 10 to 30. To be concise, SPP-EC(EAC-AL) reaches the highest SC and CH rates of 0.2242 and 462.1137 when $M = 30$. These are substantially increases from the default setting ($M = 10$), with which the scores are 0.1885 and 450.1179, respectively. Henceforth, a bigger number of centroids put forward by each segment may better support the process of consensus clustering. Again, raising M beyond 30 may bring about an improved score, but with addition demand on computing resources. At this point, it is appropriate to emphasise the complexity of selecting M clustering from a pool of B candidates. With a simple greedy forward search, it iteratively adds one from this pool to the target set until reaching the size of M. This is simple compared to other swarm intelligence counterparts with the complexity around $O(MB)$, for M rounds of B comparisons (i.e., adding each of the candidates in the pool to the target collection, then identify the best choice). Despite the chance of getting a sub-optimal solution, it can be more efficient than an optimisation method like artificial bee colony or ABC [46], which repeatedly refine the set

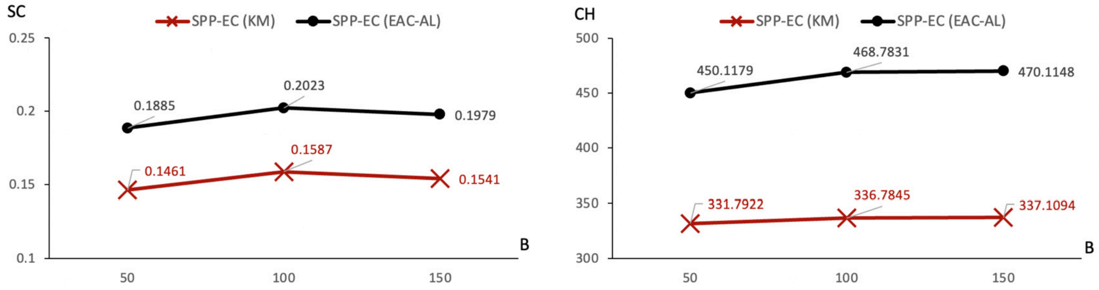

Fig. 12. Averaged SC and CH scores obtained by SPP-EC(KM) and SPP-EC(EAC-AL) across 30 trials on Data2-simulated (sampling rate of 1%, $M = 10$ and $M^* = 30$), categorised by different $B \in \{50, 100, 150\}$.

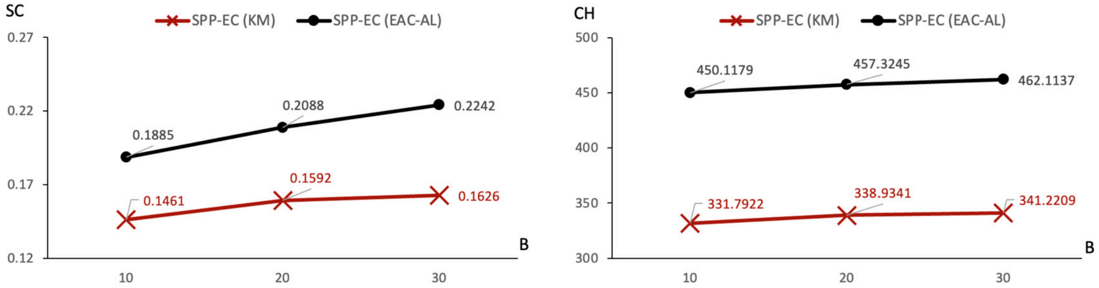

Fig. 13. Averaged SC and CH scores obtained by SPP-EC(KM) and SPP-EC(EAC-AL) across 30 trials on Data2-simulated (sampling rate of 1%, $B = 50$ and $M^* = 30$), categorised by different $M \in \{10, 20, 30\}$.

possible solutions of food sources (F) up to the maximum number of iteration (I). In each of these I iterations, each of the all food sources has gone through three processing stages with different bee types, thus resulting the complexity of around $O(3FI)$. Assumed that F is the same as M, it is possible to suggest that the greedy search can be less complex, with I in the order of hundreds is common in the literature.

After disclosing an optimal configuration of parameters at the level of segment-based analysis, the same experiment is executed once more on Data2-simulated (sampling rate of 1%) using $B = 100$ and $M = 30$. Then, Fig. 14 compares the results of this fine-tuned setting to the best scores provided in the last figure, i.e., SPP-EC(KM) and SPP-EC(EAC-AL) with $B = 50$ and $M = 30$. This shows enhanced scores for both models using a larger pool of size 100. For instance, SPP-EC(KM) achieves the highest SC score of 0.1734, which is used to be 0.1626 using a smaller pool of $B = 50$. This trend has been similarly obtained for other SPP-EC driven methods, but not included here due to the space constraint. Assuming that accurate and diverse M clusterings are made available for each segment, the corresponding set of centroids is combined with those obtained from other segments to form the dataset of representatives. Its size is approximated by the multiplication between M, H and β, with the last being an average number of clusters (i.e., a constant between $[2, 50]$). Provided this, the complexity of SPP-EC(KM) for the consensus clustering stage is simply $O(MH\beta)$, which is greatly reduced from $O(N)$ with a big dataset. It is more expensive with SPP-EC(CSPA) and SPP-EC(EAC-AL) that construct a pairwise similarity matrix among those centroids, from which the final clustering is determined. As a result, their complexity can be expressed as $O(M^2H^2\beta^2)$ that is far better than $O(N^2)$ of the conventional baselines. Actually, it seems impractical to manage this sort of data matrix as N increases beyond a million. SPP-EC(HBGF) provides a less resource-demanding alternative to SPP-EC(CSPA) and SPP-EC(EAC-AL), with its core data matrix representing a binary relation between centroids and clusters they are assigned to. Therefore, its complexity is $O(MH\beta^2)$, where the number of clusters generated at this level can also be modelled by β (the same set of generation strategies is employed at both segment and this higher levels). Based on the results and issues discussed in this section, the new framework of SPP-EC has proven effective to create clustering results of quality comparable to baseline processes on the whole big dataset. This can be useful for case studies with social media data, including the tourism subject examined in this work. Also, a rich collection of working models are introduced with tradeoffs between accuracy and complexity. This can be tailored to match requirements and supporting resources, which are different from one to another task.

4.4. A case study of noisy data clustering: robustness vs. possible information hiding

In addition to the report of experimental results and discussion presented thus far, this final part embarks on another issue of applying the proposed framework to a real-world problem. The datasets investigated in the previous sections are assumed to be perfect without any flaws to possibly deteriorate the quality of ensemble clustering. However, real data, especially those involving human responses, often exhibits problems of missing values and various input errors. Hence, this section aims to investigate the robustness of SPP-EC models and related methods on noisy data. Variants of Data1, Data2 and Data2-simulated are created by randomly injecting missing values into a matrix X at the γ percentage of $N \times D$ data entries. For instance, with $N = 980$ and $D = 10$

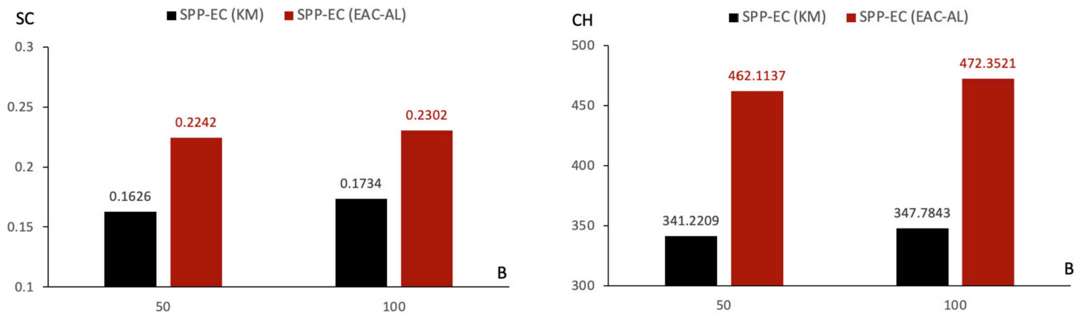

Fig. 14. Averaged SC and CH scores obtained by SPP-EC(KM) and SPP-EC(EAC-AL) across 30 trials on Data2-simulated (sampling rate of 1%, $M = 30$ and $M^* = 30$), categorised by different $B \in \{50, 100\}$.

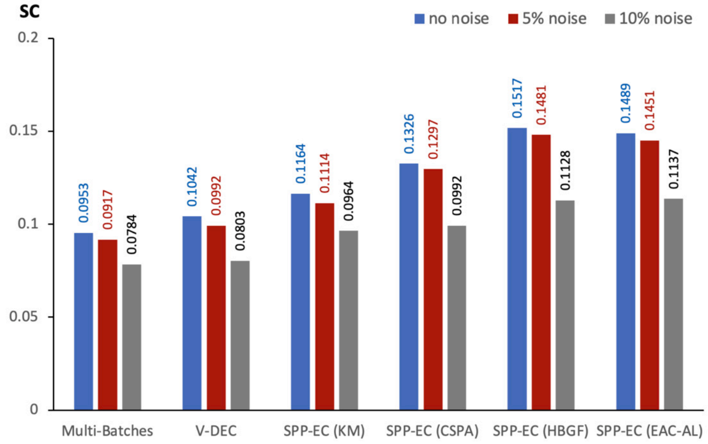

Fig. 15. Averaged SC scores obtained by proposed and other ensemble clustering methods across 30 trials on 10 noisy versions of Data1 (sampling rate of 10%), categorised by different $\gamma \in \{5, 10\}$.

in Data1, $\gamma = 1\%$ will be equivalent to 98 entries in X. Having known this amount, those entries are arbitrarily chosen to achieve the desired collection. Then, their values are considered missing and filled with 0, which is one the basic approaches to handle this problem [47]. The previous experiment with a default setting is repeated here for 30 trials on each of the 10 noisy data versions. This applies to those three datasets with $\gamma \in \{5, 10\}$ and the sampling rate of 10%.

Specific to Data1, Fig. 15 reports average SC measures obtained by different models and compare these to the scores previously presented for a 'no noise' case. At the level of $\gamma = 5\%$, almost all the ensemble clustering methods included in this study are able to sustain good performance with small drops in SC. With SPP-EC(EAC-AL), the score decreases from 0.1489 without noise to 0.1451. Likewise, this change is between and 0.1042 and 0.0992 for V-DEC. However, when γ gets to 10%, these methods become less accurate where the scores of 0.1137 and 0.0803 are obtained by the two models mentioned above. Note that similar results have also been observed with CH index. In addition, Fig. 16 similarly illustrates those scores recorded for Data2. Looking at EC(EAC-AL) again, its performance only declines from 0.2012 to 0.1987 as $\gamma = 5\%$ but further drops to 0.1402 if noisy entries occupy up to 10% of the data matrix. This tendency has been witnessed with the biggest of three datasets, i.e., Data2-simulated, which is shown in Fig. 17. In particular, the best and the worst scores of EC(EAC-AL) are 0.1918 and 0.1327, respectively.

It is fair to say that the proposed framework is robust against small amount of noisy feature values, at least up to 5% of entries in a given data matrix. Let us turn this investigation to another perspective of information hiding that is implemented by one of the related works [13] and identified in the recent survey on data privacy in machine learning systems [48]. It might be possible to provide the new framework with a noisy variation of the dataset under investigation, thus allowing the true signatures of some instances to be partly hidden. This helps to improve its capacity to preserve data privacy in addition to the use of centroids at the global clustering stage. However, as pointed out above, there is a definite tradeoff between the level of noise injected and the quality of final data partition obtained. Given this quest, Fig. 18 compares the SC scores averaged across Data1, Data2 and Data2-simulated using the sampling rate of 10% like before and $\gamma \in \{5, 6, 7, 8, 9, 10\}$. This focuses on evaluating the new models against the basic result achieved by k-means with the whole noise-free datasets, denoted in this figure as 'Baseline (KM)'. Most of SPP-EC techniques, remain comparable to Baseline (KM) up to $\gamma = 8\%$, while SPP-EC(KM) becomes sub-optimal as $\gamma > 5\%$. Based on these, it is recommended to inject noises around 5-8% to the original data prior an application of the SPP-EC approach, with the most reliable results achieved by SPP-EC(EAC-AL), SPP-EC(HBGF) and SPP-EC(CSPA).

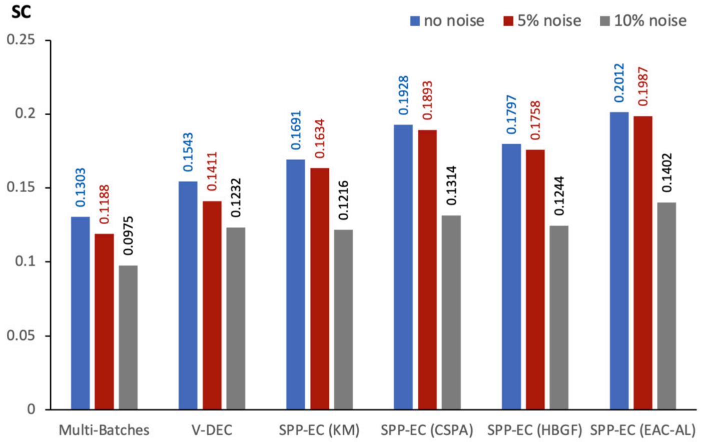

Fig. 16. Averaged SC scores obtained by proposed and other ensemble clustering methods across 30 trials on 10 noisy versions of Data2 (sampling rate of 10%), categorised by different $\gamma \in \{5, 10\}$.

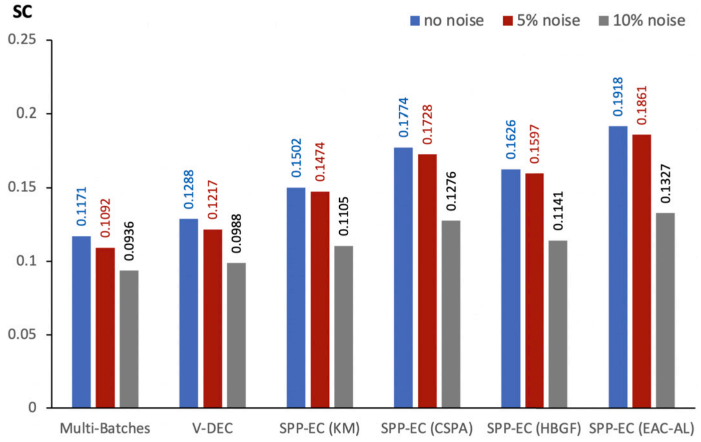

Fig. 17. Averaged SC scores obtained by proposed and other ensemble clustering methods across 30 trials on 10 noisy versions of Data2-simulated (sampling rate of 10%), categorised by different $\gamma \in \{5, 10\}$.

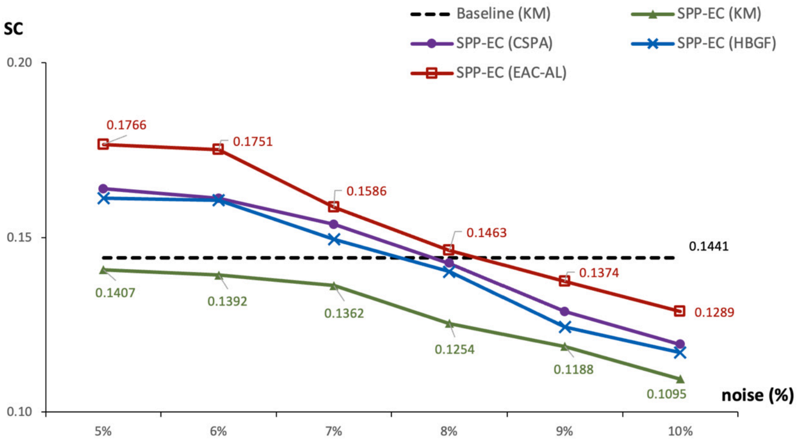

Fig. 18. SC scores obtained by the proposed methods as averages across 30 trials on 10 noisy versions of three datasets (sampling rate of 10%), categorised by different $\gamma \in \{5, 6, 7, 8, 9, 10\}$.

5. Conclusion

This paper has introduced a new framework of scalable and privacy-preserving clustering, with the focus on analysing big social media data in the tourism domain. The resulting approach called SPP-EC attempts to fill a research gap in the current literature, where only a few studies have been reported to leveraging ensemble clustering (EC) to obtain those desired properties. Mostly, previous works mitigate the potential of EC to a big data collection by either reducing dimensions or implementing original methods on a distributed computing platform. Recently, segment-based techniques have been put forward as a way to manage and analyse a big data volume, regardless of a physical setting (i.e., implemented in a single or multiple machines). However, only single partition with a unified number of clusters is drawn from each segment to create the consensus clustering, which will then be mapped to all instances across segments. The proposed SPP-EC approach seeks to boost the diversity of these segment-based inputs by applying the concept of multiple clusterings to select representatives from a pool of candidates (each with a possible different number of clusters). Having achieved those, different methods are exploited to build the final result, including a classical k-means and benchmark ensemble clustering algorithms, e.g., CSPA, HBGF and EAC-AL. In fact, the framework is generalised to accommodate other advanced EC techniques, which are not feasible to apply directly to a very large dataset.

According to experimental results systematically run on published tourism datasets with different parameter settings, the SPP-EC based models generally outperform baselines and relevant state-of-the-art techniques. Despite this initial success, it is important to evaluate them on other big sets of data, including those in tourism and other fields where the need to obtain a cluster analysis exists [49]. Other issues worth further investigation include a possible use of different swarm intelligence algorithms [19] to select representative clusterings from a pre-generated pool. This may improve the quality of segment-specific centroids, which can be sub-optimal given the current greedy search. In addition, the goodness of those candidates in a clustering pool might be enhanced by exploring noise-induced and feature-selection based generation of multiple clusterings [50].

CRediT authorship contribution statement

Natthakan Iam-On: Writing – original draft, Validation, Software, Methodology, Investigation, Data curation, Conceptualization. **Tossapon Boongoen:** Writing – original draft, Validation, Methodology, Investigation, Conceptualization. **Nitin Naik:** Writing – review & editing, Validation, Conceptualization. **Longzhi Yang:** Writing – review & editing, Validation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

This research work is based on the collaboration between Aberystwyth, Aston and Northumbria Universities. It is also partly supported by UK FCDO: Research and Innovation for Development in ASEAN (RIDA) programme 2023-24 (Grant no. RSA-03160), with the proposed method reducing the complexity of a clustering-based undersampling process (prior training a burn scar detector).

References

[1] T.K. Balaji, C.S.R. Annavarapu, A. Bablani, Machine learning algorithms for social media analysis: a survey, Comput. Sci. Rev. 40 (100395) (2021) 1–32.

[2] Z.M. Obeidat, R.S. AlGharabat, A.A. Alalwan, S.H. Xiao, Y.K. Dwivedi, N.P. Rana, Narcissism, interactivity, community, and online revenge behavior: the moderating role of social presence among Jordanian consumers, Comput. Hum. Behav. 104 (106170) (2020) 1–15.

[3] Z. Xiang, Q. Du, Y. Ma, W. Fan, A comparative analysis of major online review platforms: implications for social media analytics in hospitality and tourism, Tour. Manag. 58 (2017) 51–65.

[4] E.V. Vishnevskaya, The impact of digital technologies on the development of the tourism market, Res. Result Bus. Serv. Technol. 5 (2019) 12–24.

[5] S. Renjith, C. Anjali, A personalized mobile travel recommender system using hybrid algorithm, in: Proceedings of IEEE International Conference on Computational Systems and Communications, 2014, pp. 12–17.

[6] S. Jiang, X. Qian, T. Mei, Y. Fu, Personalized travel sequence recommendation on multi-source big social media, IEEE Trans. Big Data 2 (1) (2016) 43–56.

[7] J. Kim, M. Hastak, Social network analysis: characteristics of online social networks after a disaster, Int. J. Inf. Manag. 38 (1) (2018) 86–96.

[8] R. Mendes, J.P. Vilela, Privacy-preserving data mining: methods, metrics and applications, IEEE Access 5 (2017) 10562–10582.

[9] T. Boongoen, N. Iam-On, Cluster ensembles: a survey of approaches with recent extensions and applications, Comput. Sci. Rev. 28 (2018) 1–25.

[10] X. Peng, H. Tang, L. Zhang, Z. Yi, S. Xiao, A unified framework for representation-based subspace clustering of out-of-sample and large-scale data, IEEE Trans. Neural Netw. Learn. Syst. 27 (12) (2016) 2499–2512.

[11] C. Boutsidis, A. Zouzias, P. Drineas, Random projections for k-means clustering, in: Proceedings of International Conference on Advances in Neural Information Processing Systems, 2010, pp. 298–306.

[12] C. Zhang, H. Fu, S. Liu, G. Liu, X. Cao, Low-rank tensor constrained multiview subspace clustering, in: Proceedings of International Conference on Computer Vision, 2015, pp. 1582–1590.

[13] Y. Zhang, N. Liu, S. Wang, A differential privacy protecting k-means clustering algorithm based on contour coefficients, PLoS ONE 13 (11) (2018) e0206832.

[14] Y. Zhao, S.K. Tarus, L.T. Yang, J. Sun, Y. Ge, J. Wang, Privacy-preserving clustering for big data in cyber-physical-social systems: survey and perspectives, Inf. Sci. 515 (2020) 132–155.

[15] A. Rosato, R. Altilio, M. Panella, A decentralized algorithm for distributed ensemble clustering, Inf. Sci. 578 (2021) 417–434.

[16] R.M. Alguliyev, R.M. Aliguliyev, L.V. Sukhostat, Efficient algorithm for big data clustering on single machine, CAAI Trans. Intell. Technol. 5 (1) (2020) 9–14.

[17] P. Panwong, T. Boongoen, N. Iam-On, Improving consensus clustering with noise-induced ensemble generation, Expert Syst. Appl. 146 (2020) 113–138.

[18] S. Renjith, A. Sreekumar, M. Jathavedan, Evaluation of partitioning clustering algorithms for processing social media data in tourism domain, in: Proceedings of IEEE International Conference on Recent Advances in Intelligent Computational Systems, 2018, pp. 127–131.

[19] J. Liu, S. Anavatti, M. Garratt, K.C. Tan, H. Abbass, A survey, taxonomy and progress evaluation of three decades of swarm optimization, Artif. Intell. Rev. 55 (5) (2022) 3607–3725.

[20] M. Mariani, R. Baggio, Big data and analytics in hospitality and tourism: a systematic literature review, Int. J. Contemp. Hosp. Manag. 34 (1) (2022) 231–278.

[21] A. Ahani, M. Nilashi, O. Ibrahim, L. Sanzogni, S. Weaven, Market segmentation and travel choice prediction in spa hotels through tripadvisor's online reviews, Int. J. Contemp. Hosp. Manag. 80 (2019) 52–77.

[22] L. Serrano, A. Ariza-Montes, M. Nader, A. Sianes, R. Law, Exploring preferences and sustainable attitudes of airbnb green users in the review comments and ratings: a text mining approach, J. Sustain. Tour. 29 (7) (2020) 1134–1152.

[23] L. Esmaeili, S. Mardani, S.A.H. Golpayegani, Z.Z. Madar, A novel tourism recommender system in the context of social commerce, Expert Syst. Appl. 149 (113301) (2020) 1–11.

[24] X. G-r, C. Lin, Q. Yang, W. Xi, Z. H-J, Y. Yu, Z. Chen, Scalable collaborative filtering using cluster-based smoothing, in: Proceedings of Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2005, pp. 114–121.

[25] C. Vicient, D. Sanchez, A. Moreno, An automatic approach for ontology-based feature extraction from heterogeneous textual resources, Eng. Appl. Artif. Intell. 26 (3) (2013) 1093–1106.

[26] L. Castillo, E. Armengol, E. Onaindıa, L. Sebastia, J. Gonzalez-Boticario, A. Rodrıguez, S. Fernandez, J.D. Arias, D. Borrajo, SAMAP: an user-oriented adaptive system for planning tourist visits, Expert Syst. Appl. 34 (2) (2008) 1318–1332.

[27] J. Lucas, N. Luz, M. Moreno, R. Anacleto, A. Figueiredo, C. Martins, A hybrid recommendation approach for a tourism system, Expert Syst. Appl. 40 (9) (2013) 3532–3550.

[28] M.A. Mahdi, K.M. Hosny, I. Elhenawy, Scalable clustering algorithms for big data: a review, IEEE Access 9 (2021) 80015–80027.

[29] X.Z. Fern, C.E. Brodley, Random projection for high dimensional data clustering: a cluster ensemble approach, in: Proceedings of International Conference on Machine Learning, 2003, pp. 186–193.

[30] X. Peng, J. Feng, S. Xiao, W.Y. Yau, J.T. Zhou, S. Yang, Structured AutoEncoders for subspace clustering, IEEE Trans. Image Process. 27 (10) (2018) 5076–5086.

[31] F. Nie, J. Li, X. Li, Parameter-free auto-weighted multiple graph learning: a framework for multiview clustering and semi-supervised classification, in: Proceedings of International Joint Conference on Artificial Intelligence, 2016, pp. 1881–1887.

[32] Y. Zhao, L. Yang, J. Sun, Privacy preserving tensor-based multiple clusterings on cloud for industrial iot, IEEE Trans. Ind. Inform. 15 (4) (2019) 2372–2381.

[33] Y. Wang, S.K. Saraswat, I.E. Komari, Big data analysis using a parallel ensemble clustering architecture and an unsupervised feature selection approach, J. King Saud Univ, Comput. Inf. Sci. 35 (2023) 270–282.

[34] S. Maitrey, C.K. Jha, An integrated approach for cure clustering using map-reduce technique, in: Proceedings of, Elsevier, 2013, pp. 563–571.

[35] Q. Zhang, Z. Chen, A weighted kernel possibilistic c-means algorithm based on cloud computing for clustering big data, Int. J. Commun. Syst. 27 (9) (2014) 1378–1391.

[36] Y.I. Kim, Y.K. Ji, S. Park, Big text data clustering using class labels and semantic feature based on hadoop of cloud computing, Int. J. Softw. Eng. Appl. 8 (4) (2014) 1–10.

[37] D. Mittal, D. Kaur, A. Aggarwal, Secure data mining in cloud using homomorphic encryption, in: Proceedings of IEEE International Conference on Cloud Computing in Emerging Markets, 2014, pp. 1–7.

[38] A. Alabdulatif, I. Khalil, M. Reynolds, H. Kumarage, X. Yi, Privacy-preserving data clustering in cloud computing based on fully homomorphic encryption, in: Proceedings of Pacific-Asia Conference on Information Systems, 2017, pp. 1–13.

[39] H. Liu, T. Liu, J. Wu, D. Tao, Y. Fu, Spectral ensemble clustering, in: Proceedings of IEEE International Conference on KDD, 2015, pp. 715–724.

[40] S. Salloum, J.Z. Huang, Y.L. He, Random sample partition: a distributed data model for big data analysis, IEEE Trans. Ind. Inform. 15 (11) (2019) 5846–5854.

[41] X. Du, Y. He, J.Z. Huang, Random sample partition-based clustering ensemble algorithm for big data, in: Proceedings of IEEE International Conference on Big Data, 2021, pp. 5885–5887.

[42] N. Iam-On, Clustering data with the presence of attribute noise: a study of noise completely at random and ensemble of multiple k-means clusterings, Int. J. Mach. Learn. Cybern. 11 (3) (2020) 491–509.

[43] N. Iam-On, T. Boongoen, Diversity-driven generation of link-based cluster ensemble and application to data classification, Expert Syst. Appl. 42 (21) (2015) 8259–8273.

[44] P. Keerin, W. Kurutach, T. Boongoen, A cluster-directed framework for neighbour based imputation of missing value in microarray data, Int. J. Data Min. Bioinform. 15 (2016) 165–193.

[45] P. Keerin, T. Boongoen, Estimation of missing values in astronomical survey data: an improved local approach using cluster directed neighbor selection, Inf. Process. Manag. 59 (2) (2022) 102881.

[46] S. Kessentini, I. Naas, Absolute versus stochastic stability of the artificial bee colony in synchronous and sequential modes, Nat. Comput. 20 (2021) 443–470.

[47] P. Keerin, N. Iam-On, J.J. Liu, T. Boongoen, Q. Shen, Summarising multiple clustering-centric estimates with OWA operators for improved KNN imputation on microarray data, Fuzzy Sets Syst. 473 (2023) 108718.

[48] S.Z.E. Mestari, G. Lenzini, H. Demirci, Preserving data privacy in machine learning systems, Comput. Secur. 137 (2024) 103605.

[49] T. Sangpetch, T. Boongoen, N. Iam-On, Profiling astronomical objects using unsupervised learning approach, Comput. Mater. Continua 74 (1) (2023) 1641–1655.

[50] C. Pimsarn, T. Boongoen, N. Iam-On, N. Naik, L. Yang, Strengthening intrusion detection system for adversarial attacks: improved handling of imbalance classification problem, Complex Intell. Syst. 8 (2022) 4863–4880.