Unsupervised and Semi-Supervised Learning

Series Editor: M. Emre Celebi

Bader Alyoubi
Chiheb-Eddine Ben Ncir
Ibraheem Alharbi
Anis Jarboui *Editors*

Machine Learning and Data Analytics for Solving Business Problems

Methods, Applications, and Case Studies

Springer

Unsupervised and Semi-Supervised Learning

Series Editor
M. Emre Celebi, Computer Science Department, Conway, AR, USA

Springer's Unsupervised and Semi-Supervised Learning book series covers the latest theoretical and practical developments in unsupervised and semi-supervised learning. Titles – including monographs, contributed works, professional books, and textbooks – tackle various issues surrounding the proliferation of massive amounts of unlabeled data in many application domains and how unsupervised learning algorithms can automatically discover interesting and useful patterns in such data. The books discuss how these algorithms have found numerous applications including pattern recognition, market basket analysis, web mining, social network analysis, information retrieval, recommender systems, market research, intrusion detection, and fraud detection. Books also discuss semi-supervised algorithms, which can make use of both labeled and unlabeled data and can be useful in application domains where unlabeled data is abundant, yet it is possible to obtain a small amount of labeled data.

Topics of interest in include:

- Unsupervised/Semi-Supervised Discretization
- Unsupervised/Semi-Supervised Feature Extraction
- Unsupervised/Semi-Supervised Feature Selection
- Association Rule Learning
- Semi-Supervised Classification
- Semi-Supervised Regression
- Unsupervised/Semi-Supervised Clustering
- Unsupervised/Semi-Supervised Anomaly/Novelty/Outlier Detection
- Evaluation of Unsupervised/Semi-Supervised Learning Algorithms
- Applications of Unsupervised/Semi-Supervised Learning

While the series focuses on unsupervised and semi-supervised learning, outstanding contributions in the field of supervised learning will also be considered. The intended audience includes students, researchers, and practitioners.

** Indexing: The books of this series indexed in zbMATH **

Bader Alyoubi • Chiheb-Eddine Ben Ncir •
Ibraheem Alharbi • Anis Jarboui
Editors

Machine Learning and Data Analytics for Solving Business Problems

Methods, Applications, and Case Studies

Springer

Editors

Bader Alyoubi
University of Jeddah
Jeddah, Saudi Arabia

Chiheb-Eddine Ben Ncir
University of Jeddah
Jeddah, Saudi Arabia

Ibraheem Alharbi
University of Jeddah
Jeddah, Saudi Arabia

Anis Jarboui
University of Sfax
Sfax, Tunisia

ISSN 2522-848X ISSN 2522-8498 (electronic)
Unsupervised and Semi-Supervised Learning
ISBN 978-3-031-18482-6 ISBN 978-3-031-18483-3 (eBook)
https://doi.org/10.1007/978-3-031-18483-3

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

This volume highlights the state of the art and challenges related to the design and application of machine learning and data-analytics methods to solve decision-making problems faced by a business. Innovative and creative computational solutions and data analytics methods have been designed to support managers in solving decision problems. The volume provides a systematic understanding of the scope in-depth and rapidly builds an overview of these recent methods and applications.

The volume opens with a Chap. 1 entitled "Predicting Salaries with Random-Forest Regression". In this chapter, Frank Eichinger and Moritz Mayer deal with the issue of estimating employees' salary in order to help big firms to propose market competitive salary for their current and prospective employees. The authors investigate the use of machine learning techniques, particularly regression trees and random forest regression, to estimate market competitive salary for each profession. They explore the use of ensemble random forest regression for the improvement of salary prediction. The study is performed on a large real dataset containing more than three million employees and more than 300 professions. Obtained results have shown the effectiveness of ensemble random forest regression to achieve high-quality salary predictions.

In Chap. 2 entitled "Data-Driven Analysis of Microfinance and Social Loans Before and During the COVID-19 Pandemic Using Exploratory Analysis and Decision Tree Classifiers," Chiheb-Eddine Ben Ncir et al. mainly investigate the changes in demographic, social, and economic characteristics of the beneficiaries using both a bivariate exploratory analysis and a decision tree classifier. A machine learning decision tree classification model has been built, for individual and business microcredits, to easily visualize the main beneficiary characteristics of each credit category before and during the COVID-19 pandemic. The built decision trees allowed to deeply understand the main characteristics of each credit category in order to help managers to design more suitable and fitted microfinance products.

The application of machine learning and data analytics to the banking domain is also studied in Chap. 3 entitled "Identification of Credit Risks Using Cluster Analysis and Behavioral Scoring During the COVID-19 Pandemic." Waad Bouaguel and

Taghrid Al Silimani investigate the use of behavioral scoring to identify risk levels for defaulting credit accounts. A new approach is proposed in order to identify different risk levels for existing credit accounts and to find the principal factors affecting credit risks before and during the COVID-19 pandemic. The proposed approach is based on k-means cluster analysis and supervised learning with a decision tree.

In Chap. 4, "Improving Sales Prediction for Point-of-Sale Retail Using Machine Learning and Clustering," Chibuzor Udokwu et al. investigate the use of data analytics to improve the performance of point of sales retailing by better predicting product sales and optimizing product availability. The authors try to identify the main factors for clustering retail stores and examine model combinations of clustering and prediction algorithms that improve sales forecasts in retail stores. Obtained results show some statistically selected factors for organizing stores and present best performing algorithms for predicting product sales.

The problem of customer segmentation is addressed in Chap. 5 entitled "Telecom Customer Segmentation Using Deep-Embedded Clustering Algorithm." Jothi Ramasamy and K. Muthukumaran investigate the improvement of a telecom customer segmentation process by using deep learning. The authors propose a new telecom customer segmentation process using a deep-embedded clustering algorithm. Obtained results on Kaggle's telecom customer churn dataset have shown that deep-embedded clustering can attain better segmentation results compared to conventional clustering algorithms such as partitional and hierarchical clustering.

The deep learning technique is also exploited by Sonia Ouni et al. to improve catalog management in retail E-commerce websites. The authors propose a no-reference image semantic quality assessment approach using a conventional neural network algorithm described in Chap. 6 entitled "Semantic Image Quality Assessment Using Conventional Neural Network for E-Commerce Catalogue Management." The authors integrate a deep learning process for an automatic and semantic image quality assessment. This deep learning process aims to replace the subjective, complex, and time-consuming image quality assessment process. Performed experiments have shown the effectiveness of their proposed approach in automatically assessing the quality of images and in improving E-commerce catalog management.

In Chap. 7 entitled "Contextual Recommender Systems in Business from Models to Experiments," Khedija Arour and Rim Dridi give an overview of recent Contextual Aware Recommender Systems (CARS). These systems incorporate contexts (e.g., time, location, occasion, etc.) into consideration to suggest items that better fit the user's needs. The authors introduce the state of the art in CARS while reviewing associated types, challenges, limitations, and business adoptions. The authors also give measures, metrics, and benchmark datasets that can be used for the evaluation of CARS. Furthermore, the authors give an empirical comparison of CARS and baseline recommender system methods in two benchmark datasets.

The next two chapters give recent advances in the improvement of unsupervised learning methods and their application to real-life problems. In Chap. 8 entitled "An Overview of Multi-View Methods for Text Clustering," Maha Fraj et al. give an overview of recent methods for organizing multi-view textual data. They propose a new categorization model of the existing clustering methods based on the main properties pointed out in the multi-view textual data. The authors also give an interesting empirical evaluation of multi-view text clustering methods on real-world textual datasets. The next Chap. 9 entitled "Real-Time K-Prototypes for Incremental Attribute Learning Using Feature Selection" deals with the issue of online clustering of mixed data streams. Siwar Gorrab et al. propose an incremental attribute and object learning clustering method based on k-prototypes algorithm and using the feature selection technique in a streaming data environment. Experiments conducted on various real mixed data streams have shown the effectiveness of their proposed approach in terms of quality and time consumption.

The last Chap. 10 "Applications of Industry 4.0 on Saudi Supply Chain Management: Technologies, Opportunities, and Challenges" reviews the different Industry 4.0 technologies and their recent applications in the supply chain management sector in Saudi Arabia. Taha M. Mohamed et al. give an overview of these Industry 4.0 recent technologies including robotics, artificial intelligence, additive manufacturing, blockchain, Internet of Things, and many other technologies. The authors also describe the different opportunities and challenges of adopting these technologies in the Saudi market.

We hope that the volume will help readers to deepen their knowledge in solving business problems by using recent machine learning and data analytic techniques. We also hope that the proposed chapters will obviously help managers and researchers to propose more creative solutions to solve business problems.

Jeddah, Saudi Arabia	Bader Alyoubi
Jeddah, Saudi Arabia	Chiheb-Eddine Ben Ncir
Jeddah, Saudi Arabia	Ibraheem Alharbi
Sfax, Tunisia	Anis Jarboui

Contents

About the Editors

Bader Alyoubi is a full professor of Management Information Systems and Dean of the College of Business at the University of Jeddah, Saudi Arabia. His research interests include decision support systems and knowledge management methods and their application in business, government, and health domains. He is the author and coauthor of more than 50 publications in the field of specialization. He established the Saudi Center for the Preparation and Empowerment of Entrepreneurs at the University of Jeddah. He has many contributions to the structuring of colleges and scientific disciplines at the University of Jeddah and is chairman of several committees.

Chiheb Ediine Ben Ncir received his PhD in computer science and management from Higher Institute of Management, University of Tunis in 2014 and an HDR degree (Habilitation for the Supervision of Doctoral Research) in 2021. He occupied the position of assistant professor at the Higher School of Digital Economy (University of Manouba, Tunisia) from 2015 to 2018. Currently, he is an assistant professor at the University of Jeddah, Saudi Arabia, since 2018 and a member of LARODEC laboratory (University of Tunis). He is also a business intelligence and big data instructor at IBM North Africa and Middle East. His research interests concern machine learning methods and data mining tools with a special emphasis on Big Data clustering, disjoint and non-disjoint partitioning, and kernel methods, as well as many other related fields. He is the author or coauthor of more than 50 publications in several prestigious journals and conferences.

Ibraheem Mubarak Alharbi received a BA degree from King Abdul Aziz University, Saudi Arabia, in 2002, and Master's and PhD degree from La Trobe University, Australia, in 2009. Currently, he serves as an associate professor in the Department of Management Information Systems, College of Business, University of Jeddah, Jeddah, Saudi Arabia. His research interests include business and information ethics, information privacy, and electronic commerce. He has published many research articles in reputed journals and participated in many international conferences.

Anis Jarboui is a full professor in business administration at the University of Sfax. He holds a PhD in Finance from the University of Nice Sophia Antipolis – France (Université Côte d’Azur 2004) and an HDR degree (Habilitation for Supervising Doctoral Research) in 2008. He was Dean (College of Business Administration) of the Higher Institute of Business Administration of Sfax, from 2011 to 2017. He has previously served as a researcher and/or professor in numerous other universities and business schools including IAE Nice (2005), IAE Lille (2006–2007), and EM NormandieFrance (2017–2021). He is cofounder and vice-president of Latige-Lab. in Technology, Governance, and Entrepreneurship. He currently serves as member of editorial and scientific committee of various academic international conferences and has been invited as a speaker or moderator at numerous international conferences. His research interests involve several aspects of finance and accounting and entrepreneurship such as corporate governance, voluntary disclosure, earnings quality, entrepreneurial finance, and behavioral finance. He authored numerous papers listed among the top 5 most cited articles in refereed journals such as FRL. He published more than 100 articles in peer-reviewed/indexed journals and conferences.

Chapter 1
Predicting Salaries with Random-Forest Regression

Frank Eichinger and Moritz Mayer

1.1 Introduction

Paying competitive salaries is essential for companies of any size to retain current and attract new employees. At the same time, paying more than the market price is equally undesirable from a company perspective. Determining the market value for a particular employee or candidate is challenging, as salaries are influenced by many factors. These include the profession, the region, the age, the work experience, the company industry and the company size. Estimating competitive salaries requires a database of (close to) real salaries in a good data quality. If large amounts of data records are available, comparison groups can be built to benchmark salaries and to visualise salary distributions, for instance, of a certain profession, and provide key numbers such as median values (see Fig. 1.1 for an example). This can help employees, employers and consultants to find out if a certain salary is within the usual range. For the German market, there is the "Entgeltatlas" [19] of the German Federal Employment Agency and the commercial product "Personal-Benchmark online" [11] from DATEV eG which provide benchmark services based on large volumes of real data. These tools display salary distributions based on profession and region and partly on age and gender. To ensure statistical validity and privacy, distributions need to enclose a certain number of individuals. Hence not all

F. Eichinger (✉)
DATEV eG, Nuremberg, Germany
e-mail: frank.eichinger@datev.de

M. Mayer
DATEV eG, Nuremberg, Germany

University of Bamberg, Bamberg, Germany
e-mail: moritz.mayer@datev.de

B. Alyoubi et al. (eds.), *Machine Learning and Data Analytics for Solving Business Problems*, Unsupervised and Semi-Supervised Learning,
https://doi.org/10.1007/978-3-031-18483-3_1

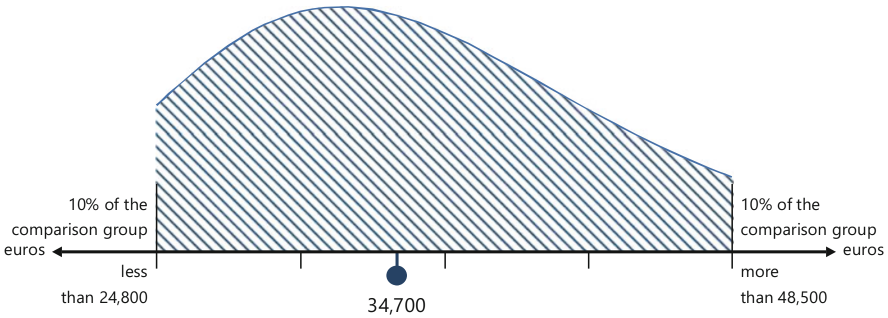


Fig. 1.1 Example salary distribution from [11] including median and percentiles

combinations of the factors mentioned are valid and can be selected. However, these solutions do not consider further factors than profession, region and demographics. Other relevant attributes such as the company industry and size may not be looked at.

If fewer data records are available or if a broader range of attributes should be used to be more specific, comparison-group benchmarks cannot be used. Alternatively, regression models may be used to predict salaries. This has the advantage that the output is a numerical value in a currency which refers to a competitive salary. Such values may be easier to interpret than a salary distribution alone. Further, when a company hires a new employee, a regression-based approach can propose an adequate salary. Particularly when the regression model uses more attributes than the distribution of a comparison group, the predicted value typically is closer to the competitive salary than the average or median of a distribution. This is an advantage not only for human-resources managers using salary prediction tools but also, for instance, for tax advisers who get enabled to offer business consultancy to their clients. Several commercial providers publish market overviews and individual salary predictions derived from salary data obtained from surveys or interviews of employers or employees. However, the exact statistic approaches employed are usually confidential. One approach, the “Gehaltsvergleich BETA” [21] of the German Federal Office of Statistics uses a relatively small sample of real salary data [24] and employs a specialised linear regression model to predict salaries. However, if more data records are available, machine learning, concretely more sophisticated regression models, may lead to better predictions. A general problem, particularly in the scientific literature (see Sect. 1.2), but also for commercial offers, is the limited availability of salary data in good quality. Therefore, most studies focus on very specific industries or markets or use quite outdated or small datasets.

In this study, we investigate whether and how machine learning, particularly regression trees and random-forest regression, can achieve high-quality salary predictions on a large dataset of salary data. As a dataset for learning and evaluation, we use a sample of roughly three million real payslips each month over one year including more than 300 professions originating from the payroll software from the German company DATEV eG. In a nutshell, besides several pre-processing steps including outlier removal, we propose an ensemble regression approach which learns –for each profession– a random-forest regression model to predict salaries. In our comprehensive evaluation, we show that this approach based on a large real dataset (a) performs better than related work on smaller datasets (comparing the error measures published) and (b) that the prediction errors can be reduced by 17.8% compared to our baseline.

The contributions of this study are as follows: (1) We show that more sophisticated machine-learning models than linear regression, namely random-forest regression, are suitable to predict salaries on a large dataset. (2) We demonstrate that an ensemble of one regression model per value of a categorical independent variable may clearly outperform a single regression model in situations where this variable has many values. (3) Our evaluation is based on a real-world dataset of millions of payslips, which cannot be found in the scientific literature in a comparable data quality and quantity.

The remainder of this chapter is organised as follows: Sect. 1.2 reviews related work, Sect. 1.3 describes our approach, Sect. 1.4 presents our results, Sect. 1.5 discusses them and gives directions for future research and implementation, and Sect. 1.6 concludes.

1.2 Related Work

The state of research on statistical models for salary prediction is rather weak. Although there is some work on salaries and their influencing factors as well as on machine-learning approaches for prediction, the focus is usually on specific aspects. This is either a relatively narrow selection of employees, for example, only managers [34], or a relatively narrow selection of industries or sectors, for example, only hospitals [43] or universities [4]. The goal of developing a statistical predictive model that (a) covers a broad picture of salary factors and (b) is valid for as many employees as possible has been pursued in comparatively few studies.

The work of Chakraborti [9] compares the predictive performance of five machine-learning algorithms for salaries of U.S. census-data individuals. Tree-based algorithms achieved the highest performance. Yet the census data used in the study is quite outdated –from 1994– and the salaries entailed are only available in the form of a categorical variable (>50,000$ and <=50,000$). Chakraborti addresses this situation and reasons that having to use such an outdated dataset is illustrating the lack of available datasets and research by other authors on the topic of salary prediction. A study by Viroonluecha and Kaewkiriya [39] applies various machine-learning algorithms to assess their salary prediction performance on data crawled from a job platform in Thailand. The crawled data is relatively new –from 2018– but also limited: only employees with academic education are considered and the dataset is relatively small (39,000 employees). The used neural network achieves the highest performance, closely followed by the random-forest model of the study. The performance metric used is the root-mean-square error (*RSME*). The result of 7740฿ (approximately 210 €) is difficult to assess, as there is no further information on, for example, a relative deviation from the target, which a metric like the mean absolute percentage error (*MAPE*) would give. Neither does the study provide information on distribution metrics of the actual salaries in the dataset such as the mean, the median or percentiles.

A related field of research which may use the same kind of data is employee churn prediction [38, 41]. However, the studies in this field likewise are struggling with the lack of available data and do not provide any additional insights into the selection of machine-learning algorithms for salary data. In essence, the analysis of the academic work in the field of salary prediction leads to the conclusion that related work on salary prediction is scarce. A significant lack of publicly available, high-quality datasets on employee salaries is hindering researchers to conduct more studies researching the effectiveness of machine-learning algorithms in salary prediction.

Apart from the scientific literature, there are several commercial solutions for salary predictions. However, most of them do not publish how they compute them. One exception is the “Gehaltsvergleich BETA” [21] of the German Federal Office of Statistics. The tool employs specialised linear regression models [37] to predict salaries and is based on a relatively small sample (600,000 employees) of real salaries [24]. It is therefore the product and study closest to the research presented in this chapter.

1.3 Data Preparation and Random-Forest Regression Modelling

In this section, we describe our dataset, the steps for pre-processing including outlier handling, the selection of machine-learning models and our choice of an ensemble of ensemble (random forests) approach for predicting salaries.

1.3.1 Data Source, Data Analysis and Feature Selection

In this study we work with a subset of a dataset of pseudonymised payslips extracted from the payroll accounting solutions of DATEV eG. This dataset comprises the payslips of 3.14 million employees in Germany over one year not including any trainees, working students, marginal employment or employees working less than 15 h per week. The dataset available is limited to the 330 professions which occur most frequently. In order not to be affected by any effects of the Covid-19 pandemic and the resulting large shares of short-time allowances in certain industries such as gastronomy, we have used data from the year 2019.

1.3.1.1 Dependent Variable

The dependent variable "salary" needs to be specified in more detail. We have chosen to predict the *annual gross income*, a numerical variable in Euros. The reason is that this value includes all special payments such as holiday pay, Christmas bonus, further bonuses, etc. This makes it easier to compare as monthly salaries do not include such payments and some employees may not have them and may have higher monthly payments instead. We have extrapolated months with missing or only partial payments, for instance, when the employee was sick. Furthermore, we have extrapolated values for employees working in part time (less weekly working hours than the company default) to the company default weekly working hours (typically 40 h) to have comparable numbers. Our preliminary experiments have shown that extrapolating to the company default leads to better results than extrapolated to a fixed number such as 40 h. Our dependent variable, the *annual gross income*, has a log-normal distribution, which is in line with the literature [28]. This means that the income distribution is skewed to the right and displays a long right tail.

1.3.1.2 Independent Variables

We have chosen the attributes listed in Table 1.1 from our dataset as independent variables as an input for our prediction models. Our data analyses have shown that all independent variables have a medium to high correlation with our dependent variable. If we look at their correlations for each profession, it is obvious that our variables have very different correlation values for the various professions. As one example, the *level of education* is correlated with the salary. However, looking at information-systems professionals as an example, the *level of education* only has a negligible influence on the salary as the vast majority of them has a college degree.

The categorical variables *company industry* and *federal state* have many possible instances. As this is not optimal for many machine-learning algorithms, we have grouped them. Our preliminary experiments have shown that this may slightly increase predictive performance. For the *company industry*, we use the hierarchical structure of the official taxonomy [22] and assign one of 23 sections to a company. Regarding the *federal state*, we cluster the states into four groups by minimising differences in the median of the *annual gross income* within a cluster and maximising it between the clusters. Also, the variable *profession* has many possible instances. We propose a specific handling in Sect. 1.3.4.

1.3.2 Outlier-Handling Strategy

Our dependent variable, the *annual gross income*, has a large spread. While the distributions of the variable are quite different for the various professions, also the

Table 1.1 Independent variables

	Variable	Type
Employee	*Profession* [23]	Categorical
	Age	Numerical
	Gender	Categorical
	Level of education [20]	Ordinal
	Level of professional training [20]	Ordinal
	Contract type (full time/part time,[a] temporary/permanent) [20]	Categorical
Employer	*Federal state*[b]	Categorical
	Degree of urbanisation[b,c] [14]	Ordinal
	Company size (number or employees)	Numerical
	Company industry [22]	Categorical

[a] Our analyses have shown that this may influence results even if we extrapolate to full time.
[b] We derive the *federal state* and the *degree of urbanisation* from the company zip code.
[c] The *degree of urbanisation* has three possible values which are ordered:

1. *Cities* (densely populated areas)
2. *Towns and suburbs* (intermediate density areas)
3. *Rural areas* (thinly populated areas)

spread within a profession can be quite large. Many salaries exceed the median salary by a factor of more than 1.5. For example, while the median value for a secretary is around 35,000€ per year and the 90% percentile is around 55,000€ per year, there are individuals earning 100,000€ and more. Very likely, these data points are outliers resulting from mistakes when entering the profession into the payroll software or not updating it when the employee has climbed-up in career. Another example is salaries below the German minimum wage probably caused by weekly working hours incorrectly entered into the payroll software. As outliers may affect prediction models quite heavily, a well-chosen strategy for outlier removal is essential. We apply our outlier handling to our whole dataset before splitting it in training, validation and test sets.

Instead of a simple approach for outlier removal which removes the highest and lowest, say, 5% of data points per profession, we choose a more thorough approach. The inter-quartile-range –the difference between the 75% and the 25% percentile– is commonly used in box plots to create the whiskers which determine the upper and lower threshold for outliers. We have investigated the percentage of outliers in comparison to different inter-quartile-range factors (*IQRF*), which allowed us to understand the outlier situation in the data more deeply. Initially we have chosen a conservative *IQRF* of 3 removing only 1% of the data, and we switch in our evaluation (see Experiment 4 in Sect. 1.4.2) to the commonly used *IQRF* of 1.5, removing 4% of the data.

1.3.3 Selection of a Machine-Learning Approach

While relatively simple linear regression models have been used in the related work [24] on smaller datasets having similar attributes, we assume that more sophisticated machine-learning models may achieve better performances when more data is available. While several models such as support-vector machines and neural networks may be used for predicting numerical salaries, we have chosen to investigate regression trees [8] and ensembles of such models, random forests [7], which have performed well for salary predictions in [39]. We detail on random forests in the subsequent section. These models offer several advantages: Regression trees and random forests are said to be good in handling categorical attributes, missing values, noise and outliers. In addition, they are said to be robust against overfitting, no separate feature selection, scaling or transformation steps are necessary, and correlated independent variables do not affect the models by much. Furthermore, random forests are one of the most accurate machine-learning methods [5]. Couronné et al. [10] and Fernández-Delgado et al. [16] demonstrate this for the related classification problem in large-scale evaluations.

1.3.3.1 Learning and Applying Random Forests

Random forests [7] are a machine-learning technique for classification or regression that constructs an ensemble of decision or regression trees. The idea behind such ensemble methods is that the prediction accuracy is increased by combining the results from multiple –possibly diverse– models [1, 5, 27, 36]. Frequently, diversity is achieved by introducing randomness into the learning algorithms. The algorithm for random forests applies, besides other modifications, the bagging technique [6] to tree-learning algorithms [33]. Details regarding decision and regression trees can be found in several textbooks, for instance, in [1, 5, 36]. Algorithm 1 describes the general process of learning a random forest *RF* from a training dataset *TS* consisting of *n* trees. It internally employs an arbitrary decision or regression tree-learning algorithm *tree_learner*() without pruning which is modified in order to internally use a small random subset of the independent variables at each split. Hence, random forests add two kinds of randomness to model-building: Firstly, the bootstrapped sampling approach of bagging creates permutated training datasets. Secondly, using random subsets of independent variables leads to more diverse trees. The reason for using random subsets of variables is that otherwise even bagged trees having differently permutated training sets tend to choose the same independent variables at the top level, resulting in relatively similar trees [1]. The random-subset approach attempts to reduce the variance and the correlation of the predictions of the individual trees, ultimately leading to better predictions than achieved by individual or bagged trees [1, 5, 27].

To deduce classifications or numeric predictions (in case of regression) from a random forest, all contained trees are used to predict a data record. The result is

Algorithm 1 Construction of random forests

```
Input: training set $TS$; number of trees $n$; a tree-learning algorithm $tree\_learner()$ without pruning
    using a small random subset of independent variables at each split
Output: random forest $RF$ (a set of trees)
$RF \leftarrow \emptyset$
for 1 to $n$ do
    $TS' \leftarrow$ a bootstrap sample of $TS$
    $RF \leftarrow RF \cup tree\_learner(TS')$
end for
return $RF$
```

then derived by employing a majority-vote strategy (classification) or calculating an average over all n trees (regression).

1.3.4 An Ensemble of Random-Forest Regression Models

We now describe our approach for predicting salaries with an ensemble of random-forest regression models.[1] Our analyses regarding the correlation of independent variables with the salary (see Sect. 1.3.1) as well as the experiments in our evaluation (see Experiment 1–3 in Sect. 1.4.2) have shown that there is one independent variable that outweighs the others by a considerable margin. This variable is the *profession* of an employee, a categorical attribute having a high cardinality (330 possible distinct values). This is challenging, as categorical features with high cardinality are problematic for tree-based machine-learning approaches. The reason is that the learning algorithms of decision and regression trees are unlikely to determine the best split with this type of data [17, 35]. We propose to solve this problem with an ensemble of random-forest regression models where we train one random-forest model per possible value of such a categorical feature with high cardinality and feature importance as described in the following. We have chosen this approach as possible alternative strategies such as grouping or clustering the values[2] of the high-cardinality feature would lead to the loss of potentially relevant information of such a highly predictive feature.

We call a categorical independent variable having a high cardinality (denoted m) and high feature importance P. We denote the distinct values of P $p_1, \ldots, p_m$. In our case, the *profession* is the independent variable P and the $m = 330$ different professions are $p_1, \ldots, p_m$. We partition our training dataset TS by assigning all

[1] While random forests are our main machine-learning technique for salary predictions, regression trees can be used in our approach as an alternative. They are less complex and perform worse than random forests. We compare the prediction performances of random forests versus regression trees in Sect. 1.4.2 in detail.

[2] We use such a strategy for the less predictive features *company industry* and *federal state* as described in Sect. 1.3.1.

tuples $t \in TS$ having the same value p_i of P to the same partition $TS_{P=p_i}$: $TS = \bigcup TS_{P=p_i}$. We then train one random forest RF_i for each $TS_{P=p_i}$ using Algorithm 1. Correspondingly, to derive predictions, we use P to decide which random forest RF_i to use. Algorithm 2 describes our approach for the construction of an ensemble model EM of random-forest regression models. It internally calls a function *random_forest_learner*() which learns a random forest as described in Algorithm 1.

Algorithm 2 Construction of an ensemble of random-forest regression models

```
Input: training set $TS$ containing a categorical independent variable $P$ having a high cardinality
    and high feature importance; a random-forest algorithm random_forest_learner(), for exam-
    ple, Algorithm 1
Output: ensemble model $EM$ (a set of random-forest-regression models)
$EM \leftarrow \emptyset$
partition $TS$ into $m = |P|$ partitions $TS_{P=p_i}$ where all tuples have the same value of $P$ $p_i$
for $i = 1$ to $m$ do
    $RF_i \leftarrow random\_forest\_learner(TS_{P=p_i})$
    $EM \leftarrow EM \cup RF_i$
end for
return $EM$
```

In this study, we learn 330 random-forest models, one per profession. Each random forest consists internally of many regression trees. To derive a salary prediction for a specific employee, we use the employee's profession to select the corresponding random-forest model.

1.4 Evaluation

In this section, we present the evaluation of our approach as described in Sect. 1.3.4 on the dataset described in Sect. 1.3.1.

1.4.1 Experimental Setup, Measure of Prediction Accuracy and Baseline

We have implemented our approach in an Apache Spark cluster [42] on standard central processing units (CPUs) using the scikit-learn library [31] for machine learning. For our experiments presented in the following, we have divided our dataset into an 80% training set, a 10% validation set and a 10% test set, which is a standard procedure in machine learning [36]. For all but the last of our experiments, we use the training set and perform a standard fivefold cross validation to obtain the experimental results. We then use the validation set for hyper-parameter tuning and use the test set to derive our final results.

We measure the prediction accuracy using the standard mean absolute percentage error (*MAPE*), which has also been used in the related work closest to ours [24]. The *MAPE* is the average of all absolute deviation values of the predictions from the actual values divided by the actual value. As we employ an ensemble of prediction models in many of our experiments, we calculate the average weighted by the number of employees predicted by a model in this situation. The reason why we have chosen the *MAPE* is that we are convinced that it is intuitive and makes more sense from a business perspective for the problem of salary predictions than to use other measures. For example, the probably most popular measure for regression, the mean squared error (*MSE*), is not intuitive as squared Euros or Dollars do not make sense for humans. Further, as some professions have much higher average salaries than others, an accuracy measure using absolute values (for example, the *MSE*) is not a well-enough indicator for the prediction accuracy. For instance, a deviation of 1000 Euros from the actual salary is a much better prediction for an actual salary of 100,000 Euros than for a 20,000 Euro salary. Therefore, a percentage-based accuracy measure like the *MAPE* is a better choice for our business problem.

To compare our results from the machine-learning models, we define the baseline as follows: The baseline predicts the salary of an employee with the median salary of all employees in that profession while ignoring all further variables. This is a very simple approach, but it simulates the first guess for a salary one would probably have when looking at salary distributions as shown in Fig. 1.1. All models developed in this study are expected to perform better than this baseline. This simple baseline approach already yields a *MAPE* of 20.8%. This is not far off from the *MAPE* of 19.3% published in the related work [24] using linear regression on many variables, a by far more complex approach (but obtained on a different dataset, see Sect. 1.5.1).

1.4.2 Experimental Results

In Fig. 1.2, we present the results of our five experiments. The results show that training individual models per profession and a more extensive outlier handling both are significant steps in improving the predictive performance. We always train and evaluate a regression tree and a random-forest regression model and compare it below. Note that Experiment 1 and 2 can be seen as preliminary experiments to demonstrate the effect of our full approach (as described in Sect. 1.3.4) in Experiments 3–5.

Experiment 1: One Model for All Professions In our first experiment, we train one model for all professions. The result of the random-forest model is already better than our baseline. The difference is around one percentage point, which is a relatively small improvement. As described in Sect. 1.3.4, to better deal with

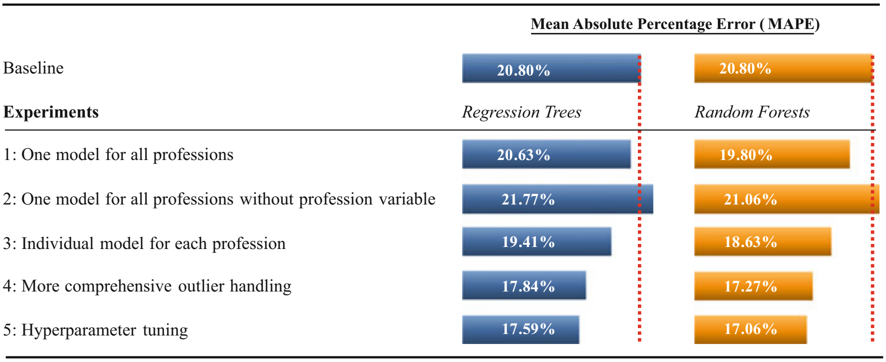


Fig. 1.2 Mean absolute percentage error (*MAPE*) of the baseline and our experiments

our important categorical independent variable *profession* and its large number of possible values, we switch to an ensemble approach in Experiment 3.

Experiment 2: One Model Without the Profession To demonstrate the influence of the independent variable *profession* for salary predictions, we run this experiment, which is the same as Experiment 1, but without using the variable *profession*. The results are more than one percentage point worse than Experiment 1 and even worse than our baseline which makes use of the *profession* only. This shows that this variable is essential and needs to be treated adequately.

Experiment 3: Separate Models for Each Profession The ensemble approach (random forest) reduces the *MAPE* by more than another percentage point compared to Experiment 1. This is a little more than two percentage points better than our baseline.

Experiment 4: More Comprehensive Outlier Handling As discussed in Sect. 1.3.2, resulting from the fact that our dataset inherently contains some incorrect data points, we deal carefully with the outliers in our dataset. In this experiment, we switch from our conservative approach of outlier removal ($IQRF = 3$) to a less conservative approach ($IQRF = 1.5$). This yields another improvement. The random-forest performance (*MAPE*) improves by one percentage point compared to Experiment 2 and is 3.5% points better than the baseline.

Experiment 5: Hyper-Parameter Tuning Regression trees and random forests come with several parameters to control and steer the machine-learning process. These should be adopted to the dataset. We try several settings for the three parameters with the highest possible impact [32] in the scikit-learn implementation [31] using the training set for learning and the validation set for evaluation: (1) number of variables randomly sampled as candidates at each split, (2) minimum number of samples required to be at a leaf node and (3) number of trees in the forest.

The result of the hyper-parameter tuning yields another slight increase in predictive performance compared to Experiment 4. The resulting *MAPE* of 17.06% is roughly four percentage points better than the baseline, a relative improvement of 17.8% (random forests).

Finally, we have trained our models with the new parameters on the unified training set (training set and validation set; 90% of all data) and have evaluated the results on the test set (10% of all data). To ensure the model fit, we compared the *MAPE* on the unified training set (16.57% for the regression trees, 16.39% for the random forests) with the *MAPE* on the up to this moment unseen test set (17.60% for the regression trees, 17.08% for the random forests). The differences of 6.2% for the regression trees and 4.2% for the random forests are marginal. Hence, it can be concluded that our models do not overfit, and it can be assumed that the models generalise well to unseen data.

In all experiments, the random forests perform better than the regression trees. This confirms the findings from the literature [5, 10, 16] which have been obtained from classification problems. Our results show that random forests also increase the predictive performance when it comes to regression.

1.4.3 Runtime

Learning our ensemble of random forests as described in Sect. 1.3.4 comes with a considerable computational cost. We have measured runtimes in the range of a few to several hours when learning our 330 random forests. As we have done all experiments on standard central processing units (CPUs), and as random forests are known to benefit from parallel computations in graphical processing units (GPUs), it can be expected that GPUs can speed-up computations considerably. Preliminary experiments of ours with neural networks on CPUs have shown that they perform considerably worse than our random forests in terms of runtime (runtimes more than doubled on the same hardware). This, however, depends to a large degree on the chosen network topology, and neural networks likewise largely benefit from GPUs. Particularly as salary information is usually not updated more frequently than monthly, spending some hours of computation time each month for model learning seems to be not problematic. Besides model learning, predicting salaries for individual employees can be done faster than in one second, which allows for integration in interactive software.

1.5 Discussion of the Results and Future Directions

We now discuss the results from the evaluation (Sect. 1.4) and give future directions.

1.5.1 Comparison to Related Work

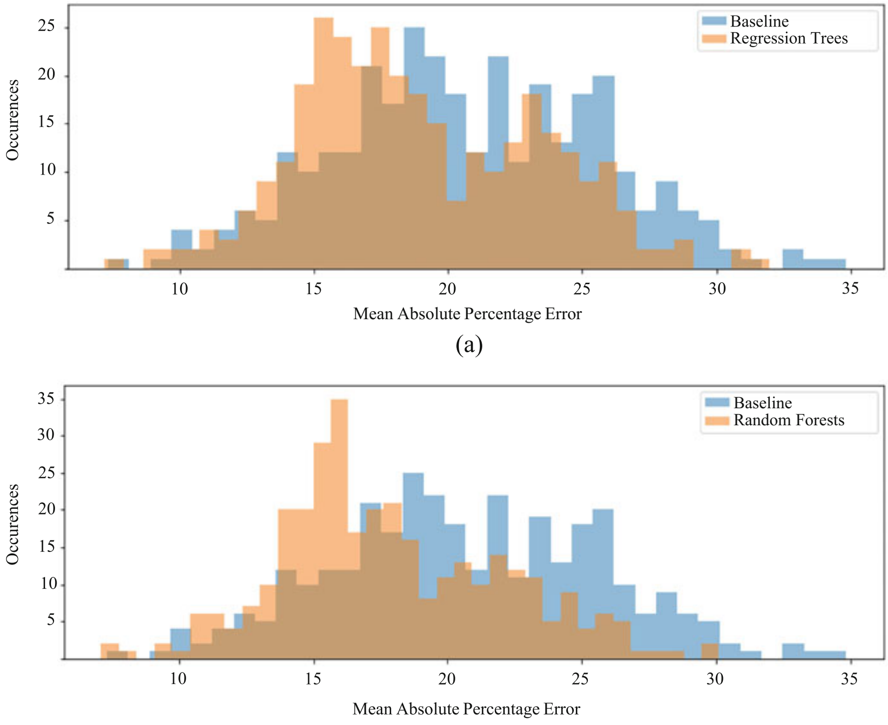
Comparing the mean absolute percentage error (*MAPE*) with values obtained from different datasets is not easy, as the dataset itself and questions of pre-processing –in particular outlier handling– affect the results (as shown in Experiment 4). However, if there are *MAPE* values published, one might obtain a rough idea if the *MAPE* values are in the same magnitude or not.

The authors of [24] publish a *MAPE* of 19.27%, which is a little more than two percentage points higher than our results. However, the authors have used a smaller dataset (roughly factor 5). As we do not know how this dataset was assembled and if and how they have possibly eliminated outliers, a direct comparison is not possible. The values nevertheless suggest that using random-forest regression on a large dataset of salary data is worth the computational effort in comparison to the results from the simpler linear regression model on a smaller dataset. The other studies discussed in the related work do not publish *MAPE* values or have employed classification algorithms to predict classes of salaries. This makes it impossible to obtain meaningful error percentages. Furthermore, the same problems regarding datasets and pre-processing apply. From all results published, we got the impression that our *MAPE* of 17.1% is a respectable result, and that it will be hard to obtain much better results using a data-driven approach without incorporating further data than the data from payslips as investigated in our study.

1.5.2 Analysis of the Regression Trees and Random-Forest Results

In Sect. 1.4.2, we have presented the (average) *MAPE* of 17.1% for our random-forest regression ensemble. Following up on the promising overall performance of our approach, we are interested in how the individual models perform (one per profession).

Figure 1.3 presents histograms of the distributions of the *MAPE* values for the baseline and our regression trees (Fig. 1.3a) and the baseline and our random forests (Fig. 1.3b). The latter illustrates that both the distributions of the baseline and the random forests have a relatively large range from roughly 10% to 30% (baseline) or 35% (random forests), while the distribution of the random forest *MAPE* values is clearly shifted to the left. As the average *MAPE* of the random forest ensemble approach is lower, this left shift is plausible. It can be an indication that the input parameters and data used in our models are more useful predictors for the salaries of some professions than they are for others. For the models yielding comparably weak prediction performances, factors not reflected in the data, but influencing the salary of those professions in reality, may be the cause. The wide *MAPE* value range can be observed in the regression trees, random forests and baseline approach alike. We



(a)

(b)

Fig. 1.3 Distribution of the mean absolute percentage error (*MAPE*), the vertical axis shows the number of models having a certain *MAPE* value. (**a**) Regression trees. (**b**) Random forests

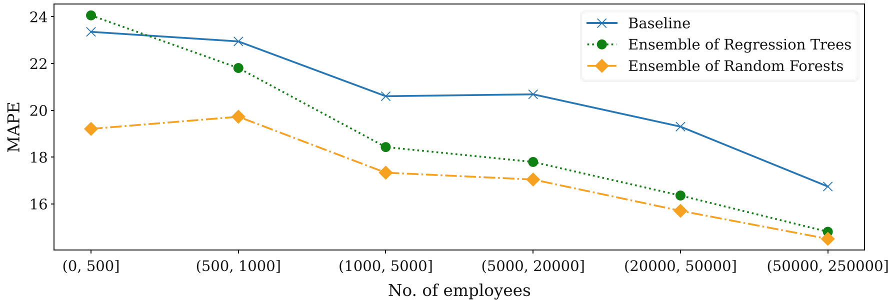


Fig. 1.4 The mean absolute percentage error (*MAPE*) versus the number of employees per model (per profession)

now discuss possible reasons for the characteristic that not all jobs can be predicted with the same *MAPE*.

Figure 1.4 displays the *MAPE* in relationship to the number of employees per model. In general, professions with fewer employees have larger *MAPE* values,

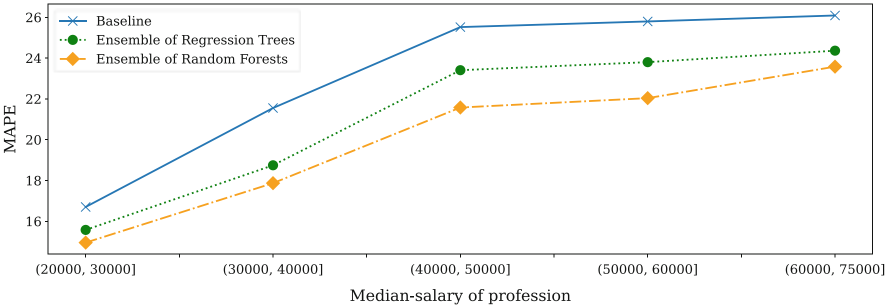


Fig. 1.5 The mean absolute percentage error (*MAPE*) versus the median salary (dependent variable *annual gross income*)

which seems to be intuitive as there are fewer training examples. Figure 1.5 displays the *MAPE* in relationship to the median salary. Here we can clearly see the best performance in the lower incomes. High incomes of 40,000€ per year and more are difficult to predict (the general median salary in Germany is around 39,000€ [25]). The reason is probably that many of the lower-paid professions are kind of more standardised, and the salaries do not vary much. Probably, there are more collective agreements in the professions where lower salaries are paid. The variation of salaries is a lot higher in the professions with a higher median salary, leading to the fact that they are harder to predict.

While most salaries were predicted more accurately by the random-forest ensemble than with the baseline model (in 309 out of 330 professions), we found that the salaries of 21 professions were on average predicted more accurately by the baseline model using nothing but the median of the profession. While there is no significant influence of the number of employees per profession on the *MAPE* of the respective model, the expertise level of those professions may be an explaining factor: Eleven out of the mentioned 21 professions are “helper activities” (lowest expertise level) and seven are of “professional expertise” (second lowest expertise level, according to the German Classification of Occupations [23]), having both a below average income. Probably, these employees have quite differing backgrounds and are thus hard to predict.

1.5.3 Feature Importance

In order to better understand our random-forest models, we have conducted permutation-feature-importance analyses [18] using the implementation of scikit-learn [31]. Due to the high computational costs of these analyses, we have done this for the models of a small sample of professions.

Figures 1.6 and 1.7 contain the results of two permutation-feature-importance analyses of two random-forest models (professions). For these two models, the *company size* is by far the most important independent variable, and the *federal state* is important in both models. For the *company industry*, the importance is high in Fig. 1.6 and low in Fig. 1.7. One explanation for the low value for the computer scientists (Fig. 1.7) is that many of them work in the same industry. Hence, the industry does not say much about the salary. For the *age* and the *level of professional training* it is the other way round: Both variables are important for the computer scientists (Fig. 1.7) where the more experienced and better educated employees earn more. They are less important for the textile sewers (Fig. 1.6) where the spread of salaries is not as big. The two examples from Figs. 1.6 and 1.7 illustrate that the feature importance vary quite largely in the different professions. Taking more than these two examples into account, it is observable that some variables –in particular the *level of education* and the *gender*– have a comparably low influences on the salary.

1.5.4 Usage of the Variable gender *and Further Improvements*

In this study, we make use of the variable *gender*. This makes sense from the perspective that it is known that there is a gender pay gap [15] which we have confirmed in our analyses (see Sect. 1.3.1). It might otherwise not be desirable to use this variable in a real product, as those should not contribute to increase the gender pay gap. In other settings, such as the "Gehaltsvergleich BETA" of the German Federal Office of Statistics [21], the gender is used as well. The rationale is that its provider has the mission to create transparency of salaries including the gender. As

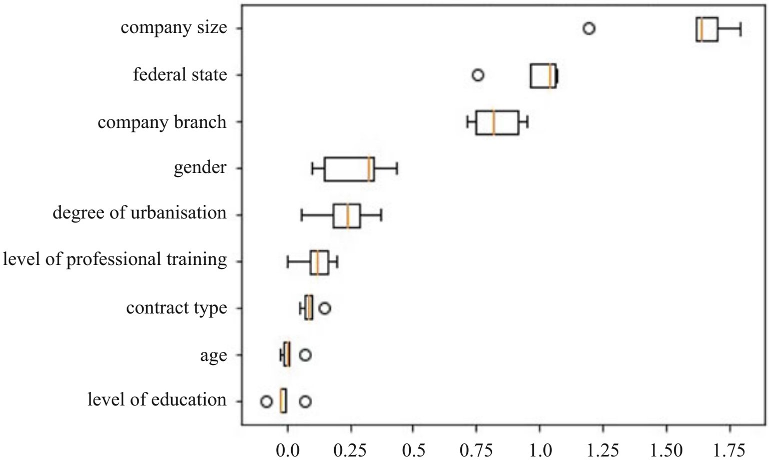


Fig. 1.6 Box plot of the permutation feature importance in percentage points of the random-forest regression model representing the profession "textile sewer"

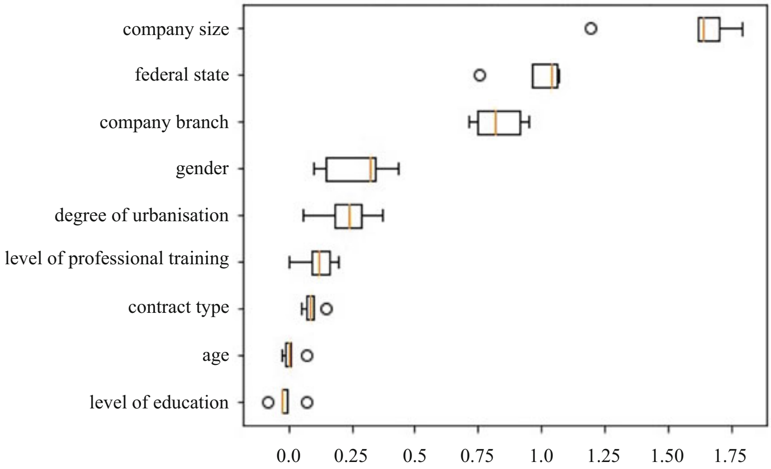


Fig. 1.7 Box plot of the permutation feature importance in percentage points of the random-forest regression model representing the profession "computer scientist"

we have observed the –maybe surprising– low variable importance of the *gender* in the permutation-feature-importance analyses in Sect. 1.5.3, we have conducted additional experiments: We have investigated the influence of the variable *gender* on the predictions for all professions by removing it from our dataset. This decreases the error values from the regression trees and the random forests in Experiment 5 (see Sect. 1.4.2) by less than half a percentage point. Hence, our approach can work without the variable *gender* as it affects the predictive performance not by much.

To predict salaries more accurately, it would be desirable to incorporate the work experience as another independent variable, as more experienced employees usually have a higher salary. When working with payroll data, this information is typically not available. Therefore, we have used the variable *age* as a surrogate and will in the future additionally incorporate the period of employment with the same employer. This might slightly improve the predictive performance as it approximates the work experience. However, it would be desirable to capture the work-experience information when hiring new employees and to keep it in the payroll software.

Our further ideas for performance improvements focus on pre-processing and data engineering. One idea is to use more complex methods in the outlier-handling process. For instance, clustering or anomaly detection. Another idea is to cluster the different professions into groups of professions. For example, software developers and programmers are two distinct professions in the classification of professions [23], having very similar characteristics (for instance, regarding salary and level of professional training). We plan to apply clustering techniques to merge such similar professions into clusters to capture more professions, to have more data per model to learn from and to possibly reduce the number of individual models. We see this as a promising approach as we could already show that the number of employees in the training data of the individual models correlates negatively with the prediction error (see Sect. 1.5.2).

Other data-engineering ideas of ours concern the regional information: We plan to further enrich our dataset with external data such as the population or the purchasing-power index of the place of employment. It could also be interesting to calculate a regional salary index based on our dataset and to use this as additional information to characterise the place of employment. These ideas aim at capturing the regional information better than with the federal state and the degree of urbanisation alone. This characterisation might be necessary for improving the predictive performance, as we know that the region has a significant influence on the salary, and as using zip codes or names of municipalities directly in the models might not work due to the extremely large number of possible values of these categorical variables.

1.5.5 Assessment of Prediction Quality for Real-World Software Products and Questions of Deployment

Even if further improvements as described before including enrichments with external data will likely lead to improved values of the mean absolute percentage error (*MAPE*), we do not expect these improvements to be large without having new data sources describing the employees. One promising new source to improve predictive performance could be the results of systematic employee ratings if companies have implemented respective evaluation and assessment processes, but this would be hard to compare between companies and would raise privacy questions. The assumption that predictions might not be improved largely leads to the question whether a *MAPE* of around 17.1%, which is roughly four percentage points better than the simple baseline, justifies a rather complex and computationally expensive machine-learning effort and will be accepted by customers. The question becomes even more severe, when we improve our baseline approach, for instance, by using median values of the professions in the same region.

Deploying and running machine learning in productive environments is still a rather complicated endeavour [12]. It includes monitoring and detection of concept drift [30], model management and re-training [3] and questions of data privacy. Attackers might, for instance, use an implementation of our approach to reconstruct the original learning data. This needs to be prevented by data-security measures such as limiting the number of requests that can be sent to the model API or by advanced privacy technology [2, 29]. Solutions in this domain could be differential privacy [13] –in particular subsampling [40]– or privacy-guided training [26].

Even if results may be improved by some degree, customers might still assume that a *MAPE* of, say, 15% or more is relatively high. It is therefore important to explain to (potential) customers of a product implementing our approach, how the *MAPE* values are calculated. First, *MAPE* values are strongly affected by incorrect data, and we must assume that not all wrong inputs regarding, for example, the profession or the weekly working hours, can be captured by outlier handling. Second, we have derived all *MAPE* values by predicting the salaries of real employees which we have not used for learning. In consequence, large deviations do not necessarily mean that the model is wrong, but might indicate that there are employees being underpaid or overpaid, which likely happens in real world. Besides this, from a customer perspective, predictions based on regression are more useful than having nothing but the median of a distribution of a rather large population (as in [11], see Fig. 1.1). Furthermore, the results of our random forests outperform the baseline approaches in 309 out of 330 professions (93.6%).

Taking all arguments in this section into account, we conclude that our approach with the predictive performance described is valuable for the customers when integrated into suitable software tools and justifies the efforts described.

One limitation to our approach might be the applicability to large companies and possibly certain under-represented professions, as our dataset focusses on small and medium-size companies. However, if a more complete dataset than ours is available for learning, we see no obstacles in applying our approach to it.

1.6 Conclusion

In this study, we have investigated an ensemble-of-ensembles approach for predicting salaries based on salary data, where we have learned one random-forest regression model per profession. In our comprehensive evaluation on a large real dataset, we have achieved a mean absolute percentage error (*MAPE*) of 17.1%. This is an improvement of 17.8% compared to our baseline, and it is two percentage points better than the results published of the related work (on a different dataset). Thus, we have shown that sophisticated machine-learning models are suitable to predict salaries on a wide range of professions and employees and that our ensemble-of-ensembles approach clearly outperforms other approaches, such as simply setting the prediction to the median per profession, or using linear regression. Our approach can be integrated into salary-analysis solutions to help HR managers and tax consultants to determine market prices for current and prospective employees.

Acknowledgments We thank Professor Dr. Sven Overhage for his ongoing support when conducting this research.

References

1. C.C. Aggarwal, *Data Mining: The Textbook* (Springer, Berlin, 2015)
2. M. Al-Rubaie, J.M. Chang, Privacy-preserving machine learning: threats and solutions. IEEE Secur. Priv. **17**(2), 49–58 (2019)
3. E. Ameisen, *Building Machine Learning Powered Applications* (O'Reilly UK Ltd., Farnham, 2020)
4. D.A. Barbezat, J.W. Hughes, Salary structure effects and the gender pay gap in academia. Res. High. Educ. **46**(6), 621–640 (2005)
5. M.R. Berthold, C. Borgelt, F. Höppner, F. Klawonn, *Guide to Intelligent Data Analysis: How to Intelligently Make Sense of Real Data*, vol. 42. Texts in Computer Science (Springer, Berlin, 2010)
6. L. Breiman, Bagging predictors. Mach. Learn. **24**(2), 123–140 (1996)
7. L. Breiman, Random forests. Mach. Learn. **45**(1), 5–32 (2001)
8. L. Breiman, J.H. Friedman, R.A. Olshen, C.J. Stone, *Classification and Regression Trees* (Wadsworth International Group, Fairview, 1984)
9. S. Chakraborti, A comparative study of performances of various classification algorithms for predicting salary classes of employees. Int. J. Comput. Sci. Inform. Technol. **5**(2), 1964–1972 (2014)
10. R. Couronné, P. Probst, A.-L. Boulesteix, Random forest versus logistic regression: a large-scale benchmark experiment. BMC Bioinform. **19**(1) (2018)
11. DATEV eG. Personal-Benchmark online. https://datev.de/web/de/mydatev/online-anwendungen/datev-personal-benchmark-online/. Accessed 23 Jan 2022
12. T. Davenport, K. Malone, Deployment as a critical business data science discipline. Harvard Data Sci. Rev. (3.1), Winter 2021 (2021)
13. C. Dwork, Differential privacy, in *International Colloquium on Automata, Languages, and Programming (ICALP)* (2006)
14. Eurostat, European Commission, Degree of Urbanisation. https://ec.europa.eu/eurostat/web/degree-of-urbanisation/methodology. Accessed 23 Jan 2022
15. Eurostat, European Commission, Gender Pay Gap Statistics. https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Gender_pay_gap_statistics. Accessed 23 Jan 2022
16. M. Fernández-Delgado, E. Cernadas, S. Barro, D. Amorim, Do we need hundreds of classifiers to solve real world classification problems? J. Mach. Learn. Res. **15**, 3133–3181 (2014)
17. J.J. Filho, J. Wainer, Using a hierarchical bayesian model to handle high cardinality attributes with relevant interactions in a classification problem, in *International Joint Conference on Artifical Intelligence* (2007)
18. A. Fisher, C. Rudin, F. Dominici, All models are wrong, but many are useful: learning a variable's importance by studying an entire class of prediction models simultaneously. J. Mach. Learn. Res. **20**(177), 1–81 (2019)
19. German Federal Employment Agency, Entgeltatlas. https://con.arbeitsagentur.de/prod/entgeltatlas/. Accessed 23 Jan 2022
20. German Federal Employment Agency, Occupation Codes for Statistical Messages in Germany. https://www.arbeitsagentur.de/betriebsnummern-service/taetigkeitsschluessel. Accessed 23 Jan 2022
21. German Federal Office of Statistics, Gehaltsvergleich BETA. https://service.destatis.de/DE/gehaltsvergleich/. Accessed 23 Jan 2022
22. German Federal Office of Statistics, German Classification of Economic Activities 2008. https://www.destatis.de/DE/Methoden/Klassifikationen/Gueter-Wirtschaftsklassifikationen/Downloads/klassifikation-wz-2008-englisch.html. Accessed 23 Jan 2022
23. German Federal Office of Statistics, German Classification of Occupations 2010. https://statistik.arbeitsagentur.de/DE/Navigation/Grundlagen/Klassifikationen/Klassifikation-der-Berufe/Klassifikation-der-Berufe-Nav.html. Accessed 21 Oct 2022

24. German Federal Office of Statistics, Interaktiver Gehaltsvergleich. https://www.destatis.de/DE/Service/Statistik-Visualisiert/Gehaltsvergleich/Methoden/Methodenbericht.pdf. Accessed 24 Jan 2022
25. German Pension Insurance, Durchschnittseinkommen. https://www.deutsche-rentenversicherung.de/SharedDocs/Glossareintraege/DE/D/durchschnittseinkommen.html. Accessed 23 Jan 2022
26. A. Goldsteen, G. Ezov, A. Farkash, Reducing risk of model inversion using privacy-guided training. Computing Research Repository (CoRR), abs/2006.15877 (2020)
27. L.I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms* (John Wiley & Sons, Hoboken, 2004)
28. E. Limpert, W.A. Stahel, M. Abbt, Log-normal distributions across the sciences: keys and clues. BioScience **51**(5), 341–352 (2001)
29. X. Liu, L. Xie, Y. Wang, J. Zou, J. Xiong, Z. Ying, A.V. Vasilakos, Privacy and security issues in deep learning: a survey. IEEE Access **9**, 4566–4593 (2021)
30. J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, G. Zhang, Learning under concept drift: a review. IEEE Trans. Knowl. Data Eng. **31**(12), 2346–2363 (2019)
31. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, É. Duchesnay, Scikit-learn: machine learning in python. J. Mach. Learn. Res. **12**(85), 2825–2830 (2011)
32. P. Probst, M.N. Wright, A.-L. Boulesteix, Hyperparameters and tuning strategies for random forest. WIREs Data Min. Knowl. Discovery **9**(3), e1301 (2019)
33. J.R. Quinlan, *C4.5: Programs for Machine Learning* (Morgan Kaufmann, Burlington, 1993)
34. R. Rahim, T. Husni, Yurniwati, Desyetti, The relation between cash compensation of banking executives, charter value, capital requirements and risk taking. Int. J. Bus. **25**(5), 399–420 (2020)
35. R. Ravi, One-Hot Encoding is making your Tree-Based Ensembles worse, here's why? https://bit.ly/3Fg81tS. Published in Towards Data Science. Accessed 04 May 2022
36. S.J. Russell, P. Norvig, *Artificial Intelligence: A Modern Approach*, 4th edn. (Pearson, London, 2020)
37. SAS Institute Inc., The SURVEYREG procedure, in *SAS/STAT 13.1 User's Guide*, chapter 98 (SAS Institute Inc., Cary, 2013), pp. 8353–8442
38. D.S. Sisodia, S. Vishwakarma, A. Pujahari, Evaluation of machine learning models for employee churn prediction, in *International Conference on Inventive Computing and Informatics (ICICI)* (2017)
39. P. Viroonluecha, T. Kaewkiriya, Salary predictor system for thailand labour workforce using deep learning, in *International Symposium on Communications and Information Technologies (ISCIT)* (2018)
40. Y.-X. Wang, B. Balle, S.P. Kasiviswanathan, Subsampled renyi differential privacy and analytical moments accountant. J. Mach. Learn. Res. **89**, 1226–1235 (2019)
41. I.O. Yigit, H. Shourabizadeh, An approach for predicting employee churn by using data mining, in *International Artificial Intelligence and Data Processing Symposium (IDAP)* (2017)
42. M. Zaharia, R.S. Xin, P. Wendell, T. Das, M. Armbrust, A. Dave, X. Meng, J. Rosen, S. Venkataraman, M.J. Franklin, A. Ghodsi, J. Gonzalez, S. Shenker, I. Stoica, Apache spark. Commun. ACM **59**(11), 56–65 (2016)
43. C. Zhang, Y. Liu, The salary of physicians in Chinese public tertiary hospitals: a national cross-sectional and follow-up study. BMC Health Serv. Res. **18**(661) (2018)

Chapter 2
Data-Driven Analysis of Microfinance and Social Loans Before and During the COVID-19 Pandemic Using Exploratory Analysis and Decision Tree Classifiers

Chiheb-Eddine Ben Ncir, Bader Alyoubi, and Roaa Alrazyeg

2.1 Introduction

Financial institutions have established different techniques and channels for attracting and sustaining customers by developing and providing reliable banking programs like microfinance. This business model refers mainly to the direct relationship between the lender and borrower without the objective of any positive social impact produced [1]. Later on, especially in developing countries, a related term has emerged referred to as "social banking" which indicates a subsidized government or development banking mainly related to microfinance or microcredit and has necessarily a positive social impact. In this context, social banks are specialized in providing financial products and services to create social and sustainable benefits for people, corporations, and organizations. The main principles that guide the operations of social banks are transparency, resilience, sustainability, and inclusivity [2]. As a result, social banks operate in societies and communities where they give support to individuals, corporations, and organizations and help them to profit from their social and economic activities.

C.-E. B. Ncir (✉)
College of Business, University of Jeddah, Jeddah, Saudi Arabia

Larodec Laboratory, ISG Tunis, University of Tunis, Tunis, Tunisia
e-mail: cbenncir@uj.edu.sa

B. Alyoubi
College of Business, University of Jeddah, Jeddah, Saudi Arabia
e-mail: balyoubi@uj.edu.sa

R. Alrazyeg
Saudi Central Bank, Jeddah, Saudi Arabia
e-mail: ralrazyeg@sama.gov.sa

B. Alyoubi et al. (eds.), *Machine Learning and Data Analytics for Solving Business Problems*, Unsupervised and Semi-Supervised Learning,
https://doi.org/10.1007/978-3-031-18483-3_2

Unlike commercial banks which focus on business transactions and profit maximization, social banks focus on the ethical and the social nature of banking. They emphasize their relationship with the client by providing financial and nonfinancial support to develop their businesses [3]. This allows social banks to take the leadership role between financial institutions and individuals, corporations, or organizations which stimulates the economic growth throughout the country and raises the people living level. The spillover effect of enhancing the financial independence of people is extended to communities and societies without neglecting the financial sustainability which will have a broad impact on the whole society [3].

Our work focuses on the main characteristics and profiles of the beneficiaries of social and microfinance loans before and during the COVID-19 pandemic. We investigate any changes in the social, demographic, and economic characteristics of the beneficiaries of social loans during the COVID-19 pandemic. Our study is based on the analysis of social and microloans granted by one of the most important social banking actors in the Kingdom of Saudi Arabia, the Social Development Bank (SDB), during the last 2 years: 2019 and 2020. In fact, the SDB was established in 1971 and is considered a key element in supporting Saudi citizens and Saudi entrepreneurs to build their own business and overcome their financial difficulties. It mainly intends to enhance and improve the financial independence of individuals and families via providing financial and nonfinancial services which allows to build an active community and a productive society at large. A deep analytical analysis of the granted loans by the SDB during the years 2019 (period before the COVID-19 pandemic) and 2020 (period during the COVID-19 pandemic) will be performed. This analytical process investigates the changes in the characteristics (demographic, social, and economic) of the beneficiaries of social loans, subsidies, and microfinance during the last 2 years in Saudi Arabia.

The analysis study focuses upon the repercussions of the COVID-19 pandemic on the characteristics of social and microfinance beneficiaries based on intelligent and recent machine learning and data analysis tools including exploratory bivariate and multivariate techniques. Appropriate and auto-fitted decision tree classification models will be built, evaluated, and interpreted in order to extract patterns and hidden knowledge from the SDB social loan data. The objective is to build the main characteristics (profiles) of the beneficiaries of social and microfinance loans before and during the COVID-19 pandemic. These build profiles will help financial institutions and authorities (banks, government, ministry of finance, etc.) to design more efficient fitted programs and measures for such types of loans.

The rest of this paper is organized as follows. Section 2.2 describes the data analytics methodology performed to identify the main changes of the beneficiaries' characteristics of microloans during the COVID-19 pandemic and gives statistics and an overview of the build data. Then, Sect. 2.3 presents the results of the bivariate exploratory data analytics process performed to identify the main changes based on credit classes for individual and business microcredit separately. Results of the multicriteria analysis study using decision tree classifiers are presented in Sect. 2.4, while the fifth section reports and discusses the main profiles of the beneficiaries before and during the pandemic. Finally, Sect. 2.5 presents the conclusion and future works to improve this study.

2.2 Data Analytics Methodology for the Analysis of Microloans and Beneficiaries' Characteristics

2.2.1 Data Analytics Methodology

Data analytics is a collection of tools, techniques, and fundamental methods and principles which allows the extraction of information and knowledge from data. It entails the use, development, and application of algorithms, procedures, and techniques for improving decision-making by comprehending historical data. Business problems can be modeled as exploratory or predictive problems and then solved using appropriate data analytics methods. The finance domain can be considered one of the world's largest applications of data analytics.

In this study, two types of data analytic techniques are performed. In a first analysis, a descriptive bivariate analysis is performed by studying changes in each variable respecting to credit categories before and during the COVID-19 pandemic. For each credit category, we report a mosaic of a bivariate analysis of the descriptive variables for the years 2019 and 2020, respectively. Our objective is to first analyze social loans based on the credit classes. After that, a second exploratory multicriteria analysis is performed by using machine learning decision trees. This process aims to identify the main changes in the beneficiaries' characteristics by highlighting the beneficiary's profile before and during the pandemic. The identification of the main changes in the characteristics of the beneficiaries is modeled as a binary classification problem where defined classes are the beneficiaries before and during the pandemic. The choice of the best decision tree model, for individual and business credits, is done by evaluating three decision tree algorithms which are respectively J48 [4], J48 Consolidated [5], and REPTree [6]. The selected tree model is the one that gives the minimal classification error.

2.2.2 Data Collection and Preprocessing

Our study is based on the microfinance and social loans granted by the Saudi SDB for the years 2019 and 2020. We collected all published data related to loans and their beneficiaries for the years 2020 and 2019 available on the Saudi open data platform.[1] Two main microcredit categories are considered: individual and business loans. Individual microloans are those provided to individuals to overcome

[1] https://data.gov.sa/Data/en/organization/social_development_bank

Table 2.1 Data description of the SDB microloans for the years 2019 and 2020

	Variable name	Variable type	Variable levels
Demographic	Beneficiary sex	Binary	“Male,” “Female”
	Beneficiary age	Categoric	*“<30,” “>=30,” “>=40,” “>=60”*
Social	Social situation	Categoric	*“Abandoned Woman,” “Divorced,” “Married,” “Single,” “Widower”*
	Special needs	Binary	*Yes, No*
	Fam. members	Categoric	*“<02,” “>=02,” “>=05,” “>=10”*
Economic	Saving credit	Categoric	*Yes, No*
	Inc. amount	Categoric	*“<5000,” “>=10.000,” “>=5000,” “>=7500”*
	Credit Amount	Numeric	*[0,+∞]*
	Credit type	Categoric	*“Individual,” “Business”*
Credit class	Credit class	Categoric	*Individual loans:* 1. Association 2. Cash 3. Family 4. Marriage 5. Private 6. Restoration *Business loans:* 1. Emerging projects 2. Excellence 3. Franchise program 4. Graduate program 5. Private CAB 6. Quaem 7. Solution 8. Vending cars 9. Others

their financial difficulties and improve their social situation. This category of loans is subclassified into six main classes (subcategories) which are respectively as follows: (1) Association, (2) Cash, (3) Family, (4) Marriage, (5) Private, and (6) Restoration. The second category of loans is the category of business microloans which are granted as funding to support a new or an existing business project. The business microloans category is subclassified into nine subcategories as in the following: (1) emerging projects, (2) excellence, (3) franchise program, (4) graduate program, (5) private CAB, (6) Quaem, (7) solution, (8) vending cars, and (9) others. Microloans are described using several descriptive variables of different types including numeric, binary, and categoric. These variables (characteristics) can be classified into demographic, social, and economic characteristics and are described in Table 2.1 within the considered levels for each variable.

2.3 Exploratory Analyses of Microloans Based on Credit Classes

Experiments are conducted on individual and business credits separately. For each credit category, we simultaneously studied the distribution of each variable level and the respective subcategories before and during the COVID-19 pandemic. A mosaic of a two-axes-analysis of each characteristic by the credit categories for the years 2019 and 2020 are reported. Our objective is to investigate any changes in the main characteristics of the beneficiaries of social loans given the wide repercussions of the COVID-19 pandemic on people, society, and the whole economy.

2.3.1 Bivariate Analysis of Individual Credit Characteristics Based on Credit Classes

Figures 2.1 and 2.2 show the distribution of individual loans for the years 2019 and 2020, respectively. Results are schematized for each variable (characteristic) projected on the different credit subcategories which are respectively as follows: (1) Association, (2) Cash, (3) Family, (4) Marriage, (5) Private, and (6) Restoration. In the following, we give a description of each characteristic for the years 2019 (before the pandemic) and 2020 (during the pandemic) simultaneously:

- *Bivariate analysis by beneficiary sex*: We notice from Figs. 2.1a and 2.2a that the total amounts of loans provided to women have increased for the period during the COVID-19 pandemic. These two figures also show that loans granted to males have three dominant classes, which are respectively family, cash, and marriage. However, we show only one category, cash loans, which was mostly granted to females.
- *Bivariate analysis by beneficiaries' age*: We notice from Figs. 2.1b and 2.2b a decrease in the total amounts of social loans granted to the age group over 60 years during the COVID-19 pandemic compared to the period before the pandemic. We also notice that age groups "<30" and "30–40" represent more than 65% of beneficiaries for both years 2019 and 2020. This shows that most of the beneficiaries of microloans are young people which proves the difficulties that the young people find in building their financial independence compared to the other age groups. Most of the granted loans for the age group "<30" were of type "marriage." A decrease is reported during the year 2020 compared to 2019 for this credit category. However, for the age group "40–60," most social loans were of type "family" and have shown an increase during the COVID-19 pandemic compared to the period just before the COVID-19 pandemic.
- *Bivariate analysis by the social situation*: We notice from Figs. 2.1c and 2.2c an increase in total amounts of social loans provided to "single" people for the year 2020 compared to 2019. However, the "married" category remains the majority

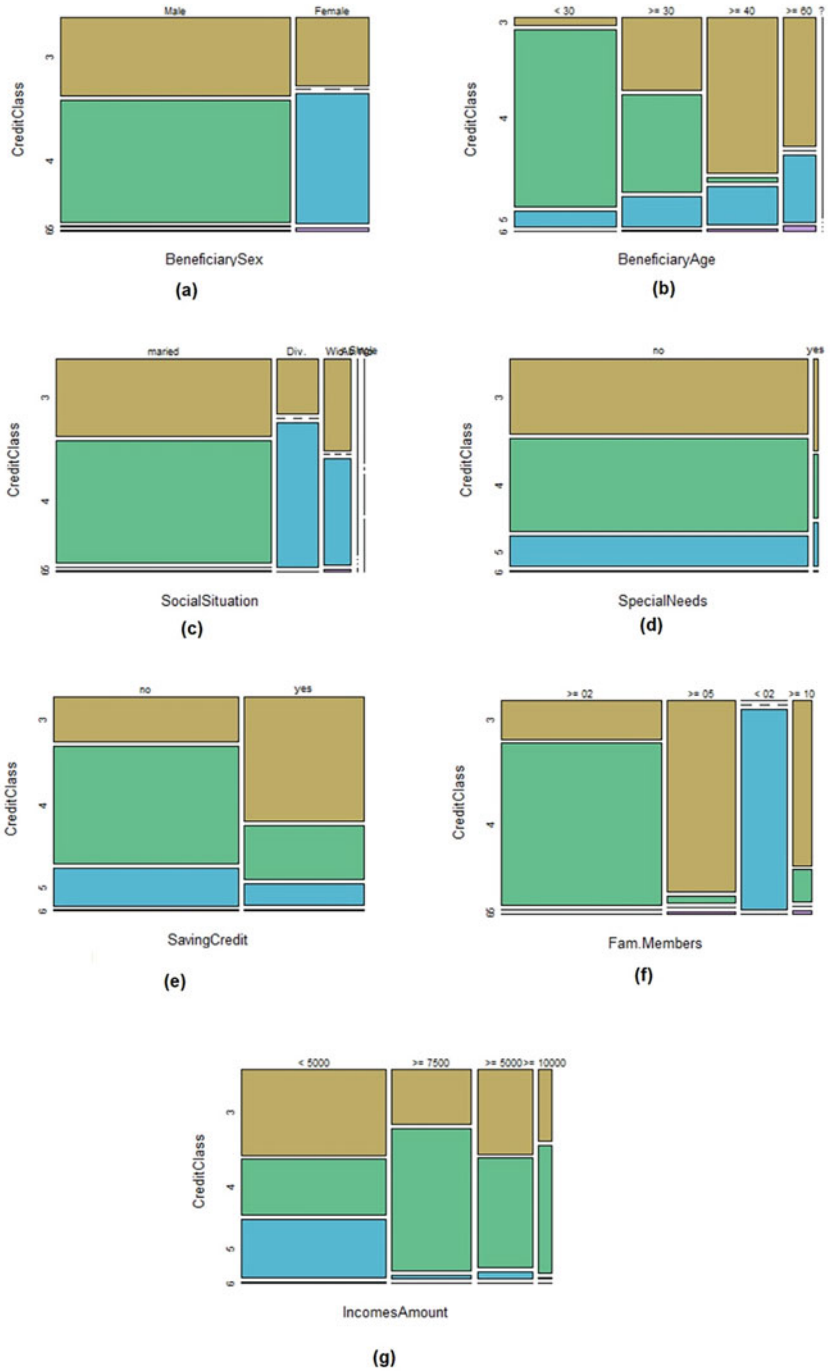


Fig. 2.1 Bivariate analysis of individual microloans' characteristics by credit class before the COVID-19 pandemic (for the year 2019)

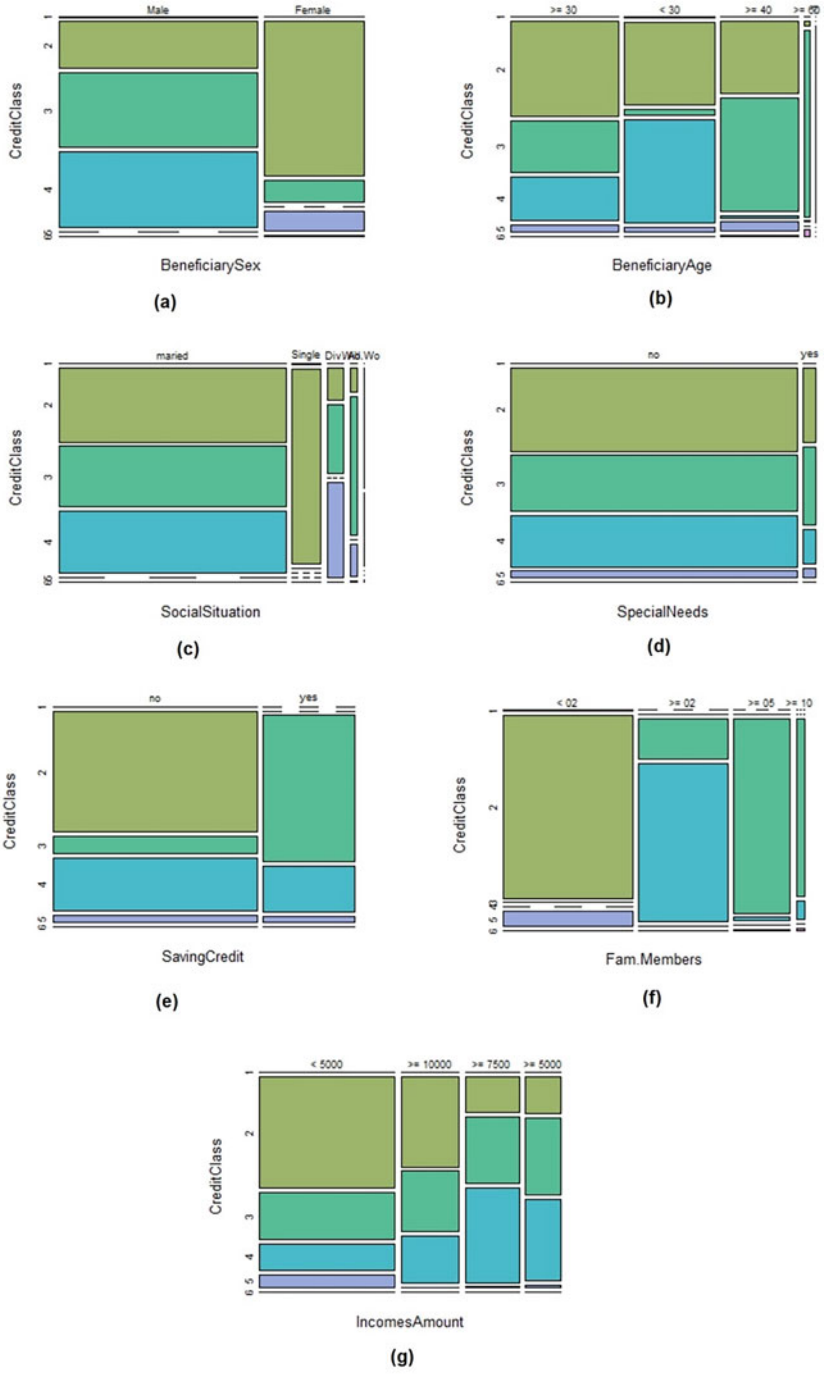


Fig. 2.2 Bivariate analysis of individual microloans' characteristics by credit class during the COVID-19 pandemic (for the year 2020)

group for those benefiting from social loans before and during the pandemic. Furthermore, we notice that “married” beneficiaries mostly benefited from two microloan products before the pandemic (“family” and “cash”), while during the COVID-19 pandemic, we show an increase of “marriage” microloans.

- *Bivariate analysis by special-needs individuals*: We notice from Figs. 2.1d and 2.2d a growth in total amounts of microloans granted to special-needs individuals during the COVID-19 pandemic compared to the period just before the pandemic. We remark that special-needs individuals have benefited from microloan products similarly to normal persons before and during the COVID-19 pandemic. This shows the absence of any customized microloan product for special-needs individuals. However, the increase of microloans granted to this category of people during the COVID-19 pandemic can be explained by the higher priority that was given to special-needs people’s credit requests during the COVID-19 compared to the period just before the pandemic.
- *Bivariate analysis by saving credit*: We notice from Figs. 2.1e and 2.1e a slight decrease in total amounts of social loans granted to individuals having a saving program for the year 2020 compared to 2019. This may be explained by the unexpected financial difficulty of many people during the pandemic given the closure of several commercial activities to fight the spread of this virus. We also show from these figures that clients with a saving program have almost benefited from the “family” microloan product before and during the COVID-19 pandemic. However, we show some changes for mostly commercialized products for the beneficiaries without a saving program during the pandemic. Before the pandemic, most granted loans were “marriage” microloans, while it becomes “cash” microloans during the pandemic.
- *Bivariate analysis by family members*: We notice from Figs. 2.1f and 2.2f a remarkable change in the group of family members who mostly benefited from social loans before and during the COVID-19 pandemic. Before the pandemic, most beneficiaries belonged to the family members category “5–10” while during the pandemic, most beneficiaries belonged to the category “<2.” In 2019, most beneficiaries of the category “5–10” benefited from social loans of type “family,” while in 2020 most beneficiaries of the category “<2” benefited from social loans of type “cash.”
- *Bivariate analysis by income amount*: We show from Figs. 2.1g and 2.2g that the category with monthly incomes less than 5000 SAR is the category that most benefited from microloan products before and during the COVID-19 pandemic. However, we show some changes in types of mostly granted products before and during the CVID-19 pandemic. Before the pandemic, social loans of type “family” were widely granted for the category having an income amount less than 5000 SAR, while during the pandemic we notice two types of loan products that were mostly granted, namely, “family” and “cash.”

2.3.2 *Bivariate Analysis of Business Credits Based on Credit Classes*

Figures 2.3 and 2.4 show the distribution of business credit characteristics before and during the COVID-19 pandemic. Results are reported for each variable in the dataset and analyzed using the nine credit classes which are respectively as follows: (1) emerging projects, (2) excellence, (3) franchise program, (4) graduate program, (5) privateCAB, (6) Quaem, (7) solution, (8) vending cars, and (9) others. We give in the following an analysis of each credit beneficiary characteristic before and during the COVID-19 pandemic. We note here that the variable "Saving credit" is ignored since all business microloans are provided without the requirement of any saving program:

- *Bivariate analysis by beneficiary sex*: We notice from Figs. 2.3a and 2.4a that parts of loans provided to males and females are the same before and during the pandemic. We notice for males that "emerging projects," "private CAB," and "solution" are the three most granted microloan products before and during the pandemic. However, for females, we notice the emergence of a new credit product "Quaem" which has been widely granted to women during the COVID-19 pandemic.
- *Bivariate analysis by beneficiaries' age*: We notice from Figs. 2.3b and 2.4b that there is no change in the distribution of business loans before and during the COVID-19 pandemic based on the beneficiary's age category. There is a balance between three age groups that were mostly benefited from social business loans, which are respectively "<30," "30–40," and "40–60."
- *Bivariate analysis by social situation*: We notice from Figs. 2.3c and 2.4c that there is no change in the distribution of business loans before and during the COVID-19 pandemic based on the social situation. Most of social business loans were provided to "married" and "single" people for both years 2019 and 2020. However, during the COVID-19 pandemic, we notice an increase of total amounts of "private CAB" microloans during the COVID-19 pandemic that were mostly provided to "single" entrepreneurs.
- *Bivariate analysis by special-needs individuals*: We notice from Figs. 2.3d and 2.4d a growth of total amounts of microfinance loans provided to special-needs individuals during the COVID-19 pandemic. Before the pandemic, the most dominated type of business microcredits granted to special-needs individuals was "private CAB." However, during the COVID-19 pandemic, we notice a balance between four microfinance business loans, which are respectively "emerging projects," "graduate program," "private CAB," and "solution."
- *Bivariate analysis by family members:* We do not notice any change in the distribution of business loans based on the family members before and during the COVID-19 pandemic as reported in Figs. 2.3e and 2.4e. However, we notice a major difference between individual and business microloans: we show that more than 80 percent of the beneficiaries of business loans belong to the family

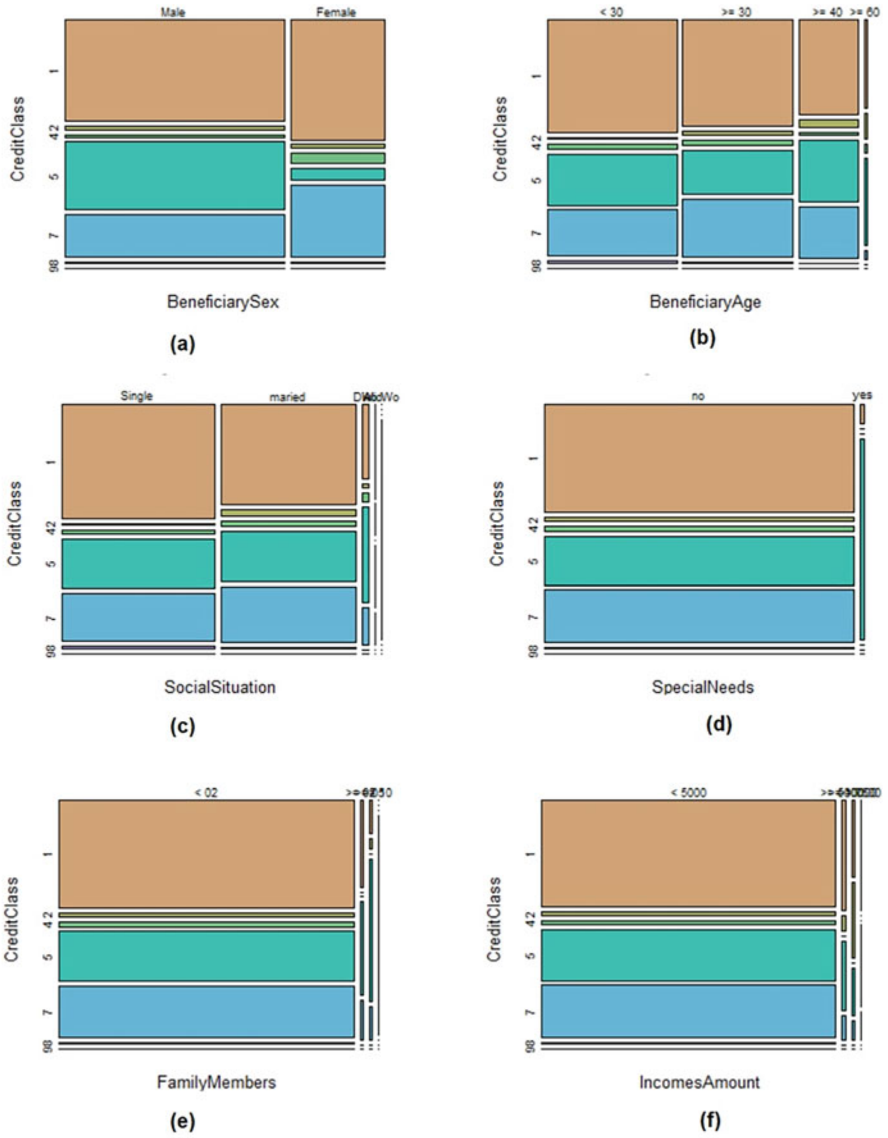


Fig. 2.3 Bivariate analysis of business loans' characteristics by credit classes before the COVID-19 pandemic (for the year 2019)

members' category "<2." This is different for individual social loans where it was a balance between three family members' categories which are respectively "<2," "2–5," and "5–10." This show that most of the beneficiaries of business microloans have a small family member.

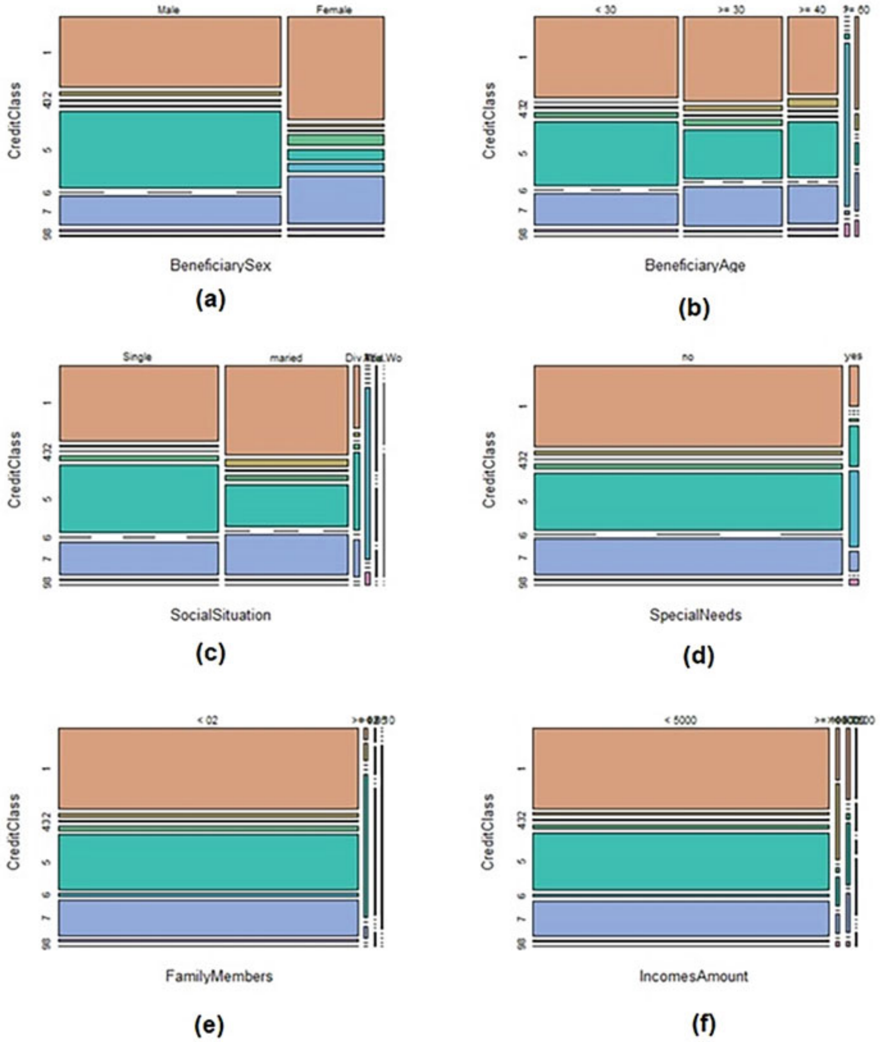


Fig. 2.4 Bivariate analysis of business loans' characteristics by credit classes during the COVID-19 pandemic (for the year 2020)

- *Bivariate analysis by income amount*: We do not notice any change in the distribution of business loans based on the income amount variable before and during the COVID-19 pandemic as reported in Figs. 2.3e and 2.4e. For both periods, the category with a monthly income less than 5000 SAR benefited more than 85% of the total amount of business loans. The three most dominated classes of business credits were "emerging projects," "private CAB," and "solution."

2.4 Identification of the Main Changes in the Characteristics of Microloans Before and During the COVID-19 Pandemic Using Decision Tree Classifiers

In this section, we designed a decision tree-based model to automatically identify main characteristics (profiling) of the beneficiaries of individual and business microfinancing before and during the COVID-19 pandemic. First, we give a brief description of decision trees, and then, we present the best obtained trees for individual and business microfinancing and the different interpretations and discussions.

2.4.1 Decision Tree: A Machine Learning Model to Solve Complex Classification Problems

A decision tree is a reliable and effective machine learning and data mining technique that provides high classification accuracy with a simple representation of gathered knowledge [7]. The decision tree consists of a set of nodes that form a rooted tree. It begins from a node called "root" with no incoming edges, while the other nodes have exactly one incoming edge. A node with outgoing edges is called test node, while the others are called leaves. The leaf nodes are considered terminal nodes representing the most appropriate target values (classes). In a decision tree, each test node splits the instance space into two or more subspaces according to a certain discrete function of the input attribute values. Instances are classified by navigating them from the root of the tree down to a leaf, according to the outcome of the tests along the path.

Several decision tree algorithms were proposed in the literature such as ID3 [8] C4.5 [4], random forest [9], and many others. Among these algorithms, ID3 is considered the basic one which can only deal with categoric data [10]. Quinlan solved this problem and proposed C4.5, also known as J48, which is a tree-based classification algorithm extending the ID3 algorithm to deal with both categoric and numeric data. It uses the gain ratio for building the tree by dividing the values of a numeric continuous attribute into two subsets. Another well-known decision tree algorithm is the random forest, which builds random trees from a given dataset. Random forest combines multiple decision trees which are merged for more accurate classification. Random forest is based on the idea that multiple uncorrelated models perform much better as a group than they do alone. When using random forest for classification, each tree gives a classification considered as a "vote." The forest chooses the classification with the majority of the "votes."

Given the simplicity and the interpretability of decision trees, they are used in a wide range of industries and disciplines such as in healthcare industries [11, 12] and in the banking sector [13–15]. In healthcare industries, decision trees can tell whether a patient is suffering from a disease or not based on conditions such as age, weight, sex, and other factors. In the banking sector, decision trees are usually used to decide if a person is eligible for a loan or not based on his financial status, family member, salary, etc. Other applications may include credit card frauds [14], bank schemes and offers [16], and loan defaults [13] which can be prevented by using a proper decision tree. We show in the next section how decision trees are used to build profiles of the beneficiaries of each credit category.

2.4.2 Experiments Design and Empirical Results

We performed two multivariate analyses on individuals and business microcredits separately before and during the COVID-19 pandemic (for the years 2019 and 2020, respectively). We modeled the classification process as a binary classification problem, for individual and business microfinancing separately, containing two categoric classes: *Individual2019* and *Individual2020* for individual social loans provided before and during the COVID-19 pandemic and *Business2019* and *Business2020* for business microfinancing granted before and during the COVID-19 pandemic. Most important characteristics of each credit class can be interpreted from the tree by describing paths toward the leaves of the tree representing credit beneficiaries' characteristics either before or after the pandemic. In order to build the best decision tree model for individual and business credits, we evaluated three types of algorithms for each credit category separately. The evaluated machine learning algorithm decision tree algorithms are J48 [4], J48 Consolidated [4], and REPTree [6] with a maximum depth equal to 30 for all algorithms. This limit is configured to make the tree easily interpretable when visualized. The objective is to determine the best tree in terms of classification error which can help to give visual profiles of the beneficiaries of social loans. Table 2.2 reports the classification error of each evaluated algorithm for the years 2019 and 2020. This table shows that there is no algorithm that outperforms the others on both datasets and for all years despite the good result of the consolidated J48 algorithm. In the following, we discuss the best obtained decision trees for each period and for each credit class.

Figures 2.5 and 2.6 report the best obtained trees in terms of classification errors for individual and business microcredits separately. We present in the following

Table 2.2 Best obtained classification errors for the evaluated classification algorithms build on the SDB data using a ten-time cross validation

Credit type	Year	Algorithm	Minimal classification error
Individual	2019/2020	J48	22.45%
		J48 Cons.	**21.37%**
		REPTree	22.56%
Business	2019/2020	J48	24.89%
		J48 Cons.	25.15%
		REPTree	**24.16%**

important findings that can be interpreted from each figure. In the first figure, we show that during the COVID-19 period, new microcredit products have been largely granted to the beneficiaries estimated by 27% of all granted microloans during the two studied years 2019 and 2020 (node 1 in Fig. 2.5). These new microloan products are "cash" and "association." These two individual microloan products are targeting citizens who are estimated able to work and did not find a job opportunity or citizens with high professional skills who want to increase their monthly income for personal reasons. Second, we show that beneficiaries of usual microcredit products (family, marriage, private, and restoration) during the COVID-19 period are characterized by their high monthly incomes (greater than 10.000 SAR). The beneficiaries of usual microcredits having a monthly income greater than 10.000 SAR is estimated by 82% during the COVID-19 pandemic compared to only 18% before the pandemic. This shows that the health crisis has aggravated the financial difficulties of individuals even for the segment whose monthly income exceeds 10.000 SAR. The third finding that can be interpreted from Fig. 2.5 is the young age of the beneficiaries of usual loans, whose income is less than 7.500 SAR during the COVID-19 period. We show in node 8 that the percentage of the beneficiaries whose age is over 60 is estimated by 72% before the pandemic compared to only 28% during the pandemic. As a result, we can clearly observe that the build tree contains social, demographic, and economic characteristics which indicate the existence of remarkable changes in these characteristics for the beneficiaries of social microloans during the COVID-19 pandemic period compared to the period just before.

Concerning the analysis of business microcredits, Fig. 2.6 firstly shows that the loan "solution" is no longer available during the COVID-19 period as reported in node 2. Although 11% of the beneficiaries of corporate microloans have benefited from this loan in 2019, it is no longer commercialized by the bank during the year 2020. The "solution" loan had the objective to finance resettlement projects which provide financial and nonfinancial services for 12 business activities that started in early 2018. The funding ranged between 50.000 SAR and 1 million SAR. This can be explained by the priority of the bank to finance other freelancer projects during the COVID-19 period. For example, new corporate microloan products have been largely provided during the year 2020 such "FranchiseProgram" and "Quaem" which are both estimated by nearly 10% of corporate loans provided during the COVID-19 pandemic as reported in nodes 3 and 7. The "FranchiseProgram" program is provided to qualified citizens to invest in small-sized business activities with high success rates and low risk ratios in coordination with national and international brands to grant a commercial franchise.

The second finding that can be stated from Fig. 2.6 is the increase of credit amounts provided during the COVID-19 period compared to those granted before the pandemic. We show from nodes 3, 6, and 12 that nearly 93% of corporate loans with an amount less than 78.000 SAR and having the category "emerging projects," "excellence," "graduate program," "private CAB," and "vending cars" were granted before the pandemic. Node 55 confirms these results and reports that 97% of microloans with large amount exceeding 101.000 SAR were granted during the COVID-19 pandemic. This can be explained by the exceptional efforts of banks and

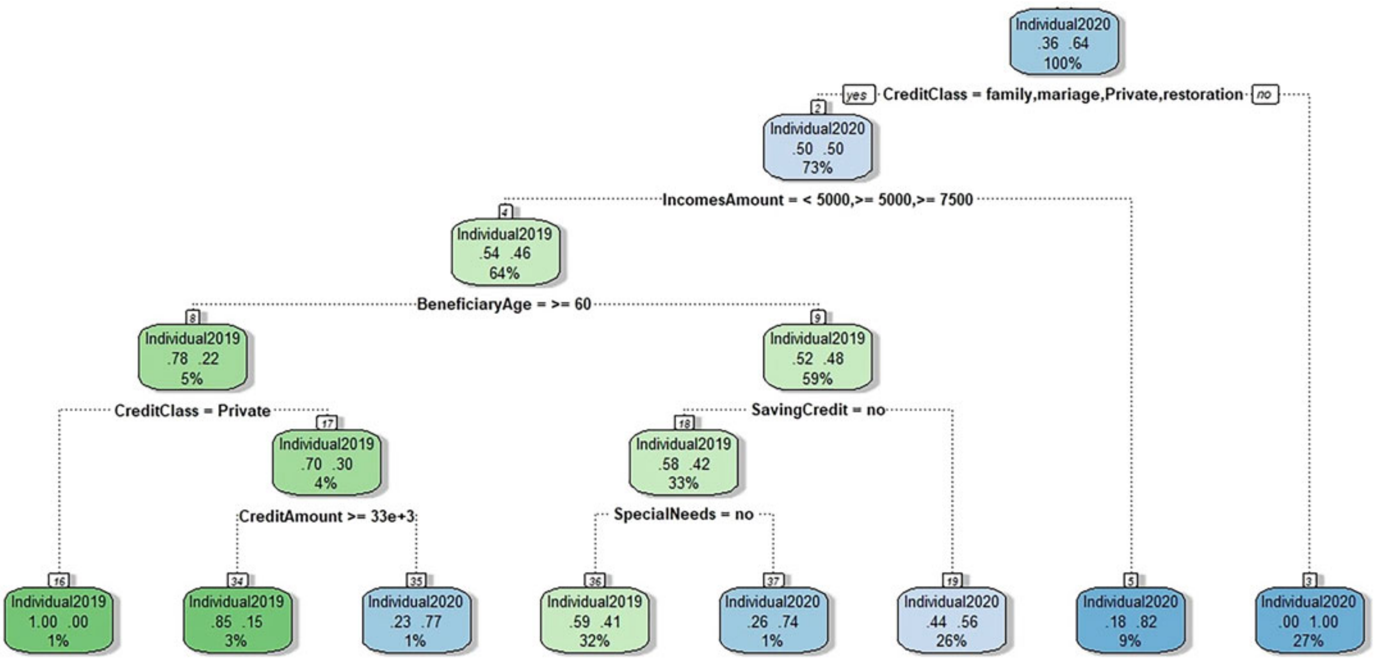


Fig. 2.5 Decision tree of the classification of business microloans based on demographic, social, and economic credit beneficiaries' characteristics

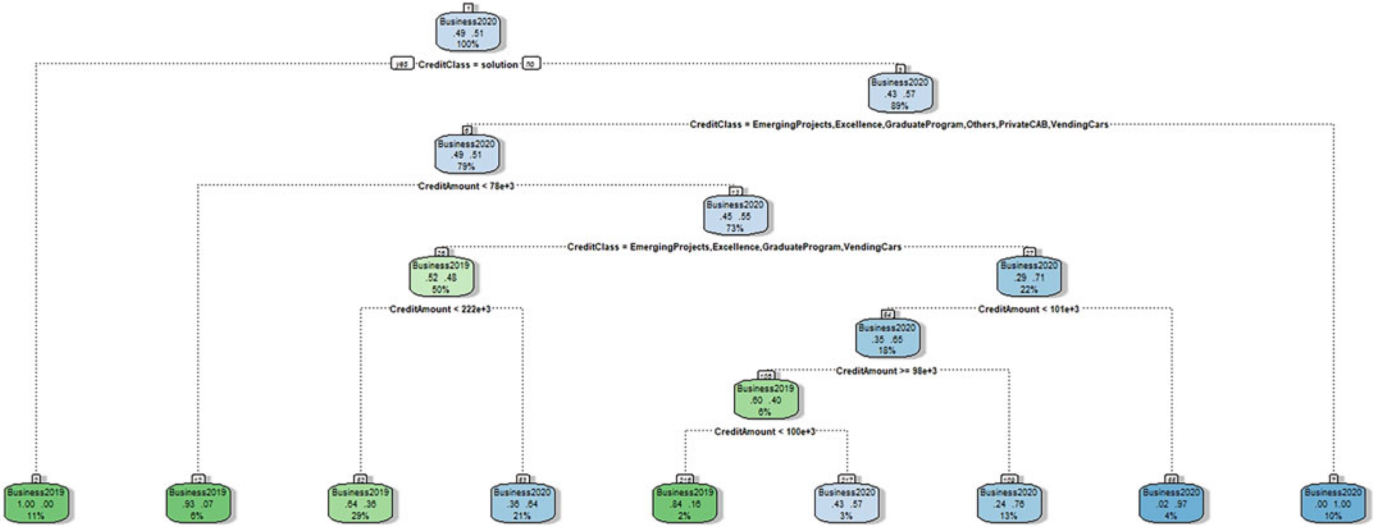


Fig. 2.6 Decision tree of the classification of business microloans based on demographic, social, and economic credit beneficiaries’ characteristics

government organizations to support microprojects and entrepreneurs to overcome the financial repercussions of the COVID-19 pandemic. A third interpretation that can be stated from Fig. 2.6 is the absence of any demographic and social characteristics in the build tree. This indicates that there was no noticeable change of demographic and social characteristics of corporate microcredit beneficiaries during the COVID-19 pandemic. The noticeable changes are observed only for economic characteristics mainly the credit amount and the credit type characteristics.

2.5 Conclusion and Perspectives

We presented in this paper a data-driven analysis of social and microfinance loans before and during the COVID-19 pandemic. Changes in social, demographic, and economic characteristics of the beneficiaries of individual and business microloans were investigated based on exploratory binary data analysis and multivariate decision tree classifiers. We separately analyzed individual and corporate microfinance loans provided by the Saudi Social Development Bank before and during the COVID-19 pandemic. Concerning individual social loans, this study has shown the financial difficulties that young people with reduced number of family member find in building their financial independence during the COVID-19 period compared to the period just before. In addition, this study has shown the financial difficulties of the category of people having large incomes (exceed 10.000 SAR) during the COVID-19 pandemic. An important increase of the percentage of people whose monthly incomes exceed 10.000 SAR has been shown. This fact indicates the financial difficulties found by people having a medium monthly income during the pandemic, who resorted to borrowing to overcome their financial difficulties. On the other side, the analysis of business microcredits does not show important changes of the beneficiaries' characteristics compared to those identified for individual microloans. We do no show noticeable changes in the demographic and social characteristics of the beneficiaries of such microloans before and during the pandemic. However, an increase in the number and amounts of corporate microcredits has been shown. This may indicate the exceptional government effort to support entrepreneurs and microprojects to overcome the repercussions of the COVID-19 crisis by providing more financial help to small and medium enterprises.

Our study is limited to social and microfinance loans granted by the Saudi SDB. However, other non-banking financial institutions, accredited by the Saudi Central Bank, play a very important role in supporting citizens and small-to-medium-sized enterprises. An interesting way to improve this study is to extend these analyses to the loans granted by these microfinancial firms. Another interesting way to improve this study is to include quality service variables in order to evaluate the quality of services and the facilities that social banks offer for the beneficiaries. This will emphasize on the new digital microfinance platforms which facilitates the credit processes and then increased the number of consumers benefiting from microfinancing services.

References

1. O. Weber, S. Remer, Social banking – Introduction, in *Social Banks and the Future of Sustainable Finance*, ed. by O. Weber, S. Remer, (Routledge, London, 2011), pp. 1–14
2. O. Weber, Social banking: Concept, definitions and practice. Global Soc. Pol. Interdiscipl. J. Publ. Pol. Soc. Dev **14**(2), 265–267 (2016). https://doi.org/10.1177/1468018114539864
3. D.W. Ariani, Social capital moderating roles towards relationship of motives, personality and organizational citizenship behaviour: Cases in Indonesian banking industry. South-East Asian J. Manag **4**(2) (2016). https://doi.org/10.21002/seam.v4i2.5637
4. J.R. Quinlan, Improved use of continuous attributes in C 4.5. J. Artif. Intell. Res. **4**, 77–90 (1996)
5. I. Ibarguren, J.M. Pérez, J. Muguerza, I. Gurrutxaga, O. Arbelaitz, Coverage-based resampling: Building robust consolidated decision trees. Knowl. Base Syst **79**, 51–67 (2015). https://doi.org/10.1016/j.knosys.2014.12.023
6. Weka official documentation. (2021). Available online at: https://weka.sourceforge.io/doc.dev/weka/classifiers/trees/REPTree.html
7. S. Sathyadevan, R.R. Nair, Comparative analysis of Decision Tree Algorithms: ID3, C4.5 and random forest, in *Computational Intelligence in Data Mining - Volume 1. Smart Innovation, Systems and Technologies*, vol. 31, (Springer, New Delhi, 2015) https://doi.org/10.1007/978-81-322-2205-7_51
8. J.R. Quinlan, Induction of decision trees. Mach. Learn. **1**, 81–106 (1986)
9. S. Rajesh, A. Chandrasekar, Esteemed software patterns for banking system. Clust. Comput. **22**, 11087–11099 (2019) https://doi-org.sdl.idm.oclc.org/10.1007/s10586-017-1304-7
10. I. Chaabane, R. Guermazi, M. Hammami, Enhancing techniques for learning decision trees from imbalanced data. Adv. Data Anal. Classif **14**, 677–745 (2020) https://doi-org.sdl.idm.oclc.org/10.1007/s11634-019-00354-x
11. D. Saraswat, P. Singh, Comparison of different decision tree algorithms for predicting the heart disease, in *Communications in Computer and Information Science*, vol. 1241, (Springer, Singapore, 2020). https://doi-org.sdl.idm.oclc.org/10.1007/978-981-15-6318-8_21
12. M. Teixeira Cazzolato, M. Xavier Ribeiro, C. Yaguinuma, M. Terezinha Prado Santos, A statistical decision tree algorithm for data stream classification, in *Proceedings of the 15th International Conference on Enterprise Information Systems - Volume 3*, (ICEIS, 2013), pp. 217–223. https://doi.org/10.5220/0004447202170223
13. S. Bhatore, L. Mohan, Y.R. Reddy, Machine learning techniques for credit risk evaluation: A systematic literature review. J. Bank Financ. Technol **4**, 111–138 (2020) https://doi-org.sdl.idm.oclc.org/10.1007/s42786-020-00020-3
14. A.G.C. de Sá, A.C.M. Pereira, G.L. Pappa, A customized classification algorithm for credit card fraud detection. Eng. Appl. Artif. Intell **72**, 21–29 (2018). https://doi.org/10.1016/j.engappai.2018.03.011
15. J.V. Devi, K.S. Kavitha, Fraud detection in credit card transactions by using classification algorithms, in *Proceedings of 2017 International Conference on Current Trends in Computer, Electrical, Electronics and Communication (CTCEEC)*, (IEEE, Mysore, INDIA, 2017), pp. 125–131
16. O. Ozgur, E.T. Karagol, F.C. Ozbugday, Machine learning approach to drivers of bank lending: Evidence from an emerging economy. Finan. Innov. **7**(20) (2021) https://doi.org/10.1186/s40854-021-00237-1

Chapter 3
Identification of Credit Risks Using Cluster Analysis and Behavioural Scoring During the COVID-19 Pandemic

Waad Bouaguel and Taghrid Al silimani

3.1 Introduction

With the appearance of the new coronavirus pandemic near the beginning of 2020, the Saudi banking sector was opposed to significant challenges. The consequences of the Covid-19 pandemic on countries' economies, and in particular on banking and financial sectors are still unclear. Hence, in order to avoid side effects of coronavirus on the Saudi economy, the Saudi government initiated a set of proactive support packages and initiative measures covering various economic sectors with issues related to financing, employment, tax payment deferrals, and a moratorium on debt repayments [1–5].

Banks facilitated the government and SAMA's financial support mechanisms, while at the same time weathering their own storms though business continuity management systems. Indeed, banks can only be part of the solution if they have enough capital to absorb losses, but an increase in credit risk could put pressure on their capital positions. Such an increase could arise from the uncertain economic situation caused by the pandemic, which could leave debtors struggling to pay back their loans. Banks' balance sheets with high amounts of non-performing loans would undermine the economic recovery. Hence, proactive and strong credit risk management practices are vital.

W. Bouaguel (✉)
University of Jeddah, Jeddah, Saudi Arabia

LARODEC, ISG, University of Tunis, Tunis, Tunisia
e-mail: wabouaguel@uj.edu.sa

T. Al silimani
University of Jeddah, Jeddah, Saudi Arabia
e-mail: tsalsilimani@uj.edu.sa

B. Alyoubi et al. (eds.), *Machine Learning and Data Analytics for Solving Business Problems*, Unsupervised and Semi-Supervised Learning,
https://doi.org/10.1007/978-3-031-18483-3_3

In order to avoid credit risk, banks should have a strategy in place to engage with borrowers as soon as they show signs of distress, and any issues should be adequately addressed. Good strategies can only be developed if banks are able to differentiate between viable, non-viable, and viable but distressed debtors at a granular level, grouping borrowers with similar characteristics and resolving them comparably. Behavioural scoring is one way to study the risk levels for defaulting credit accounts and to help banks in implementing the right policies at the right time [6]. In this kind of scoring, we try to find decisions for already existing customers with credit accounts [7]. Hence, we look for robust credit scoring models that can analyse the behaviour of already acquired credit customers [6]. Thus, behavioural scores can be used for various business purposes. In this paper, we show how to use a mixture of machine learning algorithms in order to take behavioural scoring into the next level, namely we present a new strategy for building behavioural scorecard by using cluster analysis of account behaviours in the context of Covid-19 pandemic in Saudi banking sector. The proposed approach combines both supervised and unsupervised learning. On the one hand, clustering analysis is first used by banks and financial organizations in order to identify particular segments of their customers. Then, for each customer segment, a special marketing policy will be developed. On the other hand, supervised learning will be used in order to mine the meaningful patterns in the data segments and produce new rules representing the characteristics of each customer segment.

The remainder of this paper is organized as follows. In Sect. 3.2 we discuss related works on behaviour scoring and machine learning usage for building credit behavioural scorecards. Then, in Sect. 3.3 we introduce the basic steps of our proposed approach for identifying credit risk using cluster analysis of account behaviours and supervised learning. In Sect. 3.4, the empirical framework and data collection methods are presented. Then, Sect. 3.5 summarizes the empirical finding, and, finally, Sect. 3.6 outlines the research conclusions and future directions.

3.2 Related Works on Behavioural Scoring and Machine Learning Usage for Building Credit Behavioural Scorecards

Maintaining creditworthy customers is critical to banks and financial institutions. Not all customers behave similarly regarding financial behaviour; hence different strategies should be prepared to each group of customers based on their repayment behaviour [8]. Many researchers focused on behavioural scoring, such as [9] who proposed a behavioural scoring model that classify customers into high or low contribution customers based on data analysis. This later research was extended by and [10] were the paper presents a two-stage cardholder behavioural scoring model, with merits of artificial neural networks and data analysis. Other researchers, such as [7], that presented a complete survey of credit and behavioural scoring related to forecasting the financial risks of lending consumers. [11] studied how to build behavioural scorecards with machine learning components. Three well known machine learning approaches were used: linear discriminant analysis, back propagation neural networks, and support vector machine (SVM).The proposed study explores the performance of behavioural scoring using machine learning methods and demonstrated the effectiveness of behavioural scoring using the presented techniques.

Machine Learning offers various algorithms and methods that can be used for building behavioural scorecards. There are many advantages of applying machine learning methods for behavioural scorecards, such as increasing models accuracy and capturing hidden relation in data even the non-linear one. The need for machine learning algorithms become a necessity in the last few years. Indeed, the huge number of loan applicants data produce a large quantity of information to be processed. Hence, it is often difficult to use traditional based mathematical and statistical scorecards to handle this large quantity of information. Then, machine learning is the solution. Behavioural analytic use machine learning to study a customer behaviour at a highly detailed level across each aspect of a transaction [12]. The information is tracked in profiles that represent the behaviours of each individual, merchant, account, and device. These profiles are updated with each transaction, in real time, in order to compute analytic characteristics that provide informed predictions of future behaviour [13].

3.3 Identification of Credit Risk on the Basis of Cluster Analysis of Account Behaviours and Supervised Learning

Identifying different risk levels is important, in order to choose the best financial policies. When financial institutions understand the value of credit applicants, they can provide customized products and services to different categories of credit applicants. A common way of doing this is by developing different clusters of credit accounts based on a cluster analysis of the behaviour of credit accounts, in a way that the credits applicants that are in the same cluster have the same behaviours. According to [14], clustering methods are commonly classified into the following categories: hierarchical methods, partitioning methods, density-based methods, and grid-based methods. The K-means is one the simplest clustering algorithm used to find the best clusters in a fast way. K-means belong to partitioning clustering methods, in this category, we divide the data into groups of k disjoint clusters and then, a criterion function is iteratively optimized by moving observations from one cluster to another. K-means method can be used by financial institutions for segmenting their customers and to develop marketing strategies for each segment. However, after applying the K-means clustering algorithm on a dataset, it is difficult for one to interpret and to extract required results from these clusters. Hence, decision trees (DT) could be merged with K-means method to produce meaningful rules for each customer segment. J48 DT algorithm is then used for the interpretation of the clusters produced by K-means algorithm because of its speed, simplicity, and capability to generate understandable rules. In the following, the different steps of our proposed approach are given.

- Step I: data about accounts behaviours before and during the pandemic is segmented using K-means algorithm to three clusters: credit applicants with high creditworthiness, credit applicants with middle creditworthiness, and credit applicants with low creditworthiness.
- Step II: DT model is built over the clusters obtained in Step I.
- Step III: Interpretation of DT rules and distinction of the main characteristics of the clusters that drive credit risk prior and during the pandemic.

3.4 Empirical Framework and Data Collection

3.4.1 Data Set Description

To test the validity of the proposed model, ten datasets representing several banks operating in Saudi Arabia under the supervision of SAMA were used. For each bank we randomly extracted 1000 instances. The study period spans two years before the pandemic (2018 and 2019) and the pandemic year itself (2020). The total number of instances in all data sets is more than 35,000, with about 14 qualitative features describing each credit applicant, such as the region, housing type, personal status, etc. We also used three numerical features that provide information on the credit applicant's total loan amount and outstanding credit amount with the binary categorical target variable "credit default" (for more details, see Tables 3.1 and 3.2).

After collecting data we move to the first stage of data exploration in which we perform an univariate descriptive analysis of all the variables of interest. In this stage we try to define and summarize the data and find the related hidden patterns.

The histograms shown in Fig. 3.1 display the proportions of loans by credit default, type of housing, type of property, and region. It can be observed that the number of awarded credits has decreased from 2018 to 2020. It can also be observed from Fig. 3.1 that the number of credits with default has increased during the pandemic. In fact, approximately 45% of the defaulting credit was observed in 2020, the year of the pandemic. One explanation is that during the pandemic, credit consumers shifted to personal loans when they needed cash, making this type of financing one of the fastest-growing forms of debt. However, with millions of Saudi citizens and residents losing their jobs, it became even more difficult to afford monthly payment obligations. Figure 3.1 shows It can be noticed that Riyadh, Makkah, and Eastern Province regions have the most significant proportion of credit borrowers. We can also notice that the majority of the credit applicants live in villas and in an owned house.

Table 3.1 Description of all features and univariate analysis

Feature	Input
Credit default	Yes
	No
Region	Riyadh
	Makkah
	Madinah
	Qassim
	Eastern Province
	Asir
	Tabuk
	Hail
	Northern Borders
	Jazan
	Najran
	Al Bahah
	Al-Jawf
Housing type	Rented
	Owned
	Other
Property type	Apartment
	Villa
	Other
Credit history	No credits were taken before
	All past credits have been paid back duly
	Existing credits are being paid duly until now
	There was a delay in paying off in the past
Age	25
	25–35
	36–45
	46–55
	56–65
	>65
Total monthly income	5K
	5K–10K
	11K–15K
	16K–20K
	21K–25K
	>25K
Job	Government
	Private
	Retired

Table 3.2 Description of all features (continue)

Feature	Input
Loan toner	12 months
	13–24 months
	25–36 months
	37–48 months
	49–60 months
	>60 months
Purpose of personal loan	Renovation & home improvement
	Cars & automobiles financing
	Furniture and consumer durable and goods
	Education
	Health care
	Travel & tourism
	Other personal loans
Gender	Male
	Female
Personal status	Divorced
	Married
	Widowed
	Single
	Other
Nationality	Saudi
	Non-Saudi
Number of dependants	0
	1
	2–4
	5–9
	≥10
Type of collateral	None
	Salary transfers
	Cash collateral
	Other
Total loan amount (including profits)	Numeric value that describes the total size of the loan granted, including profits
Outstanding credit amount	Numeric value that describes the outstanding loan value

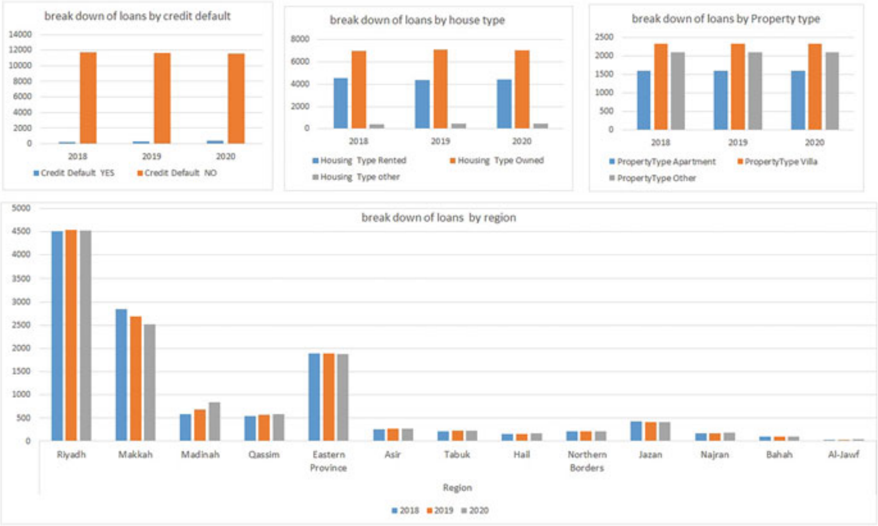


Fig. 3.1 Break down of loan by credit default, house type, property type and region

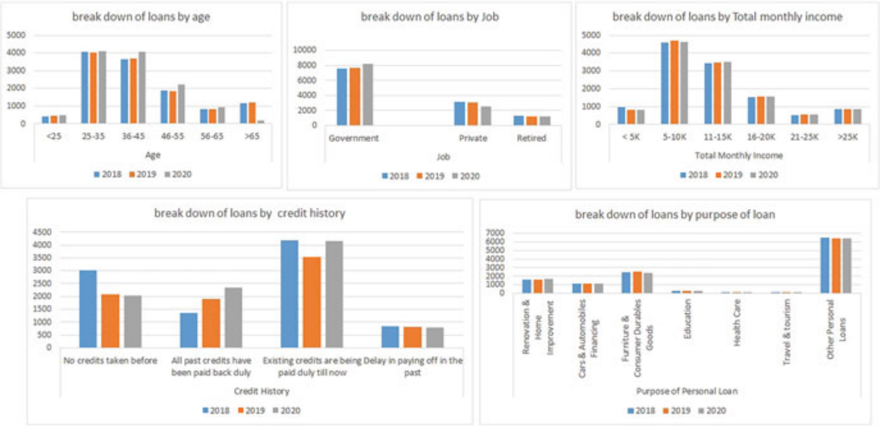


Fig. 3.2 Break down of loan by age, job, total monthly income, credit history and purpose of loan

Figure 3.2 shows that borrowers aged between 25 and 35 obtained a significant number of credits, followed by the category aged between 36 and 45. The same was observed for the total monthly income, wherein the largest category of borrowers was those with an income between SAR 5K and SAR 10K. From Fig. 3.2 we notice that borrowers with government job have the biggest chair of credits. This chair have increased from 2018 to 2020. On the contrary Fig. 3.2 shows that the number of applicants from the private sector has decreased during the pandemic compared to the pandemic period, meaning that the COVID-19 pandemic have impacted the

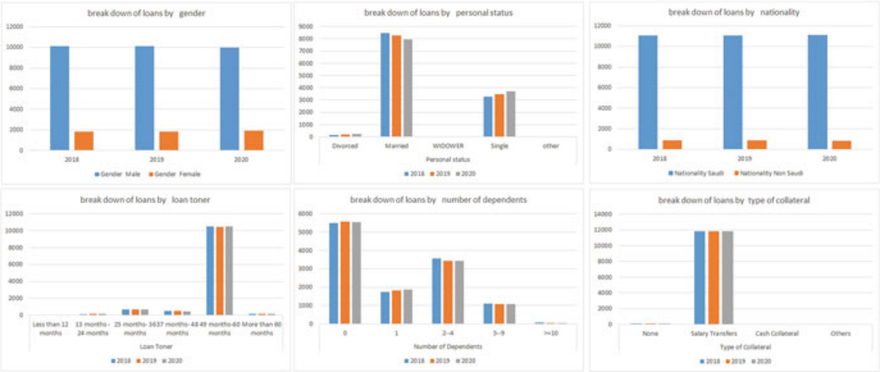


Fig. 3.3 Break down of loan by gender, personal status, nationality, number of dependant, loan toner and type of collateral

borrowers from the private sector. It is also obvious from Fig. 3.2 that the majority of borrowers are those with existing credits that are being paid duly until now. In addition, we found that approximately 6000 credits from 2018 to 2020 under the label other personal loan.

From Fig. 3.3, we notice that the majority of borrowers are males. In fact the credit behaviour related to gender did not change from 2018 to 2020. We also noticed that the credit behaviour related to gender did not change before and during the pandemic. Approximately 90% of the credit applicants before and during the pandemic are Saudi citizens where the majority are married. Approximately more than the half of borrowers have no dependants under their responsibility. In addition, it can be observed that the majority of loans before and during the COVID-19 pandemic were taken for a repayment period that varies between 49 and 60 months.

3.5 Empirical Results

In this section we present the empirical results for our proposed approach. First we use k-means clustering with WEKA software. All defaulting instances were grouped in one single data set. Then, the variable credit default was removed in order to perform clustering task. We use then K-means algorithm implementation to cluster the defaulting borrowers in segments. The WEKA implementation of K-Means algorithm uses the Euclidean distance to measure the distances between the different instances in the data set. The obtained result in WEKA shows the centroid of each cluster along with the number of instances in different clusters. When we use K-means algorithms for clustering, the choice of appropriate number of clusters is not a simple task. While there are some existing approaches that can help with this task, the choice of the number of cluster is out of the scoop of this paper. Hence, we manually fixed to three as the final number of clusters.

Table 3.3 Clustering setting using WEKA software

Clustering method	2018/2019	2020
Number of clusters	3	
Initial starting points	Random	
Percentage of instances in each cluster	C 0 (25%)	C 0 (31%)
	C 1 (44%)	C 1 (42%)
	C 2 (31%)	C 2 (27%)

Table 3.3 gives the distribution of instances over the three clusters obtained when the k-means algorithm was run on the data collected before and during the pandemic. We notice from Table 3.3 that before the pandemic 44% of the defaulting borrowers belong to cluster C1, 32% belong to cluster C2, and 25% belong to cluster C0. For the pandemic period the cluster C1 contains 42% of the defaulting borrowers. For the clusters C0 and C2 we notice a little change from the results obtained before the pandemic where the percentage of borrowers in the cluster C0 moved from 25% before the pandemic to 31% during the pandemic and C2 moved from 31% to 25%.

Once all new groups of credit applicants are obtained we move to studying the characteristics of each group by using the J48 algorithm implementation in WEKA software using the weka.classifiers.trees.J48 classifier class. Figures 3.4 and 3.5 show, respectively, the obtained tree for the period before and during the pandemic. Both figures show that the root attributed to this model is personal status. For any decision tree, we can read off an equivalent set of rules, we just go along the leaves in Figs. 3.4 and 3.5. Then for each leave, we just read off the conditions above that leaf get you from the root to that leaf. After removing the empty leaves, the rule set generated from each tree are given in Tables 3.4 and 3.5.

From Table 3.4 we can notice that Personal status, outstanding credit amount, purpose of loan, housing type, number of dependants, total loan amount and job are the most present features on the obtained rules. For the first cluster C0 we notice that all credit applicants in this category are married. The purpose of loan is on of this reason “renovation and home innovation,” “furniture and consumer durable and goods,” or “cars and automobiles financing.” For the second cluster C1 we found that the credit applicants in this cluster could be married peoples asking for a cars and automobiles financing or they may be singles with a outstanding credit amount ≤140,545. For the third cluster C2 we notice that if the credit applicants in this segment were married they generally apply for two purpose of loan: “furniture and consumer durable and goods” or “other personal loans.” If the credit applicants are singles they generally live in an owned house and have outstanding credit amount ≤140,545.

From the previous rules, we can conclude that the most important characteristics of credit applicants before the pandemic are related to their personal status and the purpose of the loan. The results show that most of the applicants were married and that the purpose of the loan in most cases was to buy furniture and consumer durable and goods.

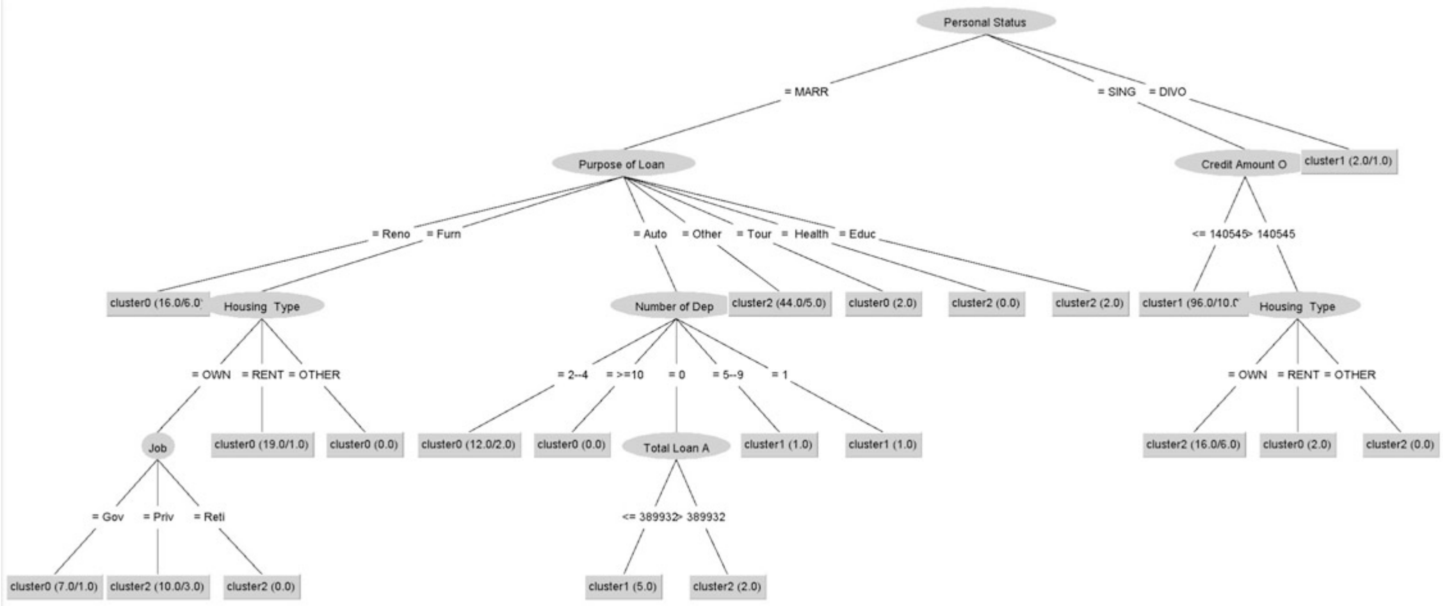


Fig. 3.4 Decision tree produced using WEKA for the period before the Covid-19 pandemic

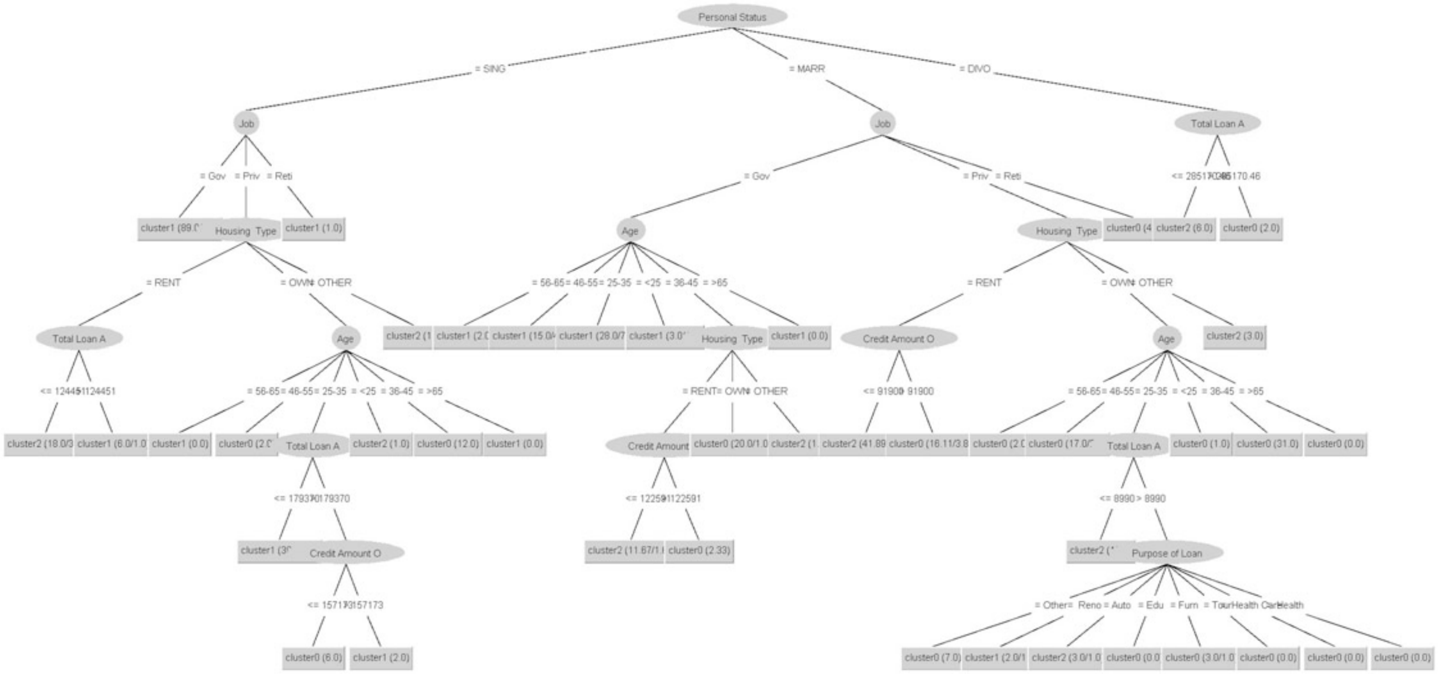


Fig. 3.5 Decision tree produced using WEKA during the Covid-19 pandemic period

Table 3.4 Rule set generated from the decision tree of Fig. 3.4

Applicants belonging to the first cluster (C0) were found to follow one of the following rules:
IF Personal status = married **AND** purpose of loan = renovation and home innovation
IF Personal status = married **AND** purpose of loan = furniture and consumer durable and goods **AND** housing type = rented
IF Personal status = married **AND** purpose of loan = cars and automobiles financing **AND** number of dependants = two to four
IF Personal status = married **AND** purpose of loan = furniture and consumer durables and goods **AND** housing type = owned **AND** job = government
Applicants belonging to the second cluster (C1) were found to follow one of the following rules:
IF Personal status = single **AND** outstanding credit amount $\leq$140,545
IF Personal status = married **AND** purpose of loan = cars and automobiles financing **AND** number of dependants = two to four **AND** total loan amount (including profits) $\leq$ 389,932
Applicants belonging to the third cluster (C2) were found to follow one of the following rules:
IF Personal status = married **AND** purpose of loan = other personal loans
IF Personal status = married **AND** purpose of loan = furniture and consumer durable and goods **AND** housing type = owned **AND** job = private
IF Personal status = single **AND** outstanding credit amount $>$140,545 **AND** housing type = owned

From Table 3.5 we can notice a little difference to what has been reported regarding the data before the pandemic. Personal status, Job, total loan amount, Age, housing type, and outstanding credit amount are the most present features on the obtained rules. For the first cluster C0 we notice that all credit applicants in this category are married, they work either in the private sector or have a governmental job. In the majority of cases they are aged between 36–45 years or from 46–55 years, with a rented or owned type of housing. For the second cluster C1 we found that the credit applicants in this cluster are either married persons, working in a governmental job and aged between 36–45 years or from 46–55. They could be also singles, working in the private sector and living in a rented house with a total loan amount (including profits) >124,451.For the third cluster C2 we notice that all the credit applicants are married, living in rented type of housing. They may either be working in a private sector with outstanding credit amount $\leq$91,900 or working with the government with outstanding credit amount $\leq$122,591.

Overall, the rules obtained reveal a slight change from those obtained for the tree reporting the behaviour of non-creditworthy applicants before the pandemic. Clearly, the applicants' characteristics are heavily influenced by their job types, while before the pandemic defaulting was highly correlated to the purpose of the loan.

3.6 Conclusion

In this study, we introduced a new semi-supervised segmentation approach in order to identify credit risk segments. The proposed approach combines Cluster analysis

Table 3.5 Rule set generated from the decision tree of Fig. 3.5

Applicants belonging to the first cluster (C0) were found to follow one of the following rules:
IF Personal status = married **AND** job = private **AND** housing type = owned **AND** age = 36–45 years **IF** Personal status = married **AND** job = private **AND** housing type = owned **AND** age = 46–55 years
IF Personal status = married **AND** job = government**AND** age = 36–45 years**AND** housing type = owned
IF Personal status = married **AND** job = private **AND** housing type = rented **AND** outstanding credit amount >91,900
Applicants belonging to the second cluster (C1) were found to follow one of the following rules:
IF Personal status = married **AND** job = government **AND** age = 25–35 years
IF Personal status = married **AND** job = government **AND** age 46–55 years
IF Personal status = single **AND** job = private **AND** housing type = rented **AND** total loan amount (including profits) >124,451
Applicants belonging to the third cluster (C2) were found to follow one of the following rules:
IF Personal status = married **AND** job = private **AND** housing type = rented **AND** outstanding credit amount ≤91,900
IF Personal status = married **AND** job = government **AND** age = 36–45 years **AND** housing type = rented **AND** outstanding credit amount ≤122,591

of account behaviours using K-means algorithm and supervised learning based on J48 decision tree algorithm in the context of COVID-19 Pandemic in banking sector in Saudi Arabia. The main results obtained with the proposed approach indicate that the most significant characteristics of credit applicants before the pandemic are related to their personal status and the purpose of the loan. We also found that the applicants' characteristics are heavily influenced by their job type during the pandemic year. As future works we plan to extend the empirical analysis by using other clustering and classification approaches.

Acknowledgments This work was funded and supervised by the Saudi Central Bank (SAMA) in a joint research program. The authors, therefore, acknowledge with thanks all SAMA members for technical and financial supports.

References

1. Minister of Finance, With more than sar 120 bn: Government of saudi arabia implements urgent measures to mitigate the impact of coronavirus on economic activities and private sector. https://mof.gov.sa/en/MediaCenter/news/Pages/News20032020.aspx
2. KPMG, Kingdom of saudi arabia, government and institution measures in response to covid-19. https://home.kpmg/xx/en/home/insights/2020/04/saudi-arabia-government-and-institution-measures-in-response-to-covid.html
3. T.S.C. Bank, SAMA: value of private sector financing support program initiatives exceeds sar 51 billion. https://www.sama.gov.sa/en-US/News/Pages/news-584.aspx
4. T.S.C. Bank, SAMA announced the extension of the deferred payments program to the end of 1st quarter of the year 2021. https://www.sama.gov.sa/en-US/News/Pages/news-631.aspx

5. T.S.C. Bank, SAMA extends deferred payments program for 6 months. https://www.sama.gov.sa/en-US/News/Pages/news-605.aspx
6. B. Baesens, Business applications and limitations of analytical credit scoring (2017). https://www.dataminingapps.com/2017/05/business-applications-and-limitations-of-analytical-credit-scoring
7. L.C. Thomas, Int. J. Forecast. **16**(2), 149 (2000). https://EconPapers.repec.org/RePEc:eee:intfor:v:16:y:2000:i:2:p:149-172
8. M. Alaraj, M. Abbod, M. Majdalawieh, L. Jum'a, Neural Comput. Applic., 1–28 (2022)
9. I. Chen, C.J. Lu, T.S. Lee, C.T. Lee, *Behavioral Scoring Model for Bank Customers Using Data Envelopment Analysis* (Springer, Berlin, 2009), pp. 99–104. https://doi.org/10.1007/978-3-540-92814-016
10. IFei, Chen, Int. J. Adv. Comput. Technol. **3**, 87 (2011)
11. H.I. Hsieh, T.P. Lee, T.S. Lee, in *2010 International Conference on Computational Intelligence and Software Engineering* (2010), pp. 1–4. https://doi.org/10.1109/CISE.2010.5677005
12. S.B. Pereira, Modelling credit card customer behaviour (2019). Work Project presented as a partial requirement for Degree of Master of Statistics and Information Management, with a specialization in Information Analysis and Management
13. N. Jiang, N. Novik, Leveraging big data and machine learning in credit reporting (2021). https://blogs.worldbank.org/developmenttalk/leveraging-big-data-and-machine-learning-credit-reporting. Last Accessed 16 Sept 2017
14. J. Han, M. Kamber, J. Pei, *Data mining concepts and techniques*, 3rd edn. (Morgan Kaufmann Publishers, Waltham, Mass., 2012)

Chapter 4
Improving Sales Prediction for Point-of-Sale Retail Using Machine Learning and Clustering

Chibuzor Udokwu, Patrick Brandtner, Farzaneh Darbanian, and Taha Falatouri

4.1 Introduction

Point-of-sale (PoS) retail represents the final element of the retail supply chain and provides products to the customers. Although there has been an immense growth in online retailing within the last decade, the importance of physical retail stores cannot be neglected. Consumers will continue to patronize and use physical retail stores in the coming years due to positive in-store experiences and the increased perceived value customers get simply by visiting the shopping stores [1]. As a result, retailers continue to improve their shopping environment since improved shopping experiences positively increase customer spending [2]. Such experiences also include the availability of products in the stores' inventories whenever the customer wants to buy the product. Thus, it is necessary to continue the optimization of the performance of these stores. In this context, a wide variety of datasets are gathered in business processes in physical retailing. Such data offers huge potential to be exploited and analysed to improve the performance of these stores by ensuring product availability. Different data analytics methods can be used to improve product availability in physical retail stores. The study [3] explored related literature to identify the benefits of applying ML in forecasting the sales of fast-moving consumer goods. The benefits identified include better sales forecast accuracy, improved inventory management, greater product availability and higher customer satisfaction.

Machine learning (ML) models provide techniques for forecasting product demand based on historical data. Several ML models have been applied to predict

C. Udokwu · P. Brandtner (✉) · F. Darbanian · T. Falatouri
Logistics Department, University of Applied Sciences Upper Austria, Steyr, Austria
e-mail: chibuzor.udokwu@fh-steyr.at; patrick.brandtner@fh-steyr.at; farzaneh.darbanian@fh-steyr.at; taha.falatouri@fh-steyr.at

B. Alyoubi et al. (eds.), *Machine Learning and Data Analytics for Solving Business Problems*, Unsupervised and Semi-Supervised Learning,
https://doi.org/10.1007/978-3-031-18483-3_4

the demand and sales of products in physical retails. The study [4] compared different ML models in predicting the sales demand in retail by applying classification models and regression models in demand forecasting. The review article [5] shows the application of regression models and deep learning methods in retail demand forecasting. Some of the classification models that have been applied in literature for demand forecasting in physical stores include Bayesian network machines (BNM), random forests (RF) and neural network multi-layer perceptron (MLP) [6–8]. For regression algorithms, they include linear regression (LR), auto-regression and auto-regression integrated moving average (ARIMA), and multivariate regression (MR) [3, 9, 10]. For the application of deep learning artificial intelligence (AI) methods in forecasting, some of the commonly used models include artificial neural networks (ANN), such as multilayer perceptron (MLP) and extreme learning machines (ELM), and recursive neural networks (RNN) such as long short-term memory (LSTM) [4, 5].

Forecasting future sales for PoS retailing with several product lists across many stores is a complex task. Such complexities negatively affect the quality of the forecast predictions. Large PoS retail chains have several stores scattered across different demographic locations. Several products and groups of products are in the inventory of these stores. Also, the sale behaviour of these products differs across stores. Building single models for each product and each PoS store is a practically difficult task to scale. Thus, to better forecast product sales, it is necessary to develop ML models specific for unique products or groups of stores.

The study [11] shows that combining clustering methods and machine learning forecasting models can help improve the quality of prediction in retail stores. This is because clustering reduces the complexity of datasets of large retail chains by organizing the data inputs into different groups of clusters. The study [9] applied basket analyses for clustering and ARIMA for forecasting product sales in omnichannel retails. The study [12] applied clustering methods such as self-organizing maps (SOM) and K-means for clustering stores with classification and deep learning methods to predict product sales. The study [13] applied a bipartite graph for store clustering and a hybrid combination of statistical model and ML classification algorithm for demand forecasting. These presented studies show consistent use of clustering methods and machine learning models in forecasting product sales in physical retail. Still, the research in these studies has failed to identify the main factors for organizing retail stores and model combinations of clustering and prediction algorithms that will result in the best performance for a forecasting system for product sales in retail. Therefore, the main research question for this paper is how to apply ML clustering and prediction algorithms to improve sales prediction for products in PoS retails. For separation of concerns and regarding the identified research gap, the following subquestions are defined:

1. What are the factors for clustering PoS stores in different clusters?
2. What are the group of stores developed by applying store clustering factors on different ML clustering algorithms?

3. What are the clustering and prediction model combinations that improve sales forecasting in retail?

The rest of the paper is organized as follows. Section 4.2 introduces the background of the paper by discussing various ML approaches in physical retailing. Section 4.3 provides the answer to the first research question by outlining factors that affect product sales in PoS retail that can be used in clustering retail stores. These factors are gathered from relevant literature and then classified into performance-related and demographic-related factors. Section 4.4 provides an answer to the second research question by applying the performance and demographic store properties on ML clustering algorithms to develop a distinct group of stores that have similar behaviour. Section 4.5 provides the answer to the third research question by examining different model combinations to determine selected models that have the best prediction results in forecasting product sales in retail. In Sect. 4.6, discussions on key findings in comparison with similar studies are presented. The conclusion of the paper and future work is presented in Sect. 4.6.

4.2 Background

This section presents literature discussions on physical retailing and the various machine learning approaches that have been applied in improving processes in physical retail. Also, the physical retailing case study used in the evaluation of the forecasting approaches applied in this paper is presented.

4.2.1 PoS Retail

4.2.1.1 Description of Physical Retailing

PoS retailing comprises a physical environment where customers can make purchases and pay for the purchased items at checkout counters [12]. This sector has generally low digitization since its main purpose is to provide the final point of product delivery to customers [13]. Also, innovation in retailing is mostly experienced in online retailing, and physical retailers are now moving towards hybrid retail channels to catch up with innovations in the retail sectors [14, 15]. These hybrid channels seek to provide a shopping experience that combines both online and physical retailing concepts. Still, the importance and future of physical retailing cannot be neglected due to the positive customer experience encountered in physical stores [1]. The study [13] examined technology integration in physical retailing and prospects of retailing and identified store management technologies as one of the innovation drivers in retail. The goal is to apply technologies in managing stock levels and product availability to improve the productivity of the store managers and the satisfaction level of the customers.

4.2.1.2 Fast-Moving Consumer Goods and ML

Managing stock levels and availability of perishable products is area store managers need the most help due to the problematic nature of these types of fast-moving consumer goods [16]. This is because, for such goods, orders are performed daily or weekly and have a very short shelf-life span. Excess order of such products results in overstock and wastage, and low order amounts result in out of stock and missed sales opportunities which negatively influences customer satisfaction [17, 18]. The study [19] evaluated various SC structures for perishable products and found that advanced inventory management approaches are needed for managing the stocking of such products. Thus, retail managers need technological help in estimating the sales and demand of such perishable products. Machine learning techniques provide approaches for forecasting such product demands, thereby helping managers in making better decisions in their store management.

4.2.2 ML Approaches Applicable in Retail

This section presents various machine learning approaches which have been applied in literature for predicting product demand in the retail sector. These approaches are classified into clustering, classification, regression and deep learning algorithms.

4.2.2.1 Clustering Approaches

Clustering involves grouping elements that seem to have similar properties close together and further away from elements that do not share such properties. Some of the clustering algorithms that have been applied in literature for grouping stores and products in physical retailing include the self-organizing map (SOM), growing hierarchical SOM (GHSOM), K-means, bipartite graph and network graph clustering.

The SOM technique groups similar data together by reducing high-dimensional data to a low-dimensional dataset while retaining the network structure of the data [4]. They are also referred to as the Kohonen network. SOM has a good performance for organizing highly dimensional data and providing comprehensive results that can easily be visualized usually as two clusters. The GHSOM is a variation of the SOM that addresses its limitation in representing hierarchical relationships in data.

K-means is a common clustering approach for portioning a given number of observations (*n*) into a fixed number of clusters (*k*) such that observations are placed closest to the clusters with the nearest mean [3]. This clustering approach has been applied in grouping products and modelling customer behaviour in retail [10, 20].

The bipartite graph represents a graph such that the edges are divided into two sets. The approach can be applied in clustering by connecting data in a variable with data in another variable for which affiliation exits [21]. This clustering approach has been applied in organizing warehouses for the delivery of products to retail customers [13].

4.2.2.2 Classification Algorithms

Classification ML algorithms organize and group a set of data into different classes that have been provided as an output. The classification algorithms presented include SVM and BNM.

SVM produces two classes for grouping the input datasets provided and can be applied to classify both categorical and continuous datasets. The BNM is a modelling approach where a probabilistic graph model is used in representing a set of variables. The edges in the graph represent conditional probabilities for the variables in the dataset [22]. Other common classifiers include decision trees and random forests (RF).

4.2.2.3 Regression Algorithms

Regression algorithms are used to address problems that involve the prediction of quantity where the input variables are usually discrete or continuous values. Auto-regression and related approaches such as ARIMA and SARIMA are common machine learning models that have been used in forecasting the quantity of product sales/demand in physical retailing [23].

The autoregressive (AR) model is an ML model for predicting time-dependent processes where an output variable is dependent on its previous values [24]. Thus, it's very useful for predicting the quantity of product demands for time-varying or seasonally dependent products. ARIMA and SARIMA are adaptations of the AR modelling approach. The former is AR integrated with moving average, while the latter also captures the seasonal attributes of the time-dependent dataset [25].

4.2.2.4 Deep Learning and Neural Network Methods

Deep learning is a broad classifier for representing deep neural network classifiers such as an artificial neural network. Some of the common approaches for neural networks are feedforward and recurrent neural networks. Neural networks (NN) generally consist of nodes ordered in a layered structure where each node receives, processes and transmits a signal to the other nodes in the next layer of the network. Learning is achieved by adjusting the weight of signals transmitted by the nodes [26]. The following deep learning algorithms have been applied in predicting product demand in retailing: MLP, LSTM and ELM.

MLP is a type of feedforward NN that uses backward propagation in training the network. In MLP, at least a minimum of a three-layered network is used in training and prediction, and they are the input, hidden and output layers. The ELM is another example of feedforward NN for classifying data where the hidden layer of the network can contain one or multiple layers. In ELM, the hidden nodes are randomly initialized. The LSTM is a type of recurrent NN and is used for predicting time series data where the unknown duration of time lags can occur between two important events in the dataset. Studies have applied ELM and MLP in the forecasting of product demand in physical retailing [5, 10, 12].

4.2.3 Case Presentation

The case selection for this paper is a retail chain with hundreds of physical stores located across Austria. The stores provide PoS grocery retail services for customers. The sales data provided for this study comprises 2 years of product sales, from 2017 to 2019, thereby representing the main input for the analyses conducted in this paper. The focal product group for this paper are fruit and vegetable products since they are fast-moving consumer goods and provide a serious challenge for store managers in estimating the optimal stock levels and order amounts.

4.3 Factors Affecting Product Sales in PoS Retail

To identify factors for clustering stores, we examined related literature and derived factors that affect store locations and their product sales patterns in physical retail. The factors identified are analysed to determine their applicability to the physical retailing case presented in this paper. First, the factors are categorized into performance- and demography-related ones. A factor is considered a performance factor if is related to actual sales in the retail stores and a demographic factor if it is related to the physical location of the store. The applicability of each factor is assessed by checking if it is relevant to the properties of the PoS retail case presented. For the applicable factors, a method to describe an automated or manual gathering of data for any given store location is described. Then, the presented data collection method is rated by its difficulty level.

The studies [5, 9] conducted systematic reviews to determine performance and demographic factors that affect sales and location of retail stores. The first conducted expert interviews to identify and evaluate demographic factors that affect store location, while the latter analysed related literature papers to identify performance criteria relevant for classifying store locations. The demographic and performance factors for classifying retail stores outlined in these two articles provide the basis for the analyses conducted in this part of the paper. Table 4.1 shows the factors for clustering PoS stores and properties that describe the selected factors.

Table 4.1 Performance and demographic factors for clustering physical retail stores

Category	Factor	Description	Case applicability	Automated(semi) data collection description
Demographics [5]	Competitor's competition	Number of competing stores close to the store	Yes	Script to evaluate the Euclidean distance between stores and competitors on a Google Map
	Crowd points (hospital, market, hotel, restaurant)	Crowd-pulling points located close to the store	Yes	A script to evaluate the actual distance between stores and crowd points
	Culture and education organization (study centre, library)	Cultural and education centres close to the store	No	–
	Relaxation (recreation centre, cinema, park)	Location of the store in a recreation or relaxation environment	No	–
	Government and business organization (government and office building)	Type of building that houses the store	No	–
	Vehicle maintenance	Vehicle maintenance within or close to the store	No	–
	Parking convenience	Parking space availability in the store building	Yes	From Google Map, parking availability can be identified
	Pedestrian crossing	Pedestrian crossing close to the store	Yes	From Google Map
	Sidewalk width	Sidewalk close to the store	Yes	From Google Map
	Road width	Width of the road connecting to the store	Yes	–
	Bus stop	Bus stop(s) close to the store	Yes	From Google Map
	Car flow	Rate of car movement around the store	Yes	–

(continued)

Table 4.1 (continued)

Category	Factor	Description	Case applicability	Automated(semi) data collection description
	Located near road intersection?	Nearby intersections	–	–
	Store visibility	Store visibility and easiness to locate	–	–
	Storefront area	Size of the front area of the store	No	–
	Population size	Population size where the store is located	Yes	From the city’s statistical data
	Income/consumption levels	Income and consumption level of people living in the store area	Yes	–
	Population and population density	Population density of the store location	Yes	From statistical data and Google Map
Performance [9]	Store profits (cash flow after discount and tax)	Cash flow from product sales in the store	Yes	Store sales data
	Customers (number of customers, amount of items in customers receipt, average customer spend)	Number of customers, average quantity and amount of items they purchase	Yes	Store sales data
	Market share (sales volume in particular product group)	Share of the store for a given product group or product type	Yes	Store sales data
	Price elasticity	How sensitive the customers are towards changes in price	Yes	Store sales data

4.3.1 Main Factors Affecting Sales and Location of Retail Stores

To determine the main factors for clustering the physical retail stores, we identified the performance and demographic properties of stores available in the literature and examined their applicability to the current retail chain case as presented in Sect. 4.2.3. These factors are then applied in clustering algorithms presented in Sect. 4.4 to develop different clusters of the stores. Table 4.1 presents the list of relevant factors for store clustering. The first column shows the category under which a factor is organized, and they are performance-related and demographic-related. The second and third columns provide a label and high-level description for the identified factors. The last two columns provide the applicability of the factors and description of (semi)automated methods for collecting data related to a given store clustering factor.

4.3.1.1 Demographic Factors for Clustering Stores

The first part of Table 4.1 shows the list of demographic factors that can be used in organizing retail stores into groups. The relevant factors applicable to the case adopted in this paper include competitors, crowd points, parking convenience, pedestrians, sidewalk, road width, bus stop, car flow, city population/density and income levels. The other listed factors in the table are not considered relevant, because, for the selected case of the retail chain, all stores share the same type of building with the same physical/structural characteristics. These unapplicable factors include culture and education, office building type, relaxation centres, vehicle maintenance and storefront area.

Some demographic factors are considered important but were not considered in the next step of this analysis in Sect. 4.4 due to difficulty in automating data gathering for such factors. The important clustering factors that are not applied in clustering the stores include income levels, crowd points, pedestrians, sidewalk, road width and bus stops.

4.3.1.2 Performance Factors for Clustering Stores

The second part of Table 4.1 shows the performance-related factors for clustering stores. All the listed factors are applicable in the retail case adopted in this paper. This is because sale activity differs across the stores irrespective of the physical structure and location of the stores. The important performance factors for clustering retail stores include store profit; customer-related factors such as purchase amount (and quantity), number of customers and average customer spending; market share; and price elasticity. For the market share, the sales share of each store in vegetable and fruits products in comparison with other stores in the retail chain is considered as the market share factor. This, therefore, includes the profit and customers purchasing the unique product groups.

Although price elasticity is considered an important clustering factor since it applies to the case adopted for this paper, this factor is not applied in the next step. This is due to the complexity of automating data gathering for this type of data for the selected store products in different city locations.

4.4 Store Clusters

Two different unsupervised clustering approaches are applied in this paper for organizing stores into groups that share similar behaviour. These clustering methods are SOM and K-means, which are then combined with ML predictive algorithms in Sect. 4.5 to determine the selection of models that produced the best result in forecasting product sales in retail.

K-means and SOM are considered in this paper because they are the commonly used clustering approaches applied in many works of literature for analysing retail stores [11, 27, 28]. The clusters produced by these algorithms are assessed for their quality and important predictors that apply in each clustering approach. The quality of the clusters is measured using silhouette measures to evaluate the cohesion and separation of developed clusters. The silhouette value ranges from -1 to 1. Values above 0.2 are considered fairly acceptable clusters, while values above 0.5 are considered good clusters [29].

4.4.1 K-means Store Clusters

To identify the store clusters using K-means, a k value from 10 downwards is applied to the input data, and the quality of the clusters produced is measured. The k value of 6 and 5 produced the best results. K value of 6 instead of 5 is selected because it produces more clustering information, also considering the small number of stores in one of the clusters. The sample in the biggest clusters contains 26.2% of the stores, while the smallest cluster contains 1.6% of the stores. Figure 4.1 shows the percentage distribution of stores in each cluster.

Further examination of the clusters with the k value of 6 shows that a store's market share in vegetable and fruit products and the number of customers for each store and quantity of items purchased are the most important predicting factors for the clusters. Assessing the cluster produced by the K-means shows a silhouette value of 0.6 which is considered a good cluster.

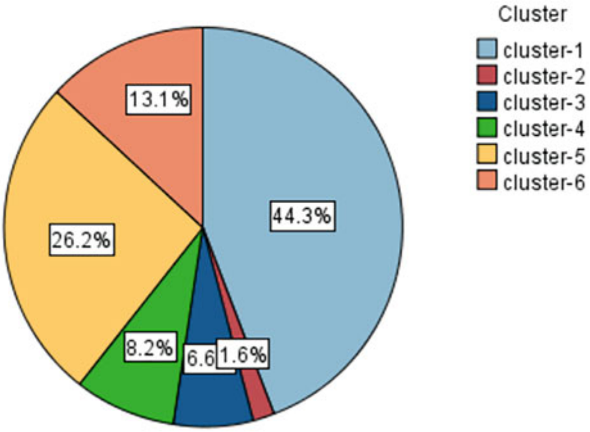


Fig. 4.1 K-means store clusters and sample size

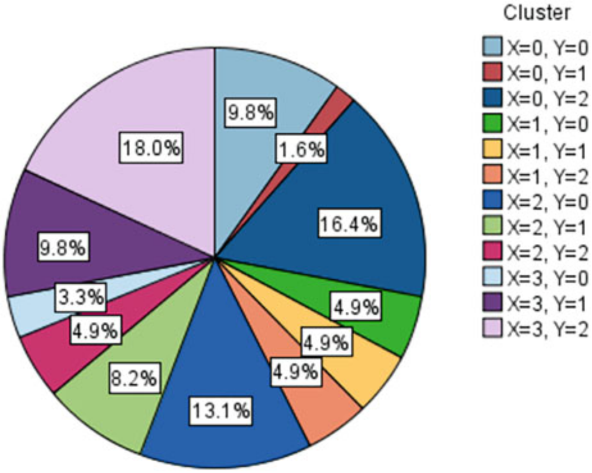


Fig. 4.2 SOM-Kohonen store clusters and sample size

4.4.2 SOM (Kohonen) Store Clusters

Organizing stores using the Kohonen clustering produced 12 groups of stores that share similar behaviour. The biggest cluster contains 18% of the sample size, while the smallest cluster contains 1.6% of the sample size. The result for the Kohonen clustering is based on the default settings of the model.

Further examination of the clusters shows that an individual store's market share in the sale of fruit and vegetable products as well as the quantity of (general) product sales and the number of customers for each store are the main predictor factors for clusters. The silhouette value for the Kohonen clustering is 0.3 which is considered an acceptable cluster result (Fig. 4.2).

4.5 Evaluation of Model Combinations for Sales Forecast

The evaluation of models is performed by combining the clustering models with different machine learning models. For this paper, we only consider machine learning classification algorithms as well as deep learning methods. This is because these types of models produce better results when external datasets are considered in modelling the data and prediction [4]. The selected ML classification models are neural network multilayer perceptron, Bayesian network models and random forests. These models are selected because they have successfully been used as classification algorithms in forecasting product sales in retail [6, 7, 19]. These prediction algorithms in combination with the clustering algorithms determine the best model selection that improves sales forecasting in retail.

Considering that products used in the case presentation of this paper, which are fruits and vegetable products, are generally classified as fast-moving consumer products, their daily sales are highly dependent on external factors such as weather and special calendar event days [4, 10]. However, we apply only the calendar data such as weekdays, weekends, months, years, seasons, effective days and holidays to provide features that describe the daily sales. The features that described the stores are represented by the cluster groups they belong to. The features described are the high-level classes that identify the product groups and the unique product identification code. Each product class contains a set of unique products. To evaluate the performance of the model combinations of the forecasting approach presented in this paper, cases that present different combination scenarios for clustering and prediction algorithms are described and then assessed.

4.5.1 Cases of Model Combinations

Different combinations of clustering and forecasting models represent the cases used in evaluating the forecasting models. Two main possibilities for combining ML models in this paper are the hybrid approach and the cluster group approach. In the hybrid approach, cluster groups produced by the clustering algorithms are considered a feature in the input data for the prediction phase. For the cluster group method, each cluster is forecasted independently. Stores that belong to the same clusters are filtered, and prediction algorithms are applied to them separately. Thus, different cases of K-means and SOM have been applied hybrid and grouping approaches, respectively:

- *Case 1*: For the first case, which serves as the base for measurement, all input factors consisting of different stores, product groups and codes as well as calendar data are forecasted with NN- (MLP), RF and BNM. In this first case, no clustering feature is added to the input dataset.

- *Case 2*: For the second case, all input factors, as well as the clustering results of K-means providing additional input features for the cluster, are used. The prediction results for the selected prediction algorithms are then measured.
- *Case 3*: The third case is the same as the previous case; however, the clustering results from Kohonen are used in providing additional input features for the prediction.
- *Case 4*: This case applies the same hybrid approach as cases 2 and 3; however, the results from both clustering methods (K-means and SOM-Kohonen) provide additional input features for the prediction.
- *Case 5*: This represents the grouping approach where the largest cluster group from K-means is predicted separately by applying the selected prediction algorithms on this cluster. The largest K-means cluster is represented by cluster 1.
- *Case 6*: This is the same approach as case 5; however, the second-largest cluster produced by K-means is predicted. The second-largest cluster of K-means is represented by cluster 5.
- *Case 7*: This is the same as the grouping approach for case 5; however, the cluster considered is based on the Kohonen clusters, and the largest cluster in the group is represented by $X = 3$, $Y = 2$.
- *Case 8*: This is the same as the grouping approach for case 5; however, the second-largest cluster of Kohonen is predicted. This cluster is represented by the label $X = 0$, $Y = 2$.

4.5.2 Input Data Preparation

The problem of predicting the daily product sales in retail is traditionally a regression problem because the daily sales quantity of a given store is represented as a continuous variable. Therefore, to solve this problem with a classification algorithm, it is necessary to convert the input data such as daily sales into categorical data. The following steps were performed in converting the continuous sales data into categorical data: first, we transformed the total daily sales for each product into boxes of products. This is achieved by dividing the product total sales by the maximum number the product box can contain. Furthermore, we reduced the product boxes into categories that represent the maximum and minimum daily sale boxes for each product. Thus, a given category can represent a range of boxes of products (min/max). A total of 12 categories are created that represent the daily product sales.

4.5.3 Evaluation of Model Combinations

Table 4.2 shows the prediction accuracy obtained by applying different cases of a combination of ML clustering and prediction algorithms. The accuracy is represented by the percentage of correct predictions produced by selecting prediction algorithms such as neural network multilayer perceptron, random forests and Bayesian network models.

The result shows that adding clustering features generally improves the forecasting results for product sales in retail. Minimal prediction accuracy improvements are recorded for forecasting cases that adopted hybrid approaches of K-means and SOM-Kohonen separately. For K-means combination with prediction algorithms (case 2), 1.3%, 1.0% and 5.0% forecast improvements are recorded in NN, RF and BN, respectively. For SOM combination with prediction algorithms (case 3), no noticeable improvements were recorded in NN and BN respectively. Combining K-means and SOM in a hybrid approach (case 4) also resulted in a 5% accuracy improvement for BN.

The forecasting cases that applied the grouping method produced better results than the hybrid approach. For grouping forecasting approaches that applied K-means with other prediction algorithms, up to 8.3%, 3.3% and 18.3% accuracy improvements are recorded for NN, RF and BN, respectively (case 6). For grouping forecasting approaches that applied SOM-Kohonen, up to 6.9%, 2.2% and 15% accuracy improvements are recorded for NN, RF and BN, respectively (case 7).

In general, forecasting approaches that adopted grouping cluster applications performed better than hybrid forecasting. K-means clusters in combination with other prediction algorithms performed better than SOM-Kohonen clusters that were combined with other prediction algorithms. Random forest forecasting algorithms produced the best prediction results for all the ML classification models evaluated. For all the cases examined, combining clustering algorithms with ML prediction algorithms produced better prediction results when compared with the base prediction except for case 8. The prediction results from case 8 can be considered an anomaly and provide an opportunity for further examination. The best combination of clustering and prediction algorithms to produce the best forecasting results in retail product sales is K-means and random forest algorithms.

4.6 Discussion of Results

This part of the paper presents comparative discussions on the main findings of this paper. Building on [30–32], the results of the paper constitute theoretical contributions in two forms: (i) examining previously tested theories in a new context and (ii) considering new assumptions or axioms in previously studied models. More precisely, existing approaches of clustering were applied to the new context of the retail domain to identify characteristics and features of how to cluster

Table 4.2 Performance of sales forecasting for different model combinations

Case	Explanation	Cluster application	Cluster model	NN (%)	RF (%)	BN (%)
1	Select all input without cluster feature	None	None	72	91	60
2	Select all input with a cluster feature	Hybrid	K-Means	73	92	63
3	Select all input with a cluster feature	Hybrid	SOM-Kohonen	72	92	60
4	Select all input with multiple cluster features	Hybrid	K-Means a& SOM-Kohonen	72	92	63
5	Select the largest cluster group	Grouping	K-Means	74	93	67
6	Select second-largest cluster group	Grouping	K-Means	78	94	71
7	Select the largest cluster group	Grouping	SOM-Kohonen	77	93	69
8	Select second-largest cluster group	Grouping	SOM-Kohonen	65	90	57

physical retail stores. Furthermore, existing approaches of sales prediction based on ML and clustering were applied to and evaluated for the retail sector and more specifically for point-of-sale retail. Regarding the consideration of new assumptions in previously studied models, we combined clustering and prediction models and analysed the potential of improving sales forecasting accuracy in the retail sector based on hybrid prediction approaches.

The main findings are as follows:

(i) Applying clustering with other predictive models generally improves the forecast accuracy. This result is similar to the findings of the study [11] which shows that combining clustering and other predictive analytics methods improves the quality of forecasting results.

(ii) Grouping approaches for clustering and predictive model combinations performed better than the hybrid approach. This is also an expected result since grouping reduced the complexity of the dataset by grouping stores with similar behaviour and predicting their daily sales independent of stores in other clusters. The study [33] shows that an increase in model complexity results in a general reduction in model performance and thereby leads to problems such as overfitting.

(iii) The combined forecasting approaches that use K-means clustering produced better results than SOM clusters. Two different parameters from the experiments carried out in this paper show that K-means performs better than SOM-Kohonen clusters for the retail data considered. First, the analyses of cohesion and separation values of the clusters produced show that K-means produced better clusters in comparison with SOM. Also, prediction results of the forecasting approaches that applied K-means produced better accuracy than approaches that adopted SOM. This result is the opposite of the findings in the study [11] that show that a variation of the SOM model in combination with other predictive models performed better than K-means. Still, the K-means clustering approach has been adopted by different studies in analysing and categorizing behaviour in retail stores. The studies [27, 28] have applied K-means for classifying the behaviour of retail store customers.

(iv) Random forest algorithms performed better than other predictive algorithms. Also, RF combination with K-means produced better forecasting results. The study [11] applied combined common clustering approaches (such as K-means and SOM) and quantitative prediction methods such as time series, regression and neural networks in predicting product sales in retail stores. The results from this paper show that classification-based algorithms can also be used in predicting product sales. This can be achieved by converting a quantitative problem, such as the amount of product sales, into categories, such as the number of boxes sold, and the ranges of boxes sold for each product.

(v) Abnormal behaviour is observed for a single case out of the examined cases where the combination of clustering and predictive models did not improve the forecasting result. This could represent a potential problem in the stores in this cluster, where their order pattern is irregular. Thus, the prediction algorithms can't figure out a systematic pattern in the ordering of such stores. Such irregular order patterns can potentially be caused by inexperienced store managers or sales artificially driven by discounts. Therefore, it is necessary to optimize the input data to develop theoretical sale data that provides an actual (true) representation of daily sales to address the impact of irregular sales due to discounts and products out of stock. Preparing datasets to address these types of abnormalities in the input data is considered preprocessing and performing such tasks can improve the performance of the prediction model [34].

4.7 Conclusions, Limitations and Future Work

This paper seeks to provide a systematic approach to improve the performance of sales prediction for product sales in retail. To achieve this, we addressed the main research question of how to apply ML clustering and prediction algorithms to improve sales forecasting for products in PoS retails. This paper provides answers that represent the main factors for clustering retail stores and shows the store clusters developed by applying these factors as input data for clustering algorithms such as K-means and SOM. Furthermore, this research examines different clustering/predictive classification algorithms that produce the best results in forecasting product sales in retail.

The result of the clustering analyses performed in this paper shows that the market share of stores in specific product groups analysed as well as the number of customers in the stores and the total amount of product sales for the stores are the most important predictors for classifying the stores. Other important factors include the number of stores' competitors, parking availability, the population density of the stores' location, product price elasticity and general cash flow of the stores. Applying these input factors to the two selected clustering methods produced different cluster sizes and quality. The analysis of cluster quality shows that the K-mean model produced clusters with better cohesion and separation values. Furthermore, assessing the forecasting results produced by combined clustering and classification-based predictive algorithms shows that K-means/random forest produced the best forecasting result.

The main limitation of the paper is the possible generalization due to single-case selection. Also, the data analysed are selected from sales data that represented a single regional state in Austria. Thus, the result may differ when different and other retail chain stores are considered with data collected across several regions (and countries). Another weakness of the result is that the impact of possibly irregular sale patterns in the input data has not been considered during the data preparation stage. Thus, the impact of problems such as out of stock and discount (promotion)-driven sales was not properly accounted for in the input data. Thus, as future work, we consider a study that focuses on developing a modelling method that optimizes the input data (preprocessing) in retail data analytics to address problems in the datasets caused by out of stocks and sales being artificially driven by discounts.

References

1. N.S. Terblanche, Revisiting the supermarket in-store customer shopping experience. J. Retail. Consum. Serv. **40**, 48–59 (2018) https://doi.org/10.1016/j.jretconser.2017.09.004
2. I. Sachdeva, S. Goel, Retail store environment and customer experience: A paradigm. J. Fash. Mark. Manag. (2015). https://doi.org/10.1108/JFMM-03-2015-0021
3. A. Likas, N. Vlassis, J.J. Verbeek, The global k-means clustering algorithm. Pattern Recogn. **36**(2), 451–461 (2003). https://doi.org/10.1016/S0031-3203(02)00060-2
4. T. Kohonen, The self-organizing map. Proc. IEEE **78**(9), 1464–1480 (1990). https://doi.org/10.1109/5.58325
5. R.J. Kuo, S.-C. Chi, S.-S. Kao, A decision support system for selecting convenience store location through integration of fuzzy AHP and artificial neural network. Comput. Ind. **47**(2), 199–214 (2002). https://doi.org/10.1016/S0166-3615(01)00147-6
6. İ. İşlek, Ş.G. Öğüdücü, A retail demand forecasting model based on data mining techniques, in *2015 IEEE 24th International Symposium on Industrial Electronics (ISIE)*, (IEEE, 2015). https://doi.org/10.1109/ISIE.2015.7281443
7. Odegua, R, Applied machine learning for supermarket sales prediction. Research gate (2020). https://www.researchgate.net/publication/338681895_Applied_Machine_Learning_for_Supermarket_Sales_Prediction
8. İ. İşlek, Ş.G. Öğüdücü, A retail demand forecasting model based on data mining techniques, in *2015 IEEE 24th International Symposium on Industrial Electronics (ISIE)*, (IEEE, 2015, June), pp. 55–60. https://doi.org/10.1109/ISIE.2015.7281443
9. G. Turhan, M. Akalın, C. Zehir, Literature review on selection criteria of store location based on performance measures. Proc. Soc. Behav. Sci **99**, 391–402 (2013). https://doi.org/10.1016/j.sbspro.2013.10.507
10. K. Kusrini, Grouping of retail items by using K-means clustering. Proc. Comput. Sci **72**, 495–502 (2015). https://doi.org/10.1016/j.procs.2015.12.131
11. I.-F. Chen, L. Chi-Jie, Sales forecasting by combining clustering and machine-learning techniques for computer retailing. Neural Comput. & Applic. **28**(9), 2633–2647 (2017) https://doi.org/10.1007/s00521-016-2215-x
12. S.S. Kolhatkar, S.V. Joshi, Transforming the point of sale to point of service-applying SOA in the Indian retail scenario. J. Business Retail Manag. Res **4**(2) (2010). https://jbrmr.com/details&cid=33
13. D. Pederzoli, ICT and retail: State of the art and prospects, in *Information and Communication Technologies in Organizations and Society*, (2016), pp. 329–336. https://doi.org/10.1007/978-3-319-28907-6_22
14. S. Chopra, How omni-channel can be the future of retailing. Decision **43**(2), 135–144 (2016). https://doi.org/10.1007/s40622-015-0118-9
15. D. Vale, I.C.-L. Guillaume, X. Lecocq, The new retail model: Global reach demands omni-channels. J. Bus. Strateg. (2021). https://doi.org/10.1108/JBS-02-2021-0026
16. H. Liu et al., Optimal purchase and inventory retrieval policies for perishable seasonal agricultural products. Omega **79**, 133–145 (2018) https://doi.org/10.1016/j.omega.2017.08.006
17. C. Joseph Udokwu, F. Darbanian, T.N. Falatouri, P. Brandtner, Evaluating technique for capturing customer satisfaction data in retail supply chain, in *2020 the 4th International Conference on E-Commerce, E-Business and E-Government*, (2020), pp. 89–95. https://doi.org/10.1145/3409929.3414743
18. P. Brandtner, F. Darbanian, T. Falatouri, C. Udokwu, Impact of COVID-19 on the customer end of retail supply chains: A big data analysis of consumer satisfaction. Sustainability **13**(3), 1464 (2021). https://doi.org/10.3390/su13031464
19. T. Thron, G. Nagy, N. Wassan, Evaluating alternative supply chain structures for perishable products. Int. J. Logist. Manag (2007). https://doi.org/10.1108/09574090710835110
20. P. Anitha, M.M. Patil, RFM model for customer purchase behaviour using K-Means algorithm. J. King Saud Univ. Comput. Inf. Sci (2019). https://doi.org/10.1016/j.jksuci.2019.12.011

21. R. Diestel, *Graph Theory, Graduate Texts in Mathematics* (Springer, 2005) ISBN 978-3-642-14278-9
22. N. Friedman, D. Geiger, M. Goldszmidt, Bayesian network classifiers. Mach. Learn. **29**(2), 131–163 (1997). https://doi.org/10.1023/A:1007465528199
23. C. Catal et al., Benchmarking of regression algorithms and time series analysis techniques for sales forecasting. Balkan J. Electr. Comput. Eng **7**(1), 20–26 (2019). https://doi.org/10.17694/bajece.494920
24. R. Shumway, D. Stoffer, *Time Series Analysis and its Applications: With R Examples*, 3rd edn. (Springer, 2010) ISBN 144197864X)
25. S. Wang, C. Li, A. Lim, Why are the ARIMA and SARIMA not sufficient. arXiv preprint arXiv:1904.07632 (2019)
26. J. Schmidhuber, Deep Learning in Neural Networks: An Overview. Neural Netw. **61**, 85–117 (2015) arXiv:1404.7828
27. T. Kansal et al., Customer segmentation using K-means clustering, in *2018 International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS)*, (IEEE, 2018). https://doi.org/10.1109/CTEMS.2018.8769171
28. V. Holý, O. Sokol, M. Černý, Clustering retail products based on customer behaviour. Appl. Soft Comput. **60**, 752–762 (2017). https://doi.org/10.1016/j.asoc.2017.02.004
29. P.J. Rousseeuw, Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. J. Comput. Appl. Math. **20**, 53–65 (1987). https://doi.org/10.1016/0377-0427(87)90125-7
30. A. Salamzadeh, What constitutes a theoretical contribution? J. Org. Cult. Commun. Confl. **24**(1), 1–2 (2020) https://ssrn.com/abstract=3599931
31. G. Fillion, V. Koffi, J.P.B. Ekionea, Peter Senge's learning organization: A critical view and the addition of some new concepts to actualize theory and practice. J. Org. Cult. Commun. Confl. **19**(3), 73–102 (2015) https://www.researchgate.net/publication/304824741_Peter_Senge's_learning_organization_A_critical_view_and_the_addition_of_some_new_concepts_to_actualize_theory_and_practice
32. S. Arbour, C.T. Kwantes, J.M. Kraft, C.A. Boglarsky, Person-organization fit: Using normative behaviours to predict workplace satisfaction, stress and intentions to stay. J. Org. Cult. Commun. Confl. **18**(1), 41–64 (2014) https://www.researchgate.net/publication/287549304_Person-organization_fit_Using_normative_behaviors_to_predict_workplace_satisfaction_stress_and_intentions_to_stay
33. H. Wu, J.L. Shapiro, Does overfitting affect performance in estimation of distribution algorithms, in *Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation*, (2006, July), pp. 433–434. https://doi.org/10.1145/1143997.1144078
34. C.C. Tu, P.Y. Chen, N. Wang, Improving prediction efficacy through abnormality detection and data preprocessing. IEEE Access **7**, 103794–103805 (2019). https://doi.org/10.1109/ACCESS.2019.2930257

Chapter 5
Telecom Customer Segmentation Using Deep Embedded Clustering Algorithm

R. Jothi and K. Muthukumaran

5.1 Introduction

Telecommunication businesses sought to strengthen client loyalty as a result of greater competition among operators and rising customer churn rates. Generally, telecommunication companies collect a large amount of data. Mobile phone usage, records, network equipment, server logs, billing, and social networks are all examples of this data. This information gives them a lot of insight into their clients and network. Most telecom companies use customer segmentation to increase customer satisfaction, which entails dividing targeted customers into different groups based on demographics or usage perspectives such as gender, age group, buying behaviour, usage pattern, special interests, and other characteristics that represent the customer. According to studies, the cost of finding new consumers for a firm is five times more than the cost of keeping current customers, therefore identifying loyal customers is an important task for the telecom companies. Most telecom companies use customer segmentation to identify the loyal customers [1].

In the telecom industry, which is a typical data-intensive industry, machine learning applications will provide significant recommendations on marketing tactics, and cluster analysis can be used for customer segmentation. Clustering is the process of putting similar data objects together in a group such that objects in one cluster are likely to be similar from objects in another. Cluster analysis is one of the most significant strategies for analysing unsupervised data, and it is employed in a wide range of applications such as computer vision [2], pattern recognition [3] and bioinformatics [4].

R. Jothi (✉) · K. Muthukumaran
Vellore Institute of Technology, Chennai, India
e-mail: jothi.r@vit.ac.in; muthukumaran.k@vit.ac.in

B. Alyoubi et al. (eds.), *Machine Learning and Data Analytics for Solving Business Problems*, Unsupervised and Semi-Supervised Learning,
https://doi.org/10.1007/978-3-031-18483-3_5

Many clustering algorithms have been suggested over the previous years for telecom customer segmentation [1]. Among them, K-means and hierarchical clustering algorithms are extensively used in helping telecom operators to implement customer segmentation and accurately locate customers' market needs. K-means is a partitional clustering algorithm that obtains k-partitioning of the given dataset with the objective of minimizing the sum of squared distances between cluster centers and the objects within the clusters. Starting from a random partitioning of the dataset, the K-means algorithm iteratively reassigns the data objects to nearest centers and this process is repeated until there is no change in the objective criterion. Due to its simple implementation, K-means has been widely employed in telecom customer segmentation [5]. However, the clustering effect is greatly influenced by the number of clusters and the starting center objects chosen, and it may also result in a local minimum until convergence or a predetermined number is reached [6].

Hierarchical clustering is another widely used algorithm for customer segmentation. While K-means directly partitions the dataset into k clusters, hierarchical clustering obtains a tree like representation of the dataset, where root node represents the dataset consisting of all the objects, and the leaves represent the individual objects (also termed as singleton clusters). The intermediate nodes at ith-level in the tree are the result of merging more similar clusters present in the level $(i - 1)$. The tree will be cut at any level to get the desired number of clusters. The visual illustration of clusters in the form of hierarchical tree helps in understanding the relationship between different customer groups in more effective way. However, the hierarchical clustering cannot handle more than a few thousand cases effectively due to its computational complexity. Therefore its challenging to apply this algorithm for massive business data such as telecom data.

Today, with the rapid expansion of smartphones, connected mobile devices, and related services over the internet, the telecom service providers need to handle a lot of data. With more number of attributes and great sparsity of telecom data are, cluster analysis becomes difficult. Thus the traditional algorithms like K-means and hierarchical clustering are inefficient in extracting desired customer groups from such big data. Dimensionality reduction techniques such as principal component analysis (PCA) have been used to solve the problem of curse of dimensionality. PCA obtains a simplified representation of the original dataset by eliminating the redundant features, thus reducing the overall complexity of cluster analysis [1]. However, when dealing with real-world data, PCA may fail to represent and recover original data from projected features [7].

Despite the fact that the notion of segmentation is commonly discussed in literature, research findings on the segmentation of a telecom customer base are extremely rare, owing to the strategic importance of this data. Deep learning, which is a subset of machine learning paradigm, has gained huge popularity in various exploratory analysis. However, deep leaning approaches for telecom customer segmentation are uncommon. Inspired from promising results of deep learning algorithms for different machine learning tasks, this paper aims to present a study on telecom customer segmentation using deep embedded clustering algorithm (DEC). DEC is an unsupervised deep learning algorithm that simultaneously accomplishes feature transformation and clustering of big data [7]. We consider Kaggle's telco customer churn dataset for experimental purpose. Results of the study reveal that DEC algorithm attains good clustering results as compared to conventional clustering algorithms.

The rest of the paper is organized as follows. Section 5.2 presents a brief overview of clustering algorithms applied in telecom customer segmentation. Details of deep embedded clustering methodology are explained in Sect. 5.4. Experimental analysis is given in Sect. 5.4. Conclusion and future directions are given in Sect. 5.5.

5.2 Related Work

Customer clustering is an effective marketing and retention strategy in the telecom business [8]. Among the different clustering algorithms used for telecom customer clustering, K-means has gained huge popularity for its simplicity [5, 9]. Ye et al. [5] applied K-means algorithm to identify potential customer groups from Changzhou telecom data collected for Jiangsu province. The reported results indicated that the resolution of customer segmentation was effective and successful. Jinghua Zhao et al. used an improved K-means algorithm and they reported that the improved algorithm was effective in customer segmentation as compared to standard K-means algorithm.

Although K-means has shown promising results in telecom customer segmentation, the choice of initial center may result in poor clustering. Many authors have used variants of K-means such as K-means++ for segmentation. Instead of choosing all k centroids randomly, K-means++ algorithm iteratively chooses k center objects such that the tth centroid will have maximum distance from previously chosen centroids in iteration 1 to $i - 1$. Recently, Y. Qiu et al. [10] have used K-means++ algorithm for identifying silent customers from telecom data. They have used Calinski-Harabasz index to estimate the number of customer groups (k value). Results have shown that the identified customer groups have supported for the improvement of operation and maintenance management and decision-making of the precision marketing.

Cheng et al. [9] have presented a novel clustering algorithm MQSFLA-k based on k-means and Multivariable Quantum Shuffled Frog Leaping Algorithm (MQSFLA) for identifying target customers for retention. Experimented results have shown that MQSFLA-k algorithm outperformed K-means in terms of convergence rate and clustering accuracy.

Namvar et al. [11] proposed a customer segmentation framework based on Average Revenue Per User (ARPU). The framework determines the marketing strategies to target mobile subscribers intelligently in a way that pushes the ARPU criteria to a higher level and achieves higher data revenue. The framework first removes the outliers using Self-Organizing-Map (SOM) clustering and then segments the pre-processed data using K-means algorithm to identify targeted user groups.

Apart from K-means approach, hierarchical algorithm and Fuzzy clustering are also used in telecom data analysis. Zhang et al. [12] used hierarchical clustering and k-medoids to extract knowledge from sequential behaviour patterns of mobile internet services. Chen, et al. [13] divided mobile subscribers to identify distinct payment habits among customers, with a particular focus on those who had missed payments. Bose and Chen used FCM to demonstrate changes in consumer behaviour as a result of customer movement between clusters.

Being high-dimensional in nature, telecom data poses great challenges for cluster analysis. To this end, researchers have applied dimensionality reduction techniques such as Principal Component Analysis (PCA) [1]. PCA maps the given dataset with d features into a new dataset with d' where $d' \leq d$ features and then applies some clustering algorithms to identify the customer groups. Alkhayrat et al. [1] have used PCA to obtain the reduced dataset and compared the results with deep auto-encoder, which is also a dimensionality reduction technique using deep neural network. Results have shown that auto-encoders outperformed PCA in terms of retrieving most dominating features for customer segmentation.

5.3 Background

Deep learning is an area of machine learning that deals with artificial neural networks (ANN), which are algorithms inspired by the structure and function of the brain. The ANN comprises a collection of connected units called artificial neurons which mimic the behaviour of biological neurons. The neurons are organized into different layers, namely an input layer, one or more hidden layers and an output layer. Each neuron receives input from other neurons, performs some processing, and produces an output. The input layer is the leftmost layer in this network, and the neurons within it are known as input neurons. The output neurons, or in this example, a single output neuron, are found in the rightmost or output layer. The intermediate layers are known as hidden layers where output of hidden layer i is connected as input for hidden layer $i+1$. Deep neural networks are a special kind of neural networks with several hidden layers to learn multiple degrees of abstraction for data representations as shown in Fig. 5.1. Deep learning techniques have been vastly employed in various pattern recognition problems such as speech recognition, visual object recognition, object detection, etc.

Deep learning models for unsupervised learning tasks have also been proposed in the literature [14]. Most of the deep clustering algorithms transform the inputs to a cluster-friendly latent representation and identify cluster membership from the transformed data. The inherent non-linear transformation of deep learning networks help them to achieve significant improvement in clustering performance. Majority of the deep clustering algorithms are based on deep belief neural network, deep auto-encoders, and convolutional neural network (CNN) [15]. Deep belief networks learn distribution of the data using probability-based generative graph model. An auto-encoder uses similar recurrent patterns to learn a low-dimensional representation

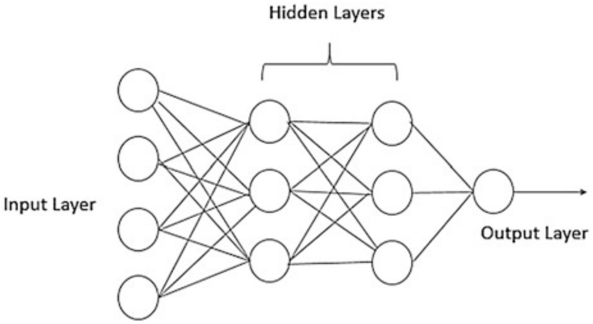


Fig. 5.1 Deep neural network architecture

of the input data. Convolutional neural networks (CNNs) are feed forward neural networks that have been designed to respond to overlapping regions in two-dimensional input fields like images or audio input. The local connectivity feature of CNN enables scalable unsupervised feature extraction from high voluminous data. In this paper, we have used Deep embedded clustering which is based on auto-encoders.

5.4 Methodology

Deep Embedded Clustering (DEC) is a deep neural network-based approach for learning feature representations and cluster assignments at the same time. Traditional clustering methods rely on distance functions to group data objects. DEC, on the other hand, maps the given set of objects into a lower-dimensional feature space and optimizes a clustering objective in this lower-dimensional space simultaneously [16].

Let $X = \{x_1, x_2, \cdots, x_n\}$ be the given dataset of objects, where x_i is a d dimensional feature vector. For customer segmentation problem, each x_i corresponds to a customer record having features such as age, gender, Senior_Citizen, etc. DEC's goal is to map the given dataset given feature space X to lower-dimensional space Z. While learning feature mapping, DEC also learns cluster assignment. The mapping parameter θ is learned using a deep neural network.

The network architecture of DEC is shown in Fig. 5.2. The network consists of encoder and decoder units, each of which implemented as a fully connected deep neural network. An input layer, three dense layers, and one embedded output layer make up the encoder unit. As illustrated in the figure, the decoder unit is configured in the opposite direction as the encoder. The encoder is trained to extract features by minimizing the reconstruction loss objective function L_{rec}. Let f_θ be the encoder function. Let g_θ be the decoder function. Then, reconstruction loss L_{rec} is computed as per Eq. (5.1) [15].

$$L_{rec} = \| x_i - g_\theta(f_\theta(z_i)) \|^2 . \tag{5.1}$$

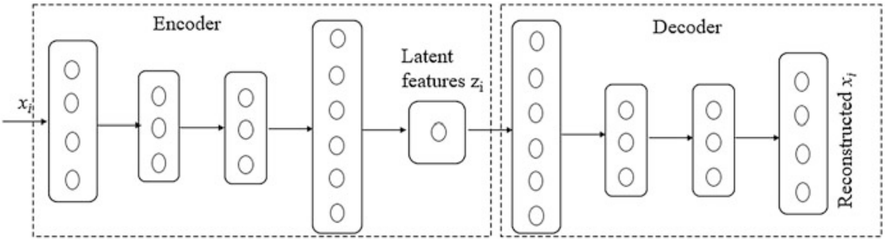


Fig. 5.2 Architecture of deep embedded clustering [17]

The DEC algorithm consists of two-phases. In phase-1, mapping parameter θ is initialized with a deep auto-encoder. Phase-2 clusters the points in Z space using Kullback–Leibler (KL) divergence method. The DEC algorithm begins with an initial estimate of θ and k cluster centroids $\{\mu_j\}_{j=1}^{k}$ in Z-space. The mapped objects are assigned to nearest cluster centroids using a soft assignment. Here student t-distribution kernel q_{ij} is used as a similarity measure to find the distance between embedded object z_i and a centroid μ_j as defined in Eq. (5.2).

$$q_{ij} = \frac{(1+ \parallel z_i - \mu_j \parallel^2 /\alpha)^{-\frac{\alpha+1}{2}}}{\sum_l (1+ \parallel z_i - \mu_l \parallel^2 /\alpha)^{-\frac{\alpha+1}{2}}}, 1 \leq l \leq k, \tag{5.2}$$

where $1 \leq i \leq n$ and $1 \leq j \leq k$. The degrees of freedom α is set to 1. If an object has more similarity to a cluster, then it will be assigned to that particular cluster. The soft assignment is refined by learning from current high confidence assignments using an auxiliary target distribution. The deviation between soft assignment and auxiliary target distribution is computed as clustering loss L_c. The entire process is repeated until convergence, i.e., there is no change in θ, and L_{rec} and L_c are negligible. The complete steps in DEC algorithm are illustrated in the flowchart shown in Fig. 5.3.

5.5 Experimental Analysis

This section presents details of telecom dataset used in the study, evaluation metrics for assessing the clustering performance, and the results of customer segmentation.

5.5.1 Dataset

Telco customer churn dataset from Kaggle repository is considered for the present study. The dataset includes 7043 samples and 20 features. Each sample represents

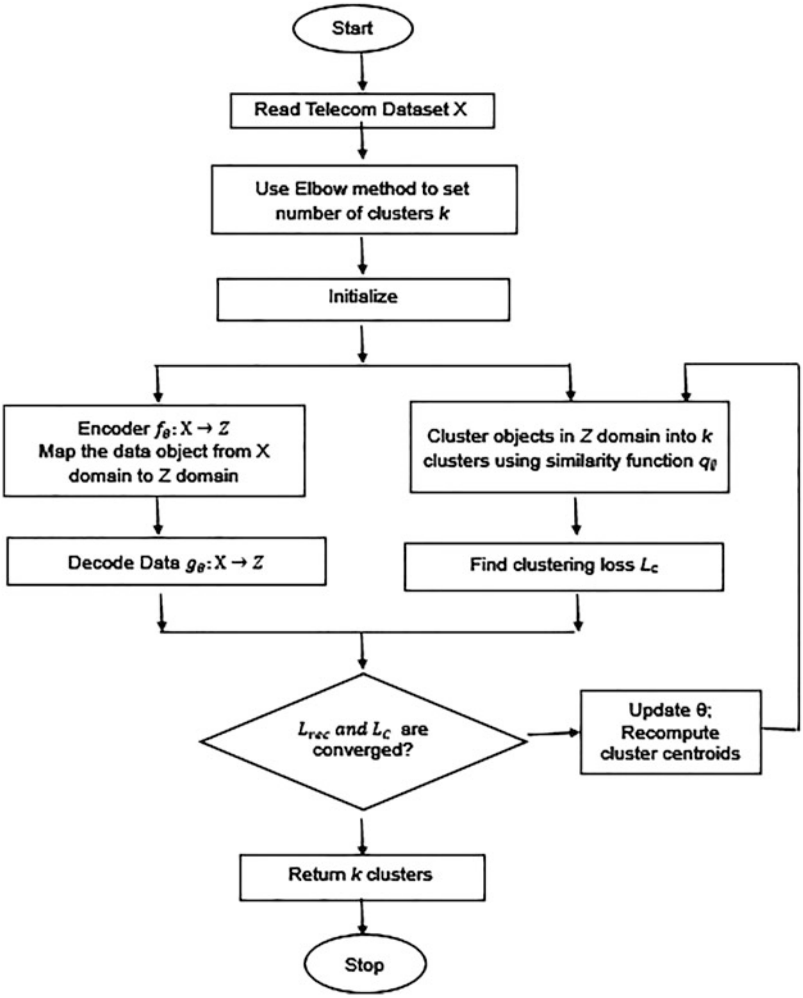

Start
Read Telecom Dataset X
Use Elbow method to set number of clusters k
Initialize
Encoder $f_\theta: X \to Z$ Map the data object from X domain to Z domain
Decode Data $g_\theta: X \to Z$
Cluster objects in Z domain into k clusters using similarity function q_θ
Find clustering loss L_c
L_{rec} and L_C are converged?
Update θ; Recompute cluster centroids
Return k clusters
Stop

Fig. 5.3 Steps in deep embedded clustering algorithm

various details of telecom customers such as their demographic details (age, gender, dependents), services that they have opted for, contract, payment method, monthly charges, total charges, etc. The last column of the dataset, called as churn, indicates whether the customer has left the telecom network or not. The dataset comprises diverse set of features such as demography, location, and services. The features are mixed set of categorical and numerical features. The heterogeneous nature of features makes it difficult for cluster analysis.

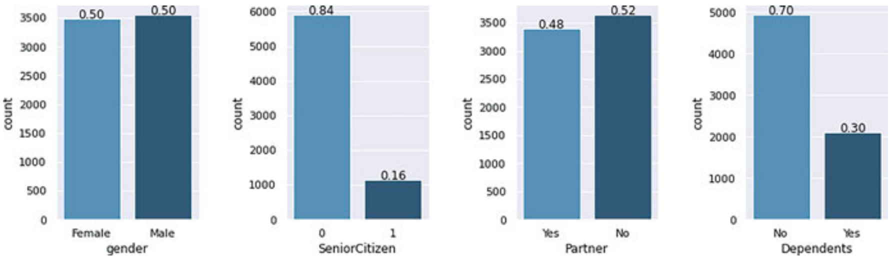


Fig. 5.4 Demographic distribution of customers with respect to their gender, senior citizen, partner and dependent

5.5.2 Exploratory Analysis of the Dataset

Exploratory analysis of the dataset is carried out and the distribution of samples with respect to different feature values are analysed. Figure 5.4 illustrates the distribution of customers with respect to their demographic information. It is seen from the figure that gender attribute seems to be having equal distribution and most of the customers are younger people. Figure 5.5 illustrates how these demographic values affect the churn rate. It is observed that Senior Citizens have a significantly higher rate of churn. Customers who have a partner appear to be less prone to leave. Customers without dependents are less likely to churn in general, yet they account for the majority of churned customers.

Similar sample distribution analysis is carried out with respect to customer account details such as tenure, contract, payment method, and billing information. Tenure, number of months the customer has stayed with the company, is also considered to be vital information in analysing customer retention behaviour. As illustrated in Fig. 5.6, the majority of consumers are on a month-to-month contract, with the proportion of customers on one-year and two-year contracts nearly equal. The electronic check method has the most customers for payment, while the others are nearly equal. Paperless billing is used by 58% of customers, and there are only 10% variances between paper and electronic billing. Services opted by the customers are also vital information in deciding the churn rate. Figure 5.7 reports that phone services have a bigger customer base, whereas other services are practically on par. Relationship between customer account information and the churn rate is depicted in Fig. 5.8. Customers who have been with the telecom company for more than 30 months have less chance of churning as observed in shown Fig. 5.8. It is evident that customers on monthly contracts are the most likely to leave among those who have churned, and the customers who opted for paper billing have less churn rate. Monthly charges also have an effect on churn rate as seen in the figure.

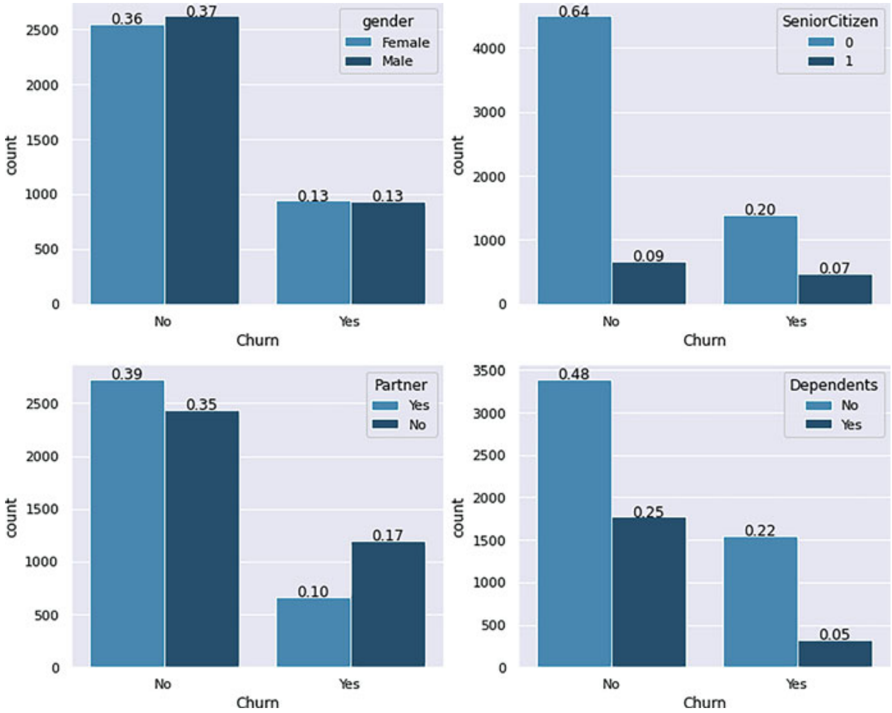


Fig. 5.5 Relationship between churn rate and demographic information

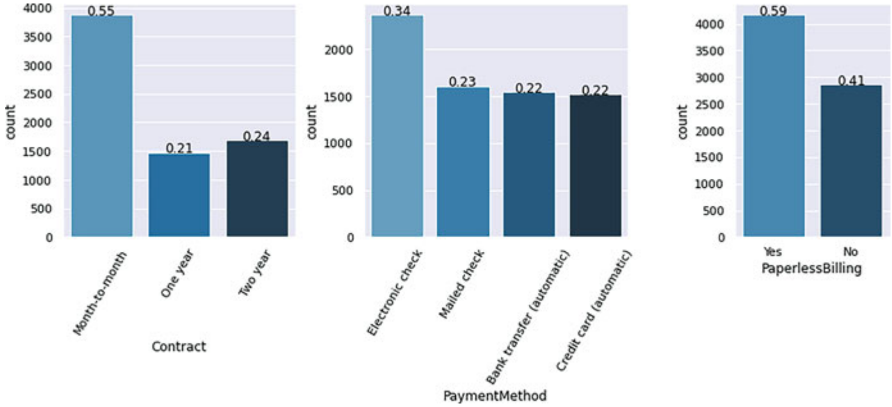


Fig. 5.6 Distribution of customers with respect to billing

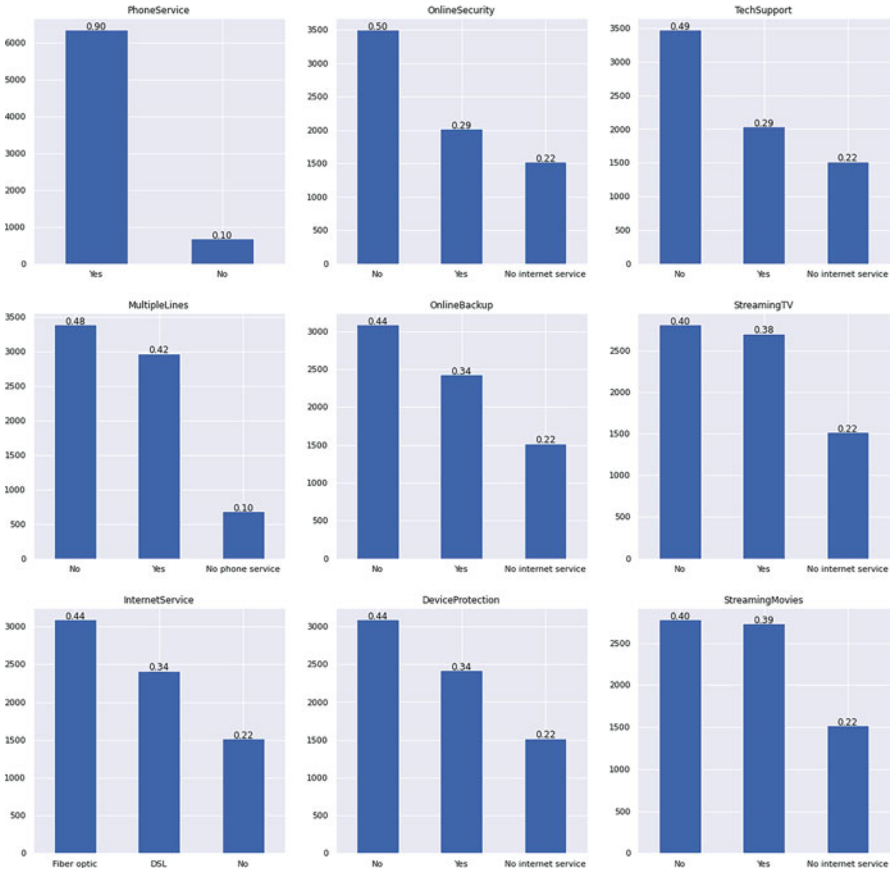


Fig. 5.7 Distribution of customers with respect to opted services

5.5.3 Number of Clusters

As we do not know the number customer groups, we have used Elbow method to determine the optimal number of clusters (k). The Elbow method tries to figure out the optimal value of k using within-cluster variance analysis. For varying values of k, total within-cluster sum of squared distances $WCSS$ is computed, and the k value for which WCS is minimum is considered as the optimal value for k. Figure 5.9 illustrates this analysis and shows the optimum value is at $k = 3$.

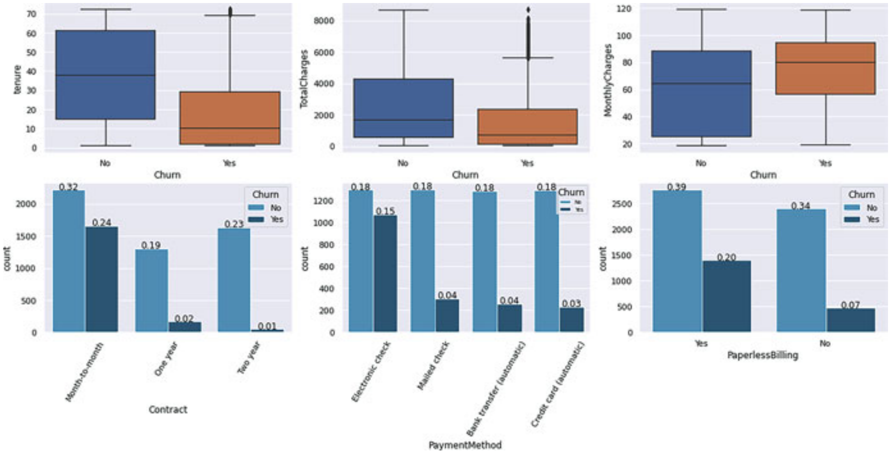
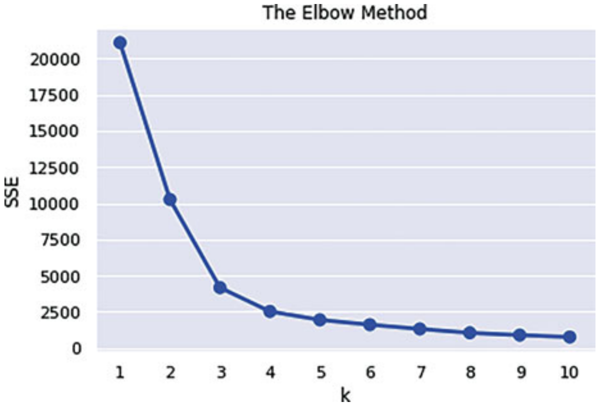


Fig. 5.8 Relationship between churn rate and account information

Fig. 5.9 Determining the number of clusters using Elbow method



5.5.4 Telecom Customer Segmentation

Deep embedded clustering (DEC) is applied in the customer segmentation process of telecom dataset to infer loyalty characteristics of different customer groups. Identification of loyal customers using Recency, Frequency, and Monetary (RFM) model has shown to provide promising results as reported in the literature [18]. For the present study, the recency could not be applied as last contract information is not provided with th dataset. So we choose tenure, frequency, and monetary value as the three loyalty characteristics of the RFM model, where total charges is considered for monetary value and frequency is computed as $tenure * contract/24$. First, the telecom dataset is partitioned into 3 clusters using DEC algorithm and then loyalty characteristics are analysed for these clusters. The results are compared with

Table 5.1 Silhouette index (SI) and Davies-Bouldin Index (DBI) scores obtained by different algorithms on Telecom customer segmentation

Algorithm	SI	DBI
K-means	0.2194	1.9283
HAC	0.2885	1.4470
DEC	0.4303	0.6785

Table 5.2 Loyalty characteristics of clusters obtained by DEC algorithm on Telecom customer segmentation. Each value in the table denotes per-cluster mean of that attribute

Algorithm	Tenure	Frequency	Monetray value
Cluster 1	31	0.43	1065
Cluster 2	55	2.56	4928
Cluster 3	16	0	565

conventional clustering algorithms such as K-means and Hierarchical agglomerative clustering.

The performance of different clustering algorithms is compared using cluster validity measures such as Silhouette index (SI) and Davies-Bouldin Index (DBI). The SI measures the degree to which the objects in a cluster are cohesive while mainlining the inter-cluster separation. Higher the SI value, more cohesive and compact clusters obtained by the clustering process. The DBI score is a function of the ratio of within cluster scatter to between cluster separation. A lower value of DBI indicated better clustering.

SI and DBI scores for K-means, HAC, and DEC algorithms on Telecom customer segmentation are measured and reported in Table 5.1. While the conventional algorithms may not be able to perform well on high-dimensional datasets, the DEC algorithm works effectively as it performs low-dimensional feature mapping and cluster assignments simultaneously. This is evident from Table 5.1 that DEC algorithm has achieved more cohesive clusters as compared to K-means and HAC.

Next, statistical measures such as mean and standard deviation of the loyalty characteristics are computed for each of the obtained clusters. Table 5.2 reports the loyalty characteristics of 3 clusters obtained by DEC algorithm. It can be concluded from the results that different customer groups have the following characteristics. The customer groups identified by the DEC algorithm clearly indicate that customers having higher tenure tend to opt more services and bring more revenues to the telecom company as indicated by their total charges. We have also analysed the three clusters with respect to other attributes such as domestic attributes and additional services opted. Customers in Cluster-1 and cluster-2 have opted for more additional services than the customers in cluster-1. However, we could not find any common characteristics among customers in a particular cluster with respect to domestic attributes. For example, young age customers are scattered in all the 3 clusters. Similarity, the customers have dependents also distributed across all the three clusters.

Table 5.3 Silhouette index (SI) and Davies-Bouldin Index (DBI) scores obtained by different algorithms on E-commerce dataset

Algorithm	SI	DBI
K-means	0.1079	4.1556
HAC	0.2122	3.0065
DEC	0.5514	1.0038

Table 5.4 Silhouette index (SI) and Davies-Bouldin Index (DBI) scores obtained by different algorithms on Mall customer segmentation dataset

Algorithm	SI	DBI
K-means	0.2967	0.9987
HAC	0.2547	1.0019
DEC	0.3405	0.3800

5.5.5 Analysis on Other Customer Segmentation Datasets

We have analysed performance of DEC algorithm on other customer segmentation datasets, namely E-commerce dataset and Mall customer segmentation dataset. Both of the datasets are available in Kaggle. Here also we have applied Elbow method to identify the number of clusters k, and the observed values of k are 3 and 5, respectively, for E-commerce dataset and Mall dataset. All the three clustering algorithms K-means, HAC, and DEC algorithms have been applied on both these datasets and the results are reported in Tables 5.3 and 5.4. It is noted that DEC algorithm outperformed on both the datasets as compared to K-means and HAC.

5.6 Conclusion and Future Scope

With the rapid development of telecom networks, telecom companies' databases now include a massive amount of business data. Analysing such huge voluminous data for identifying loyal customers to improve business solutions has become one of the challenging problems. Although customer segmentation is widely discussed, the use of deep learning for customer segmentation, especially for telecom customer segmentation, is limited. The present study explores application of both traditional and deep clustering algorithms for customer segmentation problem with the intension of finding a method which provides improved segmentation accuracy and scalability. It has been observed that deep embedded clustering algorithm (DEC) outperformed than traditional methods. The inherent non-linear transformation of deep learning networks, they achieve significant improvement in clustering performance and this is evident from the experimental analysis. The results report that DEC algorithm is able to attain better clustering performance as compared to K-means and HAC algorithms. Also, the identified clusters depict distinct customer groups with respect to loyalty attributes such as tenure, Monetary_Values, and frequency. As a future work, we explore the suitability of different clustering frameworks for customer segmentation problem.

References

1. M. Alkhayrat, M. Aljnidi, K. Aljoumaa, J. Big Data **7**(1), 1 (2020)
2. X. Zheng, Q. Lei, R. Yao, Y. Gong, Q. Yin, EURASIP J. Image Video Process. **2018**(1), 1 (2018)
3. R. Jothi, Clustering time-series data generated by smart devices for human activity recognition (2020)
4. R. Jothi, in *International Conference on Mining Intelligence and Knowledge Exploration* (Springer, Berlin, 2017), pp. 35–42
5. L. Ye, C. Qiu-Ru, X. Hai-Xu, L. Yi-Jun, Y. Zhi-Min, in *2012 7th International Conference on Computer Science & Education (ICCSE)* (IEEE, Piscataway, 2012), pp. 648–651
6. Y. Deng, Q. Gao, Inform. Syst. e-Bus. Manag. **18**(4), 497 (2020)
7. X. Guo, L. Gao, X. Liu, J. Yin, in *Ijcai* (2017), pp. 1753–1759
8. H. Wibowo, K.P. Sinaga, in *2021 3rd International Conference on Cybernetics and Intelligent System (ICORIS)* (2021), pp. 1–6. https://doi.org/10.1109/ICORIS52787.2021.9649598
9. C. Cheng, X. Cheng, M. Yuan, C. Song, L. Xu, H. Ye, T. Zhang, in *2016 16th International Symposium on Communications and Information Technologies (ISCIT)* (IEEE, Piscataway, 2016), pp. 324–329
10. Y. Qiu, P. Chen, Z. Lin, Y. Yang, L. Zeng, Y. Fan, in *2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, vol. 1 (2020), pp. 1023–1027. https://doi.org/10.1109/ITNEC48623.2020.9084976
11. A. Namvar, M. Ghazanfari, M. Naderpour, in *2017 12th International Conference on Intelligent Systems and Knowledge Engineering (ISKE)* (2017), pp. 1–6. https://doi.org/10.1109/ISKE.2017.8258803
12. X. Zhang, C. Wang, Z. Li, J. Zhu, W. Shi, Q. Wang, Electron. Commerce Res. Appl. **17**, 1 (2016)
13. C.H. Chen, R.D. Chiang, T.F. Wu, H.C. Chu, Expert Syst. Appl. **40**(16), 6561 (2013)
14. E. Aljalbout, V. Golkov, Y. Siddiqui, M. Strobel, D. Cremers, arXiv preprint arXiv:1801.07648 (2018)
15. E. Min, X. Guo, Q. Liu, G. Zhang, J. Cui, J. Long, IEEE Access **6**, 39501 (2018)
16. R. Jothi, in *Proceedings of 2nd International Conference on Artificial Intelligence: Advances and Applications* (Springer, Singapore, 2022), pp. 841–849
17. J. Xie, R. Girshick, A. Farhadi, in *International Conference on Machine Learning* (2016), pp. 478–487
18. D. Chen, S.L. Sain, K. Guo, J. Database Marketing Customer Strategy Manag. **19**(3), 197 (2012)

Chapter 6
Semantic Image Quality Assessment Using Conventional Neural Network for E-Commerce Catalogue Management

Sonia Ouni, Karim Kamoun, and Mohamed AlAttas

6.1 Introduction

Over the past decade of the twentieth century, the development of communication technologies has paved the way for innovations, fostering rapid globalization. With the convergence of telecommunications and computer technology, a new business organizational system has been born which is called the Internet presenting a revelation of ecological business development [1]. E-commerce, or electronic commercial transactions over the Internet, has evolved so quickly in the last 5 years that many experts continue to underestimate its growth and expansion [2].

However, with such easiness, we can access thousands to millions of products online for any given category. To organize products on various online channels, an e-commerce website manager should consider catalogue management solutions. It is a dynamic process in which products are categorized or structured in a consistent manner across all channels to guarantee that customers receive high-quality product data. Catalogue management is a crucial part of developing an online brand since it helps customers find what they're looking for and feel confident buying based on the given information [3]. Therefore, varying sorts of products necessitate different levels of detail and information. That's why it's important to ensure customers are given as much detail as possible such as image, color options, prices, sizes, country

S. Ouni (✉)
College of Business, University of Jeddah, Jeddah, Saudi Arabia

Laboratoire LIMTIC, Institut Supérieur d'Informatique, University of Tunis El Manar, Tunis, Tunisia
e-mail: souni@uj.edu.sa

K. Kamoun · M. AlAttas
College of Business, University of Jeddah, Jeddah, Saudi Arabia
e-mail: kkamoun@uj.edu.sa; malattas@uj.edu.sa

B. Alyoubi et al. (eds.), *Machine Learning and Data Analytics for Solving Business Problems*, Unsupervised and Semi-Supervised Learning,
https://doi.org/10.1007/978-3-031-18483-3_6

of origin, materials, specifications, and more which are all included in the product catalogue for online retailers, so they feel comfortable making their purchase [4]. Despite the convenience of Internet buying, it's usually more difficult to make purchasing judgments, especially when the image quality of the products is unclear or not good [5]. In fact, a poor image quality of products influences the effectiveness of the catalogue management which leads to imprecise or insufficient information on a product. As a result, sales are lost, and customers are more and more hesitant to buy [6]. It might also mean more returns, which could result in slow and reverse revenue streams, as well as a loss of brand loyalty. Image quality assessment (IQA) can be a solution to address this issue [7].

It is critical in image processing, particularly image enhancement. To date, there is no common approach to assessing the quality of images in different automated image processing techniques. There are two types of image quality assessment: subjective and objective. Commonly, subjective methods are a classical approach to evaluating perceived quality. The overall quality of a specified image test must be scored by human assessors. Indeed, the subjective approach reflects more faithfully human perception, but it is expensive, is time-consuming, and cannot be automated. Hence, objective metrics have received increased attention in recent years. This approach can be classified into three categories: full reference (FR), reduced reference (RR), and no reference (NR).

The full reference (FR) is based on the measurement of differences between the original image or video and the distorted one. FR metrics can be classified into two subcategories: the first is based on mathematical measures like mean squared error (MSE), mean absolute error (MAE), root mean square error (RMSE), mean absolute error (MAE), signal to noise ratio (SNR), peak signal-to-noise ratio (PSNR), and spatial color image quality metric (SCID) [8]. The second is based on characteristics of the human visual system (HVS) [9].

The reduced-reference (RR) image quality measures aim to predict the visual quality of distorted images with only partial information about the reference images. RR image quality metrics provide a solution that lies between the FR and the NR. The RR methods have a wide range of applications. For example, in real-time visual communication systems, these metrics can be used to monitor image quality degradations and manage streaming resources [10].

Finally, the no reference (NR), which looks only at the image or video under test, has no need for reference information [11]. It is complicated in such applications (catalogue, digital photography, printing, etc.) where a reference is not available to measure the quality. Since the human visual system (HVS) never requires a reference to define the quality of an image it receives, the main issue is determining how to judge image quality in the absence of a reference. However, deep learning approaches especially conventional neural networks (CNNs) have shown some promising performance in image quality assessment [12–14]. The success of CNN in IQA is inspired by the human visual cortex, and it can learn quality-aware features by itself provided with representative training data. This technique brings a new research direction with promising applications but little advance [15]. Most of these works are not interested to the color and depend on a specific distortion and especially from applications.

In this paper, we propose a new no-reference image quality assessment approach based on CNN noted: CNNs_SIQA for human perceived assess semantic quality image color. It is based on multi-deep convolutional neural networks (CNNs). The correlations between image distortions and human perceived quality are analyzed, and relevant parameters are extracted. Different transformations are realized based on the semantic quality definition. A new image database for testing the image quality assessment approach is presented. The performance is compared to another metric of the state of the art.

The rest of the paper is organized as follows: Sect. 6.2 describes the different NR IQA using neural network and discusses the related work showing the reason why objective image quality assessment is important and necessary. In Sect. 6.3, the new CNNs_SIQA is proposed. A new image database for testing image quality assessment metrics is presented in the next section. We present experimental results and correlation with subjective image quality assessment in Sect. 6.4. Finally, Sect. 6.5 draws conclusions and provides future works.

6.2 Related Work

The field of NR IQA is very young, and there are many possibilities for the development of innovative metrics. Recently, deep learning has gained researchers' attention and achieved great success on various computer vision tasks. It has attracted a great deal of attention since the turn of the century. However, only few methods using deep learning have been proposed in the literature [15, 16]. Until now, there is no generic and common approach to evaluate the quality of image. In this part, we explained and analyzed several NR IQA using deep learning.

In our previous work [17], we discuss the basic problems of NR IQA which consist of the little progress in this new research direction with promising applications. It turns out to be a very difficult task to assess the image quality objectively and blindly, although human observers usually can effectively and reliably assess the quality of distorted images without using any reference. This is mainly due to the limited understanding of the HVS and the corresponding cognitive aspects of the brain. The problem of NR IQA is made even more complex since many unquantifiable factors play a role in the human assessment of quality, such as aesthetics, cognitive relevance, learning, visual context, and so on. These factors introduce variability among human observers based on everyone's subjective quality assessment.

Most works are based to evaluate the quality assessment in the domain of compression. In the reference [18], a NR method for assessing JPEG image quality is proposed, using a sequential learning algorithm to grow and prune a radial basis function (GAP-RBF) neural network. The metric is designed to account for the 8×8 blocking artifacts that arise from JPEG encoding. This approach proceeds by extracting features to quantify the encoding distortions. These features are then used in conjunction with the GAP-RBF network model to predict image quality. Moorthy et al. in [19] propose a NR IQA approach based on a two-level process. The first one is essentially a classification stage in which the test image is assigned a probability p_i of belonging to a set of distortion classes $i = 1, \ldots, N$, where N is the number of distortion classes. The second one then performs an SVM-based quality assessment in a specific distortion class. The [20] is to perform a comparative study of seven well-known no-reference learning-based image quality algorithms. To test the performance of these algorithms, three public databases are used. As a first step, the trial algorithms are compared when no new learning is performed. The second step investigates how the training set influences the results. The authors Kang et al. [21] proposed a deep model which trains the CNN using spatially normalized image patches. The quality and distortion types are identified simultaneously using a multitask CNN architecture. Bianco et al. [22] pre-trained a deep model on the large-scale database for image classification task and then fine-tuned it for NR IQA task. They have proposed a deep learning for distortion-generic blind image quality assessment named DeepBIQ which is based on CNN. They have used an AlexNet-like architecture, which is pre-trained on ImageNet. They have extracted features from this deep network by taking multiple crops of 224 9 224 and then average-pooled the features to train a support vector regression (SVR). Gao et al. [23] have defined an effective BIQA method, code-named BLINDER (BLind 42 Image quality prediction via multi-level DEep Representations). This method used a Vgg16 architecture for quality assessment and reasoned those different levels of convolution represent image quality differently. They trained an SVR on features extracted at different levels of convolution and then averaged pooled to provide image quality. Kim et al. [24] proposed a novel NR IQA framework called deep blind image quality assessor (DIQA) composed of a two-stage image quality predictor. The first stage predicts an objective error map, and the second stage predicts a subjective quality score. It was trained using all the images from one database and then tested on another database. Four overlapping distortion types (white noise, GB, JPEG, and JPEG2000) were used in the CSIQ and TID2013 databases.

In the other work, Ravela et al. [25] have proposed a novel architecture that extracts deep features from the input image at multiple scales to improve the effectiveness of feature extraction for NR IQA using convolutional neural networks. This architecture was composed of a two-step approach. The first steps predict the distortion type, and the second stage uses a specialized deep model to predict the subjective score and perform average pooling with the prediction of other deep models. The experiment results showed the performance favorably against the state-of-the-art methods on three large benchmark datasets with authentic distortions (LIVE In the Wild, KonIQ-10k, and SPAQ).

Ma et al. [26] decomposed the BLIND image quality assessment problem into a two-stage network: the first identifies the type of distortion by classifying an image into a specific distortion type from a set of predefined categories, and the second stage performs the subjective score prediction using a specialized quality prediction network for each distortion type.

In the reference [27], the authors also proposed a two-stage framework; the first one learns feature representations and identifies nine distortion types as well as the level of distortion ranging from two to five. The second one performs quality assessment by proposing a model named a deep bilinear CNN (DBCNN). This model has focused to perform quality assessments for both authentically distorted and inauthentically distorted images.

The authors Fan et al. in [28] developed a novel NR IQA algorithm based on multi-expert convolutional neural networks (CNNs). This approach follows a two-stage by first predicting the distortion type and then performing subjective score prediction using multiple CNNs. The [29] is based on a newly collected perceptual similarity dataset, using a large set of distortions and real algorithm outputs. It contains a deep feature composed of both traditional distortions, such as contrast and saturation adjustments, noise patterns, filtering, and spatial warping operations, and CNN-based algorithm outputs, such as autoencoding, denoising, and colorization, produced by a variety of architectures and losses for image quality assessment. The reference [30] proposed both the distorted and corresponding reference images are fed into deep convolutional neural networks (CNN), and the output of each layer is utilized to model the distortion process. The features extracted from ImageNet-pre-trained models such as VGG are used to train a regression algorithm for image quality assessment. A deep CNN training approach is proposed for quality assessment by [31] using a synthetic distortion database with 18.75 million images.

In general, the main problem in the existing NR IQA is complicated by the absence of a reference image. Therefore, a lot of effort has been devoted to developing NR IQA that tries to mimic human perception. The most existing NR IQA is designed for grayscale images [32]. They ignore the correlations between color information and quality distortions, and lack of investigation of the feature representations of color information. Consequently, the prediction accuracy and generalization performance on color images are not satisfactory [32], and due to the diversity of the distortion types and image contents, it is difficult for the existing NR IQA to be applied and maintain the best performance for all cases. To solve this problem, we propose, in the next section, a semantic Color IQA with multi-deep convolutional neural networks (CNNs) noted: CNNs_SIQA.

6.3 Proposed Approach

The proposed semantic color image quality assessment with multi-deep CNNs (CNNs_SIQA) is shown in Fig. 6.1. Firstly, a group of component maps is generated with multi-transformation to the color image. These transformations are based on the definition of semantic quality image proposed in our previous work [33]. Secondly, the general CNN structures are adopted and improved to learn useful feature representations from the component maps. Each component map is the input

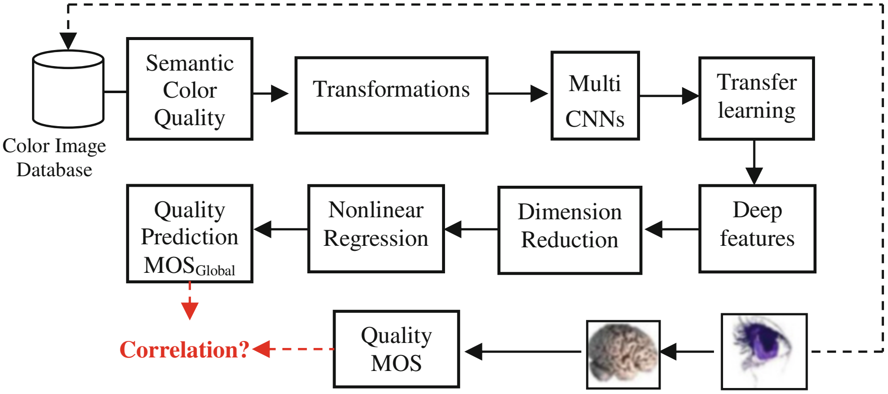


Fig. 6.1 The proposed semantic Color IQA with multi-deep convolutional neural networks

of a single CNN, thus forming a multi-CNN model. Thirdly, each CNN is trained using the transfer learning method with pre-trained models on large-scale datasets. Then, multiple output feature eigenvectors are fused as deep features. Finally, a nonlinear regression model is constructed to map the extracted deep features to the visual quality scores. The detailed steps are described as follows.

6.3.1 Transformation

Starting from the proposed definition of the semantic quality in our previous work [33], very specific metrics have been defined to model this issue. Very few works on the evaluation of image quality in the literature attempt to propose a definition of image quality in such a way as to define these questions:

1. What are images?
2. What are images used for?
3. What are the requirements and which of these images are imposed on them?

The definitions vary widely in the literature, and this fact justifies the many proposed methods for evaluating image quality. The image quality semantic is defined by (1) the usefulness of an image to be the precision of the internal representation of the image and (2) the naturalness of an image to be the degree of correspondence between the internal representation of the image and knowledge of reality as stored in memory. Using these definitions, we define the quality of an image to be the degree to which the image is both useful and natural.

Our goal is to define the transformation that models the image quality semantic (Fig. 6.2). The usefulness of an image is represented by two criteria:

- Sharpness and clarity

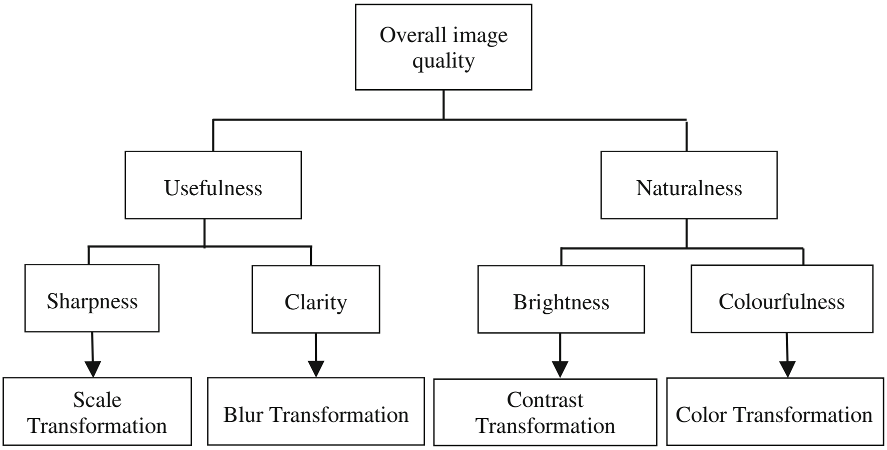


Fig. 6.2 The proposed transformations based on semantic color image quality assessment

- High level of details in dark and bright regions of the image which is provided by high local contrast

Although the key point here is the main goal for which the user intends to use the image for good naturalness, images are as follows:

- High overall contrast
- Rather uniform distribution of lightness levels meaning that image should not be too dark or too bright
- A chromatic diversity of color in the image

The main difficulty here is how to model every criterion by representing transformation that gives a value denoting image visual quality, and it should correspond to all these demands. Figure 6.2 represents the different criteria of the semantic quality image where each one is modeled by a set of transformations.

6.3.1.1 Color Transformation

The color is a powerful descriptor that often simplifies the object identification and extraction from a scene so color information also could influence human beings' judgments [34]. It can be represented in different color spaces, so the most important are:

RGB The RGB model is mostly used in hardware-oriented applications such as color monitor. In the RGB model, images are represented by three components, one for each primary color: red, green, and blue. Although the human eye is strongly perceptive to red, green, and blue, the RGB representation is not well suited for describing color images from a human perception point of view.

CIE LAB This color space is originally defined by CAE and specified by the International Commission on Illumination [35, 36]. In this color space, we have one channel for luminance (lightness) and the other two-color channels for a and b known as chromaticity layers. The a* layer indicates where the color falls along the red, green axis, and b* layer indicates where the color falls along the blue-yellow axis. a* negative values indicate green, while positive values indicate magenta, and b* negative values indicate blue, and positive values indicate yellow. The most important feature of this color space is that this is device independent [37], which means to say that this provides us the opportunity to communicate different colors across different devices.

CIE HSV The CIE HSV model is motivated by the human visual system, which is a simplified expression of Munsell color space. A three-dimensional representation of the HSV color space is a hexagon, where the central vertical axis represents the intensity. It uses hue (H), saturation (S), and value (V) to express colors. Hue (H) is defined as an angle in the range [0,2S] relative to the red axis at angle 0, green at 2S/3, blue at 4S/3, and red again at 2S. Saturation (S) is the depth or purity of the color and is measured as a radial distance from the central axis with a value between 0 at the center and 1 at the outer surface. The HSV color model is defined as follows [38]:

$$H = \begin{cases} 60\left(\frac{\mathrm{G-B}}{\gamma}\right) & \text{if}\,\mathrm{MAX} = \mathrm{R} \\ 60\left(\frac{\mathrm{G-B}}{\gamma}+2\right) & \text{if}\,\mathrm{MAX} = \mathrm{G} \\ 60\left(\frac{\mathrm{G-B}}{\gamma}+4\right) & \text{if}\,\mathrm{MAX} = \mathrm{B} \\ \text{Not defined} & \text{if}\,\mathrm{MAX} = 0 \end{cases} \tag{6.1}$$

$$S = \begin{cases} \frac{\gamma}{\mathrm{MAX}} & \text{if} \quad \mathrm{MAX} \neq 0 \\ 0 & \text{if} \quad \mathrm{MAX} = 0 \end{cases}$$

$$V = \mathrm{MAX}$$

where δ = (MAX − MIN), MAX = max(R,G,B), and MIN = min(R,G,B). It is more natural for the human visual system to describe a color image by the HSV model than by the RGB model. For $S = 0$, as one moves higher along the intensity axis, one goes from black to white through various shades of gray. On the other hand, for a given intensity and hue, if the saturation is changed from 0 to 1, the perceived color changes from a shade of gray to the purest form of the color represented by its hue.

The properties of the HSV color space are more congruent with human perception of colors than the RGB color space because it includes three aspects. To begin with, the brightness component has nothing to do with the color information in the image. Second, the H and S colors are quite like how the human eye perceives color. As a result, with HSV color space, human eyes perceive chromatic aberration more consistently. For visual perception-based image processing algorithms, the HSV color space makes sense [32].

6.3.1.2 Scale Transformation

Image quality is affected by both the local details and global composition. To capture both the global and local information, we propose to model the input image with a multi-scale representation. Also, the input image size of deep CNN is limited by the image size of the pre-training source dataset, which is 256 × 256. However, it is important to adapt the input to the network if the image has a different size by using scale transformation. This transformation depends on the size of width and height of the input image and classifies it into two cases. The first, when the width and height are similar, in this case, the transformation scale returned to zoom the image to the specific size. The second one, when the size of the width and height is different, there are a lot of methods of multi-scale transformation [32]. In our work, we adopt three types of them: scaling, filling, and cropping. The scaling method is a direct scaling by zooming the image which provides some distortion such as the degradation of visual clarity and changes the gray level. The filling approach involves average filling of the image's short edge while fixing the image's long edge. The ratio of the long edge to 256 transforms the image. The short edge's center is utilized as the center point, and the gray value 0 is used to assign the extra areas on both sides. The clipping method is based on fixing one of the short edges after the overall scaling of the image and clipping the long edges beyond 256, that is, intercepting the 256 × 256 area of the central image. However, the two lasted methods don't affect the visual quality of the original image. Therefore, there are selected in the pre-training phase.

6.3.1.3 Contrast Transformation

Image quality can be degraded due to various types of distortion. However, the contrast distortion is among the most common and fundamental distortion [39]. Contrast-distorted image (CDI) is an image with low range of grayscale. It may be caused by poor lighting condition and poor-quality image acquisition device.

Then, contrast-changed images are generated for each original image by using the two transfer curves shown in Fig. 6.3. About 16 images from each group of contrast-changed images are processed using the 2 transfer curves shown in Fig. 6.3. The remaining 8–12 contrast-changed images are created by mean shifting the original image X, either in a positive (+4X) or negative (−4X) direction. The shifts

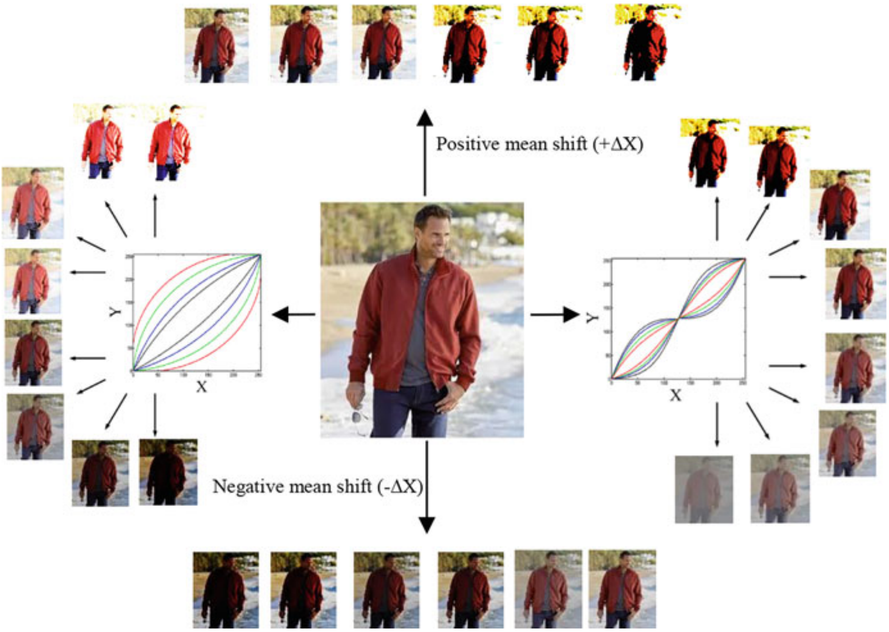


Fig. 6.3 Example of derived contrast-changed images from electronic catalogue

4X have six levels of [20, 40, 60, 80, 100, 120]. The out-of-bound values in the mean shift are clipped into the range of 0–255. Figure 6.3 presents an example of image and its derived contrast-changed images.

6.3.1.4 Blur Transformation

Blur is one of the most common distortions, and unintentional blur impairs image quality. There are a lot of situations that occur with image blur such as out of focus, object motion and camera shake, nonideal imaging systems, atmospheric turbulence or aerosol, scattering/absorption, and image compression and image denoising [40].

There are many methods to add blur in the image such as Gaussian blur, lens blur, and motion blur. For instance, we propose to use lens and motion blur in addition to Gaussian blur which is a filter with a variable Gaussian kernel. The former distortions are more likely to occur in the wild due to defocus and camera shake, whereas Gaussian blur is mostly a result of image editing. However, the lens blur is a filter with a circular kernel, and the motion blur is a filter with a line kernel. For every distortion we take five levels of degradations.

6.3.2 Multi-CNN Architecture Design

Convolutional neural networks (CNNs) are the most popular type of neural networks for working with image data because of its high representation capabilities and enhanced performance [41].

In this subsection, we explain the details of the proposed approach of the multi-CNN model. Figure 6.4 shows the overall structure design of the proposed method, starting from raw images and ending with attribute predictions. It is composed by several single-channel CNN models. Each CNN's training result can be thought of as a deep feature extractor. Multi-CNNs can extract the semantic quality feature representation of multicolor component maps and better characterize image quality distortion features, particularly image color information. The extracted from several network models are synergistically combined, and the principle of ensemble learning improves the performance of the image quality prediction even more. The different steps are as follows. Firstly, from the definition of semantic quality, the image is translated into several scales, color spaces, degrees of contrast, and blur. A group map space is made up of different scales of altered images, different color spaces, and their color components. Each component map will be utilized as an input to a single CNN, and CNN learning will be done using transfer training.

Then, to extract the feature representation closer to the IQA task, the global single CNN network structure is improved. The global network structure N_{Global} is adequately extended. The network layers that come before the full connection levels are kept, and three full connection (FC) layers are added subsequently. As an example, we can take AlexNet [42] which is a breakthrough in the history of deep learning. It consists of five convolutional and three fully connected layers. The last FC layer for outputting classification probability is deleted, while the number of full connection layers is raised after FC7. The output vector dimensions of each fully connected layer selected by the approach are 2048 (FC8), 1024 (FC9), and 512 (FC10), respectively, based on the amount of information, feature dimension, and complexity of further processing.

The fusion of output eigenvectors from several N_{SIQA} models is, after, accomplished using the Caffe framework's Concat layer. To create a multi-*CNNs SIQA*, each component map is used as the input of a single CNN network. The feature eigenvectors are created by collaborating numerous outputs after transfer learning on a single network. CNNs SIQA will be formed by different combinations of maps. Later, the effects of various feature vector combinations on image quality prediction performance will be compared and examined.

6.3.3 Network Training

There are different steps in transfer training of CNNs SIQA. In the first one, the global single CNN network N_{Global} is selected, and the related pre-trained results on

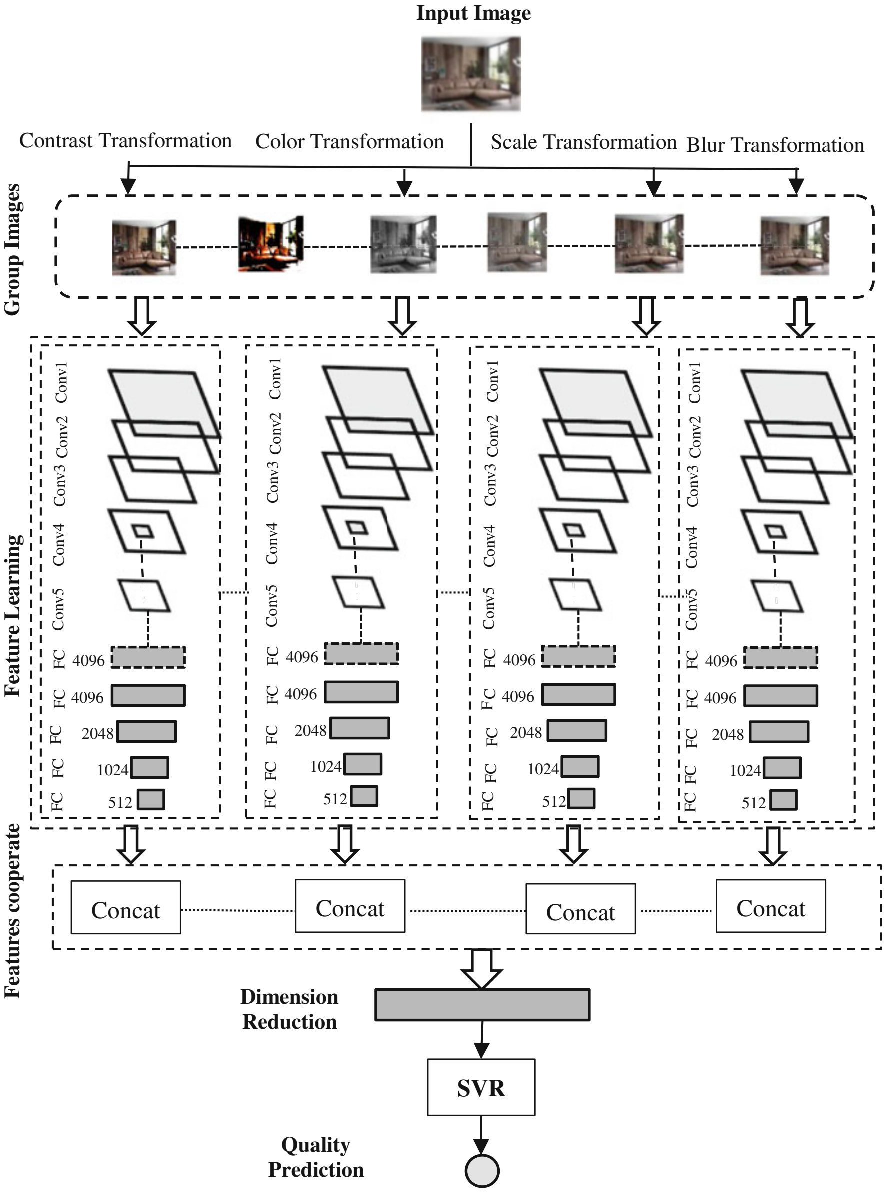


Fig. 6.4 Multi-CNN architecture design

the large-scale dataset D_{Global} are used as the initialization values of N_{Global} first few layers. The second one, using the pre-trained results D_{Global}, the back-propagation technique is used to train the new layer from the beginning, and the image quality target domain dataset D_{SIQA} is fine-tuned and optimized to obtain the updated model

N_{SIQA}. The performance of CNN is directly proportional to its ability to extract the good feature from the input data. The characteristics of the first few features learned layers aren't good for all fields. The best features learned in the network's following several layers will become more abstract and, bring you closer to your goal domain. After completing the CNN transfer training, the single CNN is used as a feature extractor to extract the image's deep features.

Mature pre-trained models can not only save time and effort by avoiding a significant number of complex network parameter adjustments, but they can also help with new network training by preventing insufficient training or training divergence related to inappropriate parameter control. To extract the deep features from the different scales of the input image, we made experiments with three different CNN networks pre-trained on the proposed database, ImageNet [43] database in a parameter study in the next section. In addition, two backbone CNN models, AlexNet [42] and VGG16 [44], are employed to learn image quality-related features, since they are a very common choice in IQA [45]. These two backbone models' depth and complexity are continually increasing, thanks to the use of various innovative convolutional network architectures, which have substantially aided deep learning to this day. A comprehensive evaluation of all possible pre-trained CNNs is proposed in the next section.

6.3.4 Deep Feature Dimension Reduction

The output features of CNNs SIQA have a large dimension. Between these features, there is a lot of repetition. Furthermore, high-dimensional features have a significant impact on later image quality degradation. As a result, the dimension of high-dimensional deep features must be reduced. The principal component analysis (PCA) [46] is an unsupervised linear dimension reduction method that reduces many possibly related variables to one or less unrelated variables known as principal components. The first principal component attempts to explain as much variability in the data as feasible, while the others attempt to interpret as much variability in the remaining data as possible.

6.3.5 Nonlinear Regression Prediction

In our approach, a support vector regression (SVR) is used to map feature space to quality scores. Many previous NR IQA, in the state of the art, works adopt SVR in quality prediction step [32] and obtain state-of-the-art result. SVR is very efficient in processing high-dimensional data [47], and we use the LIBSVM package to implement SVR with a radial basis function (RBF) kernel and obtain the optimal model parameters via cross-validation.

6.4 Experimental Results and Analysis

In this section, our experimental results and analysis are presented. First, we describe the proposed dataset and applied benchmark datasets used in this study. Second, the image quality labeling is defined, and the different image transformation is presented. Finally, we analyze the experimental results of our proposed method with parameters' design and compare it with other state-of-the-art methods.

6.4.1 Datasets

Various common databases can be used to evaluate the algorithms' performance in terms of human subjective judgments. Most of them were generated by applying synthetic distortions to high-quality photographs. However, in our case, we need to evaluate the image in an electronic catalogue, and as pointed out by Ghadiyaram and Bovik [48]: "images captured using typical real-world mobile camera devices are usually afflicted by complex mixtures of multiple distortions which are not necessarily well modeled by the synthetic distortions found in existing databases." So, we propose a new image catalogue database, and we measure the performance of our approach CNNs SIQA on various IQA databases (Fig. 6.5).

For e-commerce, catalogue management can contain different types of images such as clothing. However, we select 200 color images representing different situations with no distortions. Every image captured from many diverse mobile devices and many levels of distortion have been applied to obtain 1300 images.

In our subjective experiments, we used the single-stimulus (SS) approach. There were 22 inexperienced participants (15 males and 7 females). The majority of those

(a) Proposed catalog database

(b) LIVE database

Fig. 6.5 Sample of database. (**a**) Proposed catalogue database, (**b**) LIVE database

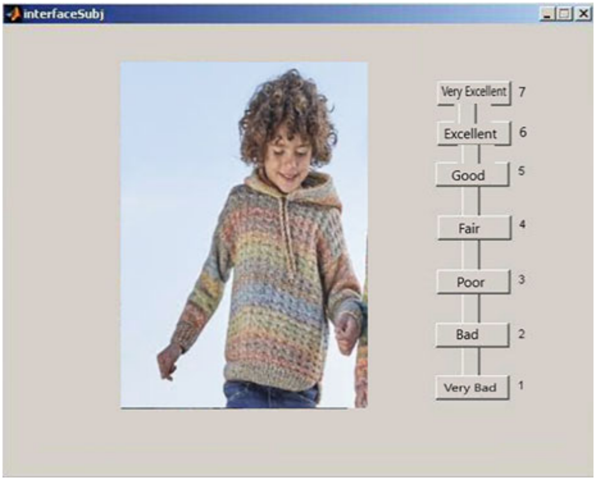


Fig. 6.6 Illustration of the interactive system used in our subjective viewing experiments

Table 6.1 Subjective test conditions and parameters

Method	Single-stimulus (SS)
Evaluation scales	Continuous quality scale from 1 to 7
Color depth	24-bits/pixel color images
Image coder	Bitmap
Subjects	Twenty-two inexperienced viewers
Image resolution	256*256
Viewing distance	Three times the image height
Room illuminance	Dark

who observed were college students. We created an interactive system, as shown in Fig. 6.6, to display the test images and collect subjective scores automatically. Three times the image height is set as the viewing distance.

The participants were asked to provide their overall quality perception on a continuous quality scale from 1 to 7, with an accuracy of up to 1/10,000 in a test environment prescribed by ITU-R BT.500-12 [49]. The order in which the test images were shown to each observer was randomized according to usual protocol. After removing outliers, the subjective scores for each changed image were calculated. Table 6.1 summarizes the abovementioned subjective test conditions and factors.

The LIVE (LIVE_Legacy) is an image quality dataset that is based on 29 reference images without distortion [50]. The LIVE dataset provides a differential mean opinion score (DMOS) which represents the subjective quality scores for each image. When the DMOS is near to zero, the image quality is the best. The images are distorted with 5 different distortion types at 7–8 degradation levels. The five distortion types included in this dataset are JP2k compression (JPEG2K), JPEG compression (JPEG), white Gaussian (WN), Gaussian blur (BLUR), and fast fading (FF). The LIVE dataset provides a differential mean opinion score (DMOS) for each distorted image. DMOS varies in the range [0,100] for images with highest

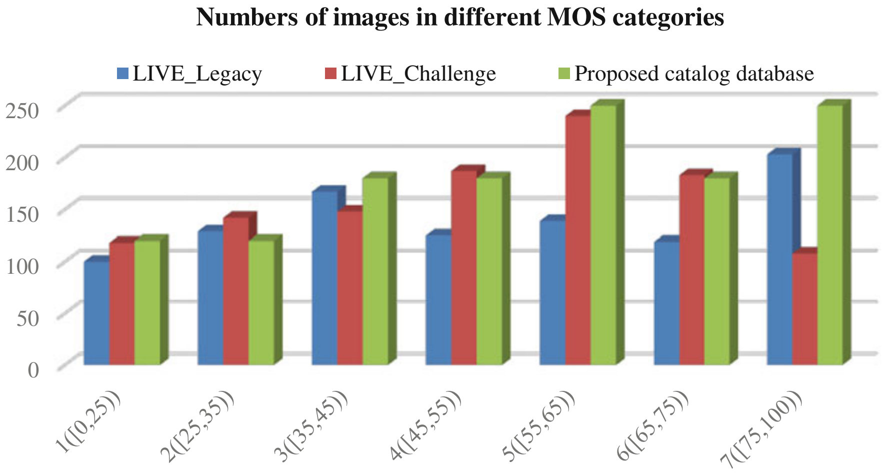


Fig. 6.7 Histogram of MOS score for different database

and lowest visual quality, respectively. In our experimental approach, we use the same database LIVE_Legacy and LIVE_Challenge proposed by [32]. From all 29 photos, 23 reference images and their distorted images were chosen as training data (LIVE_Legacy), while the remaining 6 reference images and their distorted images were used as test data. The LIVE Challenge database is a significant distortion-realistic database with 1162 authentically distorted images [51]. Thousands of people have rated database images using an online crowdsourcing method designed for subjective quality assessment. Over 350,000 opinion scores were acquired from over 8100 individual human observers. Each image's mean opinion score (MOS) is calculated by averaging individual assessments across participants and used as the ground-truth quality score. The MOS values are in the range of 1–100.

For labeling image quality, there are two main needs. To avoid the impact of an uneven training sample, the class labels should span the whole quality range of all photos in the dataset. The second is that the number of images in each category should be reasonably average. LIVE_Legacy, LIVE Challenge, and the proposed image catalogue are the databases used in this study. The link between the labels of image quality categories in the three databases and the range divisions of subjective scores is shown in Fig. 6.7. The cumulative frequency histogram distribution of the number of images in distinct subjective score ranges determines the borders of each class. The number of images in each class is relatively uniform when the number of training samples in LIVE Legacy, LIVE Challenge, and the proposed database is 7, which is also beneficial to CNN training. Although that image quality is classified, it is not an absolute classification problem. In the pooling stage, the output probability of CNN is used as the weight representation of each score.

6.4.1.1 Evaluation Criteria

The performance of NR IQA approaches is evaluated using Spearman's rank-order correlation coefficient (SROCC) and Pearson's linear correlation coefficient (PLCC) in our experiments [52, 53]. The PLCC for N testing images is defined as

$$\mathrm{PLCC} = \frac{\sum_{i=1}^{N}\left(s_i - \mu_{s_i}\right)\left(\hat{s}_i - \mu_{\hat{s}_i}\right)}{\sqrt{\sum_{i=1}^{N}\left(s_i - \mu_{s_i}\right)^2}\sqrt{\sum_{i=1}^{N}\left(\hat{s}_i - \mu_{\hat{s}_i}\right)^2}} \tag{6.2}$$

where s_i and $\hat{s}_i$ represent the ground-truth and anticipated quality scores of the i-th image, respectively, and μ_{si} and $\mu_{\hat{s}i}$ represent the average of each. The SROCC is defined as the difference between the rankings of the i-th test image in ground-truth and predicted quality scores, where d_i is the difference between the ranks of the i-th test image in ground-truth and predicted quality scores:

$$\mathrm{SROCC} = 1 - \frac{6\sum_{i=1}^{N} d_i^2}{N\left(N^2 - 1\right)} \tag{6.3}$$

The PLCC and SROCC range from -1 to 1, and higher absolute value indicates better prediction performance.

6.4.2 *Experimental Result*

In this subsection, we study the different parameters firstly in the pre-training phase and regression. Then, the impact of network structure design is presented. Finally, we compare the different design choices within the proposed approach CNNs SIQA with several NR IQA. Most of them are machine learning-based training procedures, and the rest are classic methods; in all the experiments, we randomly split the data into 80% training and 20% testing sets, using the training data to learn the model and validating its performance on the test data. We compute Pearson's linear correlation coefficient (PLCC), Spearman's rank-order correlation coefficient (SROCC), and the normalized mean absolute error (nMAE) between the predicted and the ground-truth quality scores. The nMAE is obtained by normalizing the MAE with respect to the upper limit of the MOS range to make it easier to be compared across datasets.

Experiment I, pre-trained CNNs The ImageNet-CNN and the dataset are used as source domain datasets, and we use the typical CNN models of AlexNet and VGG16 models to learn the quality features. Since these CNNs require an input with a dimensionality equal to 227 × 227 pixels, we rescale the original 500 × 500 images to 256 × 256 keeping aspect ratio. We use the stochastic gradient descent as the optimization algorithm. The hyperparameters of training are the learning rate α = 0.001 which gradually decreases the step algorithm, the momentum is μ = 0.9, and the weight decay is 0.0005.

Experiment II, regression In all the experiments, we use the Caffe open-source framework for CNN training and feature extraction and the LIBLINEAR library for SVR training. When building a nonlinear mapping model between deep features and image quality scores, Libsvm is used to implement epsilon-SVR with RBF as the kernel function, and the grid regression function is used to calculate the best regression parameters. These parameters are the penalty coefficient $c = 64$ and the RBF kernel function parameters $g = 0.0125$.

Experiment III, influence of network structure design In this paper, we adopt the backbone models used by [32] which are AlexNet and VGG16. They've been upgraded by deleting the last full connection layer and replacing it with three additional full connection layers, namely, FC8 (2048), FC9 (1024), and FC10 (512). To verify the effectiveness, tests are carried and performed on LIVE_Legacy, LIVE_Challenge, and the proposed catalogue database using a series of single CNN structures. For quality regression, the output features of a single CNN are used. Table 6.2 shows the indicators of prediction performance for each structure. Larger PLCC and SROCC, as well as a smaller MAE, reflect a better IQA model and higher prediction accuracy. From the result in Table 6.2, we can conclude that new models based on VGG16 outperform models based on AlexNet, especially when it comes to real-world distorted images in the LIVE Challenge and especially in the proposed catalogue database. Then, the accuracy of models has improved to some amount as the number of layers of full connection layers has increased. In summary, the basic

Table 6.2 Performance of different CNN structure

Backbone	FC layers	LIVE_Legacy		LIVE_Challenge		Proposed catalogue database	
		PLCC	SROCC	PLCC	SROCC	PLCC	SROCC
AlexNet	4096-4096	0.961	0.962	0.754	0.755	0.960	0.961
AlexNet	4096-4096-2048	0.964	0.962	0.754	0.769	0.965	0.963
AlexNet	4096-4096-2048-1024	0.972	0.971	0.774	0.774	0.973	0.972
AlexNet	4096-4096-2048-1024-512	0.975	0.973	0.780	0.778	0.976	0.975
VGG16	4096-4096	0.963	0.962	0.774	0.775	0.964	0.963
VGG16	4096-4096-2048	0.974	0.972	0.794	0.785	0.975	0.973
VGG16	4096-4096-2048-1024	0.978	0.973	0.804	0.781	0.979	0.976
VGG16	4096-4096-2048-1024-512	0.980	0.997	0.805	0.780	0.989	0.999

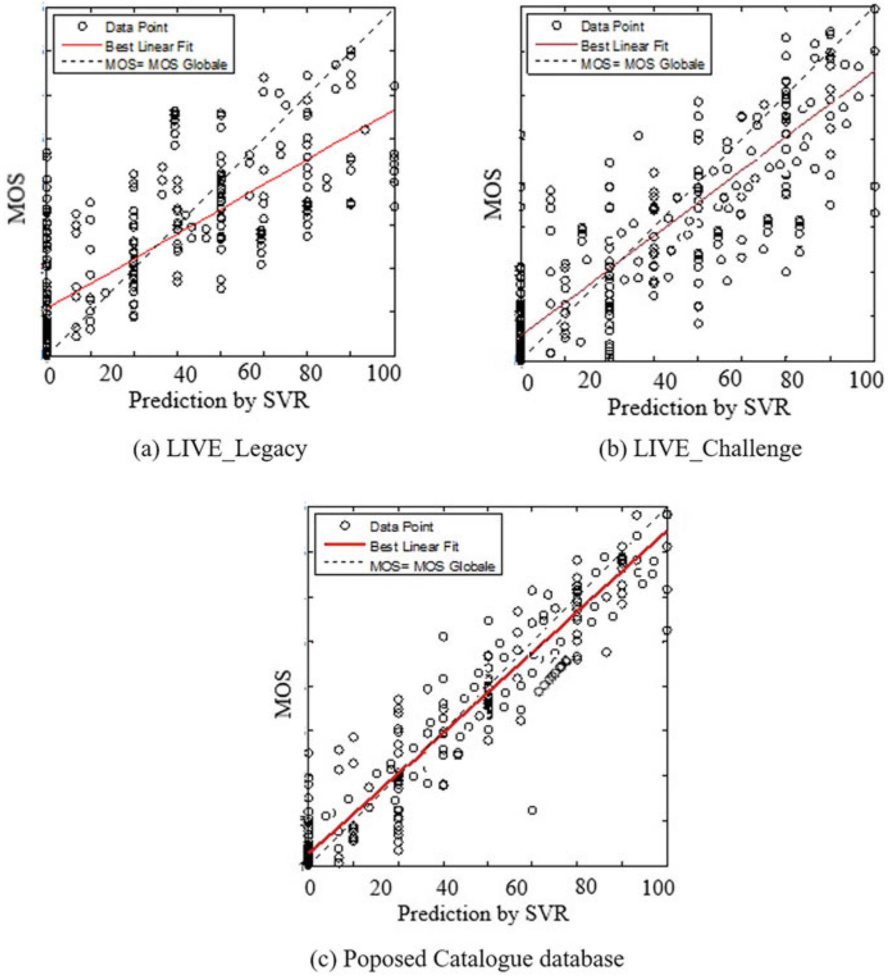
structure of future multi-CNN models is determined as "basic CNN + 4096FC-4096FC-2048FC-1024FC-512FC," which is expressed as CNNIQA AlexNet and CNNIQA VGG16, respectively.

Experiment IV, performance of the combination features This part discusses the effects of different combination of input in our proposed approach CNNs_SIQA such as sharpness, brightness colorfulness, and clarity represented respectively by scale, contrast, blur, and color transformation. This study permits to determine the optimal and the performance combination of components. Taking the LIVE_Challenge and the proposed catalogue database as experimental data, hybrid dataset as pre-trained model, CNNSIQA_VGG16 as single CNN structure, each component map and multicomponent combinations are fed to the single and multiple CNN network, respectively.

The learned features are used as the input of SVR. The median PLCC of ten random training and testing are reported. The prediction performance is shown in Table 6.3. According to the experimental results, when the CNN model is trained with different component maps, the predictions are not satisfactory, especially

Table 6.3 Pearson's linear correlation coefficient (PLCC) for each combination

Group	Component	PLCC (LIVE_CHALLENGE)	PLCC (proposed catalogue database)
1	Sharpness (scale transformation: cropping, scaling)	0.567	0.54
2	Clarity (blur transformation)	0.512	0.545
3	Brightness (contrast transformation)	0.641	0.693
4	Colorfulness (color transformation)	0.671	0.721
5	Sharpness & clarity	0.684	0.697
6	Sharpness & brightness	0.667	0.670
7	Sharpness & colorfulness	0.692	0.751
8	Clarity & brightness	0.674	0.712
9	Clarity & colorfulness	0.715	0.781
10	Brightness & colorfulness	0.748	0.791
11	Sharpness & clarity & brightness	0.795	0.804
12	Sharpness & clarity & colorfulness	0.824	0.873
13	Colorfulness & clarity & brightness	0.859	0.889
14	Sharpness & clarity & brightness & colorfulness	0.942	0.989



(a) LIVE_Legacy

(b) LIVE_Challenge

(c) Poposed Catalogue database

Fig. 6.8 Distributions of the predictions and the MOS scores of different databases. (**a**) LIVE_Legacy, (**b**) LIVE_Challenge, (**c**) Proposed Catalogue database

sharpness and clarity. The performance is effectively improved after the other component maps are merged, which indicates that the integration of features in multiple transformation is beneficial to improve the prediction accuracy.

Experiment V, performance of the proposed CNNs_SIQA According to the above analysis, the fusion of different components can improve the prediction performance by merging all transformation component. Therefore, transfer training is carried out for the component maps, and the generated model CNNsIQA_VGG16 is used for quality assessment on LIVE_Legacy, LIVE_Challenge, and the proposed catalogue database. Figure 6.8 shows the scatter distributions of the predictions and the real

Table 6.4 Performance comparison of different database

	LIVE_Legacy		LIVE_Challenge		Proposed catalog database	
Methods	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC
PSNR	0.859	0.863	N	N	0.897	0.881
SSIM	0.907	0.913	N	N	0.910	0.924
CBIQ	0.896	0.895	N	N	0.872	0.853
LBIQ	0.909	0.906	N	N	0.918	0.920
DIIVINE	0.927	0.925	0.557	0.509	0.915	0.935
BLINDS-II	0.916	0.912	0.449	0.404	0.917	0.919
BRISQUE	0.942	0.939	0.610	0.602	0.786	0.630
NIQE	0.915	0.914	0.477	0.421	0.923	0.918
C-DIIVINE	0.947	0.944	0.663	0.635	0.950	0.948
TANG MODEL	0.960	N	N	N	N	N
FRIQUEE	0.962	0.944	0.706	0.682	0.976	0.964
CNNsIQA_VGG16	0.983	0.981	0.864	0.856	0.986	0.984
CNNs_SIQA	**0.991**	**0.987**	**0.882**	**0.876**	**0.994**	**0.992**

MOS values. To further evaluate the generalization performance of the proposed model CNNs SIQA learned for the IQA, we compare it with simple and deep learning state-of-the-art-based general-purpose IQA methods.

In addition, we compare our approach, firstly, with full-reference image quality assessment (FR-IQA) algorithms such as PSNR [54] and SSIM [55]; secondly, with classical NR IQA methods like CBIQ [56], LBIQ [57], DIIVINE [58], BLIINDS-II [59], BRISQUE [60], NIQE [61], C-DIIVINE [62], Tang model [63], and FRIQUEE [51]; and finally with deep learning CNNsIQA_VGG12 [32]. Different results are presented in Table 6.4 (where N indicates that no values are reported in literature). Our model outperformed the conventional FR-IQA methods, the frontier NR IQA methods, and deep learning, for both the artificial distorted images and the real scene distorted images and electronic catalogue image.

6.5 Conclusion

The increasing growth on electronic commercial transactions over the Internet brought up many problems in catalogue management especially the image quality products which influence lost sales and a customer's hesitation to purchase. However, to solve this problem, we propose in paper a no-reference semantic quality assessment method for color images based on multi-deep CNNs named CNNs_SIQA. Based on the definition of semantic quality image, a group set of component maps are defined such as multiple scales, contrast, blur, and multiple color spaces from the original color image which are fed to the multi-CNNs. The CNN_SIQA structure is improved by adding the entire connection layers to the backbone CNN to further study image quality-associated representations. The final IQA prediction model will be developed by mapping the output feature vectors of multi-CNNs to subjective scores using nonlinear regression methods. Two principal contributions are presented. First is initializing the weights of the CNN using large-scale image classification and recognition datasets for transfer learning. It can solve two problems: designing a multi-deep convolutional network model from a different perspective that affects the quality of color images, which learns and represents image quality-related features more deeply and comprehensively and, secondly, solving the problem of insufficient images in existing and proposed IQA datasets, which results in feature learning ability limitations and overfitting of network training. The suggested technique has good measurement accuracy and generalization performance, according to systematic experimental results on image databases. In the future, it is also possible to expand the research on image restoration algorithm image to reconstruct high-quality images from distorted low-quality inputs in the electronic catalogue.

References

1. R.G. Javalgi et al., The dynamics of global e-commerce: An organizational ecology perspective. Int. Market. Rev. **22**(4), 420–435 (1983)
2. S. Hawk, The development of Russian e-commerce: The case of Ozon. Manag. Decis. **40**(7), 702–709 (1967)
3. A. Pons et al., Global e-commerce: A framework for understanding and overcoming the trust barrier. Inf. Manag. Comput. Secur. **11**(3), 130–138 (2003)
4. J.V. Mullane, K.E. Bullington, M.H. Peters, Web-mining applications in e-commerce and e-services. J. Manag. Hist. **39**(5), 388–393 (2001)
5. D. Kim, S.-g. Lee, J. Chun, S. Park, J. Oh, Catalog management in e-Commerce systems, pp. 1–6
6. P. Kalpana, M. Anitha, P.T. Selvy, An intelligent and effective model to recognize the duplicate products in catalog management system. Int. J. Sci. Technol. Res. **9**(2), 5104–5107 (2020)
7. S. Chakraborty, M.K.R. Garla, Automated catalog management and image quality assessment using convolution neural networks and transfer learning, in *Alliance International Conference on Artificial Intelligence and Machine Learning (AICAAM)*, (April 2019), pp. 236–252
8. S. Ouni, M. Chambah, M. Herbin, E. Zagrouba, SCID: Full reference spatial color image quality metric, in *SPIE/IS&T Electronic Imaging*, Proc. SPIE 7242, 72420U, California, USA, 28–30 January 2009
9. S. Ouni, E. Zagrouba, M. Chambah, M. Herbin, Vers une metrique de description objective d'une sensation subjective. Rev. Francoph. Afr. Rech. Inform. Math. Appl. **11**, 1–16 (2009)
10. J. Redi, P. Gastaldo, R. Zunino, L. Heynderickx, Reduced reference assessment of perceived quality by exploiting color information, in *Fourth International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, (2009)
11. Y. Tian, M. Zhu, L. Wang, Analysis and design of no-reference image quality assessment, in *International Conference on Multimedia and Information Technology*, (2008), pp. 349–352
12. N. Ahmed, H.M.S. Asif, Perceptual quality assessment of digital images using deep features. Comput. Inform. **39**(3), 385–409 (2020)
13. N. Ahmed, H.M.S. Asif, H. Khalid, PIQI: Perceptual image quality index based on ensemble of Gaussian process regression. Multimed. Tools Appl. **80**, 15677–15700 (2021)

14. S. Bosse et al., Deep neural networks for no-reference and full reference image quality assessment. IEEE Trans. Image Process. **27**(1), 206–219 (2017)
15. H. Zhu, L. Li, J. Wu, W. Dong, G. Shi, MetaIQA: Deep meta-learning for no-reference image quality assessment, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (June 2020), pp. 14143–14152
16. B. Bare, K. Li, B. Yan, An accurate deep convolutional neural networks model for no-reference image quality assessment, in *2017 IEEE International Conference on Multimedia and Expo (ICME)*, (2017)
17. S. Ouni, E. Zagrouba, M. Chambah, M. Herbin, No-reference image semantic quality approach using neural network, in *IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, Bilbao, Spain, (2011), pp. 106–113
18. R.V. Babu, S. Suresh, A. Perkis, No-reference JPEG image quality assessment using GAP-RBF. Signal Process. **87**(6), 1493–1503 (2007)
19. A.K. Moorthy, A.C. Bovik, A two-step framework for constructing blind image quality indices. IEEE Signal Process. Lett. **17**(5), 513–516 (2010)
20. C. Charrier, A.H. Saadane, C. Fernandez-Maloigne, Comparison of no-reference image quality assessment machine learning-based algorithms on compressed images, in *IS&T/SPIE EI, Image Quality and System Performance XII*, (2015)
21. L. Kang et al., Simultaneous estimation of image quality and distortion via multi-task convolutional neural networks, in *IEEE International Conference on Image Processing (ICIP)*, (2015)
22. S. Bianco, C. Luigi, N. Paolo, S. Raimondo, On the use of deep learning for blind image quality assessment. SIViP **12**(2), 355–362 (2018)
23. F. Gao, J. Yu, S. Zhu, Q. Huang, Q. Tian, Blind image quality prediction by exploiting multilevel deep representations. Pattern Recogn. **81**, 432–442 (2018)
24. J. Kim, A.-D. Nguyen, S. Lee, Deep CNN-based blind image quality predictor. IEEE Trans. Neural. Netw. Learn. Syst. **30**(1), 11–24 (2018)
25. R. Ravela, M. Shirvaikar, C. Grecos, No-reference image quality assessment based on deep convolutional neural networks, in *Real-Time Image Processing and Deep Learning*, (International Society for Optics and Photonics, 2019)
26. K. Ma, W. Liu, K. Zhang, Z. Duanmu, Z. Wang, W. Zu, End-to-end blind image quality assessment using deep neural ne works. IEEE Trans. Image Process. **27**(3), 1202–1213 (2017)
27. W. Zhang et al., Blind image quality assessment using a deep bilinear convolutional neural network. IEEE Trans. Circuits Syst. Video Technol. **30**(1), 36–47 (2018)
28. F. Chunling, Z. Yun, F. Liangbing, J. Qingshan, No reference image quality assessment based on multi-expert convolutional neural networks. IEEE Access **6**, 8934–8943 (2018)
29. Z. Richard, P. Isola, A.A. Efros, W. Oliver, The unreasonable effectiveness of deep features as a perceptual metric, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (2018)
30. M. Xiaoyu, J. Xiuhua, Multimedia image quality assessment based on deep feature extraction. Multimed. Tools Appl. **79**, 1–12 (2019)
31. N. Ahmed, H.M. Shahzad Asif, A. Rauf Bhatti, A. Khan, Deep ensembling for perceptual image quality assessment. Comput. Inform. **39**(3), 385–409 (2022)
32. Y. Yuan, Z. Guoqiang, C. Zhenwei, G. Yudong, Color image quality assessment with multi deep convolutional networks, in *IEEE 4th International Conference on Signal and Image Processing*, (2019), pp. 934–941
33. S. Ouni, E. Zagrouba, M. Chambah, M. Herbin, No-reference image semantic quality approach using neural network, in *2011 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, (2011), pp. 106–113
34. S. Ouni, E. Zagrouba, M. Chambah, A new no-reference method for color image quality assessment. Int. J. Comput. Appl. **40**(17), 24–31 (2012)
35. R.S. Hunter, Photoelectric color-difference meter. J. Opt. Soc. Am. **38**(7), 661 (1948)
36. R.S. Hunter, Accuracy, precision, and stability of new photo-electric color-difference meter. J. Opt. Soc. Am. **38**(12), 1094 (1948)

37. D. Jyoti Bora, A. Kumar Gupta, F. Ahmad Khan, Comparing the performance of L*A*B* and HSV color spaces with respect to color image segmentation. Int. J. Emerg. Technol. Adv. Eng. **5**(2), 192–203 (2015)
38. W. Chen, Y.Q. Shi, G. Xuan, Identifying computer graphics using HSV color model and statistical moments of characteristic functions, in *IEEE International Conference on Multimedia and Expo (ICME 2007)*, Beijing, China, July 2–5, 2007
39. I. Taha Ahmed, C. Soong Der, N. Jamil, M. Afendee, Mohamed., Improve of contrast-distorted image quality assessment based on convolutional neural networks. Int. J. Electr. Comput. Eng. **9**(6), 5604–5614 (2019)
40. D. Li, T. Jiang, Blur-specific no-reference image quality assessment: A classification and review of representative methods, in *The Proceedings of the International Conference on Sensing and Imaging*, (January 2019), pp. 45–68
41. D. Chaudhary, V. Deep, CNN model for non-screen content and screen content image quality assessment. Global J. Comput. Sci. Technol. **22**(1), 17 (2022)
42. A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks. Adv. Neural Inf. Process. Syst. **25**, 1097–1105 (2012)
43. J. Deng, W. Dong, R. Socher, L.J. Li, K. Li, L. Fei-Fei, ImageNet: A large-scale hierarchical image database, in *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL, USA, (June 20–25, 2009), pp. 248–255
44. K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556 (2014)
45. D. Varga, No-reference image quality assessment with multi-scale orderless pooling of deep features. J. Imaging **7**, 112 (2021)
46. J. Shlens, A tutorial on principal component analysis. Comput. Sci. **51**, 219–226 (2014)
47. X. Lv, M. Qin, X. Chen, G. Wei, No-reference image quality assessment based on statistics of convolution feature maps. AIP Conf. Proc. **1955**, 040034 (2018)
48. D. Ghadiyaram, A.C. Bovik, Crowdsourced study of subjective image quality, in *Asilomar Conference on Signals, Systems and Computers*, (2014)
49. ITU, *Methodology for the Subjective Assessment of the Quality of Television Pictures* (Recommendation, International Telecommunication Union/ITU Radiocommunication Sector, 2009)
50. D. Ghadiyaram, A.C. Bovik, Massive online crowdsourced study of subjective and objective picture quality. IEEE Trans. Image Process. **25**(1), 372–387 (2016)
51. D. Ghadiyaram, A.C. Bovik, Feature maps driven no-reference image quality prediction of authentically distorted images. Int. Soc. Opt. Eng. **9394**, 93940J-93940J-14 (2015)
52. S. Bosse, D. Maniry, K. Mller, T. Wiegand, W. Samek, Deep neural networks for no-reference and full-reference image quality assessment. IEEE Trans. Image Process. **27**(1), 206–219 (2018)
53. B. Yan, B. Bare, W. Tan, Naturalness-aware deep no reference image quality assessment. IEEE Trans. Multimedia **21**(10), 2603–2615 (2019)
54. Z. Wang, H.R. Sheikh, A.C. Bovik, No-reference perceptual quality assessment of JPEG compressed images, in *Processing of IEEE International Conference on Image Processing (ICIP 02)*, (IEEE, 2002), pp. 477–480
55. Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: From error visibility to structural similarity. IEEE Trans. Image Process. **13**, 600–612 (2004)
56. P. Ye, D. Doermann, No-reference image quality assessment using visual codebooks. IEEE Trans. Image Process. **21**, 3129 (2012)
57. H. Tang, N. Joshi, A. Kapoor, Learning a blind measure of perceptual image quality, in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR 11)*, (IEEE, 2011), pp. 305–312
58. A.K. Moorthy, A.C. Bovik, Blind image quality assessment: From natural scene statistics to perceptual quality. IEEE Trans. Image Process. **20**, 3350–3364 (2011)
59. M.A. Saad, A.C. Bovik, C. Charrier, Blind image quality assessment: A natural scene statistics approach in the DCT domain. IEEE Trans. Image Process. **21**, 3339–3352 (2012)

60. A. Mittal, A.K. Moorthy, A.C. Bovik, No-reference image quality assessment in the spatial domain. IEEE Trans. Image Process. **21**, 4695–4708 (2012)
61. A. Mittal, R. Soundararajan, A.C. Bovik, Making a "Completely Blind" image quality analyzer. IEEE Signal Process. Lett. **20**, 209–212 (2013)
62. Y. Zhang, D.M. Chandler, An algorithm for no-reference image quality assessment based on log-derivative statistics of natural scenes, in *Proceedings of the SPIE – Image Quality and System Performance X, SPIE*, (February 2013), pp. 1–11
63. H. Tang, N. Joshi, A. Kapoor, Blind image quality assessment using semi-supervised rectifier networks, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 14)*, (IEEE, 2014), pp. 2877–2884

Chapter 7
Contextual Recommender Systems in Business from Models to Experiments

Khedija Arour and Rim Dridi

7.1 Introduction

Recommender Systems (RS) are information filtering systems that cope with the information overload problem by filtering vital information fragment out of large amount of dynamically generated information according to users preferences about items. These systems try to give solutions to resolve this problem by searching through large volume of existing information to provide users with adapted content and services [43]. Instead of exploring an important number of items until finding the most adequate, RS became a promising area of research, thanks to their help for users to suggest the items they might prefer. However, there are usually various factors that may impact users' preferences. Therefore, research in recommender systems is starting to recognize the importance of items and the role of user's context in enhancing the recommendation output. In this respect, traditional recommender systems are extended to offer novel lines of research areas such as Context-Aware Recommender Systems (CARS). This chapter provides a survey on CARS systems and presents a standard evaluation process that can be adopted by researches on this field. We review a range of evaluation metrics and measures as well as some approaches used for evaluating recommender systems.

The layout of this chapter is structured as follows. In Sect. 7.2, the different notions and concepts are presented. Section 7.3 investigates context awareness recommender systems in detail. Evaluation process for CARS systems is given in

K. Arour (✉)
College of Business, University of Jeddah, Jeddah, Saudi Arabia
e-mail: kbarour@uj.edu.sa

R. Dridi
LISI Laboratory, INSAT, University of Carthage, Tunis, Tunisia
e-mail: rim.dridi@ensi-uma.tn

B. Alyoubi et al. (eds.), *Machine Learning and Data Analytics for Solving Business Problems*, Unsupervised and Semi-Supervised Learning,
https://doi.org/10.1007/978-3-031-18483-3_7

Sect. 7.4. Experimental results are detailed in Sect. 7.5. In Sect. 7.6, some CARS systems applied for specific business are presented. We conclude and give some future directions in Sect. 7.7.

7.2 Recommender Systems: Notions and Concepts

Some notions and concepts related to recommender systems must be presented. For example, the user, the *Item* that should be recommended, and the *Rating* that represent how much a user likes an item. So the triplet (User, Item, Rating) is the core of any recommender systems.

- User: this term depicts the set of entities to which recommendations will be given, regardless of whether they describe a person, a group of people, or other entities of interest. Generally, users designate the persons to whom items are suggested, generally presented using attributes such as the id, name, gender, age, etc. These information are modelled as "user profile" aiming to identify the user's needs for providing custom recommendations which could be suitable for the user.
  Ordinary users having a sufficient number of ratings have been distinguished from particular ones who require a special reasoning to satisfy all users' needs. In this regard, three types of particular users are identified [22]: (i) "cold start users" are the new users recently entered the system with very limited information (insufficient ratings); (ii) "grey sheep users" are the users with unusual tastes resulting low correlations with other users; and (iii) users who do not have any behavior in the current context.
- Item: the items are objects to be recommended to users, regardless of their actual representation. Generally, typical recommended items are documents, music, movies, etc. An item can be characterized by its features or descriptions and utility (positive if it is beneficial for the user and negative if not) [43]. In particular, the paper in [43] describes an item in a movie recommender system through the following attributes: title, length, genre, director, and release year.
- Rating: we denote the preference of a user toward an item as a rating. In our study of recommendation systems, we take user's rating to be the quintessential piece of information utilized to indicate a user's interest in an item. From this point of view, the rating presents the interaction between a user and the recommender system aiming to infer the user's opinions. We equate higher ratings with a greater preference (i.e., users would like better an item rated 5 rather than an item rated 2). A rating can be viewed in different forms: (i) *binary rating* that shows whether a given item is good for a user or not. As an exemplification, in YouTube[1] "like" and "follow" could be considered as a binary rating. While, binary rating is easy for the user to deal with and less ambiguous, it cannot be sufficient for

[1] https://www.youtube.com.

items comparison; (ii) *numerical rating* uses a numerical scale rating aiming to provide detailed feedback. Take Netflix as an example, it uses standard five-star rating scale to power its review system and recommendations. There are also variations like using a ten-star scale; and (iii) *ordinal rating* are basically used to clarify the meaning of each rating level with words such as 1/5 stars means “I do not like very much” and 5/5 stars means “I really like” [18].

7.2.1 Foundations of Recommender Systems

Let us first look at comprehending what a recommender system is and what types of functionalities do recommender systems have.

The concept of recommender systems was first introduced by Resnick and Varian in 1997 [41]. Indeed, the developers of the first recommender named Tapestry [20] have considered their system as a collaborative filtering system. Yet, the authors in [41] have chosen the term "recommender system" for these reasons: (i) recommenders and recipients may be unknown to each other, then may not explicitly collaborate; and (ii) recommendations may propose some particularly pertinent items, as well indicating those that should be filtered. Hence, they have considered recommender systems as an independent research area in the mid-1990s issued from different other areas such as information retrieval, approximation theory, management sciences, and also cognitive science.

Firstly, we start with a simple definition of a recommender system: A recommender system is a system able to suggest items to users [41]. More abstractly, a recommender system is a system that suggests content a user is interested in out of an enormous set of choices [41] and hence, is a system to overcome the information overload problem. For this task, a recommender system aims at predicting the most relevant items to a user and states a short list of recommendations. According to this definition, we derive two main tasks: (i) the rating prediction task and the (ii) top-n recommendations task. The latter task is based on the first one, as a recommender system orders the list of recommended items by the predicted rating representing the perceived usefulness of a user towards an item.

In another point of view, authors in [11] have differentiated between recommender systems and information retrieval systems by the power of recommenders to be personalized in addition to their ability to suggest relevant recommendations. Therefore, they propose the following definition: “A recommender system is any system that produces individualized recommendations as output or has the effect of guiding the user, in a personalized way, to interesting or useful objects in a large space of possible options” [11]. Authors assumed in their article [34] that recommendation is related to four main features. These features are important because they cover the necessary needs of users facing many set of items: Decide, Compare, Explore, and Discover. As recommender systems has been developed in different industrial domains, like: e-commerce, health-care, entertainment, etc. Many works have proposed their definition according to the application field

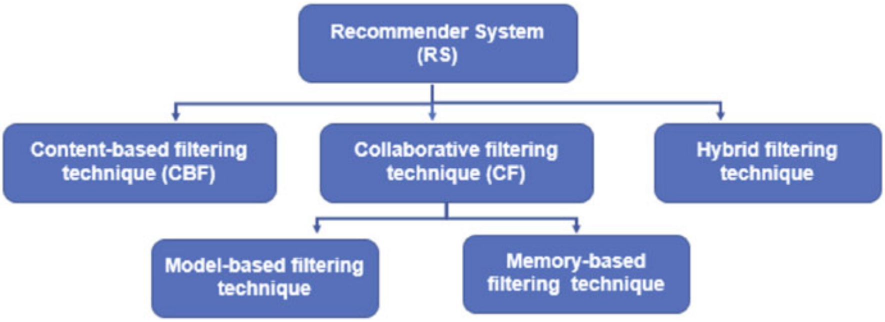

Recommender System (RS)
Content-based filtering technique (CBF)
Collaborative filtering technique (CF)
Hybrid filtering technique
Model-based filtering technique
Memory-based filtering technique

Fig. 7.1 RS classification

particularities. For instance, in e-commerce discipline, researches in [40] have considered recommender systems as computer algorithms used widely in e-commerce to propose items to a user, like what items to buy, news to read, or movies to rent.

7.2.2 Classification of Recommender Systems

Recommendation approaches are expected to predict the utilities of items for target users and offer accurate recommendations. It is possible to classify RS approaches by various ways in accordance with different criteria including the type of feedback they use (explicit or implicit feedback), the recommendation task they address (rating prediction or top-N recommendation), etc. The most common classification used in the literature is based on the type of data exploited for recommendation and establishes the following three categories (see Fig. 7.1):

- *Content-Based Filtering (CBF) approaches.* These approaches make use of knowledge related to users or items to provide recommendation.
- *Collaborative Filtering (CF) approaches.* These approaches recommend items relying on similar users and their ratings.
- *Hybrid approaches.* These approaches combine the two above-mentioned filtering approaches.

7.2.2.1 Content-Based Filtering

Content-based recommender systems are based on content information about users or items to provide recommendations. This information can take different forms like features, textual descriptions, and tags. In other words, users receive items suggestions that are similar to those they positively evaluated in the past. Particularly, recommendations are made through matching the user profile features describing the user's preferences with the items features. In content-based recommender systems, the item can be represented by a weighted terms vector extracted from its content. To define the user profile, CBF mostly concentrate on the model of the user's preference or the history of the user's interaction with the recommender.

Pandora Music Genome Project[2] is an example of a content-based approach that uses the characteristics of a song or a singer (subset of attributes describing songs) to capture the essence of music with similar characteristics and to organize them. Users' feedbacks (likes or dislikes) are adopted to filter the music station's results. Basically, a content-based recommender system comprises the following steps [32]:

1. Preprocessing of items content (e.g., Web pages, documents, product descriptions, etc.) to extract structured pertinent information (e.g., Web pages represented as keyword vectors).
2. Starting from items liked or disliked in the past, the profile of a target user is learned through machine learning techniques.
3. Matching the profile representation of the target user and that of items to be recommended computed using similarity metrics.
4. Recommending a ranked list of potentially pertinent items.

This technique presents advantages such as user independence, since CBF systems only use ratings of the active user to build the recommendation model. Additionally, when a new item appears and has not yet been rated, CBF systems are able to recommend it. However, CBF suffer from several issues such as the over-specialization, as they are not capable of finding unexpected items: the user will receive recommendations of items similar to the ones rated before.

7.2.2.2 Collaborative Filtering

To date, collaborative filtering (CF) is the most popular algorithm used to design various applications and sites for recommender systems such as Facebook,[3] Twitter,[4] Google,[5] LinkedIn,[6] and Netflix. The underlying idea behind CF is that users with common interests in the past are more likely to keep exhibiting similar interests in the future. The principal property to work with collaborative filtering are the ratings given by users for items. Therefore, the typical input of collaborative recommender systems is represented by a matrix of ratings representing users by rows and items by columns. More precisely, the user-item matrix defining users' preferences for items is used to find like-minded users by computing similarities between their profiles

[2] https://www.pandora.com.

[3] https://facebook.com/.

[4] https://twitter.com.

[5] https://www.google.com.

[6] https://fr.linkedin.com.

defining a “neighborhood” to provide recommendations. In general, a collaborative filtering system requires the following steps to generate recommendations:

1. Identification of the subject of the recommendation (ratings of the target user).
2. Identification of the most similar users to the target one using a similarity function (cosine similarity, Pearson’s correlation, etc.).
3. Identification of the rated items by the similar users and not rated by the target one.
4. Prediction of the rating of each selected item based on users’ similarity.
5. Recommendation of items according to the predicted ratings.

There are two main recommendation techniques in collaborative filtering: memory-based and model-based algorithms.

- Memory-based algorithms:
  The memory-based approach uses the entire user-item matrix to find similarities between users for estimating rating predictions. It is commonly referred to as neighborhood-based or heuristic-based approach. This approach uses previous users ratings for predicting ratings for new items using one of these two ways: user-based CF recommendation or item-based CF recommendation.
- Model-based algorithms:
  The model-based approach uses a collection of ratings in a learning phase, in which a model of user preferences is built to make intelligent rating predictions based on the observed data. Model-based CF algorithms are developed using data mining techniques and machine learning algorithms such as Bayesian networks, clustering, neural networks, linear regression and latent factor models. These latter models are known as prevalent since they use latent variables in order to explain user preferences and perform a dimensionality reduction of the rating matrix for recommendation purposes.

7.2.2.3 Hybrid Approaches

The hybrid filtering recommendation system is a system that associates two or more recommendation techniques for better recommendation performance. As stated by Burke [11], a hybrid recommender system combines multiple techniques together to obtain some synergy between them. Hybrid recommender systems have been proposed to overcome the weaknesses of collaborative filtering and content-based algorithms by combining them together instead of using them separately. This trend had also been affected in competitions such as the Netflix Prize,[7] where the winning candidate highlighted the fact that better results are often obtained when different recommendation algorithms are associated in a single model [7].

[7] http://www.netflixprize.com.

7.3 Context Awareness Recommender Systems

The use of contextual information is considered as a key component to boost the performance of systems that fall within numerous research disciplines, like mobile computing, information retrieval and recommender systems [14, 48]. In fact, the contextual information illustrated through different factors makes it possible to afford the most relevant information to the user when it is most needed. In what follows, we define the basic concepts of context and the notions that it entails.

7.3.1 Definitions

Due to the complexity and the wideness of the context concept, it has no a single definition. Indeed, context is a multifaceted concept that has been studied in various research fields and many gave multiple definitions, often different from the others and more specified than the general dictionary definition which describe context as: "conditions or circumstances that have an effect on something". Given the growing importance of context, an entire conference, CONTEXT,[8] is devoted for presenting and discussing this topic in wide range of various disciplines including artificial intelligence, cognitive science, linguistics, philosophy, and psychology. Based on a general point of view, the majority of renowned dictionaries have defined the context by almost similar definitions.

According to Oxford Advanced Learner's Dictionary,[9] "a context is the situation in which something happens and that helps you to understand it". WordNet Search 3.1[10] considers a context as "the set of facts or circumstances that surround a situation or event". For Cambridge dictionary,[11] the context is viewed as " the situation within which something exists or happens, and that can help explain it". Moreover, In Webster's dictionary[12] "a context is defined as the interrelated conditions in which something exists or occurs like environment and setting".

More specifically than the dictionaries definitions, many researchers presented and discussed several context definitions from different fields. The idea of including context in computer sciences was introduced in 1994 by Schilit [48], which defined the context as: "*location and the identity of nearby people and objects*". In accordance with Schilit, "context encompasses more than just user's location, because other things of interest are also mobile and changing". Context could also include lighting, noise level, communication bandwidth, network connectivity and

[8] http://context-07.ruc.dk.

[9] http://www.oxfordlearnersdictionaries.com/.

[10] http://wordnetweb.princeton.edu/perl/webwn.

[11] http://dictionary.cambridge.org/dictionary/.

[12] https://www.merriam-webster.com/dictionary/.

even the social situation (e.g., whether you are with your manager or with a co-worker). Later, a more abstract definition [16] presented by Dey and Abowd in 1999 states that: *context is defined as any information that can be used to characterize the situation of entities (place, people, and things), including the user and application and the interaction between them.* This is probably the most commonly and widely used definition for context in the computational sciences.

7.3.2 Context-Aware Recommender Systems Approaches

The recommendation field is one branch that adopted contextual information allowing recommender systems to be mightily contextualized to enhance the way in which these systems work.

With the goal of understanding the state of the art of this field, we provide a thorough literature review which analyses relevant Context-Aware Recommender Systems (CARS) approaches along several application domains, context types, recommendation techniques and paradigm for incorporating context.

In our discussion, we will use the term *contextual dimension* referring to a contextual factor (e.g., weather, time, etc.). The term *contextual condition* refers to a specific value in a contextual dimension (e.g., rainy, morning).

Among the earliest works on context-aware recommendation, the one proposed by Adomavicius et al. [1], who built a multidimensional recommendation model by integrating additional contextual dimensions besides the typical information on users and items. For rating prediction, this approach applied the collaborative filtering technique.

Since the early works on context-aware recommender systems, there have been many efforts made in this field where researchers have often tried to make use of contextual information to enhance standard recommendation algorithms. These recommendation approaches can generally be sub-divided by the formation of the utility function into memory-based and model-based approaches.

In the literature, many attempts have been made in order to build context-aware recommendation systems by applying memory-based methods. Two primary types of memory based have been introduced: the user-based, which builds neighbors according to users similarity; and the item-based, which constructs neighbors depending on items similarity. Typical examples of these approaches are the neighborhood-based collaborative filtering. In this respect, Lamche and co-workers [28], proposed and evaluated a context-aware recommender system in a mobile shopping scenario. It employed the nearest neighbor algorithm to recommend pertinent items according to the relevant selected contextual dimensions. For the task of Point-of-Interest (POI) recommendation, authors in [52] integrated the spatial, temporal, and the social context in their recommendation model. They exploited various contextual dimensions in a collaborative filtering algorithm by varying their weights to investigate the effect of including each dimension on recommendation accuracy. Otebolaku et al. [39] proposed an approach that emphasizes the importance of similarity between contextual dimensions. To predict user preferences, K-nearest neighbors (KNN) algorithm was adopted based on the similarity between user contexts and those of other users.

It is believed that there is still a space to enhance memory-based approaches, in order to compete with the model-based approaches. In particular, several efforts followed the evolution of model-based approaches to adapt them for context-aware recommendation. Therefore, many extended models of Matrix Factorization (MF) technique were proposed in the literature, like the contextual matrix factorization, also known as Context-Aware Matrix Factorization (CAMF). It was initially introduced in [4] to model the relatedness between the contexts and item ratings providing additional model parameters. Along with standard CAMF recommender systems, we investigate more recent CAMF researches. In [23] authors proposed a context-aware latent factor model realized using matrix factorization. This study integrated contextual information of both user and item in the absence of the historical user or item data to perform event recommendations.

However, the majority of the surveyed CAMF recommendation methods cannot fully capture the impact of the relevant contextual dimensions as well as their associations on the predicted rating. To tackle this shortcoming, an improved CAMF recommendation model on the basis of the fuzzy measures of contextual dimensions [17] was proposed. It consists of two strategies extended from the correlation based CAMF-MCS model suggested in [56]. Both of the two strategies apply a common rating prediction formula given by Zheng [56], highlighting the notion of "contextual correlation".

Besides matrix factorization-based latent factor models, others model-based algorithms have been receiving attention counting on multidisciplinary techniques such as machine learning and deep learning. These techniques have revolutionized the data mining and information retrieval techniques offering an effective impact on context-aware recommendation. For example, in [2], authors built context-aware local recommendation models where users were clustered, regarding visited destinations each period of the year. Here, the k-means clustering technique is applied to generate k clusters of countries where residents have similar behaviors according to their country of residence and to the visited destinations in different periods of visits. In reference [47], a context-aware smartphone application was developed based on artificial intelligence mechanisms to reduce the large dimensionality of context data. The principal component analysis was considered for dimensionality reduction and decision tree for building the prediction model.

7.3.3 Context-Aware Recommender Systems: Synthesis

The majority of the existing CARS follow the common classification that exists for the traditional RS: collaborative filtering, content-based filtering and hybrid recommendation approaches. That means that these works did not invent a new specific classification for CARS. In these approaches, the context is often integrated directly into the recommendation model when it is used for producing recommendations.

Another important aspect of the literature is the widespread interest in using collaborative filtering approaches, which play a principal role in the success of several CARS [29]. These recommendation systems only depend on the user past behavior. Contrary to content-based approaches which require additional information about items. In CF approaches, the most widely used algorithms are the model-based considering users ratings to build a learning model.

The matrix factorization methods are the most employed in the model-based approaches. In the presented approaches, several variations and extensions of MF methods have been used. Model-based algorithms were developed using different machine learning techniques where a recommendation approach can be viewed as a classification problem to identify what might interest the user and what might not. Various algorithms are used for this task, such as decision trees [46, 47] and clustering [24, 58].

Despite the popularity of the research around CARS, some of the existing studies still mainly rely on incomplete assumptions about how to work with contextual information. Many CARS [24, 33] assumed that all existing contextual dimensions have equal effects and should contribute to make recommendations. Some studies [31, 46] mainly focused on the approach's research area and assumed that common contextual dimensions could be selected as relevant in compliance with their application domain. Although plenty of solutions have been proposed for the problems in the area of context-aware recommendation, the majority of them represents distinct methods for discovering relevant or correlated contextual dimensions. The lack of methods that deal with both contextual information relevancy and correlation is a quite challenging process. We believe that it is essential to combine these two topics to be handled by one method for mitigating the computation complexity and the dimensionality of context representation.

7.4 Experimental Evaluation Process for CARS Systems

Evaluation is a systematic determination of merit, worth, and significance [35]. We can measure some aspects like how accurate is a recommendation?, how many users are satisfied with the system?, does the system have an impact on user actions/reactions? does the system have an impact on business value?,...

Hence, we need to identify the role of the recommender system in the business to maximize the system utility like the time on site, the profit, etc. Also, to be able to predict the rating that a user will assign an item then to predict the best recommendation.

Evaluation process plays an important role in the context of comparative evaluation of any RS or CARS systems. It is clear that the performance of any recommender system is based heavily on data. They make reliable recommendations based on the facts that they have. Also, it is important to define appropriate evaluation methodologies and metrics to measure the weakness and strength of the compared approaches.

Any RS or CARS paper claims that System X is better than System Y in terms of an effectiveness or efficiency metric M computed based on a data collection C: How reliable is this paper? More specifically, (a) What happens if C is replaced with another set of data C1? (b) How good is M?

Indeed, a recommender systems have a variety of properties that may affect user experience, such as accuracy, robustness, scalability, and so forth.

Hence, various parameters must be tuned to generate more accurate predictions. Most of the effort made when developing this work was experimenting novel solutions to upgrade the system performance results in rating prediction and recommendation performers. We present in this part, the protocol process that can be used to evaluate any CARS approach for different businesses.

7.4.1 Datasets

A dataset is a major component consists of a collection of objects related to each other to support the research evaluation [45]. In the world of recommender systems, it is a common practice to use public available datasets from different application environments in order to evaluate and compare the performance of recommendation algorithms.

In general, the evaluation of performance of any context-aware recommendation model is based on four popular contextual real-world datasets from various domains: music, food, and movie. This variation enables us to assess the performance of the proposed models across a range of different datasets, each with different characteristics. We provide in the following more details about subset of these datasets.

- **Music dataset** [5] is collected from a mobile application recommending music tracks to the passengers involved in various driving and traffic conditions. The dataset contains 8 contextual dimensions and 34 contextual conditions in total.
- **Food dataset** [38] represents a contextual food preference dataset collected from a survey containing users ratings on the food menu in the context of different degrees of hunger.
- **Movie dataset** [57] is a context-aware movie dataset collected from surveys. Students were asked to rate movies in different contexts. Three contextual dimensions were captured: Time (weekend, weekday), Location (home, cinema), and Companion (alone, family, partner).
- **LDOS-CoMoDa dataset** is a movie-rating dataset collected by Odic et al. [27]. It contains ratings acquired in contextual situations that are described as a set of different contextual conditions coming from 12 various contextual dimensions, for example, social, day type, location, and mood.

The properties of these datasets are summarized in Table 7.1.

Table 7.1 Description of the used datasets

Dataset	# of users	# of items	# of ratings	# of contextual dimensions	# of contextual conditions
Music	41	139	3940	8	34
Food	212	20	6360	6	8
Movie	97	79	5035	3	12
LDOS-CoMoDa	185	4138	2297	12	49

7.4.2 Evaluation Methodology

To evaluate a CARS, the evaluation methodology defines the followed experimental protocol that can fall into one of the two main levels: the offline or the online evaluations [6, 25].

7.4.2.1 Offline Evaluation

Offline evaluations are popular methods performed in the literature to assess recommendation approaches. This kind of evaluation is realized by using collected datasets of items gathering user interactions. User behavior when interacting with the recommendation system is simulated by using the collected dataset. Since the method deals with the users behavior collected in the past, the offline evaluation does not need any interaction with real users allowing the comparison of wide range of approaches at low cost. However, offline evaluations cannot measure the effect of the recommendation system on the user behavior, they only give a first level performance evaluation by providing a good approximation of how the system would behave with real users. The basic structure for offline evaluation process is based on the train-test and cross-validation techniques. The dataset containing the information of users, items, and ratings is often partitioned. Part of this data is used to infer the optimal utility function and referred to as training set. The other part is known as the testing set and adopted to measure the recommendations performance. When the same data is used for both training and evaluation, the dataset splitting is useful for preventing algorithms from over fitting to the evaluation testing set. To split the dataset, different ways could be adopted, knowing that the chosen manner depends on the domain of application and its constraints [3].

7.4.2.2 Online Evaluation

Online evaluation is generally conducted with real users that interact with the system and give feedback based on their experience. This type of evaluation focuses on measuring the change in user behavior during the interaction with different recommender systems. Questionnaires and user studies are provided to the user for evaluating the accuracy and performance of the RS. The risk taken when carrying out online evaluation is requiring plenty of efforts in gathering the feedback responses from users. Moreover, comparing several algorithms through online experiments is expensive and time-consuming. Besides choosing an evaluation methodology, evaluation metrics are also necessary to assess the performance of recommender systems. Numerous evaluation metrics have been proposed in the RS and CARS systems literature. However, they are generally based on the famous recall and precision metrics yet used in classical information retrieval.

7.4.3 Evaluation Metrics

We now turn our attention to the different metrics adopted to assess the performance of recommender systems. A distinction needs to be made between the evaluation metrics by taking into account the goal of the system itself. Generally, these metrics can be categorized into *prediction accuracy metrics* that determine how well a system can predict the appropriate rating for an item and *top-N metrics* that measure the suitability of top-N recommendations to users. We present in the following the commonly used evaluation metrics:

7.4.3.1 Prediction Accuracy Metrics

Prediction accuracy is considered as the most discussed property in the recommendation literature. It measures how close the recommendation system rating predictions are to the users real ratings. To date, the majority of RS are based on a rating prediction phase, where the main assumption is that a RS that produces more accurate predicted ratings will be more preferred by the user. This category of evaluation metrics comprises the well known Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE) which are considered as standard metrics for many RS such as the Netflix Prize [8]. The lower the error value, the better the predictive accuracy of the recommender system is.

- **Mean Absolute Error (MAE)** measures the average absolute deviation between the system's predicted ratings and the user's actual ratings. It is given by the following equation:

$$\text{MAE} = \frac{1}{N} \sum_{i \in N} |r_{ui} - \hat{r}_{ui}| \tag{7.1}$$

  where:

  - N: the total number of recommended items.
  - $\hat{r}_{ui}$: the predicted rating of user u for item i.
  - r_{ui}: the real rating of user u for item i.

- **Root Mean Squared Error (RMSE)** measures the quadratic error and it is hence more sensitive to large errors, since the errors are squared before they are averaged. This means that the RMSE is useful when large errors are especially undesirable. The RMSE is calculated as:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i \in N}(r_{ui} - \hat{r}_{ui})^2} \tag{7.2}$$

7.4.3.2 Top-N Metrics

For evaluating the top-N recommendations, the used evaluation metrics focus on measuring the quality of top-N recommendation lists generated by RS. In this family of measures, we found two popular metrics borrowed from the field of information retrieval: *Precision* and *Recall*.

- **Precision@N** measures the fraction of relevant recommended items in the top-N position and is defined as follows:

$$\text{Precision@N} = \sum_{i=1}^{N} \frac{\text{rel (i)}}{N} \tag{7.3}$$

  Here, rel (i) indicates the relevance level of the item at position i, rel (i) = 1 if the item is relevant and rel (i) = 0 otherwise.
- **Recall@N** calculates the ratio of selected relevant items returned in the top-N position, to the total number of available relevant items Nr. Recall can be computed with the help of the following equation:

$$\text{Recall@N} = \sum_{i=1}^{N} \frac{\text{rel (i)}}{Nr} \tag{7.4}$$

  Increasing the recommendation list size may result in a higher recall but a lower precision, since a longer recommendation list tends to include relevant items. The *F-measure* evaluates the balance between these two metrics and is described as follows:

$$\text{F-measure} = \frac{\text{2.Precision.Recall}}{\text{Precision} + \text{Recall}} \tag{7.5}$$

  Besides evaluating the relevance of items in the recommendation list, it is also important to evaluate the ranking quality. In particular, we introduce the following two widely used ranking measures *Discounted Cumulative Gain (NDCG)* and the *Mean Reciprocal Rank (MRR).*

- **NDCG@N** Normalized Discounted Cumulative Gain is calculated based on computing Discounted Cumulative Gain (DCG) which measures the effectiveness of a ranked list based on items relevance. NDCG is the normalized variant of DCG, where Ideal DCG (IDCG) is the best possible DCG.

$$\text{DCG@N} = \frac{1}{N}\sum_{i=1}^{N}\frac{2^{rel(i)}-1}{\log_2(i+1)} \quad \text{IDCG@N} = \frac{1}{N}\sum_{i=1}^{k}\frac{1}{\log_2(i+1)}$$

$$\text{NDCG@N} = \frac{\text{DCG@N}}{\text{IDCG@N}} \tag{7.6}$$

- **MRR@N** Mean Reciprocal Rank is described as the multiplicative inverse of the rank of the first relevant item, L represents the relevant items list in the testing set for each user, and $Rank_i$ denotes the position of the relevant item i in the recommendation list.

$$\text{MRR@N} = \frac{1}{|L|}\sum_{i=1}^{|L|}\frac{1}{\text{rank}_i} \tag{7.7}$$

7.4.3.3 Alternative Performance Metrics

While most research in recommender systems has focused on accuracy metrics, additional characteristics of recommendations could be taken into consideration. Thus, other performance metrics such as novelty and diversity may be measured [12]. Novelty and diversity are different though related notions.

- **Novelty** evaluates whether the recommended items are new to the user or not. It would be interesting if the user is recommended with novel items. Novelty can be measured by comparing the top-N recommendations against already used or rated recommendations. Given I_R, the set of items that have been previously recommended to a user u, and I_T, the set of the top-N recommended items to u, novelty for each user u can be defined as follows:

$$Novelty_u = \frac{|I_T \backslash I_R|}{|I_T|}$$

  The average $\frac{1}{N}\sum_{u=1}^{N} Novelty_u$ can be interpreted as the measurement of novelty, where N denotes the number of users.
- **Diversity** is related to how dissimilar the recommended items are with respect to each other. The diversity can be determined using the items content (e.g., movie or music genres) or the items ratings by measuring Intra-List Similarity (ILS) [59]. ILS calculates the similarity between two items i_n and i_m in the

recommendation list L using a similarity metric such as Jaccard similarity coefficient [9]. For a user u, ILS can be computed as:

$$ILS_u = \frac{1}{2} \sum_{i_n \in L} \sum_{i_m \in L} sim(i_n, i_m)$$

From here, the overall ILS can be calculated as the average over all users.

7.4.4 Recommender Systems Platforms

The wide array of recommendation algorithms proposed over the years brings a challenge in their reproduction and comparison. Therefore, multiple open-source frameworks exist for this purpose. Many implementations of recommender algorithms are available, especially for collaborative filtering algorithms. A lot of tools are free, open-source projects that researchers can use. However, they provide only a few classic recommendation algorithms. The most two relevant ones are LibRec (Library for Recommender system)[21] and CARSKit (Context-aware Recommender system) [57]. LibRec is depicted to baseline and social recommender algorithms, whereas CARSKit uses the implementations for no-contextual recommender algorithms from the LibRec and adds the required functionality to implement contextual recommender systems. In any experiment, the use of any implementation of the recommender algorithm and a series of evaluation metrics as provided by the two latter ones to study of the two main problems of recommender systems, rating prediction and item recommendation implements a suite of state-of-the-art recommendation algorithms as well as the traditional methods. In addition, a series of evaluation metrics are implemented including diversity-based metrics which are rarely enabled in other libraries. LibRec provides a platform for fair comparisons among different algorithms in multiple aspects, given the fact that the evaluative performance depends on data characteristic. It also provides a high flexibility for expansion with new algorithms [18].

7.4.5 Conventional Methods in Contextual Recommender Systems

To evaluate any CARS solution, a comparative study must be done with a baseline. In general, the recommendation algorithms can be chosen, for example, from the java based context-aware recommendation engine [57]. A subset of algorithms that can be chosen for comparison are described below:

1. **User-oriented K-Nearest Neighbors (UserKNN)** [53] represents a neighborhood collaborative filtering algorithm on the basis of users similarity.

2. **Item-oriented K-Nearest Neighbors (ItemKNN)** [53] represents a neighborhood collaborative filtering algorithm on the basis of items similarity.
3. **Differential Context Weighting (DCW)** [55] introduces the contextual weighting in the rating prediction process through a weighted similarity measure.
4. **Singular Value Decomposition model based on implicit feedback (SVD++)** [26] represents a matrix factorization model using users history information.
5. **List-Rank Matrix Factorization (LRMF)** [50] refers to a matrix factorization ranking model that joins the list-wise learning with MF.
6. **Context-Aware Matrix Factorization (CAMF)** [5] represents an extended MF model that integrates contextual information in the rating prediction process. We tried its three variants (CAMF-C, CAMF-CI, and CAMF-CU) and we only present the best performing one, denoted by *CAMF-Dev*.
7. **Multidimensional Context Similarity (CAMF-MCS) model** [56] refers to a CAMF algorithm considering the contextual correlation aspect using a multidimensional space.
8. **Fuzzy Weighting Recommender (FWR)** Inspired by the idea of the paper [55], the rating prediction formula of Resnick's algorithm [42] to generate contextual ratings prediction through a novel proposal called Fuzzy Weighting Recommender (FWR) is adopted [18]. In this prediction process, the notion of contextual situations similarity is introduced, where the more close the contextual situations of two ratings were given, the more reliable those ratings for further predictions. Nevertheless, this effect should be restricted since integrating contexts with low similarity can lead to adding noise to the predictions. Thus, a set of similarity thresholds are introduced to filter ratings, for the each component.

   According to FWR, the predicted rating $P_{a,i,\sigma}$ that a given user a is expected to attribute to the item i depending to his contextual situation is computed as follows:

$$P_{a,i,\sigma} = \bar{\rho}(a, \sigma_3, \epsilon_3) + \frac{\sum_{n \in N_{a,\sigma_1,\epsilon_1}} (\rho(n, i, \sigma_2, \epsilon_2) - \bar{\rho}(n, \sigma_2, \epsilon_2)) \times sim_w(a, n, \sigma_4, \epsilon_4)}{\sum_{n \in N_{a,\sigma_1,\epsilon_1}} sim_w(a, n, \sigma_4, \epsilon_4)} \tag{7.8}$$

9. **CAMF-MCS strategies** The majority of the surveyed Context-Aware Matrix Factorization algorithms (CAMF) [4, 54], cannot fully capture the impact of the relevant contextual dimensions as well as their associations on the predicted ratings. This proposal consists of two strategies extended from the correlation based CAMF-MCS model suggested in [56]. Both of the two proposed strategies [18], apply a common rating prediction formula (Eq. 7.9) highlighting the notion of "contextual correlation."

$$\hat{r}_{u,i,s_t} = \mathbf{q_i} \cdot \mathbf{p_u} . Corr(s_t, s_E) \tag{7.9}$$

In the rating formula 7.9, both items and users are characterized by vectors. In fact, each item i is associated with an item vector denoted $\mathbf{q_i}$ and each user u is associated with a user vector denoted $\mathbf{p_u}$. Those vectors values are the weights on different latent factors. Precisely, the elements in $\mathbf{q_i}$ indicate the extent to which the item i obtains those latent factors. For the vector $\mathbf{p_u}$, its elements indicate how much users like those latent factors. The function denoted $(Corr(s_t, s_E))$ predicts the correlation or the similarity between a current contextual situation s_t in which the user u consume the item i and an empty contextual situation s_E.

7.5 Experimental Results: Case Study

In this section, we present a subset of experimental studies. Before conducting the experimental evaluation, we begin by performing preliminary experiments by presenting a parameter sensitivity analysis in order to set the optimum values of these parameters to be used for the further evaluation experiments.

The neighborhood-based model (FWR) and CAMF based model (WCAMF-MCS and ICAMF-MCS strategies) are used according to MAE, Precision@N (Prec@N), Recall@N (Rec@N) and NDCG@N with N $\in$ {5,10}.

7.5.1 Analyzing Parameter Sensitivity: Impact of the Number of Iterations

We present on this part the adjustment of the number of iterations parameter. We examine the number of iterations required in the Fuzzy Weighting Recommender approach (FWR) and the CAMF-MCS strategies: the weighting strategy (WCAMF-MCS) and the interaction strategy (ICAMF-MCS). Figure 7.2 reports for each dataset the prediction accuracy measured in compliance with the number of iterations.

It is apparent from Fig. 7.2 that on Music and Food datasets, FWR requires 20 iterations to get a peak prediction accuracy. When it comes to the Movie dataset, both methods indicate reduced prediction accuracy when the iterations number goes beyond 60. For the LDOS-CoMoDa dataset, the best performance is achieved by FWR at 50 iterations. For WCAMF-MCS and ICAMF-MCS, we can note that the prediction accuracy is improved when the iterations number reaches 100. We set the suitable iterations number for each method when the best prediction accuracy is achieved.

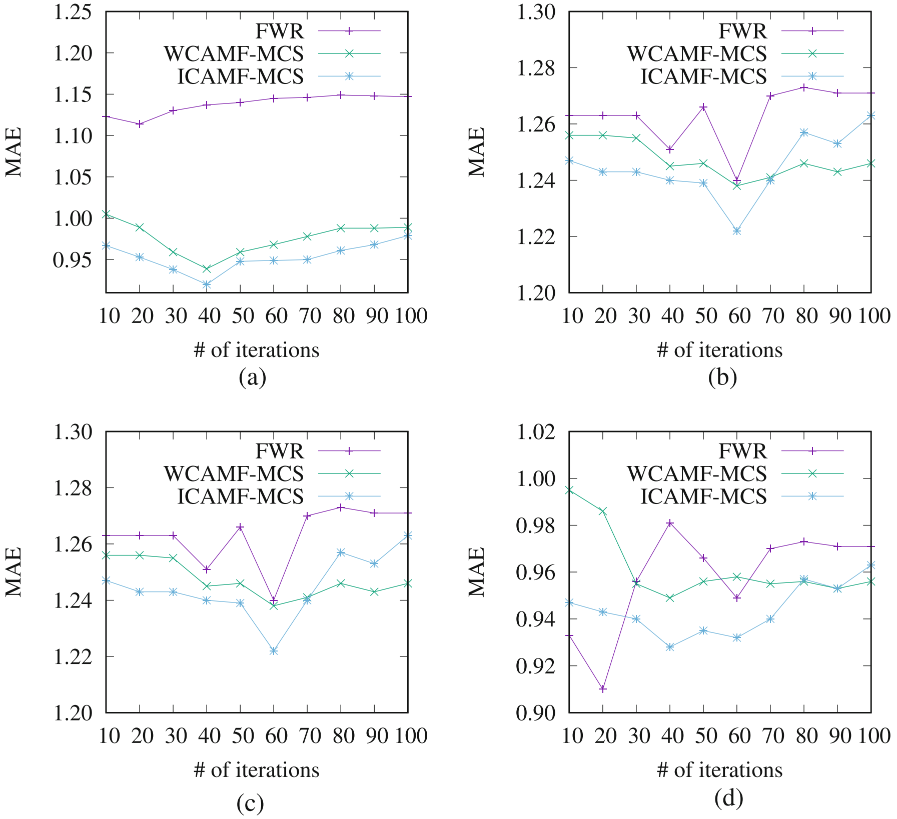


Fig. 7.2 MAE variation in different iterations numbers. (**a**) Food dataset. (**b**) Movie dataset. (**c**) Music dataset. (**d**) LDOS-CoMoDa dataset

7.5.2 Results

We present in Tables 7.2 and 7.3 the obtained experimental results between some baselines on Music, Movie, LDOS-CoMoDa and Food datasets. We can observe from the two tables below, the CAMF based model is able to outperform the neighborhood-based model. For example, ICAMF-MCS strategy gives an improvement of the Prec@5 value by 28.1%, 16.4%, 45.9%, and 14.5% over FWR, on Music, Movie, LDOS-CoMoDa, and Food datasets, respectively.

Given the fact that the neighborhood-based model can suffer from low accuracy problem due the absence of the knowledge learned about item aspects to produce accurate top-N recommendations. In addition, the neighborhood formation process, especially the user-user similarity computation step requires the calculation of user's interest similarity with all other neighbors to make predictions or recommendations which may increase the computation complexity. However, in the case of having

Table 7.2 Comparison results on the Music and Movie datasets

Dataset	Algorithm	MAE	Prec@5	Prec@10	Rec@5	Rec@10	NDCG@5	NDCG@10
Music	ItemKNN	0.983	0.015	0.014	0.043	0.079	0.040	0.045
	UserKNN	1.087	0.013	0.015	0.038	0.091	0.043	0.042
	DCW	1.064	0.058	0.052	0.090	0.144	0.121	0.123
	FWR	0.911	0.064	0.070	0.106	0.161	0.143	0.148
	SVD++	0.965	0.036	0.025	0.183	0.179	0.117	0.110
	LRMF	1.270	0.024	0.017	0.186	0.134	0.077	0.075
	CAMF-Dev	1.001	0.014	0.018	0.142	0.150	0.042	0.037
	CAMF-MCS	0.998	0.033	0.031	0.118	0.166	0.112	0.092
	WCAMF-MCS	0.939	0.078	0.071	0.191	0.172	0.128	0.129
	ICAMF-MCS	0.920	0.082	0.079	0.198	0.195	0.151	0.141
Movie	ItemKNN	1.229	0.052	0.044	0.263	0.248	0.210	0.231
	UserKNN	1.242	0.055	0.045	0.275	0.285	0.202	0.201
	DCW	1.248	0.046	0.052	0.295	0.302	0.261	0.266
	FWR	1.240	0.061	0.062	0.302	0.322	0.346	0.283
	SVD++	1.688	0.057	0.028	0.268	0.104	0.222	0.105
	LRMF	1.395	0.053	0.042	0.276	0.251	0.224	0.136
	CAMF-Dev	1.229	0.048	0.045	0.281	0.311	0.226	0.119
	CAMF-MCS	1.529	0.052	0.049	0.391	0.351	0.245	0.123
	WCAMF-MCS	1.238	0.069	0.064	0.404	0.396	0.246	0.142
	ICAMF-MCS	1.223	0.071	0.065	0.496	0.403	0.248	0.184

a sufficiently small number of users, neighborhood-based model can outperform matrix factorization based model. For example, FWR improves the best performing strategy of CAMF based model by 5% and 53.8% in terms of NDCG@10 on Music and Movie datasets, respectively. We can observe a little difference between the two strategies ICAMF-MCS and WCAMF-MCS. Most commonly, the ICAMF-MCS strategy gives a better performance than WCAMF-MCS strategy. In this respect, we can note that, ICAMF-MCS slightly improves the MAE value over WCAMF-MCS by 2%, 1.2% and 1.2% on Music, Movie and Food datasets, respectively. ICAMF-MCS strategy is also able to beat WCAMF-MCS strategy in terms of Prec@10 and Rec@10 on LDOS-CoMoDa dataset by an improvement of 23.1% and 80.7%, respectively. The obtained experimental results show the superior performance of the ICAMF-MCS strategy especially on rich contextual datasets. In fact, this latter strategy takes into account the interaction that may exist between the relevant contextual dimensions according to their fuzzy measures. Therefore, the strategy that considers correlated contextual dimensions outperforms the one considering independent contextual dimensions. As a result, the interaction among the relevant contextual dimensions may be considered as a better framework to understand and represent the contextual effects on recommendation. For instance, a user may more precisely decide a movie if the time contextual dimension is correlated with companion dimension rather than considering these contextual dimensions separately.

Table 7.3 Comparison results on the LDOS-CoMoDa and Food datasets

Dataset	Algorithm	MAE	Prec@5	Prec@10	Rec@5	Rec@10	NDCG@5	NDCG@10
LDOS-CoMoDa	ItemKNN	0.973	0.007	0.006	0.024	0.020	0.024	0.030
	UserKNN	0.952	0.004	0.005	0.019	0.025	0.021	0.027
	DCW	0.830	0.002	0.005	0.017	0.026	0.019	0.022
	FWR	0.821	0.037	0.025	0.026	0.028	0.031	0.033
	SVD++	0.871	0.024	0.025	0.018	0.017	0.021	0.040
	LRMF	2.004	0.011	0.012	0.012	0.009	0.007	0.005
	CAMF-Dev	0.867	0.022	0.020	0.027	0.022	0.006	0.006
	CAMF-MCS	1.021	0.042	0.032	0.016	0.010	0.008	0.005
	WCAMF-MCS	0.848	0.048	0.039	0.033	0.026	0.018	0.059
	ICAMF-MCS	0.804	0.054	0.048	0.043	0.047	0.033	0.061
Food	ItemKNN	1.183	0.060	0.065	0.106	0.146	0.119	0.118
	UserKNN	1.214	0.038	0.046	0.099	0.120	0.118	0.120
	DCW	1.206	0.069	0.032	0.105	0.118	0.101	0.107
	FWR	1.114	0.076	0.040	0.145	0.151	0.125	0.122
	SVD++	1.119	0.062	0.050	0.131	0.150	0.123	0.128
	LRMF	1.270	0.065	0.047	0.172	0.147	0.128	0.118
	CAMF-Dev	1.007	0.040	0.048	0.122	0.146	0.138	0.117
	CAMF-MCS	1.529	0.080	0.072	0.144	0.172	0.137	0.123
	WCAMF-MCS	0.938	0.086	0.073	0.190	0.177	0.139	0.143
	ICAMF-MCS	0.927	0.087	0.082	0.199	0.197	0.154	0.134

As expected, Tables 7.2 and 7.3 show that the neighborhood-based CF model (FWR) can significantly improves the rating accuracy metric MAE over the previous popular neighborhood-based CF approaches (ItemKNN, UserKNN and DCW). For example, FWR achieves an MAE value equals to 1.114 while the best performing neighborhood-based baseline achieves an MAE value equals to 1.183 on Food dataset. It also can be found that, on Music dataset, FWR improves the MAE value from 0.983 (the MAE of the best performing neighborhood-based baseline) to 0.911. Furthermore, when it comes to the top-N recommendation task, FWR is also able to achieve higher ranking metric values and thus outperforms the neighborhood-based baselines. For instance, on LDOS-CoMoDa dataset, FWR gives an improvement in terms of Rec@5 by 8.3% over ItemKNN, 36.8% over the UserKNN and 52.9% over the DCW. Therefore, the comparative neighborhood-based CF models always show lower results than the neighborhood based model FWR. A possible explanation for this is that these baselines ruled out the influence of contextual dimensions relevancy and interaction in determining suitable neighbors with similar contexts which may increase the computational complexity in the neighborhood formation process and thus decrease recommendation accuracy.

Regarding the comparison between the matrix factorization-based models, we can notice that CAMF based models (CAMF-Dev and CAMF-MCS) work better than MF models (SVD++ and LRMF). Nevertheless, it can be found that, in terms of MAE, MF models such as SVD++ can improve the CAMF-MCS by 36.6% on Food dataset, this may have occurred due to the small contextual conditions number in this dataset.

The two strategies (WCAMF-MCS and ICAMF-MCS) can achieve a superior recommendation performance over prior CAMF models, particularly ICAMF-MCS strategy. It outperforms Rec@5 by 41.3% and 76.5% relative to CAMF-MCS and CAMF-Dev, respectively, on Movie dataset. Moreover, on LDOS-CoMoDa dataset, ICAMF-MCS makes better Rec@5 value by 59.3% and 168.7% than CAMF-Dev and CAMF-MCS, respectively.

Let us note that ICAMF-MCS usually obtains the preferable results consistently which prove the accuracy of the interaction based CAMF strategy and confirms the efficiency of employing weighted correlated contextual dimensions in the prediction process using factorization techniques.

7.6 CARS Systems on Business

Recommender systems (RSs), initially introduced to address the problem of improving the customer experience and retention in e-commerce sites [30, 44, 51] has since become a ubiquitous and often anticipated functionality of many online interactions, from movie and song recommendations [13, 37, 49] to applications related to tourism [10, 19], social networks [36, 52], health [15], and many more. One of the important potential benefits of recommendation systems is their ability to continuously adapt to the preferences of the user.

The applications of recommender systems include recommending movies, music, television programs, books, documents, websites, conferences, tourism scenic spots and learning materials, and involve the areas of e-commerce, e-learning, e-library, e-government, and e-business services.

Collaborative filtering (CF) is the most popular algorithm used to design various applications and sites for recommender systems such as Facebook, Twitter, Google, LinkedIn and Netflix. For example, Same Mckinsey[13] study highlights that 75% of Netflix viewing is driven by recommendations. The underlying idea behind CF is that users with common interests in the past are more likely to keep exhibiting similar interests in the future. The principal property to work with collaborative filtering are the ratings given by users for items.

Amazon.com uses item-to-item collaborative filtering recommendations on most pages of their website and e-mail campaigns. According to McKinsey, 35% of Amazon purchases are thanks to recommendations systems. They suggest the most relevant items to buy and, as a result, increase a company's revenue. These suggestions are based on users' behavior and history that contain information on their past preferences.

Spotify generates a new customized playlist for each subscriber called "Discover Weekly" which is a personalized list of 30 songs based on users' unique music taste. A music recommendation engine uses three types of recommendation model: Collaborative Filtering, Natural language processing and Audio file analysis.

Many restaurant recommendation applications are available for public. For example, Google Maps,[14] helps restaurant diners to know what to order. Hence, Maps has transformed from just being a service that offers directions for a commute to more like a search engine for finding coffees, restaurants, shopping centers, etc. Google Maps uses the user's current location as the search query to rank the nearby POIs and then present them to the user. Another popular solution for Restaurant recommendation is the Yelp2app.[15] It provides users with many options, including selecting the price range, sorting restaurants by distance, and many other sophisticated options.

7.7 Conclusion

In this chapter, we introduced an overview of recommender systems and explained how basically these systems work. Therefore, we presented the basic concepts, the recommendation problem formulation, and the main recommendation techniques as well as their principal limitations. We attempted to extend existing knowledge and trace the evolution of the recommendation problem by considering recent emerging

[13] https://www.Mckinsey.com/.

[14] https://techpp.com/2020/03/09/personalised-restaurant-recommendations-google-maps/.

[15] https://www.yelp.com.

trends. We also gave an overview on performance evaluation methodology. We will focus on future work on multi-criteria decision making for CARS systems and we will discuss the main existing approaches in these areas.

References

1. G. Adomavicius et al., Incorporating contextual information in recommender systems using a multidimensional approach. ACM Trans. Inf. Syst. **23**(1), 103–145 (2005)
2. M. Al-Ghossein, Context-aware recommender systems for real-world applications. Ph.D. Thesis. Université Paris-Saclay (ComUE) (2019)
3. K. Arour, S. Zammali, A. Bouzeghoub, Test-bed building process for context-aware peer-to-peer information retrieval evaluation. Int. J. Space Based Situated Comput. **5**(1), 23–38 (2015). https://doi.org/10.1504/IJSSC.2015.067980
4. L. Baltrunas, B. Ludwig, F. Ricci, Matrix factorization techniques for context aware recommendation, in *Proceedings of the Fifth ACM Conference on Recommender Systems* (2011), pp. 301–304
5. L. Baltrunas, B. Ludwig, F. Ricci, Matrix factorization techniques for context aware recommendation, in *Proceedings of the Fifth ACM Conference on Recommender Systems*. RecSys '11. Chicago (2011), pp. 301–304. ISBN:978-1-4503-0683-6
6. J. Beel, S. Langer, A comparison of offline evaluations, online evaluations, and user studies in the context of research-paper recommender systems, in *Research and Advanced Technology for Digital Libraries*, ed. by S. Kapidakis, C. Mazurek, M. Werla (Springer International Publishing, Cham, 2015), pp. 153–168
7. R. Bell, Y. Koren, C. Volinsky, Modeling relationships at multiple scales to improve accuracy of large recommender systems, in *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2007), pp. 95–104
8. J. Bennett, S. Lanning, N. Netflix, The Netflix Prize, in *In KDD Cup and Workshop in Conjunction with KDD* (2007)
9. E. Blanchard, M. Harzallah, P. Kuntz, A generic framework for comparing semantic similarities on a subsumption hier archy, in *Proceedings of the 2008 Conference on ECAI 2008: 18th European Conference on Artificial Intelligence* (IOS Press, Amsterdam, 2008), pp. 20–24. ISBN: 978-1-58603-891-5. http://dl.acm.org/citation.cfm?id=1567281.1567291
10. M. Braunhofer, F. Ricci, Selective contextual information acquisition in travel recommender systems. Inf. Technol. Tour. **17**(1), 5–29 (2017)
11. R. Burke, Hybrid recommender systems: survey and experiments. User Model. User-Adap. Inter. **12**(4), 331–370 (2002)
12. P. Castells, N.J. Hurley, S. Vargas, *Novelty and Diversity in Recommender Systems*. Recommender Systems Handbook (Springer, Boston, 2015)
13. K. Chapphannarungsri, S. Maneeroj, Combining multiple criteria and multidimension for movie recommender system, in *Proceedings of the International MultiConference of Engineers and Computer Scientists*, vol. 1 (2009)
14. G. Chen, D. Kotz, A survey of context-aware mobile computing research. in *Dartmouth Computer Science Technical Report TR2000-381* (2000)
15. A. Civit-Balcells, L. Fernandez-Luque, F. Luna-Perejon, H. de Vries, Analyzing, analyzing recommender systems for health promotion using a multidisciplinary taxonomy: a scoping review. Int. J. Med. Inform. **1** (2017). https://doi.org/10.1016/j.ijmedinf.2017.12.018
16. A.K. Dey, G.D. Abowd, D. Salber, A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. Hum. Comput. Interact. **16**(2–4), 97–166 (2001)

17. R. Dridi et al., An improved context-aware matrix factorization model incorporating fuzzy measures, in *2018 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2018, Rio de Janeiro, July 8–13, 2018* (IEEE, Piscataway, 2018), pp. 1–8. https://doi.org/10.1109/FUZZ-IEEE.2018.8491439
18. R. Dridi et al., Effective rating prediction based on selective contextual information. Inf. Sci. **510**, 218–242 (2020)
19. D. Gavalas et al., Mobile recommender systems in tourism. J. Netw. Comput. Appl. **39**, 319–333 (2014)
20. D. Goldberg et al., Using collaborative filtering to weave an information tapestry. Commun. ACM **35**(12), 61–70 (1992)
21. G. Guo et al., LibRec: a Java library for recommender systems, in *Posters, Demos, Late-breaking Results and Workshop Proceedings of the 23rd Conference on User Modeling, Adaptation, and Personalization (UMAP 2015), Dublin, June 29–July 3, 2015*, ed. by A.I. Cristea et al., vol. 1388. CEUR Workshop Proceedings. CEUR-WS.org (2015). http://ceur-ws.org/Vol-1388/demo_paper1.pdf
22. L. Hong et al., Context-aware recommendation using role-based trust network. ACM Trans. Knowl. Discov. Data **10**(2), 1–25 (2015)
23. Jhamb, Yogesh, "Machine Learning Models for Context-Aware Recommender Systems" (2018). Engineering Ph.D. Theses. 15. https://scholarcommons.scu.edu/eng_phd_theses/15
24. M. Jin et al., Combining deep learning and topic modeling for review understanding in context-aware recommendation, in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, vol. 1 (Long Papers) (2018), pp. 1605–1614
25. Y. Kim, S.B. Cho, A recommendation agent for mobile phone users using bayesian behavior prediction, in *Proceedings of the Third International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies*, Sliema (2009), pp. 283–288
26. Y. Koren, Factor in the neighbors: scalable and accurate collaborative filtering. ACM Trans. Knowl. Discov. Data **4**(1), 1:1–1:24 (2010). ISSN: 1556-4681
27. A. Košir et al., Database for contextual personalization. Elek Trotehniski Vestnik **78**(5), 270–274 (2011)
28. B. Lamche et al., Context-aware recommendations for mobile shopping, in *Proceedings of the Workshop on Location-Aware Recommendations, LocalRec, Co-located with the 9th ACM Conference on Recommender Systems (RecSys), Vienna, September 19* (2015), pp. 21–27
29. Q.-H. Le et al., A state-of-the-art survey on context-aware recommender systems and applications. Int. J. Knowl. Syst. Sci. **12**(3), 1–20 (2021)
30. Q. Li, C. Wang, G. Geng, Improving personalized services in mobile commerce by a novel multicriteria rating approach. in *Proceedings of the 17th International Conference on World Wide Web* (2008), pp. 1235–1236
31. A. Livne et al., Deep context-aware recommender system utilizing sequential latent context (2019). Preprint. arXiv:1909.03999
32. P. Lops, M. De Gemmis, G. Semeraro, Content-based recommender systems: state of the art and trends, in *Recommender Systems Handbook* (Springer, Berlin, 2011), pp. 73–105
33. Z. Meng et al., Variational bayesian context-aware representation for grocery recommendation (2019). Preprint. arXiv:1909.07705
34. F. Meyer, Recommender systems in industrial contexts (2012). Preprint. arXiv:1203.4487
35. S. Michael, W. Peter, *War and Society in Twentieth-Century France*, ed. by M. Scriven, P. Wagstaff (Martin's Press, New York, 1991), xii, 304 p., [4] p. of plates. ISBN: 0854962921
36. M. Nilashi et al., Analysis of travellers' online reviews in social networking sites using fuzzy logic approach. Int. J. Fuzzy Syst. **21**(5), 1367–1378 (2019)
37. A. Odić et al., A.: relevant context in a movie recommender system: users' opinion vs. statistical detection, in *Proceedings of the 4th Workshop on Context-Aware Recommender Systems, CARS, September 9, Dublin* (2012)
38. C. Ono et al., *Context-Aware Preference Model Based on a Study of Difference Between Real and Supposed Situation Data* (Springer, Berlin, 2009), pp. 102–113

39. A.M. Otebolaku, M.T. Andrade, Context-aware personalization using neighborhood-based context similarity, in *Wireless Personal Communications* (2016), pp. 1–24
40. N. Polatidis, C.K. Georgiadis, Mobile recommender systems: an overview of technologies and challenges, in *2013 Second International Conference on Informatics & Applications (ICIA)* (IEEE, Piscataway, 2013), pp. 282–287
41. P. Resnick, H.R. Varian, Recommender systems. Commun. ACM **40**(3), 56–58 (1997)
42. P. Resnick et al., GroupLens: an open architecture for collaborative filtering of netnews, in *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work* (1994), pp. 175–186
43. F. Ricci, L. Rokach, B. Shapira, Introduction to recommender systems handbook, in *Recommender Systems Handbook* (Springer, Berlin, 2011), pp. 1–35
44. C. Richthammer, G. Pernul, Situation awareness for recommender systems. Electron. Commer. Res., 1–24 (2018)
45. K.V. Rodpysh, S.J. Mirabedini, T. Banirostam, Correction to: employing singular value decomposition and similarity criteria for alleviating cold start and sparse data in context-aware recommender systems. Electron. Commer. Res. **22**(1), 223 (2022)
46. I.H. Sarker, A machine learning based robust prediction model for real-life mobile phone data. Internet Things **5**, 180–193 (2019)
47. I.H. Sarker, Y.B. Abushark, A.I. Khan, ContextPCA: predicting context-aware smartphone apps usage based on machine learning techniques. Symmetry **12**(4), 499 (2020)
48. B.N. Schilit, M.M. Theimer, Disseminating active map information to mobile hosts. IEEE Netw. **8**(5), 22–32 (1994). ISSN: 0890-8044. https://doi.org/10.1109/65.313011
49. A. Sen, M. Larson, From sensors to songs: a learning-free novel music recommendation system using contextual sensor data, in *Proceedings of the Workshop on Location-Aware Recommendations, Local-Rec, Vienna, September 19* (2015), pp. 40–43
50. Y. Shi, M.A. Larson, A. Hanjalic, List-wise learning to rank with matrix factorization for collaborative filtering, in *Proceedings of the Fourth ACM Conference on Recommender Systems* (2010), pp. 269–272. ISBN: 978-1-60558-906-0
51. K. Shin et al., One4all user representation for recommender systems in E-commerce. CoRR abs/2106.00573 (2021). arXiv: 2106.00573. https://arxiv.org/abs/2106.00573
52. T. Stepan et al., Incorporating spatial, temporal, and social context in recommendations for location-based social networks. IEEE Trans. Comput. Soc. Syst. **3**(4), 164–175 (2016)
53. H.-M. Wang, G. Yu, Personalized recommendation system K neighbor algorithm optimization, in *Proceedings of International Conference on Information Technologies in Education and Learning* (ICITEL 2015), Atlantis Press. pp. 1–4
54. Y. Zheng, Context-aware collaborative filtering using context similarity: an empirical comparison. Information **13**(1), 42 (2022)
55. Y. Zheng, R. Burke, B. Mobasher, Recommendation with differential context weighting, in *21th International Conference, Rome, June 10–14* (2013)
56. Y. Zheng, B. Mobasher, R. Burke, Incorporating context correlation into context-aware matrix factorization, in *Proceedings of the International Conference on Constraints and Preferences for Configuration and Recommendation and Intelligent Techniques for Web Personalization*, Buenos Aires (2015), pp. 21–27
57. Y. Zheng, B. Mobasher, R.D. Burke, CARSKit: a Java-based context-aware recommendation engine, in *IEEE International Conference on Data Mining Workshop, ICDMW, Atlantic City, November 14–17* (2015), pp. 1668–1671
58. X. Zheng et al., A new recommender system using context clustering based on matrix factorization techniques. Chinese J. Electron. **25**(2), 334–340 (2016)
59. C.-N. Ziegler et al., Improving recommendation lists through topic diversification, in *Proceedings of the 14th International Conference on World Wide Web*, WWW '05, Chiba (ACM, New York, 2005), pp. 22–32. ISBN: 1-59593-046-9. https://doi.org/10.1145/1060745.1060754. http://doi.acm.org/10.1145/1060745.1060754

Chapter 8
An Overview of Multi-View Methods for Text Clustering

Maha Fraj, Mohamed Aymen Ben HajKacem, and Nadia Essoussi

8.1 Introduction

The rapid development of information technology and the abundant amount of available data have considerably contributed to the growth of studies on multi-view clustering [8, 32] . Multi-view data is observed from varying points resulting in different representations (views) with distinct statistical properties. In text clustering, these views can be obtained through word frequencies, topic and context based representations, and many other embedding models capable of capturing either syntactic or semantic information or both [14]. The main task of multi-view text clustering is to achieve better clustering by combining information held by each view, such information is disregarded when only a single view is used. However, efficiently integrating different views while preserving their characteristics remains a challenge. A naive solution for multi-view clustering consists in concatenating features from all views then apply a single-view clustering algorithm. Nevertheless, such combination fails to exploit the specificity of each view. Hence, multiple approaches have been proposed to optimize multi-view clustering [16, 19, 35].

This chapter reviews multi-view methods for text clustering. In fact, textual data was examined early on in the context of multi-view, particularly in cross-lingual text categorization where the data is labeled in one view and not in another, the aim is to use the information in both views to label all data [1, 28, 30]. With the abundance of unlabeled data, this process was extent to multi-view text clustering [13].

M. Fraj (✉) · M. A. Ben HajKacem · N. Essoussi
Institut Supérieur de Gestion de Tunis, LARODEC, Université de Tunis, Le Bardo, Tunisia
e-mail: maha.fraj.m@gmail.com; nadia.essoussi@isg.rnu.tn

B. Alyoubi et al. (eds.), *Machine Learning and Data Analytics for Solving Business Problems*, Unsupervised and Semi-Supervised Learning,
https://doi.org/10.1007/978-3-031-18483-3_8

The reminder of this chapter is organized as follows: Sect. 8.2 presents an overview of exiting multi-view clustering methos, specifically for text clustering. Section 8.3 evaluates the performance on real-world textual data. Finally, Sect. 8.4 presents the conclusion and current challenges.

8.2 Overview of Multi-View Textual Data Clustering

The main challenge of multi-view clustering consists in integrating the different views while taking advantage of the characteristics of each view to improve the clustering results. An intuitive solution consists of concatenating all features from all views and apply a clustering algorithm afterwards, this, however, ignores the statistical properties of each view and can conceal valuable information [3]. To this end, according to the integration scheme, existing multi-view clustering methods can be presented under three main categories [22]. The first category called late integration derives clustering results from each view individually, then a fusion step is applied to reach a consensus clustering [7, 29]. The second category is based on co-training, which incorporate multi-view integration into the clustering process directly through jointly optimizing the objective function [2, 17]. The third category is based on space learning, such that views are mapped into a new space to reveal the latent data structure. We present in the following the characteristics of each category and detail a number of existing methods.

8.2.1 Late Integration Based Methods

The late integration approach, also known as late fusion, consists of applying a clustering algorithm on each view individually and subsequently combines the results into a consensus clustering. The idea examines the relations between the clusters derived from each view rather than the relations between data points. The combination of clustering results can be obtained using different methods, such as latent probabilistic models [7] or more recently ensemble methods [9, 13, 26]. Figure 8.1 presents the overall process of late integration based methods.

8.2.1.1 Ensemble Methods for Multi-View Text Clustering

Xie et al. [31] proposed a multi-view clustering ensembles, an combination of multi-view clustering algorithms and ensemble clustering. The method extends both multi-view kernel K-means [27] and multi- view spectral clustering [16] to ensemble clustering and compares the two methods. Given a data set **X**, different clustering results $\{\pi^1, \pi^2, \ldots, \pi^H\}$ are obtained through different runs of the clustering algorithm. These clustering are then combined based on plurality voting,

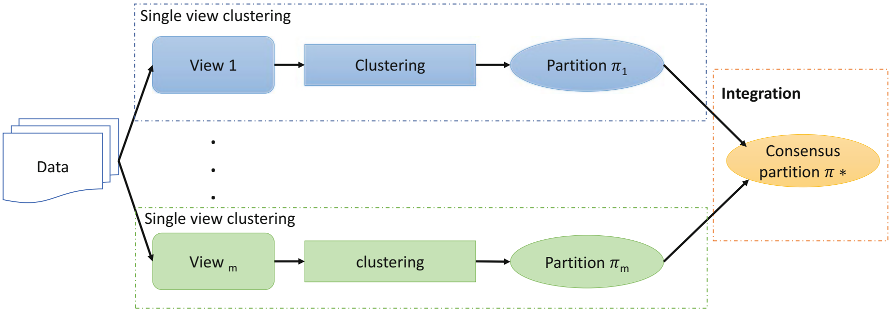


Fig. 8.1 General process of late integration based methods

Algorithm 1: Multi-view clustering ensembles

```
Input: data set X, number of clusters $k$, number of clustering ensembles $L$
Output: clustering ensembles
for each$\sigma_l \in [\sigma_{min}, \sigma_{max}]$ do
    for $v = 1$ to $m$ do
        Compute RBF $K^v = \exp \frac{-\|x_i^v - x_j^v\|^2}{2\sigma_l^2}$
    end for
    $\tilde{K}_l = [K_l^1, K_l^2, \ldots, K_l^m]$
    Run multi-view kernel K-means or multi-view spectral clustering with $\tilde{K}$ and $k$
end for
Combine the clusterings using selective voting
```

i.e., considering the majority cluster label for each data point to give the final clustering $\pi *$.

Hussain et al. [13] presented a late integration framework for multi-view document clustering based on ensemble method. The proposed method first converts views into term weighted matrices using two weighting schemes: TF-IDF and TF-ICF [24]. Hierarchical clustering is then applied on each view individually to obtain different partitions. In order to aggregate the clustering results, different ensemble techniques are adopted: the Cluster Based Similarity Partitioning (CBSP) [25], the Pairwise Dissimilarity (PD) [36], and the Affinity matrix based technique. Each ensemble technique provides a similarity matrix, which are then aggregated into an overall similarity matrix used for the final clustering. Similarly, Fraj et al. [9] proposed a multi-view ensemble methods for text clustering (MEMTC) based on multiple representations. The main idea consists of integrating different text representation models: TF-IDF, LDA, and skip-gram to generate, respectively, syntactic, topic, and semantic views. Lastly, ensemble techniques CBSP and Pairwise Dissimilarity are used to aggregate the different partitions yielded by each view. The main steps of MEMTC are presented in Algorithm 2

Algorithm 2: Ensemble methods for multi-view clustering

```
   Input: a collection of text documents $\mathbf{X}$
   Output: final consensus clustering
1  $\mathbf{X}_v \leftarrow \mathrm{R}(\mathbf{X})$    $v \in \{$TF-IDF, LDA, Skip-gram $\}$
                                                   // R: document representation
2  Apply hierarchical clustering to obtain per-view partitions. Calculate the cluster based
   similarity partitioning matrix $\mathrm{S_H}$
3  Calculate the pairwise similarity matrix $\mathrm{S_{PDM}}$
4  Aggregate the similarity matrices into one matrix S
5  Apply the hierarchical clustering on S
```

8.2.1.2 Multi-View Clustering Based on Latent Models

Bruno et al. [7] proposed an integration framework based on latent models for document clustering. In this work, documents from each view are clustered into k^v clusters. The set of clusterings $\{c_1^v, \ldots, c_k^v\}$ are then concatenated into $K \times M$ matrix $\mathbf{C}$, such that $K = \sum_v k^v$ is the total number of clusters over all views. Based on $\mathbf{C}$, a joint probability $P(c_k, c_{k'})$ is derived to deduce the pairwise cluster agreement, which represents the number of documents belonging simultaneously to clusters c_k and $c_{k'}$. The joint cluster-cluster probability is defined as follows:

$$\begin{aligned} P(c_k, c_{k'}) &= \sum_n P(c_k|x_i)P(c_{k'}|x_i)P(x_i) \\ &= \sum_n \frac{P(c_k, x_i)P(c_{k'}, x_i)}{P(x_i)} \end{aligned} \tag{8.1}$$

where the joint-cluster document probability is obtained by:

$$P(c_k, x_i) = \frac{\mathbf{C}_{k,i}}{MN}, \quad \forall k \in [1, K], \forall i \in [1, N] \tag{8.2}$$

To obtain the final clustering for each document, the Probabilistic Latent Semantic Analysis (**PLSA**) [12] is adopted to derive latent variable z_j such that

$$P(c_k, c_{k'}) = P(c_{k'}) \sum_j^L P(c_k|z_j)P(z_j|c_{k'}) \quad , j = 1, \ldots, L \tag{8.3}$$

PLSA seeks to find the relationship between the clusters observations across different views and the latent variables z. The overall clustering is established by assigning to document x_i the discrete variable z_j that maximizes the following posterior probability:

Algorithm 3: A late fusion approach using latent models

```
Input: multi-view documents $\mathbf{X}^v$
Output: Final clustering assignment $z$
Run a clustering algorithm on each view individually
Concatenate clusterings $\{c_1^v, \ldots, c_k^v\}$ into matrix $\mathbf{C}$
Apply PLSA using Eqs. 8.1 and 8.2
for $i = 1$ to $N$ do
    Assign $z_j$ to $x_i$ by maximizing Eq. 8.4
end for
```

$$P(z_j|x_i) = \frac{\sum_k P(z_j|c_k)P(c_k, x_i)}{P(x_i)} \tag{8.4}$$

To estimate the latent variables, experiments were carried using the Expectation-Maximization (**EM**) algorithm and the Nonnegative Matrix Factorization (**NMF**) [11] where both methods have performed similarly. Algorithm 3 summarizes the main step of the approach.

8.2.2 *Co-training Based Methods*

Co-training based methods seek to find a consensus by maximizing the mutual agreement across all views. Co-training was originated by Blum et al. [4] in order to tackle semi-supervised problem. Given the abundance of unlabeled data, such data can be used to enrich the training set of the labeled data, such that given two views the learning algorithm is trained on the labeled data of both views in a bootstrapping manner. Finally, based on the consensus principle, the views should agree on all labeled data. Eventually, co-training was adopted in unsupervised learning [3] and has shown good performance despite the absence of labeled data. In general, co-training based methods are based on three main assumptions: *Sufficiency*: each view is sufficient to perform the clustering task, *Compatibility*: each pair of views predicts with high probability the same label for data points with co-occurring features, and *Conditional independence*: the views are conditionally independent given the class label [16]. Figure 8.2 presents the general process of co-training based methods.

8.2.2.1 Multi-View K-Means Based Methods

Multi-view clustering was advanced by Bickel et al. [3], where the empirical results show that the proposed multi-view spherical k-means improves the quality of document clustering in comparison to the single-view version of the algorithm. The presented co-training algorithm is based on the following assumptions: given two views v^1 and v^2, each view is sufficient to output clustering results by itself, and

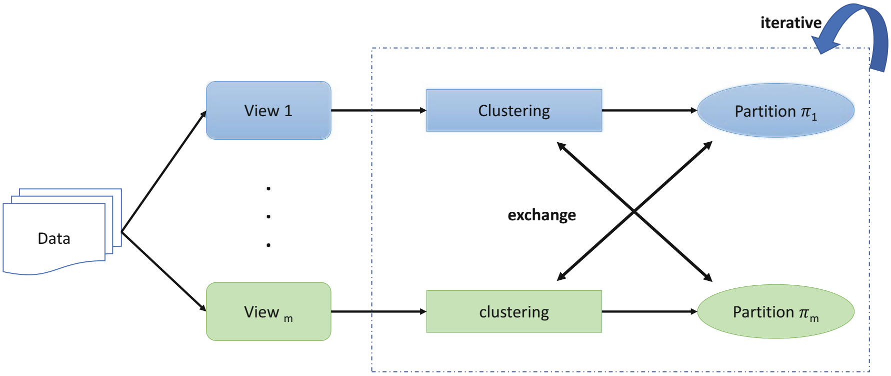


Fig. 8.2 General process of co-training

views are conditionally independent given the class label. The clustering process starts by randomly initializing the set of parameters Θ^v including the centers c_j^v, $j = 1, \ldots, k$, where k is the desired number clusters and $v = 1$ or $v = 2$. Documents are then assigned to clusters given the smallest computed distance to c_j^v. A two-step iterative process is applied afterwards taking turns between views. The first step consists of updating the clusters centers such that:

$$c_j^v = \frac{\sum_{x^v \in \pi_j^v} x^v}{\| \sum_{x^v \in \pi_j^v} x^v \|} \tag{8.5}$$

where π_j^v is the jth partition given the vth view. The assignment step consists of calculating the distance between documents and centers, and finding the new partitions. After each iteration, partitions are exchanged for an updating and assignment steps for the other view. For the final clustering, consensus centers are calculated by considering the documents that both views agree on such that:

$$cons\_c_j^v = \frac{\sum_{x_i^1 \in \pi_j^1 \wedge x_i^2 \in \pi_j^2} x_i^v}{\| \sum_{x_i^1 \in \pi_j^1 \wedge x_i^2 \in \pi_j^2} x_i^v \|} \tag{8.6}$$

The final partitioning is obtained by assigning documents to the closest consensus vector. Given that the algorithm is based on alternating partitions between views, convergence is not guaranteed. The main steps of multi-view spherical k-means are presented in Algorithm 4.

Bettoumi et al. [2] proposed a collaborative multi-view K-means CO-K-means that introduces an interconnection term to overcome the inter-view disagreement.

Algorithm 4: Multi-view spherical k-means

```
Input: multi-view documents $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \dots, \mathbf{X}^m\}$, number of clusters $k$
Output: Final clustering assignment $\pi$
Initialize randomly $\Theta^2$ and $c_j^2, \quad j = 1, \dots, k$
Assign each document to the partition corresponding to the closest center $c_j^2$
t=0
while $t < tmax$ do
    for $v = 1 : 2$ do
        $t = t + 1$
        Calculate the new centers using Eq. 8.5
        Compute the cosine distance between documents and centers
        Assign each document its closest center
    end for
    Compute Objective function by $J(\Theta_t) = \sum_{j=1}^{k} \sum_{x^v \in \pi_j^v} \langle x^v, c_j^v \rangle$
    if $(J(\Theta_t) < J(\Theta_{min}))$ then
        $t = 0$
    end
end
Calculate the consensus centers using Eq. 8.6
Find the final clustering assignment
```

Views are encouraged to reach an agreement by minimizing the contradiction across partitions. To solve this problem, the K-means objective function is altered such that:

$$\Omega = \sum_{v} \sum_{i} \sum_{k} \|\mathbf{x}_i^v - \mathbf{c}_k^v\|_2^2 + \mu\varphi \tag{8.7}$$

where μ is a modulation parameter and φ is the interconnection term denoted by:

$$\varphi = \frac{1}{|V| - 1} \sum_{v > v'} \sum_{i}^{n} \sum_{k} (\|\mathbf{x}_i^v - \mathbf{c}_k^v\|_2^2 - \|\mathbf{x}_i^{v'} - \mathbf{c}_k^{v'}\|_2^2) \tag{8.8}$$

Similarly to the classic K-means, the proposed algorithm starts by randomly initializing the clusters centers for each view, followed by an assignment step. Then, for each view, new centers are computed. The interconnection term φ aims to reduce the distance between the partitions yielded from each view. The main steps of Co-K-means are given in Algorithm 5.

8.2.2.2 Self-Organizing Map Multi-View Clustering

Fraj et al. [10] proposed a multi-view clustering method based on the Self-Organizing Map neural network [15]. Similarly to [9], each view corresponds to

Algorithm 5: Collaborative multi-view K-means

```
Input: multi-view data $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^m\}$, number of clusters $k$
Output: Final clustering assignment $\pi$
For all views, initialize the clusters centers.
repeat
    Assign data points to clusters with the smallest distance
    $\pi = \text{argmin}(\sum_v \|\mathbf{x}_i^v - \mathbf{c}^v\|_2^2 + \frac{1}{|V|-1}(\sum_{v>v'} \|\mathbf{x}_i^v - \mathbf{c}_k^v\|_2^2 - \|\mathbf{x}_i^{v'} - \mathbf{c}_k^{v'}\|_2^2))$
    for $v = 1 : m$ do
        Update centers $\mathbf{c}_k^v$ by $\mathbf{c}_k^v = \underset{c^v}{\text{argmin}} \sum \|\mathbf{x}_i^v - \mathbf{c}^v\|_2^2$
    end for
until convergence of 8.7;
```

a text representation model, i.e., TF-IDf, LDA, and skip-gram. The views are presented as input layers, such that each document has three vector representations $\mathbf{x} = \{x^1, x^2, x^3\}$. Documents are then mapped onto the output layer, such that each document is assigned to a node on the map. Consequently, each node (neuron) of the output layer is defined by v prototypes $\mathbf{w}$ each of which is associated with a view. First, the learning process consists in generating random SOM prototypes, $\mathbf{W}^v$. Secondly, an overall distance is calculated for each document $\mathbf{x}_i^v$ in the view v and the node $\mathbf{w}$ such that

$$D = \sum_i D_v(x^v, w), \ \ v \in 1, 2, 3 \tag{8.9}$$

The node with the smallest distance is considered the Best Matching Unit BMU to which the document $\mathbf{x}_i$ is assigned. The number of nodes on the output map is set empirically to boost the performance of the SOM learning, the number, however, may not coincide with the desired number of clusters which is usually less important. Therefore, the nodes on the map are clustered using agglomerative hierarchical clustering and each document is assigned to the same cluster as its corresponding SOM node. The main steps of MVSOM are presented in Algorithm 6.

8.2.2.3 Multi-View Spectral Clustering

Kumar et al. [16] have presented a co-training based spectral clustering, where two views exchange the eigenvectors resulting from the graph Laplacian of each view. The algorithm ensures consistency across views such that if two points are assigned in same cluster in one view, it should be so in all the views. On the other hand, if two points belong to different clusters in one view, they should be clustered separately across all views. The proposed algorithm first builds an adjacency matrix $\mathbf{A}^{\mathbf{v}}$ for each view, from which the graph Laplacian matrix $\mathbf{L}^v$ is obtained such that:

Algorithm 6: Self-organizing map for multi-view text clustering

```
Input: multi-view documents $\mathbf{X}^v$, number of SOM neurons $l$, learning rate $\alpha_0$, radius $\sigma_0$
Output: SOM prototypes of each view $\mathbf{W}^v$
$t \leftarrow 1$
repeat
    for $v = 1$ to $m$ do
        Initialize random SOM prototypes $\mathbf{W}^v$
        for $i = 1$ to $n$ do
            determine Best Matching Unit $BMU$ for document $\mathbf{x}_i$
            // Update SOM prototypes
            for $j = 1$ to $l$ do
                $\mathbf{w}_j^v \leftarrow \mathbf{w}_j^v + h \times \alpha \times \left(\mathbf{x}_i^v - \mathbf{w}_j^v\right)$
            end for
        end for
    end for
    // Update radius of the neighborhood
    $\sigma \leftarrow \sigma_0 \exp\left(\frac{t}{tmax}\right)$
    // Update the learning rate
    $\alpha \leftarrow \alpha_0 \exp\left(\frac{t}{tmax}\right)$
    $t \leftarrow t + 1$
until $t > tmax$
```

$$\mathbf{L}^v = \mathbf{D}^{v-1/2}\mathbf{A}^v\mathbf{D}^{v-1/2} \tag{8.10}$$

where $\mathbf{D}^v$ is the diagonal matrix such that $\mathbf{D}_{ii}^v = \sum_j \mathbf{A}_{ij}^v$. The k largest eigenvectors of $\mathbf{L}$ hold the discrimination information for clustering. Thus, the eigenvectors are exchanged across views to propagate the per-view clustering information, such that the largest k eigenvectors form the matrix $\mathbf{U}^v$. Precisely, the co-trained spectral clustering uses the eigenvectors of one view to modify the adjacency matrix of the other view and consequently the graph structure, such that each column $\mathbf{a}_i$ of $\mathbf{A}$ represents the similarity of the data point i with all point in the graph. The algorithm projects the column vectors of one view in the direction of the k eigenvectors of the other view, then back projects them to the original space to obtain the modified graph. To obtain the update adjacency matrix $\mathbf{S}^v$, a symmetrization step is performed such that:

$$\mathbf{S}^v = sym(\mathbf{U}^{\bar{v}}\mathbf{U}^{\bar{v}^T} A^v) \tag{8.11}$$

where $sym(\mathbf{S}) = (\mathbf{S} + \mathbf{S}^T)/2$. The new graph Laplacian $\mathbf{L}^v$ are obtained from $\mathbf{S}^v$, from which the k eigenvectors and $\mathbf{U}v$ are deduced. The algorithm performs these steps for a defined number of iteration. The final clustering is given by the k-means algorithm performed on matrix $\mathbf{V}$, the column-wise concatenation of $\mathbf{U}^v$. The main steps of co-training multi-view spectral clustering are given in Algorithm 7.

Algorithm 7: Co-training based multi-view spectral clustering

```
Input: multi-view documents $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^m\}$, number of clusters $k$
Output: Final clustering assignment $\pi$
// Initialization
for $v = 1 : 2$ do
    Compute adjacency matrix $\mathbf{A^v}$
    Compute normalized Laplacian matrix using Eq. 8.10
    Initialize $\mathbf{U}^{v0}$ by $\mathbf{U}^{v0} = \text{argmax}\, tr(\mathbf{U}^{vT}\mathbf{A}^v\mathbf{U}^v)$   s.t $\mathbf{U}^{vT}\mathbf{U}^v = \mathbf{I}$
end for
for $t = 1$ to $tmax$ do
    Compute $\mathbf{S^1}$ and $\mathbf{S^2}$ using 8.11
    Compute the Laplacian matrices $\mathbf{L}^v$ from $\mathbf{S}^v$
    Build $\mathbf{U}^v$ from the $k$ largest eigenvectors of $\mathbf{L}^v$
end for
Normalize the rows of $\mathbf{U}^1$ and $\mathbf{U}^2$
Build $\mathbf{V}$ as the column-wise concatenation of $\mathbf{U}^1$ and $\mathbf{U}^2$
Run $k$-means on $\mathbf{V}$ to obtain the clustering assignments
```

Lin et al. [20] proposed Multi-view Proximity Learning for Clustering (MVPL), a method that learns the proximity matrix based on data representative and spectral clustering. Given a set of multi-view data $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^m\} \in \mathbb{R}^{d^v \times n}$, a data representative matrix $\mathbf{U}^v \in \mathbb{R}^{d^v \times n}$ is associated with each view to exploit the relations between objects within the same view. The new data representative considers the proximity between each pair of data points. Therefore the learned similarity matrix is affected by these representatives, and inversely. On the other hand, MVPL considers the spectral embedding of data to integrate the different views and thus consider the inter-view relations into the similarity matrix. The goal of MVPL is to minimize the following objective function:

$$\begin{aligned}
&\min_{\{\mathbf{U}^v\},\{\mathbf{S}^v\},\mathbf{F}} \frac{1}{n}\sum_{v=1}^{m}\left(\sum_{i=1}^{n}\|\mathbf{x}_i^v - \mathbf{u}_i^v\|_2^2 + \frac{\alpha}{n^2}\left(\sum_{i,j=1}^{n}\|\mathbf{u}_i^v - \mathbf{u}_j^v\|_2^2 s_{ij} + \beta\|\mathbf{S}\|_F^2\right)\right)\\
&+\gamma\frac{1}{2n^2}\sum_{v=1}^{m}\sum_{i,j=1}^{n} s_{ij}^v\|\mathbf{f}_i - \mathbf{f}_j\|_2^2 \qquad (8.12)\\
&s.t \sum_{j=1}^{n} s_{ij} = 1, s_{ij}^v \geq 0, \forall i, j, \mathbf{F}\mathbf{F}^T = \mathbf{I}
\end{aligned}$$

where the first term considers the impact of the data representatives, while the second term models the relation between the spectral embedding matrix $\mathbf{F}$ and the similarity matrix $\mathbf{S}^v$, γ is a trade-off parameter that balances the two terms, α controls the distance between the original data features and their representatives, and β controls the sparsity of $\mathbf{S}$. Algorithm 8 describes the main steps of MVPL.

Algorithm 8: Multi-view proximity learning

```
Input: multi-view documents $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^m\}$, number of clusters $k$, parameters $\alpha, \gamma$
Output: Proximity matrices $\{\mathbf{S}^1, \mathbf{S}^2, \ldots, \mathbf{S}^m\}$
Initialize representative matrices $\mathbf{U}^v$ as $\mathbf{X}^v$
Initialize proximity matrices $\mathbf{S}^v$ by
Determine sparsity parameter $\beta$
Initialize $\mathbf{F}$ by solving $\min_{\mathbf{F}\mathbf{F}^T=\mathbf{I}} Tr(\mathbf{F}\mathbf{L}_{\mathbf{SF}}^T)$
repeat
    for $v = 1$ to $m$ do
        Update $\mathbf{U}^v$ by solving $\mathbf{U}^v(\mathbf{I} + \frac{2\alpha}{n}\mathbf{L}_{\mathbf{S}}) = \mathbf{X}^v$
        Update $\mathbf{S}^v$ by solving $\min_{\mathbf{s}_i^v} \|\mathbf{s}_i^v + \frac{\mathbf{d}_i^v}{2\beta^v}\|_2^2$
    end for
    Update $\mathbf{F}$
until converged or max iteration is reached;
```

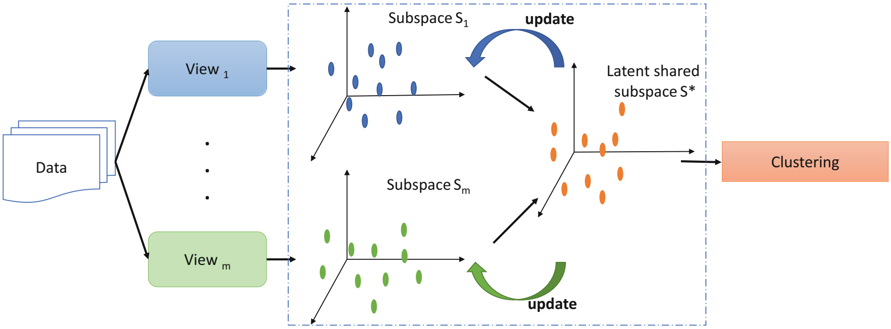


Fig. 8.3 General process of multi-view subspace clustering

8.2.3 Subspace Clustering Based Methods

The third category of multi-view clustering is based on subspace learning. Recently, more and more studies have exploited subspace clustering to extract distinct clustering features. Multi-view subspace clustering assumes that the data samples from different views share the same subspace [33]. Figure 8.3 illustrates the process of learning a shared subspace from multi-view data. The performance of subspace clustering relies on the latent representation matrix obtained from the different multi-view subspaces. Several methods have been proposed in order to identify the common subspace, we distinguish two main subcategories: NMF based methods and latent representation based methods.

8.2.3.1 Muti-View Subspace Clustering Based on Nonnegative Matrix Factorization

Liu et al. [22] proposed MultiNMF, a multi-view clustering via joint nonnegative matrix factorization. The algorithm enforces each view's indicator matrix towards a common consensus. Given multi-view data $\mathbf{X}^v \in \mathbb{R}_+^{d\times n}$, its matrix factorization is:

$$\mathbf{X}^v \approx \mathbf{U}^v \mathbf{V}^{vT} \tag{8.13}$$

where $\mathbf{V}^v \in \mathbb{R}_+^{n\times k}$ and $\mathbf{U}^v \in \mathbb{R}_+^{d\times k}$ represent the indicator matrix and the basis matrices of view v, respectively. MultiNMF adopts a normalization constraint so that all indicator matrices are comparable and significant for clustering. The problem can be defined as a joint minimization of the following objective function:

$$\begin{aligned}&\sum_{v}^{m} \|\mathbf{X}^v - \mathbf{U}^v\mathbf{V}^{vT}\|_F^2 + \sum_{v}^{m} \lambda_v \|\mathbf{V}^v\mathbf{Q}^v - \mathbf{V}^*\|_F^2 \\ &s.t\ v \in \{1,\ldots,m\}, \mathbf{U}^v \geqslant 0, \mathbf{V}^v \geqslant 0, \mathbf{V}^* \geqslant 0\end{aligned} \tag{8.14}$$

where $\mathbf{V}^*$ is the consensus matrix, and $\mathbf{Q}^v$ is a diagonal matrix such that:

$$\mathbf{Q}^v = Diag\left(\sum_{j=1}^{d}\mathbf{U}_{j1}^v, \sum_{j=1}^{d}\mathbf{U}_{j2}^v, \ldots, \sum_{j=1}^{d}\mathbf{U}_{jk}^v\right) \tag{8.15}$$

Finally, the clustering assignment of data point i is computed as $\text{argmax}_k\, \mathbf{V}_{ik}^*$. The main steps of MultiNMF are given in Algorithm 9.

Zhang et al. [34] proposed a constrained NMF based clustering (CMVNMF) that uses an inter-view must-link (*ML*) and cannot-link (*CL*) constraints in order to minimize the disagreement between each pair of views. To accomplish the clustering task, the following objective function is minimized:

$$\|\mathbf{X}^v - \mathbf{U}^v\mathbf{V}^{vT}\| + \beta \sum_{v,v'\in[1,m]} \Delta_{v,v'} \quad s.t\ \mathbf{U}^v \geq 0, \mathbf{V}^v \geq 0 \tag{8.16}$$

where β is a regularization parameter, and Δ measures the disagreement between v and v' such that:

$$\Delta_{v,v'} = \sum_{(\mathbf{x}_i^v,\mathbf{x}_j^{v'})\in ML^{v,v'}} (\mathbf{v}_i - \mathbf{v}'_j) + 2 \sum_{(\mathbf{x}_i^v,\mathbf{x}_j^{v'})\in CL^{v,v'}} \mathbf{v}_i\mathbf{v}'_j \tag{8.17}$$

The must-link and cannot-link constraints are defined by matrices $\mathbf{M}^{vv'}$ and $\mathbf{C}^{vv'}$, respectively, such that :

Algorithm 9: Multi-view NMF

```
Input: multi-view documents $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^m\}$, number of clusters $k$, parameters
    $\{\lambda_1, \lambda_2, \ldots, \lambda_m\}$
Output: Basis matrices $\{\mathbf{U}^1, \mathbf{U}^2, \ldots, \mathbf{U}^m\}$, Consensus Matrix $\mathbf{V}^*$
Normalize each view $\mathbf{X}^v$ such that $\|\mathbf{X}^v\|_1 = 1$
Initialize $\mathbf{U}^v$ and $\mathbf{V}^*$
repeat
    for $v = 1$ to $m$ do
        repeat
            Fixing $\mathbf{V}^*$ and $\mathbf{V}^v$, update $\mathbf{U}^v$ by
            $\mathbf{U}^v = \mathbf{U}^v \odot \frac{(\mathbf{X}^v\mathbf{V}^v) + \lambda_v \sum^n \mathbf{V}^v\mathbf{V}^*}{(\mathbf{U}^v\mathbf{V}^{vT}\mathbf{V}^v) + \lambda_v \sum^d \mathbf{U}^v \sum^n \mathbf{V}^{v2}}$
            // $\odot$ is the element-wise multiplication
            Normalize $\mathbf{U}^v$ by $\mathbf{U}^v = \mathbf{U}^v\mathbf{Q}^v{-1}$
            Normalize $\mathbf{V}^v$ by $\mathbf{V}^v = \mathbf{V}^v\mathbf{Q}^v$
            Fixing $\mathbf{V}^*$ and $\mathbf{U}^v$, update $\mathbf{V}^v$ by $\mathbf{V}^v = \mathbf{V}^v \odot \frac{(\mathbf{X}^{vT}\mathbf{U}^v) + \lambda_v\mathbf{V}^*}{(\mathbf{V}^v\mathbf{U}^{vT}\mathbf{U}^v) + \lambda_v\mathbf{V}^v}$
        until convergence of $\|\mathbf{X}^v - \mathbf{U}^v\mathbf{V}^{vT}\|_F^2 + \lambda_v\|\mathbf{V}^v\mathbf{Q}^v - \mathbf{V}^*\|_F^2$;
    end for
    Fixing $\mathbf{U}^v$ and $\mathbf{V}^v$, update $\mathbf{V}^*$ by $\mathbf{V}^* = \frac{\sum_v^m \lambda_v\mathbf{V}^v\mathbf{Q}^v}{\sum_v^m \lambda_v}$
until convergence of 8.14;
```

$$\mathbf{M}_{ij}^{vv'} \begin{cases} 1, & (x_i^v, x_j^{v'}) \in ML^{v,v'} \\ 0, & \text{otherwise} \end{cases}$$

$$\mathbf{C}_{ij}^{vv'} \begin{cases} 1, & (x_i^v, x_j^{v'}) \in CL^{v,v'} \\ 0, & \text{otherwise} \end{cases}$$

The distance between a pair of data points in the same cluster from different views is minimized through the must-link constraints, while the cannot-link constraints aim to maximize the distance of data points belonging to different views and different clusters. The main steps of CMVNMF are given in Algorithm 10.

8.2.3.2 Multi-View Subspace Clustering Based on Shared Latent Representation

Zhang et al. [33] proposed Latent Multi-view Subspace Clustering (LMSC), which is based on the assumption that multi-view data share a latent subspace representation. LMSC learns a common representation from the different views based on subspace clustering. First, the original multi-view data $\mathbf{X}^v$ is reconstructed based on projection models $\mathbf{P}^v$ and achieve a common latent representation $\mathbf{H}$ such that:

Algorithm 10: Constrained multi-view NMF

```
Input: multi-view documents $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^m\}$, the number of cluster $k$, the must-link
       constraints matrix $\mathbf{M}^{vv'}$, the cannot-link constraints matrix $\mathbf{C}^{vv'}$
Output: the clustering assignment $\pi$
Normalize $\mathbf{X}^v$
For each pair of views $(v, v')$, compute diagonal matrices $\mathbf{D}^{vv'}$ by $\mathbf{D}_{ii}^{vv'} = \sum_j^n \mathbf{M}_{ij}^{vv'}$ with
$i = 1, \ldots, n$
Initialize $\mathbf{U}^v$ and $\mathbf{V}^v$
repeat
    for $v = 1$ to $m$ do
        Fix $\mathbf{V}^v$, and update $\mathbf{U}^v$ by $\mathbf{U}^v = \mathbf{U}^v \odot \frac{(\mathbf{X}^v \mathbf{V}^v)}{(\mathbf{U}^v {\mathbf{V}^v}^T \mathbf{V}^v)}$
        Fix $\mathbf{U}^v$, and update $\mathbf{V}^v$ by
        $\mathbf{V}^v = \mathbf{V}^v \odot \frac{({\mathbf{X}^v}^T \mathbf{U}^v) + \beta \sum_{v=1, v \neq v'}^m (\mathbf{M}^{v,v'} \mathbf{V}^{v'}}{(\mathbf{V}^v {\mathbf{U}^v}^T \mathbf{U}^v) + \beta \sum_{v=1, v \neq v'}^m (\mathbf{D}^{vv'} \mathbf{V}^v + \mathbf{C}^{vv'} \mathbf{V}^{v'})}$
    end for
until convergence of 8.16;
```

$$\mathbf{x}_i^v = \mathbf{P}^v \mathbf{h}_i + \mathbf{e}_i^v \tag{8.18}$$

where $\mathbf{e}_i^v$ denotes the reconstruction error. Then, the latent representation is integrated into subspace clustering, such that the clustering problem is defined as:

$$\min_{\mathbf{Z}} L_r(\mathbf{H}, \mathbf{HZ}) + \alpha \Omega(\mathbf{Z}) \tag{8.19}$$

where $\mathbf{Z}$ is the subspace representation matrix, $L_r()$ is the loss function of the subspace reconstruction, $\Omega()$ corresponds to the regularization term, α balances the regularization. By introducing the parameters λ_1 and λ_2, the overall objective function of LMSC becomes as follows:

$$\begin{aligned} &\min_{\mathbf{P}, \mathbf{H}, \mathbf{Z}, \mathbf{E}_h, \mathbf{E}_r} \|\mathbf{E}_h\|_{2,1} + \lambda_1 \|\mathbf{E}_r\|_{2,1} + \lambda_2 \|\mathbf{Z}\|_* \\ &s.t \quad \mathbf{X} = \mathbf{PH} + \mathbf{E}_h, \mathbf{H} = \mathbf{HZ} + \mathbf{E}_r, \text{and } \mathbf{PP}^T = 1 \end{aligned} \tag{8.20}$$

The $\ell_{2,1}$ norm ensures robustness in the presence of noise, while the nuclear norm ℓ_* captures the underlying clustering structure. To solve Eq. 8.20, the error matrices $\mathbf{E}_h$ and $\mathbf{E}_r$ are vertically concatenated, and the Augmented Lagrangian Multiplier with Alternating Direction Minimization (ALM-ADM) strategy proposed in [21] is adopted. The main steps of LMSC are given in Algorithm 11.

Brbic et al. [6] proposed a multi-view low-rank and sparse subspace clustering (MLRSSC), with two regularization scheme: pairwise and centroid based. The first establishes a pairwise agreement across views, whereas the second coerces the representations towards a common centroid, as first introduced by Kumar et al. [17]. Both methods are based on constructing a low-rank and sparse affinity matrix from

```
Algorithm 11: Latent multi-view subspace clustering
Input: multi-view documents $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^m\}$, number of clusters $k$, parameter $\lambda$
Output: $\mathbf{Z}, \mathbf{H}, \mathbf{P}, \mathbf{E}$
Initialize $\mathbf{P} = 0, \mathbf{E} = 0, \mathbf{Z} = 0, \mathbf{Y_1} = 0, \mathbf{Y_2} = 0, \mathbf{Y_3} = 0, \mu = 10^{-6}, \rho = 1.1, \epsilon = 10^{-4}$
Initialize randomly $\mathbf{H}$
while not converged do
    Update $\mathbf{P}$ by $\mathbf{P} = \text{argmin}\ \frac{\mu}{2} \|(\mathbf{X} + \frac{1}{\mu}\mathbf{Y}_1 - \mathbf{E}_h) - \mathbf{H}^T\mathbf{P}^T\|$
    Update $\mathbf{H}$ by
        $\mathbf{AH} + \mathbf{HB} = \mathbf{C}$
        with $\mathbf{A} = \mu\mathbf{P}^T\mathbf{P}$, $\mathbf{B} = \mu(\mathbf{ZZ^T} - \mathbf{Z} - \mathbf{Z^T} + \mathbf{I})$
        $\mathbf{C} = \mathbf{P}^T\mathbf{Y}_1 + \mathbf{Y}_2(\mathbf{Z}^T - \mathbf{I})$
        $+\mu(\mathbf{P}^T\mathbf{X} + \mathbf{E}_r^T - \mathbf{P}^T\mathbf{E}_h - \mathbf{E}_r\mathbf{Z}^T)$
    Update $\mathbf{Z}$ by $\mathbf{Z} = (\mathbf{H}^T\mathbf{H} + \mathbf{I})^{-1}[(\mathbf{J} + \mathbf{H}^T\mathbf{H} - \mathbf{H}^T\mathbf{E}_r) + (\mathbf{Y}_3 + \mathbf{H}^T\mathbf{Y}_2)/\mu]$
    Update $\mathbf{E}$ by $\mathbf{E} = \text{argmin}_E\ \frac{1}{\mu}\|\mathbf{E}\|_{2,1} + \frac{1}{2}\|\mathbf{E} - \mathbf{G}\|_F^2$
    Update $\mathbf{J}$ by $\mathbf{J} = \frac{\lambda}{\mu}\|J\|_* + \frac{1}{2}\|\mathbf{J} - (\mathbf{Z} - \mathbf{Y}_3/\mu)\|_F^2$
    Update $\mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}_3$ by $\begin{cases} \mathbf{Y}_1 = \mathbf{Y}_1 + \mu(\mathbf{X} - \mathbf{PH} - \mathbf{E}_h \\ \mathbf{Y}_2 = \mathbf{Y}_2 + \mu(\mathbf{H} - \mathbf{HZ} - \mathbf{E}_r \\ \mathbf{Y}_3 = \mathbf{Y}_3 + \mu(\mathbf{J} - \mathbf{Z}) \end{cases}$
    Update $\mu$ by $\mu = \min(\rho\mu; \max_\mu)$
    Check convergence criteria $\|\mathbf{X} - \mathbf{PH} - \mathbf{E}_h\|_\infty < \epsilon$, $\|\mathbf{H} - \mathbf{HZ} - \mathbf{E}_r\|_\infty < \epsilon$ and
    $\|\mathbf{J} - \mathbf{Z}\|_\infty < \epsilon$
end
```

multi-view data. Given a set of multi-view data $\mathbf{X}^v$, MLRSSC aims to find a joint representation matrix $\mathbf{C}$ that presents an agreement across views by minimizing the following objective function:

$$\min_{\mathbf{C}^{(1)},\mathbf{C}^{(2)},\ldots,\mathbf{C}^{(m)}} \sum_{v=1}^{m}(\beta_1 \left\|\mathbf{C}^v\right\|_* + \beta_2 \left\|\mathbf{C}^v\right\|_1) + \sum_{1\leq v,w\leq m, v\neq w} \lambda^v \left\|\mathbf{C}^v - \mathbf{C}^w\right\|_F^2 \tag{8.21}$$
$$\text{s.t.} \quad \mathbf{X}^v = \mathbf{X}^v\mathbf{C}^v, \quad diag(\mathbf{C}^v) = 0, \quad v = 1, \ldots, m,$$

where $\mathbf{Z}^v$ is the representation matrix of view v, β_1 and β_2 are the balancing parameters of low-rank and sparsity constraint, λ^v is the consensus parameter. In case where all views are considered equally important, the same λ^v is used. The last term maximizes the pairwise similarity across views. To solve the problem in Eq. 8.21, the Alternating Direction Method of Multipliers (ADMM) strategy is used [5]. Algorithms 12 and 13 summarize the steps of pairwise MLRSSC and centroid-based MLRSSC, respectively.

Algorithm 12: Pairwise MLRSSC

```
Input: Multi-view documents $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \dots, \mathbf{X}^m\}, k, \beta_1, \beta_2, \{\lambda^v\}_{v=1}^m, \{\mu_i\}_{v=1}^4, \mu^{max}, \rho$
Output: $k$ clusters assignments
Initialize: $\{\mathbf{C}_i^v = 0\}_{i=1}^3, \mathbf{A}^v = 0, \{\mathbf{\Lambda}_i^v = 0\}_{i=1}^4, i = 1, \dots, m$
repeat
    for $v = 1$ to $m$ do
        Update $\mathbf{A}^v$ by solving
            $$\mathbf{A}^v = [\mu_1 \mathbf{X}^{v^T}\mathbf{X}^v + (\mu_2 + \mu_3 + \mu_4)\mathbf{I}]^{-1} \times (\mu_1 \mathbf{X}^{v^T}\mathbf{X}^v + \mu_2 \mathbf{C}_2^v + \mu_3 \mathbf{C}_1^v + \mu_4 \mathbf{C}_3^v + \mathbf{W}^{v^T}\mathbf{\Lambda}_1^v + \mathbf{\Lambda}_2^v + \mathbf{\Lambda}_3^v + \mathbf{\Lambda}_4^v)$$
        Update $\mathbf{C}_1^v$ by solving $\min_{\mathbf{C}_1^v} \beta_1 \left\|\mathbf{C}_1^v\right\|_* + \frac{\mu_3}{2}\left\|\mathbf{A}^v - \mathbf{C}_1^v + \frac{\mathbf{\Lambda}_3^v}{\mu_3}\right\|_F^2$
        Update $\mathbf{C}_2^v$ by solving $\min_{\mathbf{C}_2^v} \beta_2 \left\|\mathbf{C}_2^v\right\|_1 + \frac{\mu_2}{2}\left\|\mathbf{A}^v - \mathbf{C}_2^v + \frac{\mathbf{\Lambda}_2^v}{\mu_2}\right\|_F^2$
        Update $\mathbf{C}_3^v$ by solving
        $\min_{\mathbf{C}_3^v} \lambda^v \sum_{1 \le w \le m, v \ne w} \left\|\mathbf{C}_3^v - \mathbf{C}^w\right\|_F^2 + \frac{\mu_4}{2}\left\|\mathbf{A}^v - \mathbf{C}_3^v\right\|_F^2 + tr\left[\mathbf{\Lambda}_4^{v^T}\left(\mathbf{A}^v - \mathbf{C}_3^v\right)\right]$
        Update $\mathbf{\Lambda}_1^v, \mathbf{\Lambda}_2^v, \mathbf{\Lambda}_3^v, \mathbf{\Lambda}_4^v$
    end for
    Update $\mu_i = \min(\rho\mu_i, \mu^{max}), i = 1, \dots, 4$
    Check convergence conditions: $\|\mathbf{A}^v - \mathbf{C}_1^v\|_\infty \le \epsilon$, $\|\mathbf{A}^v - \mathbf{C}_2^v\|_\infty \le \epsilon$,
    $\|\mathbf{A}^v - \mathbf{C}_3^v\|_\infty \le \epsilon$, and $\|\mathbf{A}_t^v - \mathbf{A}_{t-1}^v\| \le \epsilon$
until Convergence or reaching the maximum number of iterations
Combine $\mathbf{C}_1^v, \mathbf{C}_2^v, \mathbf{C}_3^v$ by considering the element-wise average
Perform spectral clustering on the affinity matrix $\mathbf{S} = |\mathbf{C}_{avg}| + |\mathbf{C}_{avg}|^T$
```

Algorithm 13: Centroid-based MLRSSC

```
Input: Multi-view documents $\mathbf{X}^v, k, \beta_1, \beta_2, \{\lambda^v\}_{v=1}^m, \{\mu_i\}_{v=1}^4, \mu^{max}, \rho$
Output: k clusters assignments
Initialize: $\{\mathbf{C}_i^v = 0\}_{i=1}^3, \mathbf{C}^* = 0, \mathbf{A}^v = 0, \{\mathbf{\Lambda}_i^v = 0\}_{i=1}^4, i = 1, \dots, m$
repeat
    for $v = 1$ to $m$ do
        Update $\mathbf{A}^v, \mathbf{C}_1^v, \mathbf{C}_2^v, \mathbf{C}_3^v$ as in Algorithm 12
        Update $\mathbf{\Lambda}_1^v, \mathbf{\Lambda}_2^v, \mathbf{\Lambda}_3^v, \mathbf{\Lambda}_4^v$
    end for
    Update $\mu_i = \min(\rho\mu_i, \mu^{max}), i = 1, \dots, 4$
    Update $\mathbf{C}^* = \frac{\sum_v \lambda^v \mathbf{C^v}}{\sum_v \lambda^v}$
    Check convergence conditions: $\|\mathbf{A}^v - \mathbf{C}_1^v\|_\infty \le \epsilon$, $\|\mathbf{A}^v - \mathbf{C}_2^v\|_\infty \le \epsilon$,
    $\|\mathbf{A}^v - \mathbf{C}_3^v\|_\infty \le \epsilon$, and $\|\mathbf{A}_t^v - \mathbf{A}_{t-1}^v\| \le \epsilon$
until Convergence or reaching the maximum number of iterations
Perform spectral clustering on the affinity matrix $\mathbf{S} = |\mathbf{C}^*| + |\mathbf{C}^*|^T$
```

8.2.4 Summary of Multi-View Methods for Text Clustering

Compared to single-view data, multi-view data presents multiple advantages given its ability to describe objects from different aspects and thus give a more comprehensive representation of data. However, the manipulation and exploitation of multi-view data require further advanced algorithms in order to mine the complementarity between views and discover knowledge that is otherwise hidden in a single-view framework. Multi-view data is furthermore challenging in the case of unlabeled data given that no prior knowledge is available. The existing multi-view clustering algorithms, as the ones presented in this chapter, have shown good performance in dealing with different points of multi-view data such as finding a consensus across views, integrating the information provided by each view, discovering hidden patterns, etc.

Multiple methods for multi-view text clustering rely on a single representation model, usually the TF-IDF [16]. Although this model is capable of capture the syntactic properties of text, it is, however, unable to give an insight on semantic concepts or topically related features of text data. To this end, other methods exploited different representation models such as TF-ICF in [13] or topic models and word embeddings [9, 10]. Table 8.1 summarizes the characteristics of multi-view clustering methods.

8.3 Experiments

We evaluate in this section the performance of multi-view clustering methods on text data. We select methods from each category: MEMTC [9], MVEM [13], MVKM [3], MVSOM [10], LMSC [33], pairwise MLRSSC and centroid-based MLRSSC [6]. We also compare these methods to other baseline such as PCA and basic spectral clustering applied to concatenated views.

8.3.1 Data Sets Description

The experiments are carried on four commonly used data sets for multi-view text clustering. The *Reuters* data set is a collection of 2189 documents belonging to 8 classes. The *20 Newsgroups* consists of 2828 news articles distributed on 20 classes. The *WebKB* data set is a collection of 4168 web pages collected from computer science departments, belonging to 4 classes (student, faculty, project, course). The *BBC Sport* consists of 737 documents from the BBC Sport website corresponding to sports news articles belonging to 5 areas: football, rugby, tennis, athletics, and cricket. Before applying the clustering algorithms, a preprocessing step is performed on the data sets including stop words removal. Stop words removal consists in

Table 8.1 Summarization of multi-view clustering methods

Approach	Method	# views	Text representation	Pros	Cons
Late integration	MVEM [13]	>2	TF-IDF, TF-ICF	– Aims towards a consensus – Good empirical performance – Complementarity of views based on their clustering results	– Quality of clustering depends on the consensus technique – Does not explore the inter-relation across views – Computational cost
	MEMTC [9]		TF-IDF, LDA, Skip-gram		
	MVEC [26]		TF		
	LFALM [7]		–		
	MVCE [31]		–		
Co-training	MVKM [3]	= 2	TF-IDF	– Maximizes mutual agreement across views – Exchanges information, i.e., clustering assignment	– Sensitive to noise – Becomes challenging when number of views increases
	Co-trained spectral [16]	=2	TF-IDF		
	MVPL [20]	>2	TF		
	Co-Kmeans [2]		>2		
	MVSOM [10]		TF-IDF, LDA, Skip-gram		
Subspace based	MultiNMF [22]	>2	TF	– Explores the specificities of each view – Suitable for high-dimensional data	– Depends on the optimization of the latent subspace – Parameter tuning
	CMVNMF [34]				
	LMSC [33]		TF-IDF		
	MLRSSC [6]				

Table 8.2 Data sets description

Data set	Documents	Features	k
Reuters	2189	2577	8
BBC Sports	737	3853	5
20 newsgroup	2263	6943	20
webKB	2084	3857	4

eliminating common words that appear frequently and offer no additional semantic value. Table 8.2 summarizes the properties of all data sets.

8.3.2 Evaluation Measures

To measure the quality of the clustering and compare it with existing methods, three evaluation measures are utilized: the F-measure [18], the Normalized Mutual Information (NMI) [37], and Purity [23]. Given a set of clusters $C = \{c_1, c_2, \ldots, c_k\}$ and the gold standard classes $G = \{g_1, g_2, \ldots, g_j\}$:
F-measure is a trade-off between $Precision$ and $Recall$ such that:

$$F - measure(c_k, g_j) = 2 * \frac{Precision(c_k, g_j) \times Recall(c_k, g_j)}{Precision(c_k, g_j) + Recall(c_k, g_j)} \tag{8.22}$$

$$Precision(c_k, g_j) = \frac{|c_k \cap g_j|}{|c_k|} \tag{8.23}$$

$$Recall(c_k, g_j) = \frac{|c_k \cap g_j|}{|g_j|} \tag{8.24}$$

Normalized Mutual Information (NMI) measures the quality of clustering with regards to the number of clusters and their sizes. NMI is defined as:

$$NMI(C, G) = \frac{I(C, G)}{[E(C) + E(G]]/2} \tag{8.25}$$

where I is the mutual information and $E(C)$ is entropy.

$$I(C, G) = \sum_k \sum_j \frac{|c_k \cap g_j|}{N} \log \frac{N|c_k \cap g_j|}{|c_k||g_j|} \tag{8.26}$$

$$E(C) = -\sum_k \frac{|s_k|}{N} \log \frac{|s_k|}{N} \tag{8.27}$$

Purity: measures the number of correctly assigned documents, where each cluster is assigned to the dominant class in that cluster. The larger the number of clusters is, the higher the Purity is. Unlike NMI, Purity cannot trade-off the quality of the clustering against the number of clusters

$$Purity(C, G) = \frac{1}{N} \sum_{k} \max_{j} |c_k \cap g_j| \tag{8.28}$$

For all measures, the values range from 0 to 1, such that values closer to 0 represent poor quality

8.3.3 Experimental Results

Table 8.3 reports the performance of the different methods. Given the results, we can observe that most multi-view methods provided better clustering in comparison to concatenated views. This shows that concatenating views can result in losing the individual properties of views and affect the overall clustering. Another noticeable observation is that all methods have given their best results on the smallest data set, the BBC Sport, while the overall performance is affected on the largest data set, 20 newsgroup. We can conclude that the size and the dimension of the data set can jeopardize the performance; this may be due to noise and redundant information. Although all methods have yielded close results, we can notice that multi-view subspace clustering methods achieve relatively better results on almost all data sets, which can indicate that these methods are capable of learning a common latent representation from all views. On the other hand, both ensemble methods have performed similarly, however, MEMTC had better performance, which indicates that including other representation scheme can improve the final clustering.

Overall, late integration based method has shown good empirical performance given that the individual clustering provided by each view can compensate the clustering inaccuracy of another view. However, such methods can be computationally expensive since the clustering is performed of the number of views and the integration phase is independent from the clustering phase and can add on to the computational cost.

Co-training based on a simultaneous optimization of one unified objective function to achieve one clustering result from different views [2, 3]. However, having a unified objective function does not allow to learn from each view independently, which can result in losing the knowledge held in different views and can later be integrated to improve the overall clustering. Furthermore, co-training based method becomes intractable when the number of views is over three.

Another issue consists of integrating multiple views while maintaining their diversity. Precisely, in the clustering process reaching a consensus, or co-training based clustering can result in losing the specificity of each view. To this end subspace clustering based algorithm can present a solution [6]. However, the challenge

Table 8.3 Comparison of clustering results with multi-view methods

Data set	Method	F-score	NMI	Purity
Reuters	PCA	0.442	0.335	0.422
	Concat SC	0.476	0.227	0.436
	MVKM	0.648	0.428	0.743
	MEMTC	0.814	0.604	0.458
	MVEM	0.490	0.337	0.493
	LMSC	0.705	0.508	0.593
	Centroid MLRSSC	0.629	0.430	0.534
	Pairwise MLRSSC	0.539	0.339	0.443
	MVSOM	0.709	0.464	0.606
BBC Sport	PCA	0.613	0.388	0.606
	Concat SC	0.500	0.206	0.405
	MVKM	0.693	0.564	0.633
	MEMTC	0.797	0.730	0.771
	MVEM	0.819	0.717	0.753
	LMSC	0.804	0.711	0.767
	Centroid MLRSSC	0.838	0.708	0.833
	Pairwise MLRSSC	0.873	0.716	0.871
	MVSOM	0.821	0.728	0.744
20 newsgroup	PCA	0.356	0.302	0.290
	Concat SC	0.440	0.439	0.392
	MVKM	0.432	0.380	0.373
	MEMTC	0.511	0.534	0.458
	MVEM	0.380	0.305	0.300
	LMSC	0.539	0.470	0.525
	Centroid MLRSSC	0.540	0.531	0.494
	Pairwise MLRSSC	0.519	0.516	0.482
	MVSOM	0.445	0.446	0.382
webKB	PCA	0.578	0.304	0.558
	Concat SC	0.277	0.172	0.558
	MVKM	0.564	0.321	0.460
	MEMTC	0.596	0.406	0.465
	MVEM	0.542	0.268	0.448
	LMSC	0.394	0.160	0.294
	Centroid MLRSSC	0.622	0.418	0.561
	Pairwise MLRSSC	0.632	0.405	0.581
	MVSOM	0.618	0.255	0.597

remains in finding a shared subspace while incorporating the diversity aspect. To summarize, this experimental results help drawing the following conclusions:

- Large data set and high-dimensional data affects the performance of multi-view methods. Therefore, considering a dimensionality reduction methods can help avoid this issue.
- Taking advantage of different representation schemes can improve the clustering performance of multi-view methods.
- Subspace based methods have good performance, yet these methods include multiple parameters and the optimization scheme is not evident to achieve.

8.4 Conclusion

We have presented in this chapter a categorization of existing multi-view clustering methods based on the fusion style of multi-view data. Three main integration scheme can be distinguished: late integration, co-training based methods, and subspace based methods. For each category, we have detailed a number of multi-view clustering algorithms, and the means of managing text data. Lastly, we have discussed the advantages and the limits of these methods and raised the following issues: the representation of multi-view text data relies on terms frequencies only, the intra-view properties of each view can be further leveraged to improve the clustering results, incorporating the specificity of each view in the clustering process can provide a better understanding of data. Multiple recent research studies focus on incomplete views with missing values. Some other works rely on incorporating deep learning into multi-view clustering to further discover hidden patterns shared among views.

References

1. M. Amini, N. Usunier, C. Goutte, Learning from multiple partially observed views-an application to multilingual text categorization, in *Advances in Neural Information Processing Systems* (2009), pp. 28–36
2. S. Bettoumi, C. Jlassi, N. Arous, Collaborative multi-view k-means clustering. Soft Comput. **23**(3), 937–945 (2019)
3. S. Bickel, T. Scheffer, Multi-view clustering, in *ICDM*, vol. 4 (2004), pp. 19–26
4. A. Blum, T. Mitchell, Combining labeled and unlabeled data with co-training, in *Proceedings of the Eleventh Annual Conference on Computational Learning Theory* (ACM, New York, 1998), pp. 92–100
5. S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, et al., Distributed optimization and statistical learning via the alternating direction method of multipliers. Found. Trends® Mach. Learn. **3**(1), 1–122 (2011)
6. M. Brbić, I. Kopriva, Multi-view low-rank sparse subspace clustering. Pattern Recogn. **73**, 247–258 (2018)
7. E. Bruno, S. Marchand-Maillet, Multiview clustering: a late fusion approach using latent models, in *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (2009), pp. 736–737

8. G. Chao, S. Sun, J. Bi, A survey on multi-view clustering (2017). arXiv preprint arXiv:1712.06246
9. M. Fraj, M.A.B. Hajkacem, N. Essoussi, Ensemble method for multi-view text clustering, in *International Conference on Computational Collective Intelligence* (Springer, Berlin, 2019), pp. 219–231
10. M. Fraj, M.A.B. Hajkacem, N. Essoussi, Self-organizing map for multi-view text clustering, in *International Conference on Big Data Analytics and Knowledge Discovery* (Springer, Berlin, 2020), pp. 396–408
11. E. Gaussier, C. Goutte, Relation between PLSA and NMF and implications, in *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (2005), pp. 601–602
12. T. Hofmann, Unsupervised learning by probabilistic latent semantic analysis. Mach. Learn. **42**(1), 177–196 (2001)
13. S.F. Hussain, M. Mushtaq, Z. Halim, Multi-view document clustering via ensemble method. J. Intell. Inform. Syst. **43**(1), 81–99 (2014)
14. D. Kim, D. Seo, S. Cho, P. Kang, Multi-co-training for document classification using various document representations: tF–IDF, LDA, and Doc2Vec. Inform. Sci. **477**, 15–29 (2019)
15. T. Kohonen, The self-organizing map. Proc. IEEE **78**(9), 1464–1480 (1990)
16. A. Kumar, H. Daumé, A co-training approach for multi-view spectral clustering, in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)* (2011), pp. 393–400
17. A. Kumar, P. Rai, H. Daume, Co-regularized multi-view spectral clustering, in *Advances in Neural Information Processing Systems*, vol. 24 (2011)
18. B. Larsen, C. Aone, Fast and effective text mining using linear-time document clustering, in *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. (CiteSeer, 1999), pp. 16–22
19. Y. Liang, Y. Pan, H. Lai, J. Yin, Robust multi-view clustering via inter-and-intra-view low rank fusion. Neurocomputing **385**, 220–230 (2020)
20. K.Y. Lin, L. Huang, C.D. Wang, H.Y. Chao, Multi-view proximity learning for clustering, in *International Conference on Database Systems for Advanced Applications* (Springer, Berlin, 2018), pp. 407–423
21. Z. Lin, R. Liu, Z. Su, Linearized alternating direction method with adaptive penalty for low-rank representation, in *Advances in Neural Information Processing Systems*, vol. 24 (2011)
22. J. Liu, C. Wang, J. Gao, J. Han, Multi-view clustering via joint nonnegative matrix factorization, in *Proceedings of the 2013 SIAM International Conference on Data Mining* (SIAM, 2013), pp. 252–260
23. F. Nie, G. Cai, X. Li, Multi-view clustering and semi-supervised classification with adaptive neighbours, in *AAAI* (2017), pp. 2408–2414
24. J.W. Reed, Y. Jiao, T.E. Potok, B.A. Klump, M.T. Elmore, A.R. Hurson, TF–ICF: a new term weighting scheme for clustering dynamic data streams, in *2006 5th International Conference on Machine Learning and Applications (ICMLA'06)* (IEEE, Piscataway, 2006), pp. 258–263
25. A. Strehl, J. Ghosh, Cluster ensembles—a knowledge reuse framework for combining multiple partitions. J. Mach. Learn. Res. **3**(Dec), 583–617 (2002)
26. Z. Tao, H. Liu, S. Li, Z. Ding, Y. Fu, From ensemble clustering to multi-view clustering, in *IJCAI* (2017)
27. G. Tzortzis, A. Likas, Kernel-based weighted multi-view clustering, in *2012 IEEE 12th International Conference on Data Mining* (IEEE, Piscataway, 2012), pp. 675–684
28. X. Wan, Co-training for cross-lingual sentiment classification, in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1* (Association for Computational Linguistics, 2009), pp. 235–243
29. Q. Wang, Y. Dou, X. Liu, Q. Lv, S. Li, Multi-view clustering with extreme learning machine. Neurocomputing **214**, 483–494 (2016)

30. B. Wei, C. Pal, Cross lingual adaptation: an experiment on sentiment classifications, in *Proceedings of the ACL 2010 Conference Short Papers* (Association for Computational Linguistics, 2010), pp. 258–262
31. X. Xie, S. Sun, Multi-view clustering ensembles, in *2013 International Conference on Machine Learning and Cybernetics (ICMLC)*, vol. 1 (IEEE, Piscataway, 2013), pp. 51–56
32. Y. Yang, H. Wang, Multi-view clustering: a survey. Big Data Mining Anal. **1**(2), 83–107 (2018)
33. C. Zhang, Q. Hu, H. Fu, P. Zhu, X. Cao, Latent multi-view subspace clustering, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 4279–4287
34. X. Zhang, L. Zong, X. Liu, H. Yu, Constrained NMF-based multi-view clustering on unmapped data, in *AAAI* (2015), pp. 3174–3180
35. X. Zhao, N. Evans, J.L. Dugelay, A subspace co-training framework for multi-view clustering. Pattern Recogn. Lett. **41**, 73–82 (2014)
36. L. Zheng, T. Li, C. Ding, Hierarchical ensemble clustering, in *2010 IEEE International Conference on Data Mining* (IEEE, Piscataway, 2010), pp. 1199–1204
37. F. Zhuang, G. Karypis, X. Ning, Q. He, Z. Shi, Multi-view learning via probabilistic latent semantic analysis. Inform. Sci. **199**, 20–30 (2012)

Chapter 9
Real-Time K-Prototypes for Incremental Attribute Learning Using Feature Selection

Siwar Gorrab, Fahmi Ben Rejab, and Kaouther Nouira

9.1 Introduction

Nowadays, data accessibility is not such a trivial deal because it is continuously generated by thousands of data sources [1]. Day by day, the amount of these information becomes larger since huge volumes of sensory, transactions, and web data are unceasingly generated at high rate as data streams. These are sequences of mixed data elements from various sources which further need to be analyzed and investigated. Thus, we can admit that one of the main sources of big data can be the streaming data. By the same token, data streams are everywhere and are generated by the applications like cell-phones, cars, security sensors, televisions, and so on. Hence, its partitioning into sets of meaningful sub-classes is required for proper and efficient mining of intended data.

Different from traditional machine learning strategies which train all features in one batch, incremental attribute learning (IAL) is a feasible machine learning strategy for solving high-dimensional pattern classification problems [2]. It is a divide and conquer machine learning task that could deal with dynamic feature space. IAL is considered as the most critical and tough machine learning tasks [3]. Actually, it has been proposed to avoid retraining from the scratch once new attributes emerge with newly joined samples as streaming data. So far, IAL has been widely employed for pattern recognition based on a number of different predictive machine learning algorithms [4]. As well, some literature proved that IAL is an applicable strategy for solving numerous supervised machine learning problems in terms of new entering features. But, it is rarely mentioned using unsupervised classification techniques. For instance, in [5] a new incremental unsupervised k-

S. Gorrab (✉) · F. Ben Rejab · K. Nouira
Université de Tunis, ISGT, LR99ES04 BESTMOD, Tunis, Tunisie

B. Alyoubi et al. (eds.), *Machine Learning and Data Analytics for Solving Business Problems*, Unsupervised and Semi-Supervised Learning,
https://doi.org/10.1007/978-3-031-18483-3_9

prototypes algorithm has been proposed to avoid retraining from the scratch when dealing with IAL task. As a matter of fact, applying IAL in all the above mentioned studies leads up mostly to a better performance and efficiency than the conventional methods that train all data attributes in one batch. So, here is a request to promote the unsupervised machine learning techniques to handle IAL task when dealing with big data.

More to the point, and while the learning process, data is continuously generated by thousands of data sources as time proceeds under the form of data streams. So, raw data usually comes with many forms of imperfections. This includes missing values, redundancies, noise, and/or inconsistencies which may lead to low-quality and undermined subsequent learning algorithms. So, it is the reason why conducting properly data preprocessing techniques as a mandatory step influences significantly the performance and the coherence of subsequent automatic discoveries and decisions. Moreover, several data mining algorithms do not perform well when having data with large amounts of attributes which include irrelevant ones [6].

Most data-driven analyses and modeling techniques may not scale well to high-dimensional data. It is because the learning quality and efficiency usually degrade rapidly with increased dimension [7]. Particularly, high-dimensional data sets may lead to the curse of dimensionality in clustering. Therefore, feature selection is required for better analysis and comprehension which consists in selecting a subset of relevant attributes for mining, among all original attributes. Feature selection is the concept of improving the model performance by applying machine learning since the presentation of the model is influenced by the features of the data we used to train the model [8].

The goal here is to obtain a feature subset from all available data attributes that is far more beneficial to the further process. This is done through eliminating irrelevant and redundant attributes which may lead to undesired correlations in further mining process [9]. Consequently, this will make faster the learning process and requires less storage space by dint of identifying a subset of the most relevant features [10]. Add it to that, feature selection is rarely used in the unsupervised tasks (e.g., clustering) [9]. So here is a necessity to precede clustering mixed data streams with applying feature selection preprocessing technique. Hence, feature selection techniques have become an apparent need in diverse applications and widely used on large-scale data and online learning problems due to their performance.

In this paper, we propose a new approach that handles IAL task, namely Feature Selection Incremental K-prototypes (FSIK-prototypes). This later conserves the old knowledge which has already been extracted from the previously built model and continue learning on newly emerged instances with new added features, as data stream, in order to build a complementary model. As a result, it includes knowledge from old and newly added data stream model as well. Into the bargain, when talking about newly joined features, the question to ask here is: are they all relevant for mining and future analysis?

Our proposal gradually trains additional attributes one by one. Indeed, it precedes clustering new data streams escorted with newly joined features with applying the feature selection preprocessing technique as a mandatory step in order to select only significant ones. So, this research would contribute using the unsupervised k-prototypes algorithm to a better comprehension of the impact of feature selection preprocess in IAL view point.

The rest of the paper is organized as follows: Sect. 9.2 reviews the conventional k-prototypes algorithm and the streaming data preprocessing literature. Section 9.3 describes our contribution. Section 9.4 is devoted to the experimental results and to discuss the obtained analysis. Section 6 concludes the paper.

9.2 K-Prototypes and Streaming Data Preprocessing

Certainly, the field of mixed data clustering progresses overall areas since most of real-world applications include numerical as well as categorical data.

9.2.1 Theoretical Concepts of K-Prototypes Algorithm

The k-prototype clustering algorithm was proposed in [11] as an extension to the k-means clustering algorithm in order to deal with the mixed data by combining both k-modes [12] and k-means [13] clustering algorithms. This algorithm is well recognized in mixed data domains because it is clear, scalable, and convergent. It provides new representations of cluster centers as well as a novel definition of distance (dissimilarity measure) between a data point and a cluster center [11]:

Given a data set $X = \{x_1 \dots x_n\}$ of n data points containing m_r numeric attributes and m_t categorical attributes, the focal objective of the K-prototype clustering algorithm is to group the data set X into k clusters while minimizing the following cost function:

$$J = \sum_{i=1}^{n} \sum_{j=1}^{k} u_{ij} d(x_i - c_j), \tag{9.1}$$

where $u_{ij} \in \{0, 1\}$ is an element of the partition matrix U_{n*k}, indicating the membership of data point i in cluster j; $c_j \in C = \{c_1 \dots c_k\}$ is the center of the cluster j and $d(x_i - c_j)$ is the dissimilarity measure defined as follows in Eq. (9.2):

$$d(x_i - c_j) = \sum_{r=1}^{m_r} \sqrt{(x_{ir} - c_{jr})^2} + \sum_{t=1}^{m_t} \delta(x_{it}, c_{jt}). \tag{9.2}$$

Note that x_{ir} and x_{it} represent, respectively, the values of the numeric attribute r and the categorical attribute t for a data point i; c_{jr} represents the mean of the numeric attribute r and cluster j, calculated using Eq. (9.3):

Algorithm 1 K-prototypes clustering algorithm

```
1: Input: X: data set, k: number of clusters
2: Output: Cluster centers
3: Begin
4:    Select $k$ initial prototypes (cluster centres) randomly from the data set $X$
5:    Attribute each data point in $X$ to its closest cluster center according to Eq. (9.2).
6:    Update the cluster centres after each allocation using Eqs. (9.3) and (9.4).
7:    If the updated cluster centers are identical to the previous ones then terminate, otherwise,
      return to step 5.
8: End.
```

$$c_{jr} = \frac{\sum_{i=1}^{|c_j|} x_{ir}}{|c_j|}. \tag{9.3}$$

With $|c_j|$ is the number of data points assigned to a cluster j; c_{jt} is the most common value (mode) for categorical attributes t and cluster j, calculated as shown in Eq. (9.4):

$$c_{jt} = a_t^h, \tag{9.4}$$

where $f(a_t^h) \succeq f(a_t^z)$, $\forall z$, $1 \leq z \leq m_c$, having $a_t^z \in \{a_t^1 \ldots a_t^{m_c}\}$ is the categorical value z and m_c is the number of categories of categorical attribute t; $f(a_t^z) =| x_{it} = a_t^z \mid p_{ij} = 1 |$ is the frequency count of the attribute value a_t^z; for categorical features, $\delta(p, q) = 0$ when $p = q$ and $\delta(p, q) = 1$ when $p \neq q$.

9.2.1.1 Algorithm

The main process of the k-prototypes method is described in the following algorithm:

This non-incremental clustering algorithm requires to store and to process all the input data pattern matrix in the memory. Put differently, it necessitates the complete input data being loaded into the memory and thus results in high requirements of memory space [14]. Besides, the batch k-prototypes algorithm deals only with object learning. So, it is unable to handle IAL task. Notably, incremental clustering algorithms consider the input data pattern which is to be processed once at a time in the memory in such a way that it becomes easy to add the new input data patterns into the existing clusters [14]. Thus, we devote our work to study the k-prototypes clustering algorithm that handles mixed data attributes in IAL context. Furthermore, we aim to examine the impact of applying feature selection preprocessing technique when learning mixed data streams incrementally.

9.2.2 Streaming Data Preprocessing Techniques

Raw data is usually susceptible to missing values, noisy data, incomplete data, and outlier data. This is the reason why data preprocessing becomes a mandatory step in which data gets prepared and transformed before being mined to bring it to such a state that now the machine can easily parse it [15]. As well, it is essential to understand the nature of the data so that to enhance data efficiency and performing a more relevant data analysis after solving data problems that may, otherwise, lead to performing inaccurate data analysis. Dominated by increasingly large data sets, several data preprocessing methods have become necessary in current knowledge discovery scenarios [16]. In details, they aim at reducing the complexity inherent to real-world data sets, so that they can be easily processed by current data mining solutions [16]. Consequently, these approaches result in more understandable structure of raw data and faster and more precise learning process.

9.2.2.1 Dimensionality Reduction for Data Streams

Data preprocessing and especially dimensionality reduction have become fundamental techniques in current knowledge discovery scenarios for wide-scale and streaming data learning tasks [17]. Indeed, the classification of high-dimensional data is a challenging problem due to the presence of redundant and irrelevant features in a higher amount [17]. These irrelevant features demean the performance of machine learning algorithms and increase their computational complexity. Nevertheless, data streams impose specific constraints to be learned that are not available in batch algorithms. Actually, assuming the fact that the whole training sets are not available from the beginning of the learning process in a streaming data context, static data preprocessing techniques are not directly adapted to handle data streams [16]. In the context of data preprocessing methods for handling data streams, dimensionality reduction techniques still have a long road ahead of them. Despite online learning is growing in importance due to the development of internet and technologies for massive data collection [16].

The problem of preprocessing techniques for data streams is challenging due to the challenging nature of the data that is continuously arriving and evolving [18]. Particularly, dimensionality reduction is an essential task for many large-scale information processing problems such as classifying document sets, searching over web data sets, etc. It is commonly used to mitigate the curse of dimensionality which is a serious problem as it will impede the operation of most data mining algorithms as the computational cost rise [9]. Likewise, dimensionality reduction methods fall into two categories:

1. Feature extraction: it intends to extract attributes by projecting the original high-dimensional data into a lower-dimensional space through algebraic transformations [9]. Typically, techniques in the feature extraction category are more effective than those in feature selection category, but they may break down when

processing large-scale data sets or data streams due to their high computational complexities [9].

2. Feature selection: its intention is to find out a subset of the most representative features according to some criteria. Feature selection algorithms have been widely used on large-scale data because they are more efficient than traditional feature extraction methods [9].

Particularly, we aim to go through one particular dimensionality reduction preprocessing technique, namely feature selection, that is to select only the most relevant and prominent attributes among the new incoming ones that permeate inside emerging data streams. Despite its popularity use in numerous classification tasks in order to enhance the performance of this supervise algorithm such as [19] for improving the classification as well as prediction accuracy, [20] for classification of hyperspectral data by SVM, and unsupervised tasks (e.g., clustering) [9]. For instance, in [21], an extended k-means was proposed to cumulatively determine whether the newly arrived feature can be selected as a representative streaming feature. This is the reason why we have decided to tackle this task and we aim to supply a better insight when covering this issue.

9.3 Proposed Feature Selection Incremental K-Prototypes

Once new instances become available over time, particularly when these new input patterns include new ones, in addition to the old set of features, our two proposed methods are capable of handling these continuously arriving data streams incrementally. This novel method, entitled Feature Selection Incremental K-prototypes (FSIK-prototypes), is mainly an extension of the proposed IK-prototypes method in [5]. It deals with such dynamic object and attribute spaces, namely the skillful incremental mixed attribute and instance learning context.

9.3.1 Feature Selection Incremental K-Prototypes Through Incremental Attribute Learning Context

While seeking to ameliorate the performance of the initially proposed IK-prototypes [5] method in incremental attribute learning (IAL) context, we intend to introduce our contribution. Notably, FSIK-prototypes are an extension and an increased version of the IK-prototypes [5] algorithm, which precedes clustering incoming mixed data streams with newly joined attributes with applying the feature selection preprocessing technique before modeling data. The goal here is to select only the most relevant and prominent features among the newly joined ones. Another point is that we desire to investigate the impact of applying the feature selection in

IAL context and to check whether it will aptly influence the performance and the efficiency of this incremental unsupervised attribute learning algorithm.

9.3.1.1 Definition and Approach Presentation

As being a technique that searches to pull out the most useful and efficient features among all the available ones in a data set for a machine learning model, feature selection is one of the most important preprocessing techniques. Actually, when you supply too many variables to a model, especially variables that are not useful to the model, it affects its performance in a negative way. So, we want to explore at what point this relevant operation will boost the model performance. Despite its popularity use in numerous classification tasks in order to enhance the performance of this supervised learning algorithm, feature selection is rarely used in unsupervised tasks [9]. Besides, the problem of preprocessing for data streams is challenging due to the challenging nature of the data that is continuously arriving and evolving [18]. This is why we would like to search closely about the influence and the utility of the feature selection preprocessing operation on the IAL process and whether it would affect the algorithm performance.

To do so, we have opted to use the variance threshold selection criterion which consists in calculating the variance of the recently joined features, present in an emerging data stream. Then, we fix a variance threshold based on the alteration of the calculated variances. Finally, we will select the attributes according to their degrees of variation. The selector in this case is the variance of the features in the data set. Its intention is to find out a subset of the most representative features according to the variance criteria, then eliminates those that are below a certain threshold. Actually, what is the utility of having a variable that does not vary in all the data rows, means that is constant?

Definitely, there exist numerous other selecting techniques or what we call transformers that could be used. Nonetheless, in our study we are focusing on the efficiency of feature selection in IAL context rather than on the method itself. Thus, we would like to justify our choice of the variance threshold selection criterion with the fact that:

- Firstly, the variance threshold criterion has been chosen owing to the fact that it is a simple and effective baseline approach to feature selection.
- Besides, it removes all zero variance features by default.
- Last but not least, the variance calculation is a straightforward and a simple task which does not require complicated operations.

Figure 9.1 presents the architecture of our proposed Feature Selection Incremental k-prototypes method as follows:

1. The upper part of Fig. 9.1 sets out the data preprocessing phase that precedes the newly proposed IK-prototypes method when being applied on the newly emerging data streams. As a matter of fact, once new streaming instances came

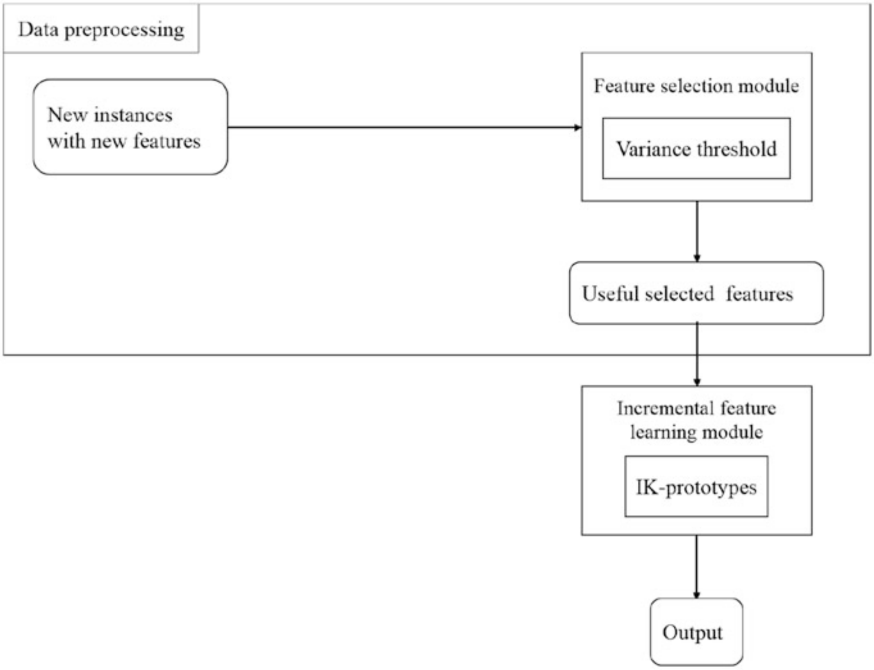


Fig. 9.1 Feature Selection Incremental K-prototypes: architecture of selected IAL based on IK-prototypes preceded by the feature selection preprocessing technique

to appear along with newly added attributes, the feature selection module is the first to be applied before the attribute learning process. More precisely, only the variance of each of the recently joined attributes will be calculated. Then, a variance threshold will be fixed, regarding the values of the calculated variances, in such a way that attributes with lower variances than the fixed variance threshold will not be selected. This threshold is responsible to distinguish the most relevant features among those that are not important. Here, we have to notice that the importance of the attribute refers to its variance in the data set.

2. The lower part of Fig. 9.1 illustrates the second step of the secondly proposed FSIK-prototypes method. Actually, the resulting selected features and the former ones would be provided to the incremental feature learning module which is based on the firstly proposed IK-prototypes algorithm. Thereby, new features joined with streaming data rows are successfully learned in an incremental way.

We take as example a data set D1, composed of 690 instances and 15 attributes. To start with, we introduce an input data composed of 400 objects and 10 attributes. Then, a data stream came to join the learning process with 290 instances and 15 attributes (10 old attributes and 5 new ones). So, the feature selection technique will be applied only on the recently joined features starting with calculating their variances as shown in Table 9.1.

Table 9.1 Calculated variances of each new added attribute

Attribute	Variance
A10	0.237
A11	10.374
A12	0.245
A13	0.243
A14	68.257
A15	14.010

Regarding the alteration of the calculated variances of the newly joined attributes, we can easily fix a variance threshold whereof only attributes' variances higher than this threshold would be selected. Put differently, the selected features are the most pertinent and significant ones since having important variances, generally higher than 40%. Means that those features are significant in the data set while being present with different values in the whole data rows.

Here from Table 1.1, we can fix the variance threshold at 0.3 and then select the attributes whose variances' are higher than this threshold. As a result, the selected features would be A11, A14, and A15 and we will neglect the A10, A12, and A13 attributes. Once the feature selection module is accomplished, we are faced with a data stream with the most essential newly added features. Now, we might proceed with the incremental feature learning module as just illustrated, using the proposed IK-prototypes [5] algorithm. This method will be accurately explained in the following subsection.

9.3.2 *Incremental K-Prototypes Through Incremental Attribute Learning Context*

At present, data availability is not such a trivial question owing to the fact that we are dealing with dynamic instance and feature spaces. So, to handle such continuous emergence, two procedures might be appealed as follows: either to ignore the previous data and take into account only the new one or to retrain from scratch. Still, both are not the best procedures. As a matter of fact, the former is about to lose knowledge while focusing only on the newly arrived chunks of data and neglecting the old ones. The latter would be a time consuming giving that a retraining from the scratch will occur each time new unbounded sequences of data arise.

Thereupon, a new clustering technique based on the k-prototypes algorithm has been proposed in [5] and which well performs when data streams appear with new joined mixed attributes, in addition to the old set ones. As data stream proceeds, the algorithm tackles both incremental object and attribute learning tasks at the same deal. Its objectives are to:

- Supply the k-prototypes algorithm with a novel property, that is to manage the incremental attribute learning task in addition to its object learning capability.

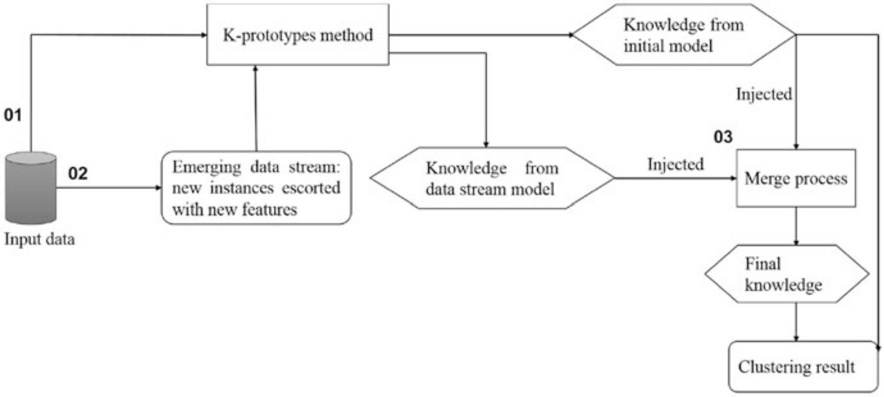

K-prototypes method
Knowledge from initial model
Injected
01
03
02
Emerging data stream: new instances escorted with new features
Knowledge from data stream model
Injected
Merge process
Input data
Final knowledge
Clustering result

Fig. 9.2 An overview of the proposed Incremental K-prototypes through IAL context [5]

- Respect better basics of clustering in terms of dispersion of elements within and between clusters.
- Reduce time processing when assigning the incoming objects with new attributes to their appropriate clusters.

The proposed method includes the old knowledge which has already been extracted from the previously built model and continues learning only on the newly added instances which include new features in order to build a complementary model that incorporates knowledge from old and newly added instances as well. Mainly, on account on these new instances escorted with new attributes that have to be learned as development proceeds, this proposed incremental object and attribute learning algorithm is conducted as shown in Fig. 9.2.

To start with, we have to mention that the proposed IK-prototypes method is established in three steps as shown in Fig. 9.2. First of all, once an initial input data is available, we start with applying the conventional k-prototypes algorithm. The result here is k different clusters as knowledge from the initial model in which each similar instances are joined together into one cluster. Afterwards, new data objects with new features penetrate as well be a data stream.

At this level, we move on to the second step that consists in applying the same conventional k-prototypes algorithm only on the newly joined data stream. Hence, the aim here is to create k' clusters using the same standard k-prototypes algorithm and then save the obtained results as a second knowledge from the data stream model. In this regard, we can admit that the same procedure takes place for the streaming data, which consists mainly in new incoming instances escorted with new features that penetrate as clustering proceeds. Accordingly, a second model have just been created once learning the new data stream. Overall, we have gained knowledge from both models: the first one is derived from the input data, while the second one is derived from the continuously emerging streaming data.

9.3.2.1 Algorithm

The main process of the proposed IK-prototypes method is mainly described in Algorithm 1:

Algorithm 2 Incremental K-prototypes algorithm

```
1: Input: $X$: data set, $k$: number of clusters
2: Output: $k$ Cluster centers
3: Begin
4:    Select $k$ initial prototypes (cluster centres) randomly from each arriving data stream $X$.
5:    Attribute each data point in $X$ to its closest cluster center according to Eq. (9.2).
6:    Update the cluster centres after each allocation using Eqs. (9.3) and (9.4).
7:    If the updated cluster centers are identical to the previous ones then terminate, otherwise, go
      back to step 5.
8:    If a new data stream arrives with new attributes then go back to step 5, Merge the resulting
      clusters from both models, otherwise, terminate.
9: End.
```

At the outset, assuming that some primarily accessible input data has been learned using the conventional k-prototypes algorithm. Afterwards, and from the moment that new sample of instances escorted with new features take place, it is suggested to keep in background the k-prototypes algorithm which would, in this step, learn only these new incoming data streams escorted with new added attributes. Subsequently, knowledge from both initial and data stream models would be incorporated into one resulting cluster after applying the merge procedure. This results in acquiring knowledge from the joined models and to achieve the IAL task based on the k-prototypes algorithm. The main difficulty here is how to fuse this knowledge, and how to make the model aware about the precedent data on which an initial model has already been built?

We are going to provide a consistent answer to this question and to deeply explain the merge process while illustrating the merge procedure in the next subsubsection.

9.3.2.2 Merge Procedure

Once, an initial model has been built based on some available data. Then, at the arrival of new features with the joined instances, the IK-prototypes keep learning the new samples. Thereafter, it injects the information extracted from the initial model in the merge process, so that it could deal with IAL task without retraining from the scratch. This method creates a new based merge [22] clustering algorithm that can handle incremental attribute learning task for mixed data streams. Additionally, whenever new input data objects came to appear, as being a joined data stream escorted with new attributes, the proposed incremental system can learn these new incoming data streams without retraining from the scratch or wasting information with neglecting the input set of data.

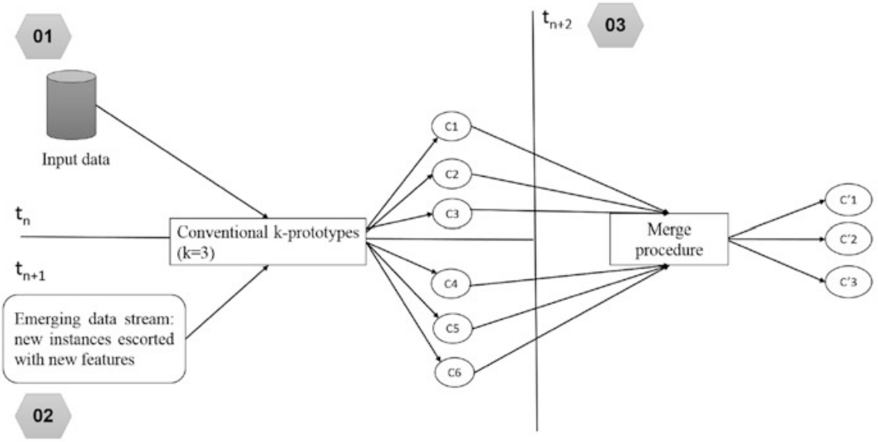

01
Input data
t_n
t_{n+1}
Conventional k-prototypes (k=3)
Emerging data stream: new instances escorted with new features
02
C1
C2
C3
C4
C5
C6
t_{n+2}
03
Merge procedure
C'1
C'2
C'3

Fig. 9.3 Workflow of the merge process

Table 9.2 Number of instances that contains each cluster before merge

Data set	D1 (690 instances)
Cluster 1	171
Cluster 2	126
Cluster 3	103
Cluster 4	63
Cluster 5	77
Cluster 6	150

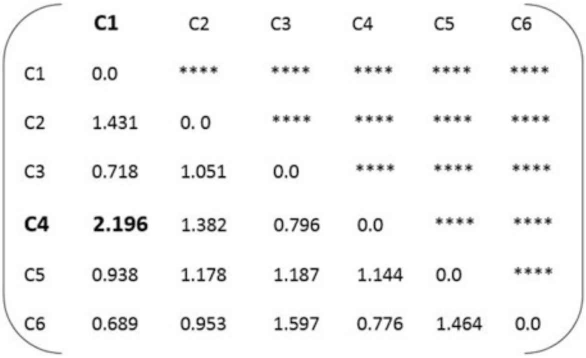
Particularly, how to make the model aware about the old knowledge which has already been extracted from the previously built model?

Actually, it is based on the merge procedure that consists in combining knowledge coming from both gained models in such a way that each two similar clusters are combined together and while guaranteeing the return to the initial number of clusters k because it does not deal with incremental class learning task. This is the main purpose behind the proposed incremental feature learning procedure.

In short, to gain the most coherent and stable clusters as shown in Fig. 9.3, we focused on merging the most two likely clusters from k and k' clusters, resulting from both initial and data stream model in order to output more coherent clusters and further to achieve the unsupervised incremental mixed attribute learning task. Figure 9.3 illustrates the main idea behind the functionality of the merge technique.

For instance, we take the example of a data set D1 which will be illustrated later on in order to give a better insight to the workflow of the merge procedure. Subsequently, Table 9.2 shows the number of instances that contains each cluster of the 690 objects data set D1 with number $K = k + k' = 6$ clusters

To this end, the similarity between the most two likely searched clusters is mainly based on the two following indexes:



	C1	C2	C3	C4	C5	C6
C1	0.0	****	****	****	****	****
C2	1.431	0. 0	****	****	****	****
C3	0.718	1.051	0.0	****	****	****
C4	**2.196**	1.382	0.796	0.0	****	****
C5	0.938	1.178	1.187	1.144	0.0	****
C6	0.689	0.953	1.597	0.776	1.464	0.0

Fig. 9.4 Davies-Bouldin Index matrix (DB) for each two clusters

1. Davies-Bouldin Index (DBI) [23]: this index is the first merge criteria that calculates the average similarity between clusters. Similarity here is based on a comparison between the distance between clusters and the size of the clusters themselves. Indeed, the lower BDI is, the better partition of clusters is. Besides, values closer to zero relate to a better partition of clusters. Thus, the two clusters corresponding to the highest DBI value will be selected. The matrix (DB) below in Fig. 9.4 displays the degree of similarity between all the K clusters, where the closest clusters coincide with the highest DBI value. The first line and the first column of this matrix are cluster centers of the different K clusters.

   The DBI similarity is calculated in the following way shown in Eq. (9.5):

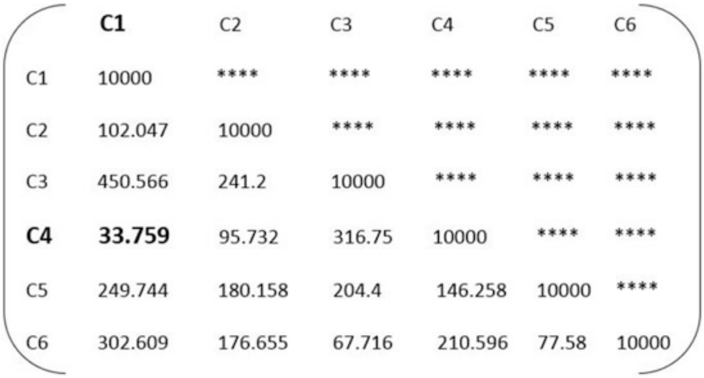
$$DBI = \frac{1}{k}\sum_{i=1}^{k} \max R_{ij}; i \neq j. \tag{9.5}$$

   DBI is defined as the average similarity between each cluster C_i for $i = 1, \ldots, k$ and its most similar one is C_j; R_{ij} is the similarity measure. In the context of the DBI, a simple choice to construct R_{ij} so that it is non-negative and symmetric is as follows in Eq. (9.6):

$$R_{ij} = \frac{s_i + s_j}{d_{ij}}, \tag{9.6}$$

   where s_i is the cluster diameter, it represents the average distance between the point i/j and its cluster center; d_{ij} represents the distance between both cluster centroids i and j.

2. Calinski-Harabasz Score (CHS) [24]: also known as the variance ratio criterion, the CHS might be used to evaluate a clustering model. This index searches for similarity between clusters. In fact, it determines the ratio of the sum of between-clusters dispersion and of inter-cluster dispersion for all clusters where the dispersion here is definitely the sum of the squared distances between each



	C1	C2	C3	C4	C5	C6
C1	10000	****	****	****	****	****
C2	102.047	10000	****	****	****	****
C3	450.566	241.2	10000	****	****	****
C4	**33.759**	95.732	316.75	10000	****	****
C5	249.744	180.158	204.4	146.258	10000	****
C6	302.609	176.655	67.716	210.596	77.58	10000

Fig. 9.5 Calinski-Harabasz score matrix (CH) for each two clusters

point to its nearest cluster center. In opposition to the DBI, a higher CHS refers to a model with better defined clusters. It is calculated as follows in Eq. (9.7):

$$s = \frac{tr(B_k)}{tr(W_k)} * \frac{n_E - k}{k - 1}, \tag{9.7}$$

where n_E is the size of the set of data **E**; k is the number of clusters; $tr(B_k)$ is trace of the between group dispersion matrix; $tr(W_k)$ is the trace of the within-cluster dispersion matrix, defined by Eq. (9.8)

$$B_k = \sum_{q=1}^{k} \sum_{x \in C_q} (x - c_q)(x - c_q)^T; \; W_k = \sum_{q=1}^{k} n_q (C_q - c_E)(C_q - c_E)^T \tag{9.8}$$

knowing that n_q is the number of points in cluster q; c_q and c_E are, respectively, the centers of the cluster q and of E.

The CHS is high when clusters are dense and well separated which leads to the standard concept of clustering. Given that, we will select the two clusters owing the lowest CHS as shown in Fig. 9.5, where the same six clusters used with the previous merge criteria are present.

Between those calculated indexes, the proposed algorithm is supposed to choose the clusters corresponding to the highest DBI coinciding with the lowest CHS. In this case, both indexes represented in bold in Fig. 9.4 as in Fig. 9.5 refer to the same couple of clusters that will be merged: cluster 1 and cluster 4. As a matter of fact, we have successfully extracted the most two similar clusters where the two latter scores coincide, and then, merge these two selected clusters so that the incremental attribute learning task will be established. Thus, this couple of clusters will be fused in one resulting cluster. This operation will be repeated until reaching the initial number of cluster k. Over and above that, the IK-prototypes [5] have effectively avoided retraining from the scratch when new instances with newly added attributes arrive.

Nevertheless, once in a while the highest DBI and the lowest CHS may not be the best choices for the merge procedure if they result in different combination of clusters. In such a case, the merge algorithm will carry on with combining both clusters resulting from the two calculated indexes and ends up with maintaining the cluster owing the lowest Sum of Squared Error value (SSE). This latter computes the dispersion of elements of a cluster in relation with their centroids, that is the sum squared distances of samples to their closest cluster center. By the same token, it is used as an evaluation criteria for clustering algorithms and its calculation is as follows in Eq. (9.9):

$$SSE_{x \in C} = \sum_{i=1}^{n} \sum_{j=1}^{m} (x_{kj} - c_j), \tag{9.9}$$

where i, j are different clusters; c_j is the centroid of the cluster j; x_{kj} is the sum of squared distances from the point k to the nearest cluster j.

Here is the merge process algorithm used in our proposed Incremental k-prototypes:

Now that the merge algorithm has been presented, we go forward to detail the used functions.

1. ***Compute (Index):***
   - **Input:** input clusters
   - **Output:** matrices of calculated Davies-Bouldin Index values and/or Calinski-Harabasz scores

   An essential function that searches to pull out the best couple of clusters to further be merged. It calculates the Davies-Bouldin and the Calinski-Harabasz scores, presented after all in the two symmetric matrices DB and CH. Each box of these matrices shows one of the indexes value between two clusters.
2. ***Max (DB/CH), Min (DB/CH):***
   - **Input:** matrices of DB indexes or CH scores
   - **Output:** highest and/or lowest corresponding index

   These two functions are used in the proposed algorithm to retrieve, respectively, the highest and the lowest indexes from the input matrices. Accurately, the Max function searches into Davies-Bouldin matrix for the largest score between two clusters. Unlike the Min function that looks for the smallest Calinski-Harabasz score between each couple of clusters.
3. ***Merge (clusters):***
   - **Input:** cluster i, cluster j
   - **Output:** cluster ij

Algorithm 3 Merge algorithm

```
Input: clusters = $\{c_1, c_2, ..c_k\}$
Output: clusters = clusters = $\{c^{'}_{i..k}\}$
Begin
  For each cluster $A_i$ in $\{c_1, c_2, c_k\}$ do
    For each cluster $B_j$ in $\{c_4, c_5, c_{k'}\}$ do
      $DB \leftarrow computeDaviesBouldinIndex(A_i, B_j)$
      $CH \leftarrow computeCalinskiHarabaszIndex(A_i, B_j)$
    end For
  end For
  For each cluster $A_i$ in $\{c_1, c_2, c_k\}$ do
    For each cluster $B_j$ in $\{c_4, c_5, c_{k'}\}$ do
      If $DB(A_i, B_j)$ = Max(DB) then
        $Cluster1 \leftarrow A_i$
        $Cluster2 \leftarrow B_j$
      end If
      If $CH(A_i, B_j)$ = Min(CH) then
        $Cluster3 \leftarrow A_i$
        $Cluster4 \leftarrow B_j$
      end If
    end For
  end For
  For $i$ in $[1..k]$ do
    If (Cluster1=Cluster3) and (Cluster2=Cluster4) then
      $c^{'}_i \leftarrow Merge(Cluster1, Cluster2)$
      Delete (Cluster1)
      Delete (Cluster2)
      i=i+1
    end If
      $SSE1 \leftarrow SSE(Merge(Cluster1, Cluster2))$
      $SSE2 \leftarrow SSE(Merge(Cluster3, Cluster4))$
    If SSE1 < SSE2 then
      $c^{'}_i \leftarrow Merge(Cluster1, Cluster2)$
      Delete (Cluster1)
      Delete (Cluster2)
    Else
      $c^{'}_i \leftarrow Merge(Cluster3, Cluster4)$
      Delete (Cluster1)
      Delete (Cluster2)
    end If
  end For
Return clusters = $\{c^{'}_{i..k}\}$
End.
```

The merge function allows to combine the most similar couple of clusters. Accordingly, it results in one cluster which incorporates elements from both selected clusters.

4. ***Delete (cluster):***

   - **Input:** cluster

   After merging the most appropriate clusters, we need to remove each one of them. Consequently, we avoid redundancy given that the new created cluster contains elements from both of them.

9.4 Experimentation

Aiming to investigate the capability of our proposed approach and to show its efficiency, we proceed by performing multiple experiments on number of real mixed data sets. Subsequently, we will study the results obtained from these different experiments so that to evaluate the performance of both proposed algorithms IK-prototypes [5] and FSIK-prototypes for mixed data streams clustering within IAL context. Additionally, our study leads to expert the scalability of our proposed FSIK-prototypes method not only by exposing the experimental results but also by measuring the evaluation metrics devoted to evaluate our work. Actively, we highly prioritize engaging our research to show how it capes with the weaknesses of the batch k-prototypes clustering algorithm.

9.4.1 Framework

We have lunched two experimental categories; in one's element is testing the efficiency of the firstly proposed algorithm IK-prototypes [5]. The former is an extension of the conventional k-prototypes algorithm that is able to deal with recently added features that occur in the incoming mixed data streams over time. Then we make focus on applying feature selection as being a mandatory preprocessing task, responsible of selecting the most effective and relevant features presented is our data sets for future analysis. For this second experimental part, we highlight as well the impact of this selection operation particularly on incremental unsupervised mixed attribute learning context. To do so, we have developed these new machine learning methods in addition to those in which our programs will be compared to using Python 3.7 and through the powerful scientific environment Spyder as being an open source cross-platform integrated development environment (IDE). Into the bargain, all experiments have been conducted on an i5 CPU, 64-bit computer with 4 GB memory.

Table 9.3 Summary details of used real mixed data sets

Data set	Number of object	Number of attributes	Acronym
German credit	1000	7 numeric, 13 categorical	Cg
Credit approval	690	5 numeric, 10 categorical	Ca
Chess (King-Rook vs.King)	28,056	2 numeric, 4 categorical	krk
sf-police-incidents	538,638	4 numeric, 3 categorical	SFPI

9.4.1.1 Real Data Sets Description

Aiming to evaluate the different simulations that have been applied on several real mixed data sets, we define for each data set its number of instances, number of attributes and its acronym. Therewith, the number of instances as well as the number of features vary from one data set to another. Likewise, the Chess and Credit Approval data sets are derived from the U.C.I repository [25]. Whereas, German Credit and the sf-police-incidents data sets are imported from openML [26] as described in Table 9.3.

- **German Credit (Cg)**: this real mixed data set classifies German people according to their credits. People could be arranged as owing good or bad credit risks.
- **Credit Approval (Ca)**: it concerns credit card applications in which all attribute names and values have been changed to meaningless symbols to protect the confidentiality of the data.
- **Chess (krk)**: it is a Chess end-game mixed data set for White King and Rook against Black King. It was generated by Michael Bain and Arthur van Hoff at the Turing Institute, Glasgow, UK.
- **sf-police-incidents (SFPI)**: it is about incident reports from the San Francisco Police Department between January 2003 and May 2018.

9.4.2 Evaluation Criteria

Intending to assess the performance of our proposal, we have basically used three indices, responsible for estimating the cluster cohesion (within or intra-variance) and the cluster separation (between or inter-variance) and combine them to compute a quality measure, in which the combination is performed by a division or a sum [27].

- **The Sum of Squared Error (SSE $\downarrow$)**: Since being a crucial criteria that gives a feedback about the inter-cluster and the intra-cluster similarity, it is calculated as earlier in Eq. (9.9). Also, the more it is closer to zero, the more clusters are well defined.
- **The Davies-Bouldin Index (DBI $\downarrow$)**: This index is probably one of the most used indices to evaluate clustering results. It might be used to evaluate a clustering

model where a lower score relates to a model with better defined cluster. This latter is calculated using Eq. (9.5).

- **The run time (RT ↓)**: Starting from the beginning of the procedure, the run time is, in all, the time needed to achieve the final clustering result: objects are distributed between k clusters.

9.4.3 Results and Discussion

For the sake of clarity, the incremental aspect is detailed as follows: admitting that an initial model has been built based on a primary sample of objects with a specific number of attributes and once new instances escorted with newly added features take place, knowledge from both constructed models will be merged to achieve the IAL task. Well, our proposed FSIK-prototypes algorithm may deal with this task. Into the bargain, it starts with selecting the main and the most effective features from the newly added ones, then continues processing similarly to the IK-prototypes algorithm based on the merge procedure. The merge process follows the next steps:

1. Calculate the DBI and the CHS for each cluster.
2. Identify the most two similar clusters to be merged.
3. Combine the two selected clusters using the merge Algorithm 3.

9.4.3.1 Sum of Squared Error Results

The following Table 9.4 exhibits the comparison between the k-prototypes, the Incremental K-prototypes and Feature Selection Incremental K-prototypes algorithms when fixing the number of clusters k at 3 (chosen randomly from the beginning of the learning process) and based on the Sum of Squared Error evaluation criteria. In fact, Table 9.4 summarizes the SSE values for each cluster of each data set by representing the total SSE values in its rows using, respectively, the last mentioned algorithms.

For a better comprehension, the IK-prototypes method outperforms the conventional k-prototypes algorithm in terms of stability of clusters for all used data sets since it gained lower SSE values. Knowing that the lower SSE value is, the

Table 9.4 The calculated SSE of K-prototypes vs. IK-prototypes vs. FSIK-prototypes per data set

Data sets	Cg	Ca	krk	SFPI
Conventional K-prototypes	2.95	4.93	3.19	4.18
Incremental K-prototypes	2.90	4.62	3.02	3.51
Feature selection incremental K-prototypes	2.82	2.25	2.94	3.01

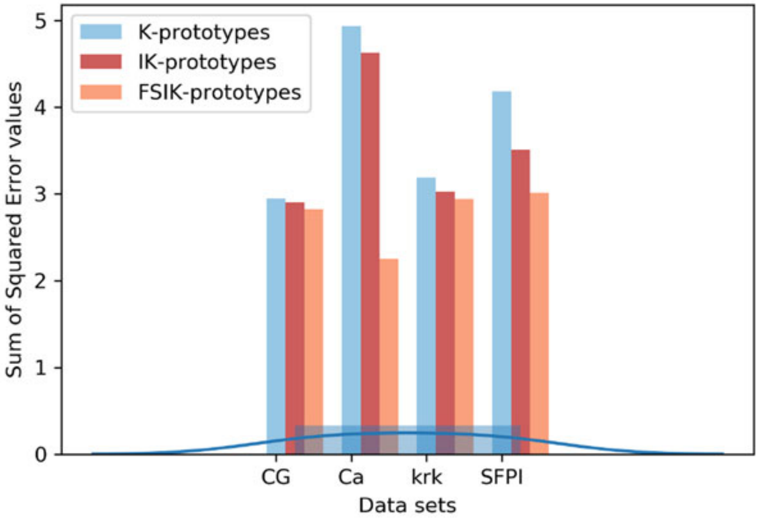


Fig. 9.6 SSE values of K-prototypes vs. Incremental K-prototypes vs. Feature Selection Incremental K-prototypes for each data set

Table 9.5 The calculated DBI of K-prototypes vs. IK-prototypes vs FSIK-prototypes per data set

Data sets	Cg	Ca	krk	SFPI
Conventional K-prototypes	3.42	2.56	2.69	2.66
Incremental K-prototypes	3.05	2.54	2.64	2.53
Feature selection incremental K-prototypes	2.26	2.05	2.10	1.97

maximum coherence within clusters and the minimum similarity between clusters are. Furthermore, our proposed FSIK-prototypes exceed both K-prototypes and IK-prototypes because it acquired the lowest SSE values for all used data sets as depicted in Fig. 9.6.

9.4.3.2 Davies-Bouldin Index Results

In the next Table 9.5, we detail the calculated DBI scores according to the diverse used data sets for the different three methods.

Looking at this Table 9.5, we can confirm that the proposed IK-prototypes method outperforms the batch k-prototypes method as long as it gained lower DBI scores for the numerous used data sets. Moreover, our proposal FSIK-prototypes represent the optimum feature learning approach while owing the smallest DBI values for all used data sets.

Figure 9.7 confirms that our proposed FSIK-prototypes method is the optimum incremental feature learning method in terms of developing a better defined model

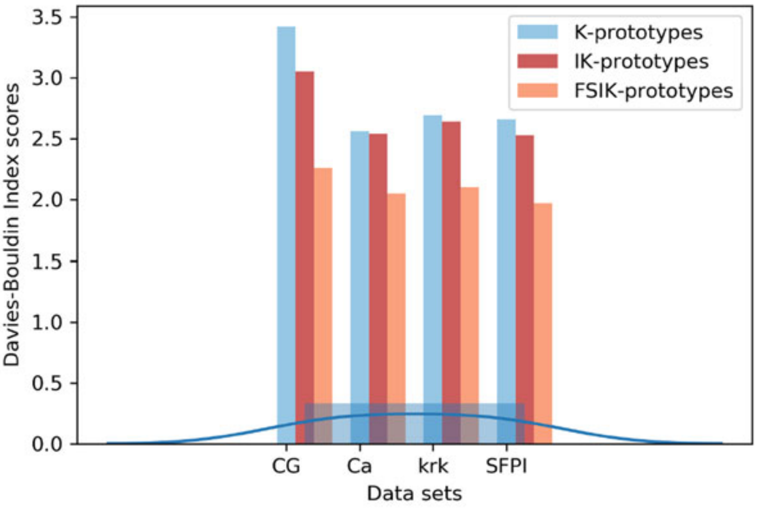


Fig. 9.7 Davies-Bouldin Index values of K-prototypes vs. Incremental K-prototypes vs. Feature Selection Incremental K-prototypes per data set

Table 9.6 Run time in seconds of k-prototypes vs. IK-prototypes vs. FSIK-prototypes per data set

Data sets	Cg	Ca	krk	SFPI
Conventional K-prototypes	20.68	10.05	354	14094.6
Incremental K-prototypes	11.24	5.23	267	10807.2
Feature selection incremental K-prototypes	10.73	4.81	126.6	5283.0

with more stability of clusters and more similarity within them. So, these obtained results highlight the relevance of our proposal in IAL context. As a matter of fact, we notice that the DBI emphasizes our proposal's efficiency and scalability.

9.4.3.3 Run Time Results

As being a significant evaluation criteria for all mentioned methods, the calculated run time in seconds of each used data set is illustrated in the next Table 9.6.

Looking at Table 9.6 that outlines the run time of the compared algorithms, we can notice the decreased results starting from the first row to the last one. Accordingly, we can observe the vantage of the IK-prototypes method in terms of time processing regarding the conventional k-prototypes algorithm. Well, the time required for its execution is almost less by half than the time needed for the batch k-prototypes execution and the offset is proportional the size of the data set. Furthermore, the FSIK-prototypes method presents also more promising results while requiring the lowest time for its execution. Thus, we can admit that the run time evaluation criteria emphasizes our proposal competence.

In a nutshell, our proposed FSIK-prototypes do not only outperform the IK-prototypes method but also do generally outperform the conventional k-prototypes algorithm by providing a well-defined model with better defined clusters. In fact, the IK-prototypes need less time processing to get final clustering results. According to the SSE values and the DB scores, it leads to a model with increased similarity within clusters and better separation between them. In the same way, our proposed FSIK-prototypes exceed the latest mentioned algorithms with generating a better partition between clusters and a well-defined model in less by half time consuming. All in all, the achieved results confirm that the incremental learning is an interesting alternative and constitutes one of the major concerns of the machine learning. Furthermore, performing feature selection before modeling data reduces training time because less data leads to train faster. In addition, it reduces over-fitting since it minimizes the possibility to make decisions based on noisy data, eases the learning process and speeds it up.

9.5 Conclusion

In this work, we introduced a novel clustering approach that avoids retraining from the scratch as data streams emerge with new added features. Besides, it selects only the most significant ones using the feature selection preprocessing technique. The efficiency of our proposal is approved across a comparison between the batch k-prototypes, the IK-prototypes and our proposed FSIK-prototypes methods, based on different evaluation criteria and numerous simulations made on four real mixed data sets. Encouraging by such promising outcome, we leave for future work to check the utility of ordered incremental feature learning algorithms based on our proposed method. Also to make use multiple other feature selection techniques for this task to further compare their results. Last but not least, we look for a dynamic k-prototypes method that deals with evolving feature, object and class learning spaces.

References

1. S. Gorrab, F.B. Rejab, Incremental-decremental attribute learning algorithm based on K-prototypes for mixed data stream clustering. Int. J. Comput. Inform. Syst. Ind. Manag. Appl. **13**, 149–159 (2021)
2. T. Wang, W. Zhou, X. Zhu, F. Liu, S.U. Guan, Integrated feature preprocessing for classification based on neural incremental attribute learning, in *2016 19th International Conference on Information Fusion (FUSION)* (IEEE, Piscataway, 2016), pp. 386–393
3. T. Wang, S.U. Guan, F. Liu, Ordered incremental attribute learning based on mRMR and neural networks. Int. J. Design Anal. Tools Integr. Circuits Syst. **2**(2), 86–90 (2011)

4. T. Wang, S.U. Guan, K.L. Man, T.O. Ting, EEG eye state identification using incremental attribute learning with time-series classification. Math. Probl. Eng. **2014**, 365101 (2014)
5. S. Gorrab, F.B. Rejab, IK-prototypes: incremental mixed attribute learning based on k-prototypes algorithm, a new method, in *International Conference on Intelligent Systems Design and Applications* (Springer, Cham, 2020), pp. 880–890
6. S. Beniwal, J. Arora, Classification and feature selection techniques in data mining. Int. J. Eng. Res. Technol. **1**(6), 1–6 (2012)
7. R. Liu, R. Rallo, Y. Cohen, Unsupervised feature selection using incremental least squares. Int. J. Inform. Technol. Decis. Making **10**(6), 967–987 (2011)
8. V. Chaurasia, S. Pal, Stacking-based ensemble framework and feature selection technique for the detection of breast cancer. SN Comput. Sci. **2**(2), 1–13 (2021)
9. E. Hancer, A new multi-objective differential evolution approach for simultaneous clustering and feature selection. Eng. Appl. Artif. Intell. **87**, 103307 (2020)
10. S. Saha, S. Rajasekaran, R. Ramprasad, Novel randomized feature selection algorithms. Int. J. Found. Comput. Sci. **26**(3), 321–341 (2015)
11. Z. Huang, Extensions to the k-means algorithm for clustering large data sets with categorical values. Data Mining Knowl. Discovery **2**(3), 283–304 (1998)
12. Z. Huang, A fast clustering algorithm to cluster very large categorical data sets in data mining. DMKD **3**(8), 34–39 (1997)
13. J. MacQueen, Some methods for classification and analysis of multivariate observations, in *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, No. 14. (1967), pp. 281–297
14. K. Balaji, K. Lavanya, Clustering algorithms for mixed datasets: a review. Int. J. Pure Appl. Math. **18**(7), 547–56 (2018)
15. S.A. Alasadi, W.S. Bhaya, Review of data preprocessing techniques in data mining. J. Eng. Appl. Sci. **12**(16), 4102–4107 (2017)
16. S. Ramírez-Gallego, B. Krawczyk, S. García, M. Woźniak, F. Herrera, A survey on data preprocessing for data stream mining: current status and future directions. Neurocomputing **239**, 39–57 (2017)
17. D. Singh, B. Singh, Hybridization of feature selection and feature weighting for high dimensional data. Appl. Intell. **49**(4), 1580–1596 (2019)
18. G. Krempl, I. Žliobaite, D. Brzeziński, E. Hüllermeier, M. Last, V. Lemaire, T. Noack, A. Shaker, S. Sievi, M. Spiliopoulou, J. Stefanowski, Open challenges for data stream mining research. ACM SIGKDD Explorations Newsletter **16**(1), 1–10 (2014)
19. V. Karunakaran, V. Rajasekar, S. Joseph, Exploring a filter and wrapper feature selection techniques in machine learning, in *Computational Vision and Bio-Inspired Computing* (Springer, Singapore, 2021), pp. 497–506
20. M. Pal, G.M. Foody, Feature selection for classification of hyperspectral data by SVM. IEEE Trans. Geosci. Remote Sensing **48**(5), 2297–2307 (2010)
21. N. Almusallam, Z. Tari, J. Chan, A. Fahad, A. Alabdulatif, M. Al-Naeem, Towards an unsupervised feature selection method for effective dynamic features. IEEE Access **9**, 77149–77163 (2021)
22. C. Ounali, F. Ben Rejab, K. Nouira Ferchichi, Incremental algorithm based on split technique, in *International Conference on Intelligent Systems Design and Applications*. (Springer, Cham, 2018), pp. 567–576
23. D.L. Davies, D.W. Bouldin, A cluster separation measure. IEEE Trans. Pattern Anal. Mach. Intell. **2**, 224–227 (1979)
24. T. Caliński, J. Harabasz, A dendrite method for cluster analysis. Commun. Statistics-Theory Methods **3**(1), 1–27 (1974)
25. A. Asuncion, D. Newman, UCI machine learning repository (2007)
26. J. Vanschoren, J.N. Van Rijn, B. Bischl, L. Torgo, OpenML: networked science in machine learning. ACM SIGKDD Explorations Newslett. **15**(2), 49–60 (2014)
27. O. Arbelaitz, I. Gurrutxaga, J. Muguerza, J.M. Pérez, I. Perona, An extensive comparative study of cluster validity indices. Pattern Recogn. **46**(1), 243–256 (2013)

Chapter 10
Applications of Industry 4.0 on Saudi Supply Chain Management: Technologies, Opportunities, and Challenges

Taha M. Mohamed, Abdulaziz Alharbi, Ibrahim Alhassan, and Sherif Kholeif

10.1 Introduction

Supply chain management (SCM) is the flow of services and goods from one location to another. It includes all processes that change raw materials into finished products for end users. SCM aims to maximize customer gain and value and makes a competitive advantage in the market. Waters [1] explains supply chain management as "series of activities and organizations that materials move through on their journey from initial suppliers to final customers." SCM is used to link warehouses, manufacturers, and suppliers. Also, transporters, retailers, and customers are involved. The aim of this connection is to ensure that the correct products/services are made ready for distribution at correct prices, locations, quantities, time, and other conditions. This will achieve the minimal costs while satisfying customer needs as well. The overall result of SCM is achieving a sustainable competitive advantage.

SCM involves the management and planning of the activities necessary for conversion, sourcing, procurement, and other logistical management activities. At the same time, SCM also includes coordination with partners, such as customers,

T. M. Mohamed (✉) · S. Kholeif
College of Business, University of Jeddah, Jeddah, Saudi Arabia

Faculty of Computers and Artificial Intelligence, Helwan University, Helwan, Egypt
e-mail: tmmohamed@uj.edu.sa; sakholif@uj.edu.sa

A. Alharbi
College of Business, University of Jeddah, Jeddah, Saudi Arabia
e-mail: aalharbi1@uj.edu.sa

I. Alhassan
Saudi Electronic University, Riyadh, Saudi Arabia
e-mail: ialhassan@seu.edu.sa

B. Alyoubi et al. (eds.), *Machine Learning and Data Analytics for Solving Business Problems*, Unsupervised and Semi-Supervised Learning,
https://doi.org/10.1007/978-3-031-18483-3_10

intermediaries, suppliers, and third-party service providers. So, SCM integrates supply and demand processes inside, and across, different companies. SCM services include activities from strategic, tactical, and also operational levels [2]. Industry 4.0 is a recent term that includes employing recent IT technologies in many fields. Industry 4.0 includes robotics, artificial intelligence, virtual reality, wearable devices, additive manufacturing, blockchain, nanotechnology, quantum computing, Internet of Things, autonomous vehicles, and many other recent technologies. One of the major fields that heavily uses Industry 4.0 technologies is SCM. For example, robotics may be used in manufacturing and packaging; blockchain may be used in security and access. Virtual reality may be used in simulating shipping. Internet of Things may be used in smart connectivity of transmitted objects and so on. SCM is merged to Industry 4.0 to introduce a so-called SCM 4.0.

The Kingdom of Saudi Arabia (KSA) has many enablers that represent key success factors for SCM and logistics industry. The KSA includes the two holy mosques in both Makkah and Madinah and has longer borders on both the Red Sea and Arabian Gulf. Additionally, the KSA has many seaports and airports such as Jeddah, Yanbu, and Dammam. All these seaports have strategic value due to their strategic position and huge shipping capabilities. For example, Jeddah airport is the gateway for the pilgrims, whereas Dammam seaport is the gateway of oil exportation. The KSA develops the Vision 2030 which includes a specific program for SCM and logistics. This program is called National Industrial Development and Logistics Program (NIDLP). The KSA Vision 2030 aims to diversify the sources of the national income which heavily relies on the logistics industry.

This chapter reviews some Industry 4.0 technologies and their adoption in SCM. The chapter is organized as follows; Section 10.1 introduces KSA National Industrial Development and Logistics Program (NIDLP). Section 10.2 introduces the different Industry 4.0 technologies and their application in SCM. Section 10.3 shows the different enablers for adopting Industry 4.0 technologies in KSA. Section 10.4 introduces the different challenges. Finally, the chapter concluded in Sect. 10.5.

10.2 Industry 4.0 and Supply Chain Management (SCM)

Industry 4.0 has been defined in [3] as a methodology that applies emerging technologies in order to transform the existing production transiting from the technological tool dominating manufacturing to digital production. It is an incorporation of technologies, including big data and cloud manufacturing, IoT, and CPS. Industry 4.0 can make the delivery of products and services faster, substantially sustainable, more efficient, and cheaper [4, 5]. In [6], the authors explored the significance of Industry 4.0 in the area of supply chain. In [7], the authors investigated the association of Industry 4.0 to sustainable development. Saudi Arabia has been identified as one of the most rapidly developing economies in the region [8]. Technologies are the central element of Industry 4.0 due to the fact that the adoption of 12 technologies facilitates the interconnection in Industry 4.0. Considering claims made in previous studies such as in [4, 5], it has been determined that the most prevailing technologies of Industry 4.0 along with their presumed contribution in supply chain management will be taken into consideration in the following subsections.

The automotive industry fulfills a major part in advancing technological development and its application. Industry 4.0 is the source of all these emergent technologies; it includes robotics, additive manufacturing, augmented reality (AR), human-machine interfaces (HMI), artificial intelligence (AI), Internet of Things (IoT), blockchains, cloud-based data storage, autonomous vehicles, drones, digital transformation, etc. Most of the previous technologies could be implemented in different industries such as engineering, production logistics, management, etc. [9]. Using Industry 4.0 in logistics and supply chains derives new capabilities in many areas such as prototyping, product design, remote control, and many other services. Genuine advantages are gained when Industry 4.0 technologies are applied. Such advantages include improved performance and delay reduction. The Industry 4.0 technologies and their applications are explained in the following section.

10.2.1 Cyber-Physical Systems

The integration of physical processes and computing is a feature of cyber-physical systems. Cyber-physical systems (CPSs) which link the physical and digital worlds are at the heart of Industry 4.0. These systems connect the physical and digital worlds in new ways, whether via the Internet or other distributed ledgers [10].

Cyber-physical systems achieve the integration of these networks via the employment of various sensors, actuators, control processing units, and communication devices. Structure-wise, control systems have two parallel networks, one physical network of linked infrastructure components and one cyber-network composed of intelligent controllers and the communication connections between them [11]. Information from the real shop floor and the virtual computational world are closely linked in a manufacturing scenario. This provides a whole new level of production control, monitoring, openness, and efficiency.

Several studies have reviewed and effectively analyzed at the influence of CPS on the supply chain. One of the primary drivers of Industry 4.0 is automated and real-time data collection, which may lead to better supply chain decision-making via the usage of CPS. Enhanced information exchange may have a substantial influence on supply chain performance, resulting in improved connection and automation [12, 13]. By permitting a high degree of integration and information flow, CPSs, for example, may provide a better understanding of varied supply chain partners' expectations and may foster collaboration and cooperation among them [11]. So, supply chain decision-making and reactivity are likely to be enhanced, leading to better product delivery and higher levels of client satisfaction.

10.2.2 Internet of Things

Internet of Things (IoT) refers to five different technologies which may include radiofrequency identification (RFID) devices, the spatially dispersed and dedicated sensors of the wireless sensor networks (WSN), cloud computing, middleware, and the IoT apps that enable M2M and C2M interactions [14]. The IoT gained popularity in the first decade of the twenty-first century and may be seen as a forerunner to Industry 4.0. Different definitions have been offered, owing primarily to the inclusion of two distinct words, Internet and things [15]. As a result, definitions are split into two distinct visions: on one side, there is an emphasis on Internet components, while, on the other, there is a focus on things components [14]. IoT allows an item to communicate with any other object at any time and from any location. As an open system, it links numerous disparate things to the Internet and constitutes a global network of interrelated elements.

IoT technologies are most typically applied in smart industry. The networked manufacturing facilities and intelligent production systems are referred to as Industry 4.0 [16]. The primary physical purpose of a light bulb, for example, is to provide localized illumination. A security system that detects human presence and changes the lighting mode in response is an example of an IT-based service. This indicates that a product may be intelligent and remotely handled by its owner's owing to IT-based digital features. It is also possible to increase the functionalities of particular items through connections to other products and overall product systems [16]. In addition, the IoT may lead to the creation of systems of systems capable of extending industrial borders and shaking competitive dynamics [17].

The Kingdom of Saudi Arabia's Vision 2030 has a major role set for the transport and logistics industries. The role of logistics and transportation management is very important for the successful development of any nation. The advances achieved in IoT technologies have made it realistically possible to have smart transport systems [18]. In Saudi Arabia, a major increase in the demand for IoT technologies has been recorded in various Saudi Arabian industries with a market estimate of more than 16 billion in 2019, growing from an estimated 4.88 billion in 2014 [19]. IoT intelligence may be incorporated into goods, processes, and supply chain infrastructures [20]. With the advent of the Internet of Things, almost every item in an IoT-enabled environment is equipped with sensors and linked to each other and to the Internet. A greater degree of connectivity may increase supply chain visibility and efficiency by offering real-time data access [21]. The Internet of Things computer-generated and real-time resource monitoring, as well as better quality control and foresight, simplify buying and sourcing. Furthermore, according to [22], the firm's usage of Internet of Things technology enables it to gather data from its suppliers, boosting the SC's ability to adapt to changing conditions. Some of the advantages may include faster lead times, improved quality, and product variety at a lower cost.

Manufacturing is also affected by the Internet of Things in terms of quality, sustainability, production planning, and scheduling all while decreasing lead times and costs, extending the life of products, and creating new income sources. The Internet of Things benefits one of South Carolina's most critical industries, delivery, which includes storage, order, and inventory management, as well as shipping and delivery [23]. As an additional benefit of IoT, devices, systems, and manufacturing processes may be linked to the Internet to allow more sophisticated forms of automation. Increased automation may result in improved productivity, efficiency, and quality control, all of which are beneficial to supply chain operations. So, we save time, space, money, and waste by minimizing our carbon footprint. However, software incompatibility between consumers and suppliers, which results in the loss of data and information, continues to be a worry for enterprises [24]. Kanban and lean manufacturing are also encouraged by the Internet of Things. Consequently, trash disposal and reverse logistics are made more convenient [22]. Businesses may benefit from Internet of Things (IoT) technologies, regardless of the challenges they face in deploying and integrating them both internally and externally [15, 24].

10.2.3 Blockchain

For the Kingdom of Saudi Arabia to be able to utilize smart IoT-based transportation and logistics systems with a reliable security level, blockchain technology is considered. Blockchain has emerged as one of the most widely accepted technologies for trusted, secure, and decentralized intelligent transportation systems [18]. According to [25], a blockchain is a decentralized, verifiable, and immutable digital record of transactions. The ledger must be maintained up to date by each peer in turn, as there is no single point of control in the network. A private-public key cryptography pair is utilized by each member to sign all transactions to ensure anonymity. The private-public key pair is unconnected to the organization's corresponding name, allowing it to be utilized independently of the organization's name. In addition, the blocks' interdependence makes them immutable. Any accepted transaction will be included in a new block, which will include the hash of previous blocks [26]. Thus, it is possible to alter it after the fact.

The blockchain is a distributed database system that employs encryption and a consensus method to preserve transactional data and other sorts of information [27]. A data structure is constructed by joining pieces together in a chain. To maintain track of a distributed ledger or list of entries known as the chain, users or participants need a network of computers. A major advantage of a corporate environment in which no one company has control is that it addresses concerns of transparency between individuals and groups whose interests are not always matched [26]. Real-time data updates may be beneficial to all stakeholders since they eliminate the need for time-consuming and error-prone reconciliation processes [28]. Consequently, everyone on the network benefits since they all have a better grasp of what's going on in the network and how to respond to it in real time. OSCM academics are very interested in it since it has the potential to deliver vast quantities of vital information to companies and supply chains. As a result, they are paying close attention to it.

Blockchain maintains transparency, efficiency, and trust by using data coding and encryption.

Blockchain has become the current fad in Fintech, with announcements of new businesses and corporate initiatives coming out virtually every day [29]. The logistics and supply chain management (SCM) community were a little slower to get on board and recognize the potential of blockchain. When it comes to create a single point of truth, blockchain is one of the most promising technologies out now. Supply chain transparency is one of the most critical and challenging areas for logistics and supply chain management (SCM) development [26]. It's no surprise that some logistics professionals believe blockchain has immense promise as a platform for economic regeneration [28]. When taken as a whole, blockchain might be considered the "holy grail" of technology [30]. However, as is typical with new technology, the excitement surrounding blockchain seems to be fueled mostly by technology vendors, consultants, and journalists. Small- and medium-sized businesses, in particular, claim to have a limited understanding of blockchain [31]. This may be explained by the technology's novelty, as well as the absence of compelling use cases that demonstrate blockchain's superiority to traditional IT solutions. Blockchain research in logistics and supply chain management is still in its early stages, and potential applications should be investigated [32, 33].

In terms of immutability, transparency, and decentralization, blockchain may deliver numerous benefits to businesses in the context of SC, especially when paired with other technologies. Starting with shipping and transportation, blockchain has the potential to reduce costs by 15–50%, especially in international transportation [34]. Through the use of blockchain technology, these documents may be digitalized, allowing each partner to regularly monitor the quality of the goods while saving money and time [25].

In addition, blockchain makes it easier to spot fake goods. A product's validity and legality may be assured by replacing paper certificates with blockchain, which displays all prior owners of the goods. Since items can be monitored throughout the SC, blockchain also ensures transparency [25]. Customers will be able to see the product's path from raw ingredients to completed items, which will increase consumer loyalty and profitability [34]. One final benefit of blockchain technology is that it decreases the likelihood of errors while boosting the efficiency with which data can be transmitted, monitored, and so on. Despite these developments, organizations continue to confront everyday issues pertaining to scalability, interoperability, security, and privacy, as well as a lack of industry standards and rules.

10.2.4 Artificial Intelligence

According to [35], one of the key technologies capable of improving communication between devices and machines, among other things, has long been artificial intelligence (AI). AI can be explained as the ability of computers to converse with people and to replicate their talents. AI is also defined as a component of software engineering that is concerned with the development of frameworks that can perform astutely and freely as people do in their daily lives [36]. Therefore, AI can assist in streamlining processes by resolving problems more quickly and accurately while also handling enormous volumes of data [37]. Rapid application of artificial intelligence has had a major impact on our way of life, and can radically change the ways in which services and products are made for us and/or how we consume them. This will result in a cascading effect on the ways we deal with such issues as employment, competitiveness, and productivity. Furthermore, artificial intelligence can revolutionize the way we innovate into the future with potentially far reaching consequences [38]. According to [39], AI can speed up and improve the precision of processes while also handling massive volumes of data.

Artificial intelligence has been acknowledged for its wide range of applications, including SCM. A smart and agile supply chain decision-making process may be facilitated by using artificial intelligence (AI). Customers are satisfied since their needed products are always delivered on schedule and in good condition, thanks to an AI system that is proactive [40]. Using AI to automate compliance reduces costs and improves a value chain network's efficiency [41]. In today's changing business environment, using artificial intelligence to increase demand forecasting abilities is critical (AI). The ability of AI-driven bots to customize client interactions makes them a powerful tool for increasing customer loyalty [42–44]. A customer service staff and echo users work together to make these bots useful for tracking down the progress of a package's arrival [45]. Warehouse operations may benefit from AI by eliminating time-consuming manual labor. Companies like Amazon and Alibaba are already using robots driven by artificial intelligence (AI) to increase supply chain productivity and efficiency [41]. Every minute and every mile counts in the supply chain, and AI employs algorithms to optimize delivery routes and shipments to help save both time and money [46]. When it comes to AI, there are several subcategories that fall under various sorts of AI technology. As there are so many terminologies and definitions of artificial intelligence, it is difficult to investigate diverse groupings, subsets, or varieties that have different views.

Machine learning (ML), big data, deep learning, and artificial neural networks (ANNs) are some of the many branches of artificial intelligence (AI). In this section, we will focus on the most important types of artificial intelligence [42–44].

10.2.4.1 Machine Learning

Applications of machine learning are spreading due to the emergence of big data, business intelligence, as well as all the applications that require workplace automation. Machine learning (ML) is a branch of artificial intelligence (AI) and computer science, which uses automated techniques for solving problems based on historical data. AI is the computational techniques and algorithms that promote smartness-like performance of machines. Such algorithms make the machine think, act, and implement tasks simulating human thinking. Samuel created the term "machine learning" (ML) in 1995 to describe the capacity of computers to learn without being explicitly taught. Machine learning is the study of how computers may learn to solve problems by acquiring information directly from data. Programming techniques that allow computers to learn from their own experiences is a primary goal in the field of machine learning (ML).

Task-specific classifications may be made in machine learning. Classification and regression are supervised learning problems. Unsupervised learning problems include clustering and association rule mining. In supervised learning, a computer program is trained using examples of known data. It is the most prevalent kind of machine learning [47]. Taking into consideration that the output is also known, this learning process aims to develop rules that connect the data from the input and output and then apply those rules to new data as it is received. The term "unsupervised learning" refers to a system that can learn on its own. There are no pre-labeled goal values in this style of learning since the correct responses are not supplied. "Learning without a teacher" is another name for this method.

Clustering is a well-known unsupervised learning activity. To classify inputs, the approach detects commonalities between the inputs. Association rules, self-organizing maps, and multidimensional scaling are complementary methods to clustering.

Recently, machine learning has gained significant attention because of its use in several distinct phases of supply chain management within the respective sector. Applications of machine learning have been recently applied in demand because of their essence in promotion and price recommendations, customized product recommendations, as well as sales forecasting. Applications of machine learning play a significant role in optimizing margins and markups along with the inventory in accurate levels of stock, warehouse, workbench, and logistics planning [48]. It has been determined that approaches of machine learning assist in assessing the substantial amount of information gathered from operations of sustainable manufacturing [49]. Additionally, the advantages of using machine learning are taken into consideration in the existing three attributes of sustainability along with some novel study scopes for practices of Industry 4.0. Artificial intelligence and machine learning contribute to revenue growth, cost reduction, and error reduction, specifically in manufacturing sectors.

10.2.4.2 Big Data Analytics

Big data analytics (BDA) has been used for increasing attention that is attained by both researchers and media [50]. The accumulation of big data has been referred to as utilizing, acquiring, and storing huge data resources that are used for capturing a huge amount of data resources by increasing value-added management. It is required that a trending new enterprise system has been placed by the global era for measuring all kinds of industrial approaches. In [51], the authors explained that big data is used for explaining the term "3Vs," named as volume, variety, and velocity. The volume recurred by big data has been produced by exceeding data that has been stored in stored for about 20 years ago. Another term, velocity, has referred to increase in the speed of data management, and variety has been included with structuring unstructured data management. BDA has maintained supply chain management which enabled quick realization for perceiving business value in an organization.

Data analytics had enabled huge impact over systematic data approach that leads to competitive advantage. Data analytics has enabled a real-time data stream that resulted in dramatic improvements that initiate problem-solving approaches for cost avoidance factors. Big data analytics has imposed a huge impact on the supply chain and risk resilience management that incurred processing of tools for occupying valuable opportunities that reach in proactive automation [52]. This has been used for improving operations and processing reliability that deliver superior qualities to be captured by the median of competitive advantage. The development of behavioral patterns has been managed through data analysis that resulted to create stream value of detailed frameworks. BDA developing pattern has captured data analysis that allows identification of operational variables [42–44]. All these competencies have shared real medium which enhanced the power of decision-making for planning through managing the supply chain process [53]. It has imposed a significant impact on maintaining logistics and supply chain processes that enabled advanced methods of data analysis. Some companies developed predictive models that are based on BDA which helped to enable requirements of the plan [54].

There are two types of scopes of application of BDA in the context of SCM [55]. BDA has been applied for improved sourcing that stable the support of SC networking sites for product designing. First, big data analysis is used for facilitating future demands that captured market trends. Companies can simply go through purchasing strategies that retain costing and reduce risk management force for better estimation [55]. BDA can settle metrics and benchmarks that can evaluate performance management force. The performance of the company must be used for delivering raw materials in correct quality and quantity management [55]. BDA has outlined the successful design based on SC management in terms of the number and location of machinery warehouses required for shipping points [56]. However, BDA has been used for identifying bottleneck points that are used in distributing works for optimizing triple trade-off barriers among cost, product, and quality differentiation. BDA has been used for developing the strategic and tactical side of SCM that is used for demand planning, inventory, production, and procurement practices. They initiate a deeper understanding for attaining production time that is considered less complicated for matching up with the demand and supply of inventory management.

In Saudi Arabia, a significant indirect effect of big data analytics (planning, procuring, manufacturing, delivering) was indicated as the mediator on process orientation (PO) and information systems programming (ISP) to improve the supply chain process in Saudi Arabian industrial organizations, as well as improve organizational effectiveness [57].

10.2.5 Cloud Technologies

Cloud computing is considered a set of dispersed computers, including servers and data centers providing the demanded resources as well as services on the Internet [58]. Cloud computing is the most crucial and ubiquitous technology. It plays an essential role in enhancing the use of superior transportation valence, broadband, portable prerequisites, and mobile devices. Cloud computing comprises of five common attributes, including measured service, fast traction, resource collection, and extensive network access [59]. In addition to this, these authors argued that cloud computing plays an efficient role in the integration of the SC with the help of resources that are dispersed geographically and could be documented faster.

A significant impact has been made by cloud computing on applications of SCM; henceforth, the adoption is presumed to grow continuously. So, spare parts management and service, logistics, sourcing, procurement, forecasting, and planning are considered to be the most significant practices, and so cloud computing can be implemented within these activities, efficiently. The integration of cloud-based networking sites can play an essential role for firms to improve levels of their service via coordination between partners of the supply chain network such as distributors, suppliers, and retailers who contribute crucially in demand forecasting [60]. Cloud computing plays a pivotal role in the management of inventory, transportation, and warehouse due to the reason that it provides logistics tracking operations in order to increase partners of the supply chain [58]. These authors further stated that operations that include global trade compliance, transportation route planning, fleet management, order processing, and replenishment planning could significantly transmit to the cloud. So, cloud technologies can achieve the significant responsiveness and visibility required in supply chain management (SCM). Also, it can help in decision-making and planning processes. In fact, this will lead to reducing delay in supply chains. They initiate a deeper understanding for attaining production time that is considered less complicated for matching up with the demand and supply of inventory management [61].

10.3 Enablers of KSA SCM 4.0

Before discussing the enabling factors of Industry 4.0 in KSA, a short description of the various Saudi Arabian industry sectors is appropriate. According to the independent Saudi government National Competitiveness Center (NCC), Saudi Arabia remains the largest economy in the MENA region and a hub that connects the three contents "Asia, Africa and Europe." Within an environment with GDP estimated at USD 1.6+ trillion in 2020, Saudi industries are in four main categories: agriculture at 2.56%, manufacturing at 12.97%, industry at 41.36%, and services at 56.24%. A total labor force of nearly 14.5 million workers in 2020, with services industries having the greater employment share at nearly 73% of the total labor force. The information technology industry comes in third place in Saudi Arabia, superseded by financial services and raw materials in first and second place, respectively.

The KSA Vision 2030 focuses on enhancement of regulations and legislations in addition to social and economic reforms. This is expected to lead to significant achievements on the national economy. The vision also aims to diversify the sources of the national economy. This will help in attracting foreign investments in many sectors especially logistics and supply chain. Consequently, a dedicated program for logistics and supply chain industry is developed as a part of Vision 2030. This program is called the National Industrial Development and Logistics Program (NIDLP). NIDLP aims to transform KSA into a global logistics hub leading in SCM. This will generate many job opportunities and makes optimal usage of resources. NIDLP targets empowering citizens, engaging private sector, and making effective initiatives execution. The overall result is to excellence in SCM. On the same time, NIDLP aims to maximize energy and mining sectors in KSA by using Industry 4.0 technologies. NIDLP also focuses on strengthening enablers in all NIDLP sectors. Such enablers include financial enablement, infrastructure development, and reforming laws and regulations. NIDLP realizes that making competitive advantages is a key driver of economic transformation and investment attraction. As a result, major regulatory reforms have been made to align with KSA Vision 2030 [62].

There are many enablers when applying SCM on KSA. Information is one of the key enablers and drivers for SCM. Information allows other SC drivers working together aiming to create an integrated, coordinated supply chain. Also, information is required in supply chain processes to carry out transactions, and helps managers make necessary decisions. Also, optimal production plans could be put based on demand and supply available information [63].

On the contrary, in the absence of information, managers will not know customers' needs or current stock levels. Also, manufacturing and shipping will not operate properly.

Regarding KSA, information is almost available on different KSA portals. There is a dedicated website for each ministry and authority in KSA. Having access to big data and achieving integration between factories is relatively easy for the Kingdom because of the high coordination between industrial clusters. Moreover, an excellent IT infrastructure exists with high-speed Internet connections either fiber optics or fifth-generation mobile connectivity. Internet penetration is almost 100% of the population. Also, availability of laptops and smart phones is almost with all population segments. In addition, the KSA possesses significant potential and resources to compete in SCM sector. The demographic composition KSA includes a large percentage of young people aiming to employment in this sector. Also, 67% of Saudis are aged under 34 years old which contributes on the workforce power of the kingdom. Those young people have great interest in advanced technology. This will increase production and competitiveness. The KSA financial resources allow the Kingdom to acquire an advanced position worldwide in SCM.

Regarding the legislative side, NIDLP works on reforming the legislative environment by developing legal frameworks necessary for adopting SCM 4.0 in KSA. NIDLP also targets improving the human resources skills on this sector. As a result, this will foster the creation of new jobs on SCM which returns on the progress and welfare of citizens. One of the great enablers is the other related programs of the KSA's Vision 2030. Such programs aim to support entrepreneurship and innovation. This will attract foreign companies working on SCM 4.0 to invest in the KSA. The strategic location of the KSA is a nice story. The KSA is strategically located at the crossroads of three continents (Asia, Africa, and Europe). The KSA has access to the most important water ways which provides a significant competitive advantage as a global logistical gateway for global market.

Additionally, the KSA is the largest economy in the Arab World. It is also the second largest in the Middle East region. The gross domestic product (GDP) exceeds SAR 6.2 trillion in 2020. The powerful purchasing power of companies and individuals is a great advantage. The free movement of capital and the stability of currency strengthen the KSA's position worldwide. It is interesting to mention that the KSA has a huge availability of natural resources such as oil and other minerals [62].

10.4 Challenges of KSA SCM 4.0

Despite the presence of several SCM 4.0 enablers in the KSA, there are some challenging issues. These challenges could be classified into technical, financial, environmental, legal, and sociocultural challenges [64]. Financial challenges include some capital attraction limitations from private and governmental sectors. Also, there are difficulties in obtaining loans from commercial banks. However, the Saudi Industrial Development Fund has launched new financing programs and considered the program sectors as strategic pathways; the Fund is still in the process of funding legislation. Solving this challenge is expected to foster the logistics and supply chain industry in KSA.

Another challenge is the exports fluctuation returns joined with the oil price fluctuations. In fact, this issue affects the national long-term plans in SCM, and other fields. The non-oil exports currently represent one third of the Kingdom's total exports, growing 7% annually. This may mitigate the risks associated with oil price fluctuations. Another challenge is the competitive value weakness in the global market. To deal with this issue, the KSA's Vision 2030 aims to increase SME contribution to the Kingdom's economy from 20% to 35%. In fact, SCM 4.0 is essential to achieve this objective.

From an educational perspective, there is a limited availability of educational disciplines and training programs on SCM. Technical and vocational training programs directly serve NIDLP sectors. To deal with this challenge, some Saudi universities have developed new disciplines and colleges on SCM. As an example, the University of Jeddah (UJ) chooses SCM as a concentration domain. Additionally, UJ introduces two specialized bachelor and master programs in SCM. Also, it creates the Applied College which is the first specialized college in SCM and logistics sciences covering SCM 4.0 as well.

Technical wise, reducing product's life cycle and introducing competitive prices are challenges. Globally, this challenge has been overcome by disruptive technologies. So, the solution is to invest in research and development in order to satisfy the required competitive advantages.

It is interesting that the KSA supports technical innovation and develops the national R&D and innovation system. Also, there is a shortage of job information, and lack of integrated database in this arena. The solution is to adopt SCM 4.0 technologies to overcome this challenge. Another challenge is shortage of awareness on SCM 4.0 applications. This can be solved by offering the necessary workshops and various marketing techniques.

A closely related sector is the energy sector as it is considered the strategic product of the KSA. This sector faces another challenge due to limited information about demand. This challenge causes ambiguity on the demand for oil exportation which in turn increases delivery costs. Another issue is delivery variation in gas consumption between different yearly seasons [62].

Finally, the current and emerging challenges such as COVID-19 and worldwide instability are big challenges to the SCM sector. These challenges affect the whole SCM industry starting from planning, production, shipment, and delivery. These challenges lead to increased supply chain costs. However, regarding COVID-19 in KSA, the SCM sector is beginning to recover through community immunization obtained by developed vaccines. However, the emerging instability and wars worldwide still represent challenging issues to the SCM sector.

10.5 Conclusion

This chapter presents an overview of the various Industry 4.0 technologies and its applications on the supply chain management (SCM). Merging the two fields is called SCM 4.0. The Kingdom of Saudi Arabia (KSA) develops the National Industrial Development and Logistics Program (NIDLP) as a program of Vision 2030. The program aims to foster SCM 4.0 industry to effectively contribute on the national GDP. The chapter shows that there are many enablers for adopting SCM 4.0 in the KSA. Some of the important enablers include the strategic location of the KSA, existence of the two holy mosques, large availability of oil and minerals, advanced infrastructure, and also availability of the necessary human resources.

However, adopting SCM 4.0 in the KSA still faces some challenges. Challenges may include training, loan facilitation, lack of awareness, and some other necessary regulations. In fact, the KSA plans to deal with these challenges to convert the KSA into a large logistic hub, which enables the SCM to effectively contribute to the national economy.

References

1. C.D.J. Waters, *Supply Chain Management: An Introduction to Logistics*, vol 2 (Palgrave Macmillan, New York, 2009)
2. V. Anca, Logistics and supply chain management: An overview. Stud. Bus. Econ. **14**(2), 209 (2019)
3. E. Oztemel, S. Gursev, Literature review of industry 4.0 and related technologies. J. Intell. Manuf. **31**(1), 127–182 (2020)
4. A.H. Sutawijaya, L.C. Nawangsari, What is the impact of industry 4.0 to green supply chain? J. Environ. Treat. Tech. **8**, 207–213 (2020)
5. M. Piccarozzi, B. Aquilani, C. Gatti, Industry 4.0 in management studies: A systematic literature review. Sustainability (Switzerland) **10**(10), 3821 (2018)
6. P. Dallasega, E. Rauch, C. Linder, Industry 4.0 as an enabler of proximity for construction supply chains: A systematic literature review. Comput. Ind. **99**, 205–225 (2018)
7. S.S. Kamble, A. Gunasekaran, S.A. Gawankar, Sustainable Industry 4.0 framework: A systematic literature review identifying the current trends and future perspectives. Process Saf. Environ. Prot. **117**, 408–425 (2018)
8. M. Nurunnabi, Transformation from an oil-based economy to a knowledge-based economy in Saudi Arabia: The direction of Saudi vision 2030. J. Knowl. Econ. **8**(2), 536–564 (2017)
9. A. Sassi, M.B. Ali, M. Hadini, H. Ifassiouen, S. Rifai, The relation between Industry 4.0 and Supply Chain 4.0 and the impact of their implementation on companies' performance: State of the Art. Int. J. Innov. Appl. Stud. **31**(4), 820–828 (2021)
10. C. Oberg, G. Graham, How smart cities will change supply chain management: A technical viewpoint. Prod. Plan. Control **27**(6), 529–538 (2016)
11. F. Strozzi, C. Colicchia, A. Creazza, C. Noè, Literature review on the "Smart Factory" concept using bibliometric tools. Int. J. Prod. Res. **55**(22), 6572–6591 (2017)
12. C. Blome, A. Paulraj, K. Schuetz, Supply chain collaboration and sustainability: a profile deviation analysis. Int. J. Oper. Prod. Manag. **34**(5): 639–663 (2014)
13. F. Wiengarten, A. Longoni, A nuanced view on supply chain integration: A coordinative and collaborative approach to operational and sustainability performance improvement. Supply Chain Management: An International Journal **20**(2), 139–150 (2015)
14. E. Fleisch, M. Weinberger, F. Wortmann, Business models and the internet of things, in *Interoperability and Open-Source Solutions for the Internet of Things*, (Springer, Cham, 2015), pp. 6–10
15. I. Lee, K. Lee, The Internet of Things (IoT): Applications, investments, and challenges for enterprises. Bus. Horiz. **58**(4), 431–434 (2015)
16. F. Wortmann, K. Flüchter, Internet of things. Bus. Inf. Syst. Eng. **57**(3), 221–224 (2015)
17. M.E. Porter, J.E. Heppelmann, How smart, connected products are transforming competition. Harv. Bus. Rev. **92**(11), 64–88 (2014)
18. M. Humayun, N. Jhanjhi, B. Hamid, G. Ahmed, Emerging smart logistics and transportation using IoT and Blockchain. IEEE Internet Things Mag. **3**(2), 58–62 (2020). https://doi.org/10.1109/IOTM.0001.1900097
19. M.A. Khan, M.T. Quasim, F. Algarni, A. Alharthi, Internet of things: On the opportunities, applications and open challenges in Saudi Arabia, in *2019 International Conference on Advances in the Emerging Computing Technologies (AECT)*, (2020), pp. 1–5. https://doi.org/10.1109/AECT47998.2020.9194213
20. J. Macaulay, L. Buckalew, G. Chung, *Internet of Things in Logistics. A Collaborative Report by DHL and Cisco* (DHL Trend Research and Cisco Consulting Services, DHL Customer Solutions & Innovation, 2015)
21. F. Kache, S. Seuring, Challenges and opportunities of digital information at the intersection of Big Data Analytics and supply chain management. Int. J. Oper. Prod. Manag. **37**(1), 10–36 (2017)

22. M. Ben-Daya, E. Hassini, Z. Bahroun, Internet of things and supply chain management: A literature review. Int. J. Prod. Res. **57**(3), 1–24 (2017). https://doi.org/10.1080/00207543.2017.1402140
23. Lopez Research, *Building Smarter Manufacturing with the Internet of Things (IoT). Part 2 of the IoT Series*. Lopez Research white paper, (2014). http://cdn.iotwf.com/resources/6/iot_in_manufacturing_january.pdf
24. P. Bowman, J. Ng, M. Harrison, T.S. Lopez, A. Illic, *Sensor Based Condition Monitoring*. Building Radio frequency IDentification for the Global Environment (BRIDGE) Euro RFID project, (2009), pp. 1–95
25. N. Hackius, M. Petersen, Blockchain in logistics and supply chain: Trick or treat? in *Digitalization in Supply Chain Management and Logistics: Smart and Digital Solutions for an Industry 4.0 Environment. Proceedings of the Hamburg International Conference of Logistics (HICL)*, vol. 23, (epubli GmbH, Berlin, 2017), pp. 3–18
26. S.A. Abeyratne, R.P. Monfared, Blockchain ready manufacturing supply chain using distributed ledger. Int. J. Res. Eng. Technol. **5**(9), 1–10 (2016)
27. N. Lomas, *Everledger Is Using Blockchain to Combat Fraud, Starting With Diamonds*, (2015)
28. M.J. Casey, P. Wong, Global supply chains are about to get better, thanks to Blockchain. *Harvard Business Review Home*, (2017)
29. D. Tapscott, A. Tapscott, *Blockchain Revolution*, 1st edn. (Penguin Random House, New York, 2016)
30. N. Popper, S. Lohr, Blockchain: A better way to track pork chops, bonds, bad peanut butter? *The New York Times*, (2017)
31. W. Kersten, M. Seiter, B. von See, N. Hackius, T. Maurer, *Trends and Strategies in Logistics and Supply Chain Management – Digital Transformation Opportunities* (DVV Media Group, Hamburg, 2017)
32. J.L. Zhao, S. Fan, J. Yan, Overview of business innovations and research opportunities in blockchain and introduction to the special issue. Financ. Innov. **2**(28), 1–7 (2016)
33. J. Yli-Huumo, D. Ko, S. Choi, S. Park, K. Smolander, Where is current research on blockchain technology? – A systematic review. PLoS One **11**(10), 1–27 (2016)
34. M. Dobrovnik, D.M. Herold, E. Fürst, S. Kummer, Blockchain for and in logistics: What to adopt and where to start. Logistics **2**(3), 18 (2018)
35. A.L. Guzman, S.C. Lewis, Artificial intelligence and communication: A Human–Machine Communication research agenda. New Media Soc. **22**(1), 70–86 (2020)
36. C. Dirican, The impacts of robotics, artificial intelligence on business and economics. Procedia Soc. Behav. Sci. **195**, 564–573 (2015)
37. R. Dwivedi, J. Edwards, A. Eirug, V. Galanos, Artificial Intelligence (AI): Multidisciplinary perspectives on emerging challenges, opportunities, and agenda for research, practice and policy. Int. J. Inf. Manag. **57**, 101994 (2021)
38. M.T. Nuseir, M.F. Basheer, A. Aljumah | P. Foroudi (Reviewing editor), Antecedents of entrepreneurial intentions in smart city of Neom Saudi Arabia: Does the entrepreneurial education on artificial intelligence matter? Cogent Bus. Manag. **7**, 1 (2020). https://doi.org/10.1080/23311975.2020.1825041
39. J. Amann, A. Blasimme, E. Vayena, D. Frey, V.I. Madai, Explainability for artificial intelligence in healthcare: A multidisciplinary perspective. BMC Med. Inform. Decis. Mak. **20**(1), 1–9 (2020)
40. R. Toorajipour, V. Sohrabpour, A. Nazarpour, P. Oghazi, M. Fischl, Artificial intelligence in supply chain management: A systematic literature review. J. Bus. Res. **122**, 502–517 (2021)
41. P. Treleaven, B. Batrinca, Algorithmic regulation: Automating financial compliance monitoring and regulation using AI and blockchain. J. Financ. Transform. **45**, 14–21 (2017)
42. S.F. Wamba, R. Dubey, A. Gunasekaran, S. Akter, The performance effects of big data analytics and supply chain ambidexterity: The moderating effect of environmental dynamism. Int. J. Prod. Econ. **222**, 107498 (2020)

43. S.F. Wamba, M.M. Queiroz, L. Wu, U. Sivarajah, Big data analytics-enabled sensing capability and organizational outcomes: Assessing the mediating effects of business analytics culture. Ann. Oper. Res. (2020). https://doi.org/10.1007/s10479-020-03812-4
44. S.F. Wamba, R.E. Bawack, C. Guthrie, M.M. Queiroz, K.D.A. Carillo, Are we preparing for a good AI society? A bibliometric review and research agenda. Technol. Forecast. Soc. Chang. **164**, 120482 (2020). https://doi.org/10.1016/j.techfore.2020.120482
45. M.H. Huang, R.T. Rust, Engaged to a robot? The role of AI in service. J. Serv. Res. **24**, 30 (2020). https://doi.org/10.1177/1094670520902266
46. J. Wen, L. He, F. Zhu, Swarm robotics control and communications: Imminent challenges for next generation smart logistics. IEEE Commun. Mag. **56**(7), 102–107 (2018)
47. S. Marsland, *Machine Learning: An Algorithmic Perspective* (Chapman and Hall/CRC, 2011)
48. L. Columbus, McKinsey's state of machine learning and AI, 2017. *Forbes*, (2017). Available online: https://www.forbes.com/sites/louiscolumbus/2017/07/09/mckinseys-state-of-machine-learning-and-ai-2017. Accessed on 17 Dec 2020
49. A. Jamwal, R. Agrawal, M. Sharma, A. Kumar, V. Kumar, J.A.A. Garza-Reyes, Machine learning applications for sustainable manufacturing: A bibliometric-based review for future research. J. Enterp. Inf. Manag. **35**, 566 (2021)
50. F. Provost, T. Fawcett, Data science and its relationship to big data and data driven decision making. Big Data **1**(1), 51–59 (2013)
51. A. McAfee, E. Brynjolfsson, T.H. Davenport, D.J. Patil, D. Barton, Big data: The management revolution. Harv. Bus. Rev. **90**(10), 60–68 (2012)
52. S. LaValle, E. Lesser, R. Shockley, M.S. Hopkins, N. Kruschwitz, Big data, analytics and the path from insights to value. MIT Sloan Manag. Rev. **52**(2), 21–32 (2011)
53. M. Dubarry, D. Beck, Big data training data for artificial intelligence-based Li-ion diagnosis and prognosis. J. Power Sources **479**, 228806 (2020)
54. D. Mishra, A. Gunasekaran, T. Papadopoulos, S.J. Childe, Big Data and supply chain management: A review and bibliometric analysis. Ann. Oper. Res. **270**(1), 313–336 (2018)
55. L. Wang, C.A. Alexander, Big data driven supply chain management and business administration. Am. J. Econ. Bus. Adm. **7**(2), 60–67 (2015)
56. M. Nelissen, R. Schip, *Industry 4.0: The Time to Start Is Now! Supply Chain Transformation* (Campgemini Consulting, 2014). https://www.capgeminiconsulting.com/blog/supply-chain-transformation-blog/2014/12/industry-40-thetime-to-start-is-now
57. A. Elgendy, The mediating effect of big data analysis on the process orientation and information system software to improve supply chain process in Saudi Arabian industrial organizations. Int. J. Data Netw. Sci. **5**(2), 135–142 (2021)
58. B. Sundarakani, R. Kamran, P. Maheshwari, V. Jain, Designing a hybrid cloud for a supply chain network of industry 4.0: A theoretical framework. BIJ **28**(5), 1524–1542 (2019). https://doi.org/10.1108/BIJ-04-2018-0109
59. M. Ghobakhloo, N.T. Ching, Adoption of digital technologies of smart manufacturing in SMEs. J. Ind. Inf. Integr. **16**, 100107 (2019). https://doi.org/10.1016/j.jii.2019.100107
60. R. Singh, N. Bhanot, An integrated DEMATEL-MMDE-ISM based approach for analysing the barriers of IoT implementation in the manufacturing industry. Int. J. Prod. Res. **58**(8), 2454–2476 (2020). https://doi.org/10.1080/00207543.2019.1675915
61. G. Schuh, T. Potente, C. Wesch-Potente, A.R. Weber, J.P. Prote, Collaboration mechanisms to increase productivity in the context of industry 4.0. Procedia CIRPI **19**, 51–56 (2014)
62. National Industrial Development and Logistics Program (NIDLP) delivery plan 2021–2025. https://www.vision2030.gov.sa/media/qysdyqxl/nidlp_eng.pdf
63. S. Chopra, P. Meindl, *Supply Chain Management Strategy, Planning, and Operation*, 5th edn. (Pearson, Boston, 2013)
64. F.C. Martins, A.T. Simon, R.S. Campos, Supply Chain 4.0 challenges. Gest. Prod. **27**(3), e5427 (2020). https://doi.org/10.1590/0104-530X5427-20

Index

B. Alyoubi et al. (eds.), *Machine Learning and Data Analytics for Solving Business Problems*, Unsupervised and Semi-Supervised Learning,
https://doi.org/10.1007/978-3-031-18483-3