International Journal of Automation and Computing
www.ijac.net

18(3), June 2021, 492-502
DOI: 10.1007/s11633-020-1275-7

Application of Machine Learning for Online Reputation Systems

Ahmad Alqwadri[1] Mohammad Azzeh[2] Fadi Almasalha[1]

[1]Department of Computer Science, Applied Science Private University, Amman 11931, Jordan

[2]Department of Software Engineering, Applied Science Private University, Amman 11931, Jordan

Abstract: Users on the Internet usually require venues to provide better purchasing recommendations. This can be provided by a reputation system that processes ratings to provide recommendations. The rating aggregation process is a main part of reputation systems to produce global opinions about the product quality. Naive methods that are frequently used do not consider consumer profiles in their calculations and cannot discover unfair ratings and trends emerging in new ratings. Other sophisticated rating aggregation methods that use a weighted average technique focus on one or a few aspects of consumers′ profile data. This paper proposes a new reputation system using machine learning to predict reliability of consumers from their profile. In particular, we construct a new consumer profile dataset by extracting a set of factors that have a great impact on consumer reliability, which serve as an input to machine learning algorithms. The predicted weight is then integrated with a weighted average method to compute product reputation score. The proposed model has been evaluated over three MovieLens benchmarking datasets, using 10-folds cross validation. Furthermore, the performance of the proposed model has been compared to previous published rating aggregation models. The obtained results were promising which suggest that the proposed approach could be a potential solution for reputation systems. The results of the comparison demonstrated the accuracy of our models. Finally, the proposed approach can be integrated with online recommendation systems to provide better purchasing recommendations and facilitate user experience on online shopping markets.

Keywords: Reputation system, rating aggregation, machine learning, consumer reliability, user trust.

Citation: A. Alqwadri, M. Azzeh, F. Almasalha. Application of machine learning for online reputation systems. *International Journal of Automation and Computing*, vol.18, no.3, pp.492–502, 2021. http://doi.org/10.1007/s11633-020-1275-7

1 Introduction

Online rating is a common way for consumers to meet their demand when choosing products in online shopping markets[1, 2]. Consumers feel confident in expressing their opinions through ratings[3]. A reputation system is an intrinsic part of recommender systems, which can facilitate product choice decision by reflecting global opinion about products[4, 5]. The process of aggregating reputation scores for online products is important part of the reputation system because it affects the choices of consumers, thus targeting consumer′s satisfaction[6, 7]. The use of reputation systems is increasingly noticed because they are free, widely available, easy to reach, and can facilitate consumer decisions[8, 9]. The accuracy of computing product reputation scores has great influence on the consumer decision because it reflects global opinion about products. In the literature, there are too many published reputation systems[8]. Amongst them, the Naive methods (i.e., average and median of ratings) are frequent methods to compute product quality because they are simple and easy to apply without additional configuration setup. But, these methods do not take into consideration the consumers profile′s data in their process or even the popularity of the product[10]. It also cannot discover unfair ratings and trends emerging from recent consumer ratings[1, 11]. Therefore, other probabilistic and statistical methods were emerged to handle these limitations[1, 4, 9, 12, 13]. These methods showed good accuracy, but they have large space of configuration possibilities. The weighted average methods are a common alternative for computing the product reputation score, where the weights are measured from different sources such as reliability of consumers[14, 15], trust[16, 17], leniency of consumer[13] or rating age[18]. The weighted average methods initially require computing quality of consumer′s ratings before calculating the product score, and then follow a predefined threshold built by the expert. These weighted methods require sophisticated processing to obtain the reputation score of products. For example, the lenient quality (LQ)[13] model calculates the weight based on reviews leniency or strictness in providing ratings. However, the majority of cur-

Research Article
Manuscript received August 28, 2020; accepted December 30, 2020; published online March 8, 2021
Recommended by Associate Editor Xun Chen
Colored figures are available in the online version at https://link.springer.com/journal/11633
© Institute of Automation, Chinese Academy of Sciences and Springer-Verlag GmbH Germany, part of Springer Nature 2021

rent weighted methods focus on a single aspect of consumer′s ratings such as time of ratings, malicious ratings, or tendency of consumer′s ratings. Also, they measure weights to form a consumer′s profile, but they do not predict them.

In summary, we can notice that most of the previous rating aggregation models focus on a few aspects of consumer data. In addition, the machine learning algorithms have not been used intensively during the rating aggregation process to predict weights from consumer′s profile data instead of statistical methods. Therefore, this paper proposes a new weighted average approach to compute product reputation score, where weights are predicted from consumer′s profiles, using machine learning algorithms. To facilitate that, various consumer-related variables are extracted from the raw rating dataset, including:

1) Consumer tendency, which measures the user behavior in providing ratings, which is expressed by three variables (number of positive ratings, number of neutral ratings and number of negative ratings given by a consumer).

2) Consumer fluctuation, which measures the variance of consumer ratings from the ratings provided by the community.

3) Consumer experience, which is the ratio of number of ratings provided by each consumer to the total number of ratings in the system.

4) Consumer reliability, which measures the average of errors for all ratings provided by a consumer. This variable shows the reliability of the consumer in providing ratings, which measures the closeness of consumer ratings to the average products rating.

The extracted dataset represents a description of consumer′s ratings where each row represents a consumer data whereas the columns represent the extracted variables. The extracted dataset is entered to the machine learning algorithm to predict consumer reliability as a form of weight. The tendency variables in addition to fluctuation and experience variables are considered input variables while reliability is considered as an output variable. Multiple machine learning algorithms are used in this paper including, linear regression (LR), support vector regression (SVR), K-nearest neighbor (KNN) and regression tree (RT). The predicted consumer reliability is treated as a consumer weight and used with a weighted average method to compute the final product quality score. The main research questions that we address in this paper are:

RQ1. Do the extract variables have great effect on computing consumer reliability?

RQ2. Does using machine learning enable us to compute consumer reliability efficiently and thus enhance accuracy of rating aggregation?

RQ3. Which machine learning method can produce better performance?

To answer RQ1, we propose various variables that reflect consumer tendency, experience, and fluctuation in providing ratings. To answer RQ2 and RQ3, we develop four machine learning algorithms to predict consumer reliability. The accuracy of each algorithm is compared to previous reputation systems in order to determine the performance and stability of our proposed model.

The paper is structured as follows: Section 2 presents the related work. Section 3 presents the choice of learning methods. Section 4 presents the used dataset. Section 5 introduces the proposed reputation systems. Section 6 presents evaluation measures. Section 7 describes evaluation measures. Section 8 presents results, and finally Section 9 ends with a conclusion.

2 Related work

Naive methods are the most frequently used methods for computing ratings in most E-commerce systems[19, 20]. The later methods are not informative as they cannot discover recent rating trend and easily influenced by unfair ratings[1, 11]. On the other hand, the weighted average methods work more efficiently than Naive methods as they consider the consumer data in computing reputation product score. Josang et al.[9] stated that the ratings age is a good factor which can reflect the importance of old or recent ratings. They demonstrated that linear and nonlinear aging discount functions can be used through a weighted average method. This technique needs involving a professional expert to specify the unit of age (i.e., day, week, month and year). A different study suggests using the number of past transactions instead of ratings age[4]. Leberknight et al.[10] demonstrated that a higher weight must be given for recent ratings, and the reputation system should take that as well the discounting factor into account during ratings computation. They proposed a model that divides the rating into a number of non-overlapping equal subsets, and then investigated the volatility in each subset with respect to the nearest subset. Finally, the variabilities in all subsets are fused together through a discounting function that is used later to compute product score.

Other studies measured weights from consumer data such as reliability, credibility and trust of consumers. Lauw et al.[13] proposed to use leniency and strictness of consumers in providing ratings. Lenient consumers are those who frequently provide positive ratings regardless of the actual product quality. Strict consumers are those who frequently provide negative ratings regardless of the actual product quality. Jøsang and Haller[21] proposed a reputation system based on a multinomial Dirichlet probability distribution. Bharadwaj and Al-Shamri[14] developed some new variables based on the work of Jøsang and Haller[21] and using fuzzy logic to compute the trust of consumer and reputation of product. Cho et al.[22] used three variables to evaluate the reliability of the consumer, namely: consumer expertise in a specific category, con-

sumer trust, and co-orientation. These factors are fused together using either arithmetic average, harmonic average, or multiplication. In the same direction, Liu and Rezvanl[11] proposed a set of variables to address the problem of unfair ratings, which are fused together using fuzzy logic. The model has been validated using single and multiple attacks procedures. In the same direction, Rezvani and Rezvanl[1] proposed a new method to detect unfair rating using randomized algorithm. On the other hand, Abdel-Hafez et al.[4, 12] used a Beta distribution function for sparse and sense datasets to efficiently compute reputation scores for none-popular items. Azzeh et al.[6, 8] proposed two reputation systems where the first one is based on moving average and the second one is based on fuzzy logic. The first approach assumes to measure variability of data within a window, that is determined based on specified thresholds, then reflects the variability to weight. Regarding fuzzy logic, Azzeh et al.[8] proposed four factors from consumer profile that serve as input for a predefined fuzzy logic system to measure consumer influence.

Other studies focused on examining various factors that affect reputation systems[2, 23–30]. Particularly, Wu et al.[23] examined the impact of initial configuration on identifying online user reputation for the user–object bipartite networks. They employed multiple datasets from two sources: Netflix and MoviLens. The results showed that the online user′s reputations increase as users rate more and more items. Yang et al.[24] found that online ratings are subject to anchoring bias where users tend to give a low rating after low rating and high rating after high rating. Gao et al,[25] proposed group-based ranking method to evaluate user′s reputations based on their grouping behaviors. This can support the reputation system and online rating ranking. They found that their proposed model is more accurate than the correlation method in the presence of spamming attacks. Chen and Gao[26] proposed a trust-based recommendation method after integrating the information of trust relations into the resource-redistribution process. They involved a tunable parameter to scale the resources received by trusted users before the redistribution back to the objects. From these studies we can notice that none of them applied machine learning to predict user reliability from user profile data, which is the main objective of this paper.

3 Choice of machine learning algorithms

In this study, four common machine learning regression algorithms are used by reason of good and stable performance in different fields. These algorithms are linear regression (LR), support vector regression (SVR), K-nearest neighbor (KNN) and regression tree (RT). SVR is a supervised machine learning algorithm that is used to predict both linear and non-linear output. The SVR is controlled by many tuning parameters which have significant impact on its accuracy. These parameters are: 1) type of kernel function, 2) hyperplane construction method. The SVR attempts to find the optimal hyperplane (Margin), which is the maximum distance between the linear model and the “support points” close to the decision boundary. If there are no points near Margin, then the derived hyperplane can perfectly separate the data with minimum error.

RT is another supervised machine learning algorithm used to predict the continuous value. The algorithm uses Gini or Entropy variables for identifying the optimal divisible features. This process is known as binary recursive partitioning, which continuously splits data into small subsets of data and stop when the algorithm cannot divide data into more coherent groups. Finally, the average of the output in each leaf node is considered as a representative point for the group.

LR is a supervised machine learning algorithm used to predict the continuous values. There are two types of this algorithm, the simple linear regression that uses one value of input to predict the output with continuous values in a constant slope, and the multiple linear regression that uses more than one value of input to predict output. To perfectly construct a linear model, all input variables must be checked against normal distribution, in case if the input variable does not meet this condition then it is transformed to another scale using logarithmic function.

KNN is a machine learning algorithm that uses similarity measures to retrieve the closest data points to the new case. The algorithm requires determining the number of nearest neighbors (k) and weighting mechanism if necessary, before running the algorithm. The Euclidean distance is usually used as a similarity measure to identify the nearest observations.

4 Proposed rating aggregation method

To evaluate the proposed model, we used three variants of MovieLens datasets[31]. Each dataset has a different number of consumer ratings for movies. We use three types of datasets to evaluate our proposed model as shown in Table 1. The first dataset is called 100 K which consists of 943 consumers that rated 1 682 movies, and the total rating count is 100 000. The second dataset is called 1 M which consists of 6 040 consumers and 3 706 movies including 1 000 209 ratings count. The third dataset is called 10 M that consists of 71 567 consumers, 10 681 movies, and the total count of ratings is 10 000 054. As shown in Table 2, each MovieLens dataset contains the

Table 1 Description of MovieLens datasets

Dataset	Consumer count	Movie count	Total rating count
100 K	934	1 682	100 000
1 M	6 040	3 706	1 000 209
10 M	71 567	10 681	10 000 054

Table 2 MovieLens dataset summary

Attributes	Type	Description
ConsumerID	Numeric (1–6 040)	Consumer ID
MovieID	Numeric(1–3 952)	Movie ID
Rating	Numeric (1–5)	Rating of the movie
Timestamp	Numeric (Unix time)	Time of rating in second

following attributes 1) ConsumerID; 2) MovieID; 3) Rating in range 0 to 5; and finally 4) Timestamp which is measured using Unix time.

5 Proposed rating aggregation method

In this paper, we propose a new weighted average reputation aggregation model which uses machine learning as a core module to predict consumer's weight as part of computing product reputation score. The general reputation system that is used to compute the product reputation score is described in (1).

$$score_i = \frac{\sum w_j \times r_j}{\sum w_j} \tag{1}$$

where w_j is the predicted weight for consumer, j who rated product and i with rating value r_j.

The machine learning algorithms here are used to predict the weight of each consumer. To facilitate that, the raw dataset is processed from the current form to a proper input form. Therefore, a set of new variables are extracted from the raw rating dataset which describe the characteristics of each consumer. We believe that these variables can help in predicting the weight of each consumer. The extracted variables are:

1) **Consumer tendency** measures the strictness and leniency of the consumer in providing ratings. This factor can be measured by three variables (number of positive ratings (pos), number of neutral ratings (nut) and number of negative ratings (ngv)). The positive variable counts the number of positive ratings that fall in range 4–5. The neutral variable counts the number of neutral rating that equals 3. Finally, the negative variable counts the numbers of negative ratings that fall in the range 1–2.

2) **Fluctuation** measures how far the rating given by a consumer deviates from other consumers for that product. This variable can be formulated as a discounting function as shown in (2). If the consumer under investigation provided ratings close to other consumers over all shared products, then she/he gets a fluctuation value close to one. Otherwise the value will be discounted according to the number of differences.

$$fluc_i = \frac{1}{m}\sum_{k=1}^{m}\frac{1}{n}\sum_{j}^{n}\lambda^{|r_{ik}-r_{jk}|} \tag{2}$$

where n is the number of consumers. λ is the fading variable that is used as a discounting factor which in our case is $\lambda = 0.95$. m is the number of shared products between the consumer i and other consumers. r_{ik} is the rating given by consumer i for product k, while r_{jk} is the rating given by consumer j for product k.

3) **Experience** measures the ratio of the rating given by a consumer i from the total rating given by consumers in the raw dataset to see the experience of the consumer in providing ratings. The higher the number the better the experience. The reviewer's experience is very important in determining the reviewer's confidence and his ability to provide true ratings. This factor can be assessed by finding the ratio between the number of ratings provided by reviewer u_i and maximum reviewer ratings in the dataset, as shown in (3).

$$f_{i4} = \frac{|u_i|}{\max\{|u_1|, |u_2|, |u_3|, \cdots, |u_n|\}} \tag{3}$$

where $|u_i|$ is the number of ratings given by a consumer i.

4) **Reliability** measures the average of errors for all ratings given by a consumer i. For each consumer, we calculate the difference between its ratings and the product's average ratings. The obtained errors are then averaged to compute the trustworthiness. This factor will be used as consumer weight (i.e., the output variable when using machine learning methods), and we can calculate the consumer weights as shown in (4).

$$rel_i = \frac{1}{m}\sum_{k=1}^{m}|r_{ik} - \bar{r}_k| \tag{4}$$

where r_k is the average of ratings for product k.

The summary of all extracted variables is shown in Table 3. The above six variables are collected for each consumer from the raw rating dataset to form a new consumer profile dataset. The consumer profile dataset is used to learn the weight of the consumer through the machine learning algorithm as shown in Fig. 1. All variables in the dataset will be used as input, except the reliability variable will be served as output. The four employed machine learning algorithms (SVM, RT, LR and KNN) will be used to build prediction models. These models will be validated using 10-folds cross validation. In each iteration 90% of the data will be used as training while the remaining data is served as testing. This process is re-

Table 3 Description of consumer profile dataset

Variable	Type
Positive rating count (pos)	Numeric
Neutral rating count (nut)	Numeric
Negative rating count (ngv)	Numeric
Experience (exp)	Numeric
Fluctuation (fluc)	Numeric
Reliability (rel)	Numeric

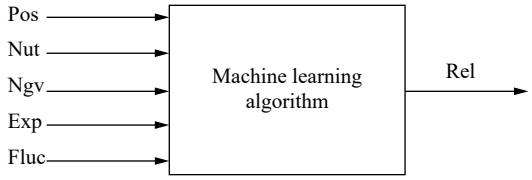


Fig. 1 Machine learning module that is used to predict consumer's reliability

peated 10 times until all data are tested. The error values are recorded in each iteration then they are averaged to obtain final error. After that, we calculate the product scores for each product as shown in (1).

6 Evaluation measures

In the literature, there is no agreed evaluation measure to validate reputation systems. However, we will use the common measures that are used by previous researchers[4, 8]. First, we use mean absolute error (MAE) that calculates how much the predicted score is close to actual ratings for the product. To find the MAE of all cases we calculate the difference between the actual rating and predicted rating as shown in (5).

$$MAE = \frac{1}{m}\sum_{k=1}^{m}\frac{\sum_{i=1}^{n}(r_{ik} - score_k)}{n} \tag{5}$$

where $score_k$ is the generated score for product k. m is the number of products in the testing data. n is the number of ratings for k-th product in the testing data.

There is another evaluation measure called the Kendall Tau coefficient which finds the correlation between two ranked list. The outcome of this analysis is a value between -1 and $+1$. If the later calculated value is close to -1 then it represents a total disagreement while it represents a total agreement if the value is closer to $+1$. If the value is close to zero, then that means there is no agreement at all. In our case, the good results are achieved when two lists have different rankings which confirms that both reputation systems are different. To investigate the sensitivity of this analysis, we compute the similarity over a specified percentage of the top ranked product. We have chosen 10%, 20%, $\cdots$, 100% as threshold points. The main objective of this analysis is that the consumers are usually concerned about top products, and to confirm that our model produces relatively different list of ranked products from other models.

7 Research methodology

As we mentioned before in Section 4 there are six new variables that have been extracted from the rating raw dataset. All variables are supposed to be normalized in order to have the same influence. We will use the min-max scaling technique to transfer all variables into scale 0 to 1. These variables form the input and output to the employed machine learning methods in order to predict consumer weight from the reliability variable. In the first step of our empirical evaluation, we divided consumer's profile datasets into groups of training and testing sets using 10-folds cross validation. In each validation step, the training dataset (90% of the entire data) is used to learn the machine learning model while the testing data (10% of the entire data) is used for consumer reliability prediction. This procedure is repeated ten times until all testing subsets are validated. The predicted weight for each consumer is stored to be used later when computing product reputation score as explained in (1). The accuracy of this procedure is assessed using MAE and Kendall Tau correlation as discussed in Section 5.

All the experiments were designed and implemented using Python. From Python we used the following libraries: Pandas to import the dataset, DataFrame to access the dataset as a data frame on python, Itertools to access all data on data frame loop, CSV library to access the dataset and to create a new CSV library file from extracted factors, NumPy to deal with numbers, Sklearn to use the machine learning algorithms and mean absolute error, mysql connector to connect and access the database on MySQL and finally SciPy to use Kendall Tau coefficient. Also, we used MySQL 7.3.12 to store the extracted variables from the original dataset, execute the SQL operations that handles the consumer weights, and to find the product scores for each product (actual product scores).

The parameter configuration for each kind of machine learning algorithm is described here. For KNN, we set nearest neighbor $k = 5$ to avoid bias, and Euclidean distance as a similarity measure. For SVR we used a radial basis function as a kernel function and gamma with auto value. For LR, we checked that variables respect the normal distribution, if not we transform it into another scale using an algorithmic function, we also set random state = 0. Finally, for RT we used the categorical/regression algorithm (CART) for building the prediction model. The constructed models are also compared to previous reputation systems that are already published in literature. Strictly speaking, we compare our model to the following previous reputation systems such as Average, Median, BetaDR[4], Bayesian[32], Dirichlet[21], IMDb, Fuzzy[11], and LQ[13].

8 Results

This section presents the results of our constructed models, in addition to the comparison with other known reputation systems mentioned before in Section 6. The MAE evaluation measure was used to assess the accuracy of reputation systems by assessing the differences between actual products scores and their predicted scores. Note that the machine learning models are used only to predict consumer weight from the reliability variable,

then the weighted average method is used to compute the final product reputation score. The results of MAE for all reputation systems are computed after calculating product reputation scores, which are presented in Table 4. We can notice that all results, over all data sets, are quite small which means that our reputation systems have capability to predict the correct weight for each consumer based on its provided ratings. Amongst them, RT surpasses other models because it has the capability to classify data into more coherent groups for which the consumer weight is predicted from the closest consumers. Surprisingly, the LR model beats KNN even though, most recommender systems favor KNN because it can identify closest consumers based on the idea of matching. However, the differences among the four machine leaning algorithms are not significant. The second important observation is the stability of results over all datasets. We can notice that RT is the superior over all datasets, followed by LR then by KNN and SVR respectively. This stability is an important factor in identifying the most accurate models.

Table 4 MAE accuracy values of the four reputation models

Dataset	LR	RT	SVR	KNN
100K	0.75	0.71	0.82	0.79
1M	0.73	0.69	0.77	0.76
10M	0.67	0.65	0.78	0.72

Table 5 shows the results of previous reputation systems from the literature. We followed the same validation procedure conducted over our models with previous reputation systems. Particularly, the Average and Median models do not require to undergoing the cross-validation procedure because they do not involve consumer weight computation. The remaining models have undergone the same 10-folds cross validation. Note here, these models measure but not predict the consumer weights from raw data. This is the main difference between our approach and previous approaches. We can observe that none of the previous models has beaten our results, therefore we can confirm that our proposed procedure is more accurate than previous model's procedures. Hence, we can notice that our models give the best accuracy in comparison with other models over sparse datasets and dense datasets. Surprisingly, the naive median method outperforms all sophisticated weighted average methods. This might confirm that the naive method is still useful in some domains, but further investigation is still needed to see if this is true for other domains. The good news from this comparison is that our ML models give higher accuracy than naive models (Average, Median), also the comparison with Bayesian and Fuzzy models gives more accuracy. In addition, our model gives higher accuracy than commercial reputation systems like IMDb and in the 1M dataset we noticed that our ML models give higher accuracy compared with other reputation systems.

To investigate the stability of all reputation systems, we rank the four machine learning models and previous reputation systems based on their MAE values as shown in Table 6. We can notice that the RT model is ranked first with high accuracy and the LQ is the lower accuracy. Notably, we can see a stable ranking for all models across all datasets despite slight rank changes for some models like Average and Bayesian.

In addition to the above analysis we performed Kendall Tau correlation to compare between two different ranked lists. The main objective of this analysis is to confirm that our model produces relatively different list of top ranked products from other models because the consumers are usually concerned about top products. The good results are obtained when two lists have different rankings which confirm that both reputation systems are different. To investigate the sensitivity of this analysis, we compute the similarity over a specified percentage of the top ranked product. We have chosen 1%, 10%, and 20%, 30%, $\cdots$, 100% as threshold points. In other words, we rank the top products based on their predicted scores, then we chose each time a threshold like 10%. For those selected products we compute Kendall Tau coefficient. This process is repeated but for other sets of thresholds (i.e., 20%, 30% to 100%). Figs. 2–5 summarize the Kendal Tau sensitivity analysis, where each figure shows a comparison between one of our reputation systems and previous published models over a specified dataset. The horizontal axis represents the percentage of top products and the vertical axis represents the Kendall Tau values. The main observation that is found from these figures is that there is a common trend in all comparisons. They begin with perfect agreement or disagreement and start declining to reach a level near to zero which indicates no similarity between two ranked lists. These results confirm that our reputation systems produce relatively different top ranked lists to our other model, which necessarily demonstrate that our models are significantly different in computing products reputation scores.

Fig. 2 shows a comparison between LR reputation model and other models over three datasets. For the 100K

Table 5 Comparison with previous using MAE evaluation measures

Dataset	Average	Median	BetaDR	Bayesian	Dirichlet	IMDb	Fuzzy	LQ
100K	0.91	0.89	0.89	0.90	0.89	0.91	0.92	1.02
1M	0.86	0.84	0.84	0.86	0.84	0.87	0.87	0.97
10M	0.84	0.81	0.83	0.84	0.84	0.86	0.85	0.96

Table 6 Ranking of models based on MAE over three datasets

Rank	100 K	1 M	10 M
1	RT	RT	RT
2	LR	LR	LR
3	KNN	KNN	KNN
4	SVR	SVR	SVR
5	Median	Median	Median
6	BetaDR	BetaDR	BetaDR
7	Dirichlet	Dirichlet	Average
8	Bayesian	Bayesian	Dirichlet
9	Average	Average	Bayesian
10	IMDb	Fuzzy	Fuzzy
11	Fuzzy	IMDb	IMDb
12	LQ	LQ	LQ

dataset as shown in Fig. 2(a), it is noticed that our model ranks 1% of top product quite similarly to Median, Fuzzy, and BetaDR models. However, the correlation degree began to decline after using top 10%. The same trend is observed for 1 M as shown in Fig. 2(b) where our model shows a relatively small similarity degree with other models, specifically Fuzzy model, at 10% which ranks top products differently from our models at various percentages of top products. Notably, our model and LQ, BetaDR and average models rank top products differently, which indicates that our model is more accurate as confirmed by MAE. For Large dataset 10 M, we can notice that our model produces a quite similar top product list to Fuzzy, BetaDR, and Bayesian when we look at top 1% and 10% of the products. Above all, we can confirm that our LR reputation system has some degree of similarity on 1% and 10% top ranked products, but this degree declined afterwards. The stability of the results over the three datasets confirm that our LR model produces significantly different results and better accuracy as confirmed by MAE.

Regarding the RT model, we can notice that our model and the three models (Fuzzy, Bayesian and BetaDR) rank only top 1% and 10% products similarly on 100 K dataset as shown in Fig. 3(a), but they decline after using 10%, which confirms that ranking lists are independent from each other. The good point here is that all similarity lines decline to reach near to zero after 20% which tell us that the RT model produces different reputation scores than the other models. For other comparisons over 1 M and 10 M datasets we observe relatively the same trend that our model ranks top products differently from other reputation systems as shown in Figs. 3(b) and 3(c). In summary, we can figure out that the ranking order of the top 10% of product list generated by our model is relatively different from other reputation systems, over three datasets.

Fig. 4 shows a comparisothree n between the KNN

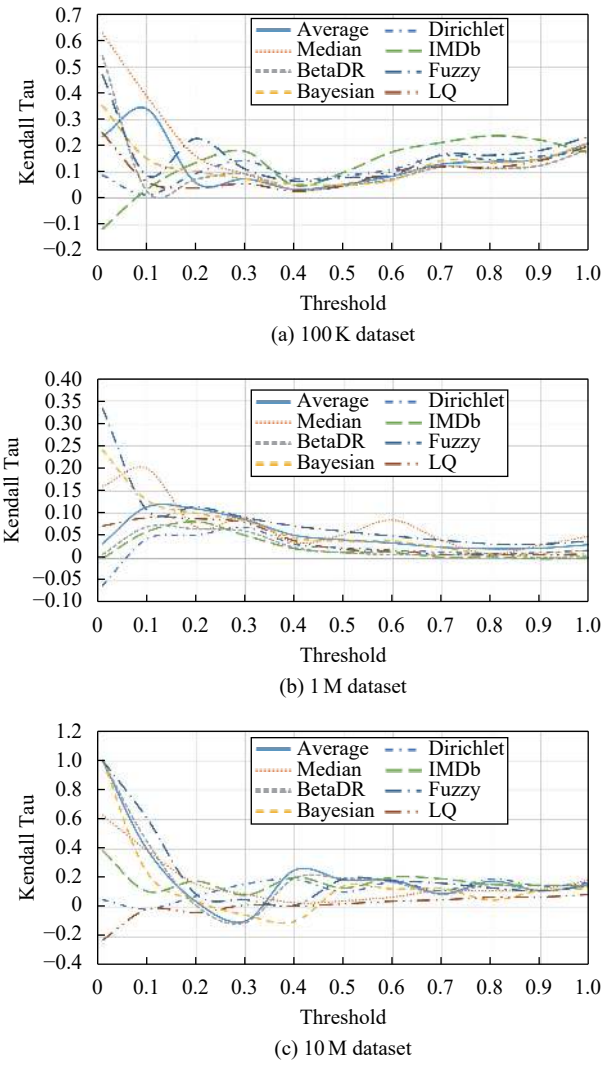
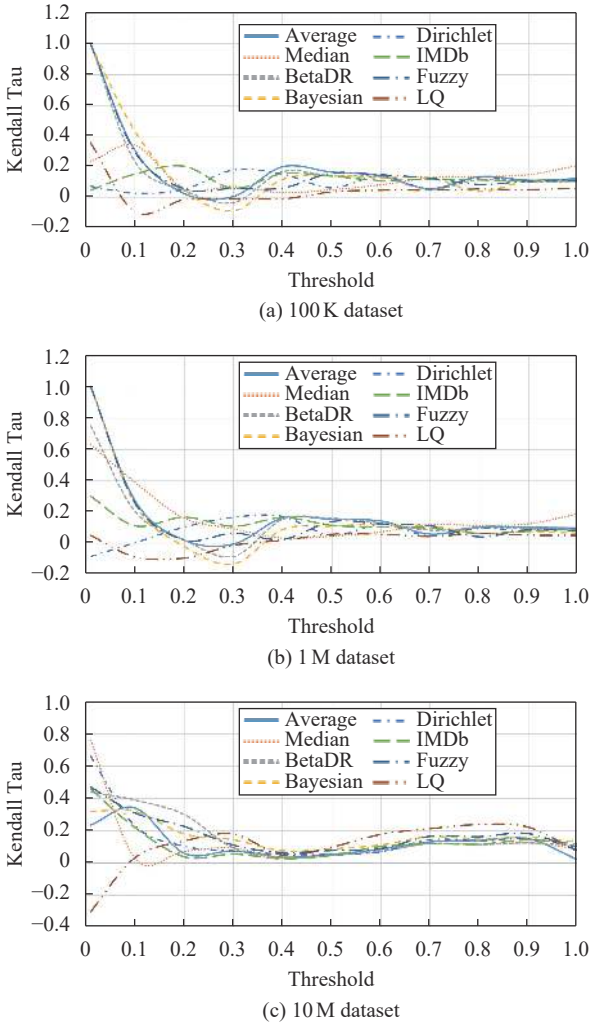


Fig. 2 Kendall Tau coefficient comparison of LR and previous reputation systems over employed datasets

reputation system and other models over datasets. For the 100 K dataset as shown in Fig. 4(a), it is noticed that our model ranks 1% and 10% of the top product relatively similarly to Median and Bayesian models. However, the correlation degree began to decline after using top 20%. The main observation here is that there is no stable relation with the Fuzzy model. The trend is slightly different over 1 M as shown in Fig. 4(b) where our model shows a relatively small similarity degree with other models at 1% and 10% which ranks top products quite similarly to BetaDR, Average and Median. Notably, our model and LQ, and IMDb models rank top products differently, which indicates that our model is more accurate as confirmed by MAE. For Large dataset 10 M we can notice that our model produces a quite similar top product list to Fuzzy, LQ and BetaDR when we look at the top 1% and 10% of the products. Finally, we can confirm that our RT reputation system has some degree of similarity on 1% and 10% top ranked products, but this degree de-

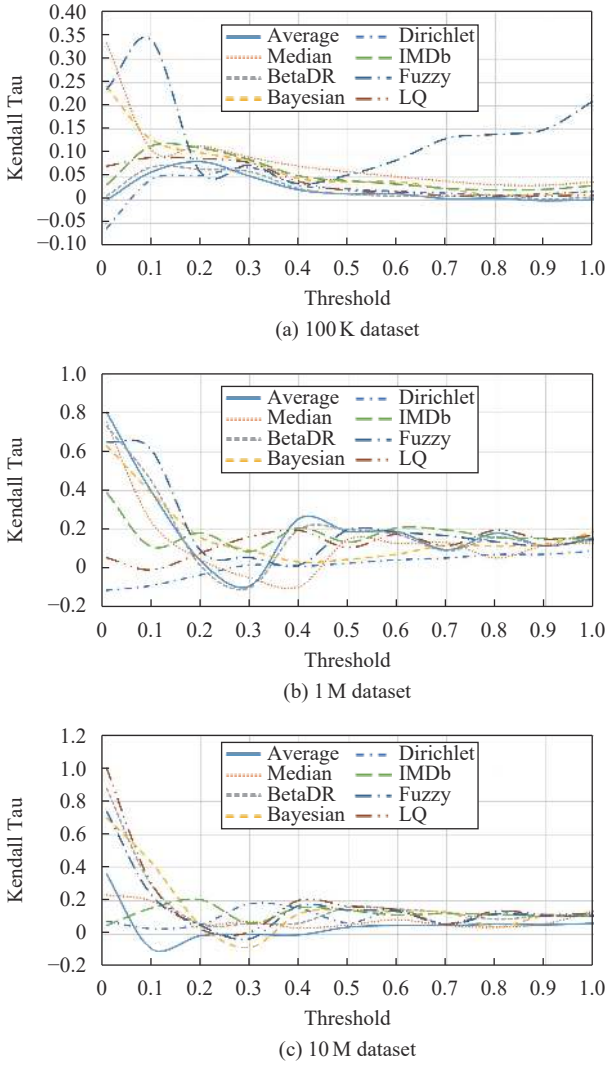


(a) 100 K dataset

(b) 1 M dataset

(c) 10 M dataset

Fig. 3 Kendall Tau coefficient comparison of RT and previous reputation systems over employed datasets



(a) 100 K dataset

(b) 1 M dataset

(c) 10 M dataset

Fig. 4 Kendall Tau coefficient comparison of KNN and previous reputation systems over employed datasets

clined afterwards. The stability of the results over the three datasets confirm that our KNN model produces significantly different results and better accuracy as confirmed by MAE.

Fig. 5 shows a comparison between the SVR reputation system and other models over three datasets. For the 100 K dataset (Fig. 5(a)), we can notice that our model ranks 1% and 10% of the top product similar to the Fuzzy model and quite similar to Median and BetaDR. However, the correlation degree began to decline after using the top 30%. For 1 M (Fig. 5(b)), the trend is similar where our model shows a relatively similar degree with Average and Bayesian at 1% and 10% which ranks top products quite similarly. Regarding the 10 M dataset (Fig. 5(c)) we can notice that our model produces quite similar top product lists to Fuzzy when we look at the top 1% and 10% of the products. Finally, we can confirm that our SVR reputation system has some degree of similarity on 1% and 10% top ranked products, but this degree declined afterwards.

Finally, we revisit the proposed research questions:

RQ1. Does the extract variables have great effect on computing consumer trust?

Answers. Yes, according to MAE and Kendall Tau results, the extracted variables have the capability to help in predicting consumer reliability based on the employed machine learning methods. Our models with four factors give high accuracy in comparison with other reputation systems that depend on one or two factors.

RQ2. Does using machine learning enables us to compute consumer trust efficiently and thus enhance accuracy of rating aggregation?

Answers. According to the MAE validation method we noticed that all results, over all data sets, are quite small which means that our reputation systems have the capability to predict the correct weight for each consumer based on its provided ratings.

RQ3. Which machine learning method can produce better performance?

Answers. According to MAE validation results we no-

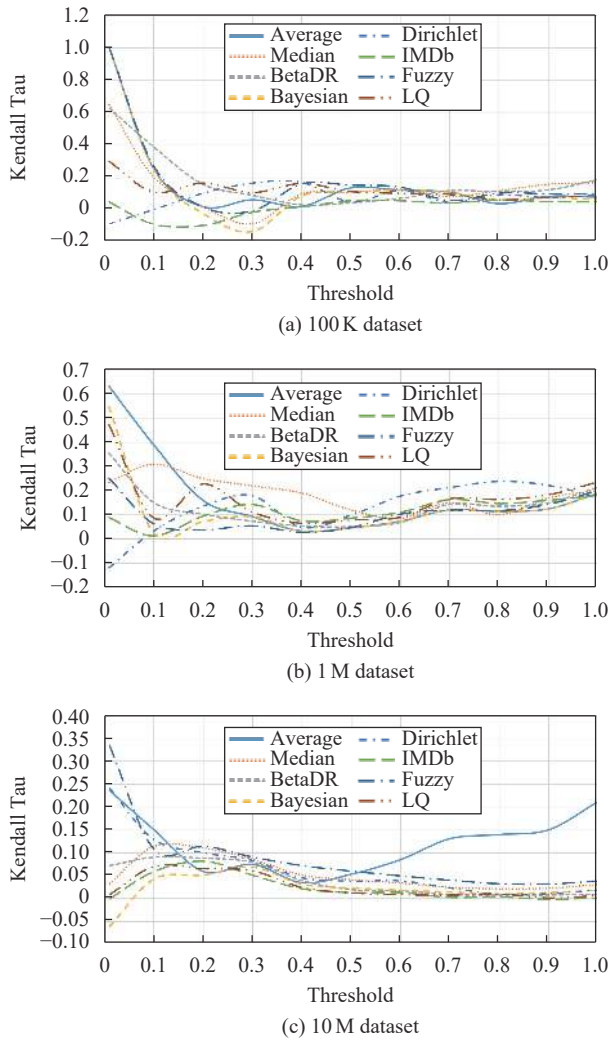


Fig. 5 Kendall Tau coefficient comparison of SVR and previous reputation systems over employed datasets

tice that the RT machine learning model gives higher accuracy over the three employed datasets.

9 Conclusions

With the increasing popularity of online shopping markets, reputation systems emerged as a solution to facilitate choices. This paper proposed a new reputation system based on a weighted average approach. The weight is predicted from consumer profile data using machine learning algorithms. In fact, four machine learning algorithms were used to predict consumer weights. These weights are used within the weighted average model to compute product reputation score. To predict weights, we constructed a new consumer profile matrix that consists of six variables: number of positive ratings, number of neutral ratings, number of negative ratings, fluctuation, experience and reliability. In this approach we focused on giving higher weights for highly trusted consumers. We believe that rating weights should relate to the reliability of ratings given by a consumer, as this reflects how end users view an item.

The constructed reputation models have been evaluated against various reputation models from the literature. The results showed that the proposed approach surpasses all previous models over MovieLens datasets using MAE evaluation measure. According to the MAE validation method, we concluded that all results, over all data sets, are quite small. In more detail, the proposed approach performs significantly better than all other models by reducing the error generated in rating predictions. Also, we noticed that the proposed approach produces a relatively different ranking for items based on the reputation scores compared with the naive and baseline methods. Besides, it provides a different ranking compared with the other sophisticated models such as LQ and Dirichlet. This indicates the significance of proposing the new reputation model based on machine learning and addresses the need to evaluate reputation models with regard to the accuracy of the ranked items list, which was performed in the second part of the experiment. According to the Kendall tau coefficient validation method, the overall results show the same trends in all figures. These results demonstrate that our proposed approach produces relatively different ranked product lists than previous models, which necessarily confirm that our models are more accurate based on MAE. These encouraging results have subsequent implications for recommender systems when they are integrated with our proposed approach. This kind of integration is supposed to provide better purchasing recommendations and facilitate user experience on online shopping markets. However, there is still a need to investigate this issue with state of art recommendation systems.

Acknowledgements

The authors are grateful to the Applied Science Private University, Amman, Jordan, for the financial support granted to cover the publication fee of this research article.

References

[1] M. Rezvani, M. Rezvani. A randomized reputation system in the presence of unfair ratings. *ACM Transactions on Management Information Systems*, vol. 11, no. 1, Article number 2, 2020. DOI: 10.1145/3384472.

[2] F. Matinfar. A computational model for measuring trust in mobile social networks using fuzzy logic. *International Journal of Automation and Computing*, vol. 17, no. 6, pp. 812–821, 2020. DOI: 10.1007/s11633-020-1232-5.

[3] S. Sharpe, D. L. Huang, T. Ravichandran. Toward an understanding of consumer feedback in the online environment: Does managerial participation help? In *Proceedings of the Academy of Marketing Science Annual Conference*, Springer, Cham, Switzerland, pp. 393–398, 2016. DOI: 10.1007/978-3-319-11815-4_110.

[4] A. Abdel-Hafez, Y. Xu. Exploiting the beta distribution-based reputation model in recommender system. In *Proceedings of the 28th Australasian Joint Conference on Artificial Intelligence*, Springer, Canberra, Australia, pp. 1–13, 2015. DOI: 10.1007/978-3-319-26350-2_1.

[5] H. A. Kurdi. HonestPeer: An enhanced EigenTrust algorithm for reputation management in P2P systems. *Journal of King Saud University-Computer and Information Sciences*, vol. 27, no. 3, pp. 315–322, 2015. DOI: 10.1016/j.jksuci.2014.10.002.

[6] M. Azzeh. Online reputation model using moving window. *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 4, pp. 508–512, 2017. DOI: 10.14569/IJACSA.2017.080467.

[7] M. Rezvani, A. Ignjatovic, E. Bertino. A provenance-aware multi-dimensional reputation system for online rating systems. *ACM Transactions on Internet Technology*, vol. 18, no. 4, Article number 55, 2018. DOI: 10.1145/3183323.

[8] M. Azzeh, M. Hijjawi, A. M. Altamimi. Online reputation model using multiple quality factors. *International Journal on Advanced Science, Engineering and Information Technology*, vol. 8, no. 6, pp. 2612–2619, 2018. DOI: 10.18517/ijaseit.8.6.6259.

[9] A. Jøsang, R. Ismail, C. Boyd. A survey of trust and reputation systems for online service provision. *Decision Support Systems*, vol. 43, no. 2, pp. 618–644, 2007. DOI: 10.1016/j.dss.2005.05.019.

[10] C. S. Leberknight, S. Sen, M. Chiang. On the volatility of online ratings: An empirical study. In *Proceedings of the 10th Workshop on E-business*, Springer, Shanghai, China, pp. 77–86, 2011. DOI: 10.1007/978-3-642-29873-8_8.

[11] S. Y. Liu, H. Yu, C. Y. Miao, A. C. Kot. A fuzzy logic based reputation model against unfair ratings. In *Proceedings of the International Conference on Autonomous Agents and Multi-agent Systems*, ACM, St. Paul, USA, pp. 821–828, 2013.

[12] A. Abdel-Hafez, Y. Xu, A. Jøsang. A normal-distribution based rating aggregation method for generating product reputations. *Web Intelligence*, vol. 13, no. 1, pp. 43–51, 2015. DOI: 10.3233/WEB-150306.

[13] H. W. Lauw, E. P. Lim, K. Wang. Quality and leniency in online collaborative rating systems. *ACM Transactions on the Web*, vol. 6, no. 1, Article number 4, 2012. DOI: 10.1145/2109205.2109209.

[14] K. K. Bharadwaj, M. Y. H. Al-Shamri. Fuzzy computational models for trust and reputation systems. *Electronic Commerce Research and Applications*, vol. 8, no. 1, pp. 37–47, 2009. DOI: 10.1016/j.elerap.2008.08.001.

[15] S. Tadelis. Reputation and feedback systems in online platform markets. *Annual Review of Economics*, vol. 8, pp. 321–340, 2016. DOI: 10.1146/annurev-economics-080315-015325.

[16] P. Bedi, R. Sharma. Trust based recommender system using ant colony for trust computation. *Expert Systems with Applications*, vol. 39, no. 1, pp. 1183–1190, 2012. DOI: 10.1016/j.eswa.2011.07.124.

[17] E. Ayday, H. Lee, F. Fekri. An iterative algorithm for trust and reputation management. In *Proceedings of IEEE International Symposium on Information Theory*, Seoul, Korea, pp. 2051–2055, 2009. DOI: 10.1109/ISIT.2009.5205441.

[18] P. Mu, M. Chang. Time-decay-based reputation method for buyers making decisions in online shopping. In *Proceedings of the 9th International Conference on Electronic Business*, Macau, China, pp. 855–863, 2009.

[19] M. Azzeh. Comparative analysis of online rating systems. *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 7, pp. 326–330, 2017. DOI: 10.14569/IJACSA.2017.080743.

[20] M. Allahbakhsh, R. L. Rafat, F. L. Rafat. A prediction-based approach for computing robust rating scores. In *the 9th International Conference on Computer and Knowledge Engineering*, IEEE, Mashhad, Iran, pp. 116–121, 2019. DOI: 10.1109/ICCKE48569.2019.8964992.

[21] A. Jøsang, J. Haller. Dirichlet reputation systems. In *Proceedings of the 2nd International Conference on Availability, Reliability and Security*, IEEE, Vienna, Austria, pp. 112–119, 2007. DOI: 10.1109/ARES.2007.71.

[22] J. Cho, K. Kwon, Y. Park. *Q*-rater: A collaborative reputation system based on source credibility theory. *Expert Systems with Applications*, vol. 36, no. 2, pp. 3751–3760, 2009. DOI: 10.1016/j.eswa.2008.02.034.

[23] Y. Y. Wu, Q. Guo, J. G. Liu, Y. C. Zhang. Effect of the initial configuration for user-object reputation systems. *Physica A: Statistical Mechanics and its Applications*, vol. 502, pp. 288–294, 2018. DOI: 10.1016/j.physa.2018.02.147.

[24] Z. M. Yang, Z. K. Zhang, T. Zhou. Anchoring bias in online voting. *EPL* (*Europhysics Letters*), vol. 100, no. 6, Article number 68002, 2012. DOI: 10.1209/0295-5075/100/68002.

[25] J. Gao, Y. W. Dong, M. S. Shang, S. M. Cai, T. Zhou. Group-based ranking method for online rating systems with spamming attacks. *EPL* (*Europhysics Letters*), vol. 110, no. 2, Article number 28003, 2015. DOI: 10.1209/0295-5075/110/28003.

[26] L. J. Chen, J. Gao. A trust-based recommendation method using network diffusion processes. *Physica A: Statistical Mechanics and its Applications*, vol. 506, pp. 679–691, 2018. DOI: 10.1016/j.physa.2018.04.089.

[27] J. Gao, T. Zhou. Evaluating user reputation in online rating systems via an iterative group-based ranking method. *Physica A: Statistical Mechanics and its Applications*, vol. 473, pp. 546–560, 2017. DOI: 10.1016/j.physa.2017.01.055.

[28] R. H. Li, J. X. Yu, X. Huang, H. Cheng. Robust reputation-based ranking on bipartite rating networks. In *Proceedings of the SIAM International Conference on Data Mining*, Anaheim, California, USA, pp. 612–623, 2012. DOI: 10.1137/1.9781611972825.53.

[29] H. Liao, A. Zeng, R. Xiao, Z. M. Ren, D. B. Chen, Y. C. Zhang. Ranking reputation and quality in online rating systems. *PLoS One*, vol. 9, no. 5, Article number e97146, 2014. DOI: 10.1371/journal.pone.0097146.

[30] K. Li, T. Xu, S. Feng, L. S. Qiao, H. W. Shen, T. Y. Lv, X. Q. Cheng, E. H. Chen. The propagation background in social networks: simulating and modeling. *International Journal of Automation and Computing*, vol. 17, no. 3, pp. 353–363, 2020. DOI: 10.1007/s11633-020-1227-2.

[31] F. M. Harper, J. A. Konstan. The MovieLens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems*, vol. 5, no. 4, Article number 19, 2015. DOI: 10.1145/2827872.

[32] A. K. Verma, R. Anil, O. P. Jain. Fuzzy logic based group maturity rating for software performance prediction. *In-*

ternational Journal of Automation and Computing, vol. 4, no. 4, pp. 406–412, 2007. DOI: 10.1007/s11633-007-0406-8.

Ahmad Alqawadri is a master student in computer science at Applied Science Private University, Jordan.

His research interests include machine learning and data mining.

E-mail: ahmad.qawadri@asu.edu.jo
ORCID iD: 0000-0002-8615-0530

Mohammad Azzeh received the M. Sc. degree in software engineering from University of the West of England, UK in 2003, the Ph. D. degree in computing from University of Bradford, UK in 2010. He is a full professor at Department of Computer Science, Faculty of Information Technology, Applied Science University, Jordan. He has published over 50 research articles in reputable journals and conferences such as *IET Software*, *Software: Evolution & Process*, *Empirical Software Engineering and Systems & Software*. He was conference chair of CSIT2016 and CSIT2018, and he is co-chair of many IT-related workshops.

His research interests include software cost estimation, empirical software engineering, data science, mining software repositories, machine learning for software engineering problems.

E-mail: m.y.azzeh@asu.edu.jo (Corresponding author)
ORCID iD: 0000-0002-0323-6452

Fadi Almasalha received the M. Sc. degree in computer science from New York Institute of Technology, USA in 2005, and the Ph. D. degree in computer science from University of Illinois, USA in 2011. He is an associate professor at Faculty of Information Technology, Applied Science Private University, Jordan. In fall of 2011, he joined Department of Computer Science at the Applied Science University, Jordan. He received his associate rank in 2016, during his appointment as the head of computer science department.

His research interests include security, Internet of things, and image processing.

E-mail: f_masalha@asu.edu.jo
ORCID iD: 0000-0002-7900-2409