MDFP: A Machine Learning Model for Detecting Fake Facebook Profiles using Supervised and Unsupervised Mining Techniques

Mohammed Basil Albayati [1], Ahmad Mousa Altamimi [2]

Department of Computer Science
Applied Science Private University
Amman, Jordan.
[1] mohammed.sabri@asu.edu.jo, [2] a_altamimi@asu.edu.jo

Abstract - **This work presents a machine learning model that utilizes a set of supervised and unsupervised mining algorithms for detecting fake Facebook profiles. Specifically, three supervised algorithms (ID3 decision tree, k-NN, and SVM) and two unsupervised algorithms (k-Means and k-Medoids) are implemented using the RapidMiner© Studio with a set of 12 behavioral and non-behavioral attributes provided in the Facebook users' profiles. To collect the related data and due to the strict privacy settings of Facebook, a special tool (CRAWLER) is developed specifically for this purpose. This ends with a dataset of 982 profiles that are used to carried out two experiments, with and without removing the missing values profiles, to determine which technique has performed best. Results showed that the supervised algorithms have best accuracy rates over the unsupervised algorithms in both experiments. In particular, ID3 algorithm outperforms other classifiers, while k-Medoids registered the lowest accuracy rate in the detection process.**

Keywords - Facebook; Fake profiles; Machine learning; Supervised techniques; Unsupervised techniques; Detection model.

I. INTRODUCTION

In recent years, social media has become ubiquitous and important for creating and sharing ideas. By design, it is an interactive Web 2.0 Internet-based application tools (e.g., Facebook, Twitter, Instagram, and alike) that facilitates electronic communications between people. In fact, these technologies change the way in which people interact with each other. It makes it easy and simple to stay connect with peoples with no boundaries in communicating even in other countries [1].

The wave of online social networks is in constant growing and expected not to stop any time soon due to the fact that these networks providing huge number of services and virtual worlds of communication with high quality, reliability, accessibility, and availability. This results in a huge registered user in these networks [2].

In this regard, Facebook is recognized as the most popular social network among the others. According to recent surveys of [3,4], the American Academy of Pediatrics showed that 84% of adolescents age in America have a Facebook accounts. Facebook administration declared that Facebook has about 2.2 billion registered users on the latest official announcements on March 2017.

Being said that, Facebook is a double edge sword, while Facebook provides a preferable platform for communication and interacting, it can be used to violate users' privacy. Fake, spamming, or cloning profiles are used by scammers to impersonate the victim identity in order to steel valuable information, defame personal reputation, or using the user's contacts for abusing. For instance, fake profiles are created based on false or fake information for various types of suspicious activities like financial fraud, hacking, or scamming [5, 6, 7].

It is worth to mention here that according to the Facebook's Statement of Rights and Responsibilities, users should provide their real and legit information once they created their profiles. Facebook urges its users to be committed to these policies and terms in order to have an experience in an environment of safety, security, and privacy [8]. However, such statement is not followed in real. Thus, in this work, we focus on the problem of detecting fake profiles in Facebook and presenting a data mining-based detection model (MDFP) to handle this problem. To this end, the Supervised and Unsupervised mining techniques have been employed based on a set of 12 numerical, behavioral, and profile's content attributes (e.g., profile picture, education information, work information, number of liked pages by that user) that are extracted from the user's friend list.

To evaluate our model, a set of three supervised techniques are used (e.g., k-NN, SVM, and ID3 Decision Tree). While, two unsupervised techniques are employed (e.g., k-Means, and k-Medoids). These techniques have been implemented using RapidMiner studio mining tool on a dataset of size 982 profiles (781 real and 201 fake). The validation method used for both types of mining techniques is k-fold cross validation method with 10 folds and two clusters for the unsupervised technique. For each mining technique two experiments have been conducted, in the first one the k-NN algorithm is used for imputing the missing values, while they are filtered out in the second experiment.

Regarding the results, we observed that the supervised algorithms (SVM, k-NN, and ID3 Decision Tree) showed the best accuracy with the values of (0.9572, 0.9145, and 0.9776), respectively, in the first experiment, and (0.9531, 0.9107, and 0.9766) in the second experiment. However, the unsupervised algorithms showed a relatively low accuracy rates with the values of (0.6731, 0.6701) in the first experiment, and (0.6808, 0.6641) in the second experiment. Further details about the results are given in section 4.

The remaining of this paper is structured as following. Section II presents the related works, the material and methodology discussed in section III, while section IV illustrates the experiments and the obtained results. In section V a discussion about the experimental results is given. Finally, the conclusion of this work is given in section VI.

II. RELATED WORKS

Reviewing the literature, many works and studies have come up with different approaches for handling the phenomena of fake profiles on online social networks, each study analyze these fake profiles from different angle based on the researchers' perspective to find a possible solution to this problem. In this regard, many works have been presented in the literature, for example the work of [9] employed a number of supervised algorithms (SVM, Naïve Bayes, and Decision Tree) and implemented using Python scripts to exploit the profile attributes (e.g., No. of friends, Education and work, Gender, and others) for detecting fake profiles on Facebook. Regarding the used dataset, the authors prepared their own dataset scrapped from one profile with 975 friends created specifically for this purpose. Authors of [10] presented a directional approach for capturing fake profiles on Facebook. They utilized the Facebook API to collect the data regarding user online activities and interactions with other users (behavioral attributes). Authors characterized these activities through an extensive set of 17 attributes like (comments, shares, tags, etc.) and applied a total of 12 supervised machine learning techniques upon datasets. The system showed an accuracy of 79%, which may not be impressive results.

Detecting Spam profiles has also considered in the literature. Authors of [11] proposed a statistical model with 14 generic features (attributes) from Facebook and Twitter dataset regarding 4 basic kinds of social interactions activities (profile interaction features, posts/ tweets, URLs, and tags / mentions). The model identifies spam profiles on Facebook and Twitter by employing three supervised algorithms (Naive Bayes, Jrip, and Decision Tree J48). Two different experiments are applied: firstly, examining the role of the whole attributes set and calculate the accuracy. And secondly, discovering the impact of each attributes to find out which one plays the key role in the classification process.

Authors of [12] presented a Social Privacy Protector (SSP) software for detecting fake profiles on Facebook, the SSP consists of three protection layers (Friends analyzer, Privacy protector, and HTTP server). The software present convenient method for restrict the users that may be suspected as fake profiles without removing it from the user's friends list. Y scanning the user's friends list and returns a credibility score. Each friend analyzed by machine learning algorithms which takes into account the strength of the connection between the user and his/her friends. The strength of each connection is based on a set of fifteen connection features such as the number of common friends, and the number of pictures and videos the user and his friend are tagged in together. Plus, eight supervised algorithms are applied (Naïve Bayes, Bagging, J48, and others). The SSP is an add-on software implemented in the Firefox browser which helps improve the user privacy with simple steps.

Another supervised machine learning approach presented in [13] for detecting and characterizing phantom (fake) profiles in online social gaming applications hosted by Facebook, the research focused on the statistical differences among a subset of features associated with genuine and phantom profiles using supervised learners as mentioned to classify fake and genuine profiles, a set of 13 attributes regarding the social activates and the game statistical information of these users is used. The classification experiment focused on Support Vector Machine algorithm. Plus, elaborating others machine learning techniques to improve some issues regarding the feature selection process. The proposed model shows promising results for detecting these types of profiles.

A framework for detecting spammers/ fake profiles using Facebook as a test case is presented in [14], by exploiting a behavioral and community-based (graph-based) attributes, including the structure of the nodes and some topological information in a machine learning approach. The framework implemented using 10 discriminative topological attributes like number of posts, number of sent/ received messages…etc. Four experiments are conducted using two datasets: Facebook dataset and Enron network (Email messages dataset). Four supervised algorithms are employed in the first and second experiments (Naïve Bayes, J48, k-NN, and Decision Tree), and J48 classifier employed in the third and fourth experiments. The results showed that the Decision tree had the highest accuracy rates and framework's detection showed promising performance.

Another example of the spam detection on twitter using a traditional machine learning classifier presented in [15]. The authors use four supervised classifiers (SVM, Naïve Bayes, k-NN, and Random Forest) applied on dataset collected from Twitter social site by using Twitter API gathering information (attributes) regarding to User-based features (No. of friends, No. of followers, and others), and Content-Based Features (No. replies/mentions, No. of hashtags, and others), collecting a total of 1000 accounts and labeled manually as spam and non-spam accounts. A number of experiments conducted in this work using Content-based attributes in the first experiment, and User-based attributes in the second. For the third experiment both types (User-based and Content-based) are employed. The experimental results showed that the Random Forest classifier had the best performance, this algorithm (Random Forest) used in additional experiments for comparing it with similar works in the literature using different attributes. The results exhibit slightly better performance.

LinkedIn site also considered in the fake profiles' detection, as the authors in [16] proposed data mining approach for detecting this type of profiles exploiting a set of 15 static/ non-behavioral attributes like: No. of Languages, Profile Summary, No. of Qualification, and others. using SVM and Neural Networks NN as a supervised classifier. The algorithms applied on limited dataset of size 74 (34 fake and 40 legitimate) profiles, collected and inspect manually due to the high restriction on the public information and the privacy policy of the LinkedIn, the collected data divided into three equal sized dataset of size 37 (20 legitimate and 17 fake) for the experiments. The both SVM and NN algorithms applied with in two experiments, the SVM showed a better performance than the NN in both experiments.

On the other side, unsupervised techniques were present in some works, like the [17] for detecting a multiple account instead of single one, where the authors investigate three methods (unsupervised learning using Katz similarity, semi-supervised learning using Katz similarity, and semi-supervised learning using graph embedding) to predict whether a pair/multiple of accounts belong to the same user on Facebook. It computes the similarity matrix to measure how closed two accounts are in the graph and predicts if they belong to the same user or not, depends on specific a threshold of this predicator. Another unsupervised approach presented by the authors of [18], where the anomaly detection model focusing on three major types of attributes: temporal, spatial, spatio-temporal features. Here, the authors combined these three types of attributes to produce a fourth one (multiple features) that aggregates these three attributes into one vector and passes it into the proposed approach. A Principal Components Analysis (PCA) technique employed for capturing anomalous behavior, where any user who does not fit the model will be flagged as fake or suspicious. The approach used real data collected from three popular networks (Facebook, Yelp, and Twitter) and achieved 66% accuracy. Authors of [19] proposed a machine learning approach for detecting spam bots in Twitter OSN through exploiting two main spam features, which are: The graph-based features (attributes) including the number of friends, number of followers and the follower's ratio (the ratio of the number of peoples following you to the number of peoples you follow). And the content-based approach which is the number of duplicated tweets, number of HTTP links, and the number of replays/ mentions.

It is important to mention here that our work is differentiated from the works presented in this section as following:

Firstly, most of the presented works employed only supervised techniques such as [9-16], while our work utilized the supervised and unsupervised mining approaches. For example, [9] proposed supervised machine learning approach (SVM, Naïve Bayes, and Decision Tree) for detecting fake profiles on Facebook using behavioral and non-behavioral attributes. [10] proposed a behavioral based approach for detecting fake profiles on Facebook using supervised machine learning algorithms (k-Nearest Neighbor, Nave Bayes, Decision Tree, and others). The work of [12] employed a set of supervised techniques (Naive-Bayes, Bagging, Random-Forest, J48, and others) for calculate the strength connection of the user and its friends detecting the profiles that impose threat on Facebook site using behavioral-based attributes. On the other hand, fewer works utilized unsupervised techniques like [17] and [18]. The work of [17] proposed for detecting multiple accounts by calculate the strong connection between these accounts (profiles). The authors of [18] used an anomaly detection technique for capturing anomalous profiles based on their social online behaviors. In contrast, our work utilized both the supervised and unsupervised learning techniques in order to detect fake Facebook profiles and using both behavioral and non-behavioral attributes (profile content information, and social online activates).

Secondly, we note that some of the presented works considered only Twitter platform. [19] for example, proposed a model for identifying spam bots by employing supervised techniques (Decision Tree, Neural Network, Support Vector Machines, and k-Nearest Neighbors). On the same vein, the work of [15] employed a number of supervised algorithms (Support Vector Machine, Naïve Bayes, k-Nearest Neighbor, and Random Forest) for detecting spam profiles. Other works as the one presented in [16] consider other platforms such as LinkedIn. However, our work considers Facebook platform.

Thirdly, most of the presented works handled the fake profiles by utilizing the behavioral-based attributes like [11], [10], [12], and [14]. While some other works used non-behavioral attributes as in [16] for detecting fake profiles on LinkedIn. Moreover, we note that few works considered both of the attribute's types (behavioral and non-behavioral) like [19], [9] and [15]. On the other hand, our work considers the two types of attributes using both supervised and unsupervised techniques to detect fake profiles on Facebook unlike [19] and [15] in which it considered the Twitter platform for detecting spam/Fake profiles using supervised techniques.

III. MATERIAL AND METHODOLOGY

In order to enable users to detect fake Facebook profiles, a machine learning model (MDFP) is developed by utilizing the data mining approach. Specifically, the supervised and unsupervised machine learning techniques (algorithms). The main objective behind the using of these algorithms is to build a model with distinguished attributes and predefining labels of known classes (Fake and Real) that is able to classify/predict a new data of unknown labels. To do so, the system firstly should collect a set of the behavioral and non-behavioral attributes from the user's friends' profiles (listed in tables 1,2, and 3).

A CRAWLER module is developed specifically for this purpose (crawls the profiles and collects their attributes). After that, the missing values are handled using the k-NN algorithm. Here, the algorithm is utilized to impute the missing values by finding k nearest neighbor of the current missing value in the dataset based on the other available information. Finally, a set of supervised and unsupervised mining algorithms are implemented using the RapidMiner data science platform to detect the fake profiles. Fig. 1 illustrate the main components of the MDFP model.

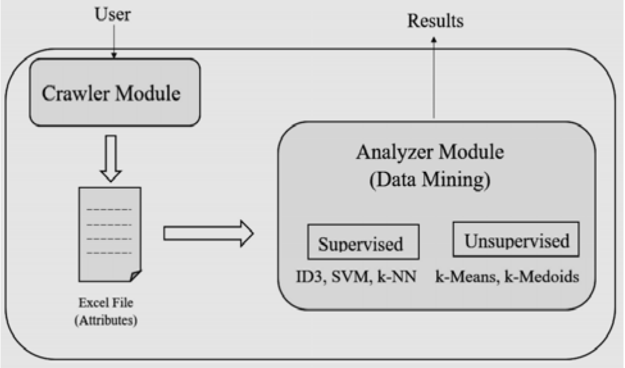


Figure 1. MDFP System Components

The MDFP model is consist of two main components: the CRAWLER and the analyzer modules. Next, a detailed description about these modules are given.

A. CRAWLER Module

A special purpose software developed for collecting the required data sets using Java scripts, PHP, html, and C# languages. Figure 2 illustrates its main components and it works as follow:

1. The user login into his/her Facebook accounts through the CRAWLER, which enables the CRAWLER to access the information for that account.
2. As a result, the CRAWLER collects the friends list (Name, Profiles' link, and FB id) and storing them in local database to manage the stored profiles.
3. After that, the CRAWLER scans each profile to collect the required attributes. This is done by searching for the class name of these attributes, where each one in the Facebook structure has a class name. The required attributes for our work appear mainly in two section on Facebook structure (*About* and *Timeline)*.
4. The collected profiles values are stored in a relational database (MySQL).
5. For the pre-processing purpose**,** a simple PHP code is built-in with the CRAWLER convert them to an applicable form for the data mining algorithm, the 12 utilized attributes in our model are pre-processed as follow:

- (Profile Picture, Work Place, Education, CheckIns, Intro., Living Place, and Family member\ Relationship) represented as Boolean or categorical value.
- Numerical attributes (No. of Likes, No. of Groups, No. of Mutual Friends, No. of Tags, and No. of Posts) represented in the model as natural N numbers. (more details will be given in the next sub-sections)

6. These values are finally stored in CSV format or Excel file to be processed by the Analyzer Module.

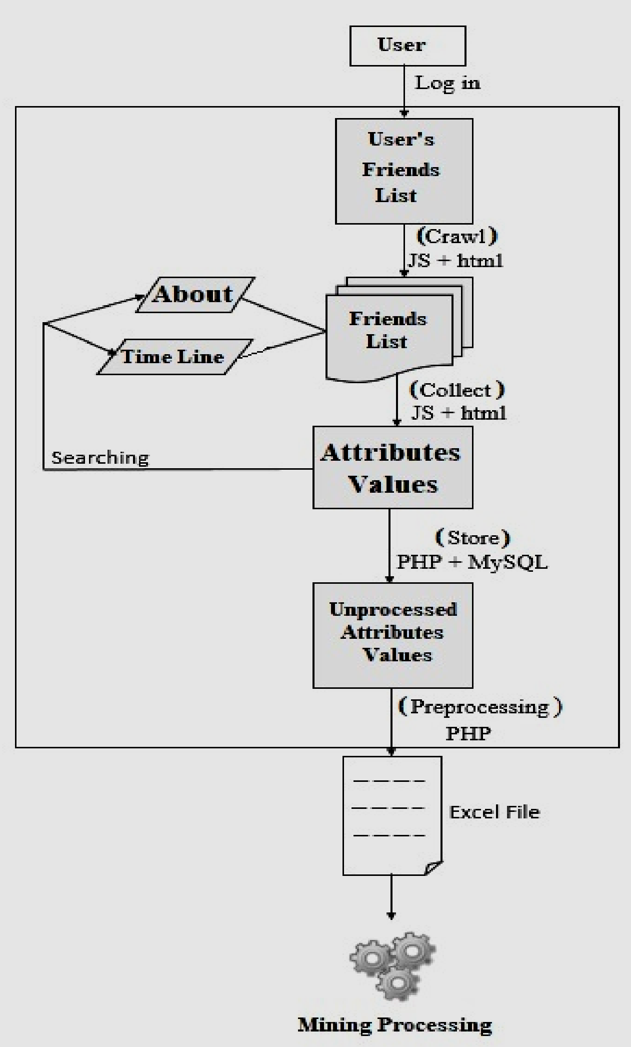


Figure 2.The CRAWLER Processes

B. Analyzer Module

The analyzer module is implemented using RapidMiner studio 8.0.1 [22]. The RapidMiner® enables us to apply a wide range of datamining algorithms (Supervised and Unsupervised). In the supervised mining, the classifier is trained with known class data (Fake and Real). However, for the unsupervised mining, statistical interestingness measures are defined for each cluster. Accordingly, profiles with similar attributes are grouped together in a same cluster, while the other profiles are grouped in a deferent one.

Further discussion about the implementation process will be presented in the next section.

C. The Employed Attributes

In this section a description discussion about each of the employed attribute will be given:

1. Profile Picture: this attribute reflects the visual identity of the profile owner. Facebook presents a set of options to add, edit, delete or change the profile picture. We utilized this attribute as one of the indicators for detecting fake profiles, which represented as categorical type (Fake/ Real) or a Boolean value (0 to represent a fake profile picture and 1 for real profile picture). Here, each user can recognize its online friends.
2. Workplace: contains the workplace information for that user. Here, categorical type categorical type (Exist/ Not Exist) or Boolean values {0, 1} are used to represent this attribute. 1 for a valid workplace and 0 for not valid workplace or an absent workplace.
3. Education: represents the last educational institute went by the user. In the same vein, categorical type (Exist/ Not Exist) or Boolean values {0, 1} are used to represent this attribute. 1 for the valid institute name, and 0 or not valid institute or an absent one.
4. Living Place: identifies the living place address (e.g. city, town, state or country) of the user. A Boolean value 1 represents a valid living place, and 0 for invalid living places or absent one or categorical type (Exist/ Not Exist).
5. Family Member/ Relationship: Facebook gives the ability to its users to share the social relationship status (e.g. Single, Engaged, Married, Divorced). MDFP model considers this attribute as a Boolean value, 1 for valid relationship status and 0 otherwise. Or again categorical type (Exist/ Not Exist).
6. Check-In: identifies the user's social activities in categorical fashion, or Boolean as 1 or 'Exist' for accounts that have registered at least one check-in, and 0 or 'Not Exist' for those who did not register any check-In.

TABLE 1. NEW EMPLOYED ATTRIBUTES

Attribute	Type
Profile Picture	Non-Behavioural \ Profile Content Attributes
Living Place	Non-Behavioural \ Profile Content Attributes
Check-In	Non-Behavioural \ Profile Content Attributes
Family Member/ Relationship	Non-Behavioural \ Profile Content Attributes
No. of Groups	Behavioural \ Numerical Attributes

7. Introduction "Bio.": shows the biography of a user and gives a clue about the real identity. A valid Bio. an attribute is represented as 1 or 'Exist', and 0 or Not 'Exist' otherwise. In our work, the Bio. that contains at least five words is considered as a valid attribute.
8. No. Mutual Friends: this attribute shows in numbers if the target (tested) profile has mutual friends with model's user, hence gives more incredibility to the target profile, this attribute is represented in the model as natural N numbers.

TABLE 2. PREVIOUSLY USED ATTRIBUTES

Attribute	Type	References
Education, Workplace, Introduction "Bio."	Non-Behavioural \ Profile Content Attributes	[12]
No. of tags	Behavioural \ Numerical	[13] [12]
No. of Mutual Friends	Behavioural \ Numerical Attributes	[15]
No. of posts (Wall Activities)	Behavioural \ Numerical Attributes	[12] [17]
No. of Pages	Behavioural \ Numerical Attributes	[14] [21]

TABLE 3. ATTRIBUTES DESCRIPTIONS AND INTUATIVE JUSTIFICATION

Attribute	Description	Justification
Profile Picture	Visual identification of the user.	Real users use their real pictures more often than fake users.
Work place	Workplace or job title's information,	Real users more often use their real workplace information than fake users.
Education	Attended (school, college, university…etc.) information.	Real users mentioned their education information in their Facebook profiles more often than fake users.
Living Place	Living place address (city, town, state…etc.) information.	Real users more often use their real living place information than fake users
Check-In	Information for announcing user location.	Real users check into places in their Facebook's profiles more often than fake users.
No. of Posts	Social online activities shared on Facebook	Real users have more online activities than fake users.
No. of Tags	Identify the user by someone else on his/ her wall.	Real Users tagged more often than fake users.
Introduction "Bio."	Introduction information about Facebook's users.	Real users are more often write something about themselves than fake users.
No. of Mutual Friends	Number of the people who are Facebook friends with both users and the target profiles.	Real users have more mutual friends with target profile than fake users, hence gives profile more incredibility.
No. of Pages	Number of pages liked.	Real users usually liked more pages than fake users.
No. of Groups	Number of groups joined.	Real users usually join groups more than fake users.
Family\ Relationship	Social relation Information\Status	Real users share their real social relation status than fake users.

9. No. of Pages: number of pages liked by the user is considered in the model as natural N number

10. No. of Groups: indicate the number of groups joined by the user, again represented as natural N numbers.
11. No. of Wall Posts: this attribute indicates the number of posts (photos, videos, links, text posts, shared, …etc.) that have been posted on his/ her profile's wall (timeline), also this attribute is represented as natural N number.
12. No. of Tags: this attribute indicates the numbers of posts that have been linked (Tagged) to the target user and showed on his/ her timeline.

D. The Machine Learning Model for Detecting Fake Facebook Profiles (MDFP)

In summary, the core idea of the proposed model for detecting the fake profiles is to design a machine learning model of two classes (Fake and Real) fed with a set of attributes extracted from Facebook's profiles (training set). This set is used to identify specific rules and relationship among the attributes of the training set in order to give the corresponding class by applying the employed machine learning algorithms to making the necessary computational analysis and measuring their performance. A set of 12 attributes is utilized, including five new attributes presented in this work (Profile Picture, Living Place, Check-In, Family Members, and No. of Groups), and (Education, Workplace, Introduction "Bio.", No. Tags, No. of Mutual Friends, No. of Posts, No. of Pages) which were used in the literature, MDFP model consist of two main parts: the CRAWLER which crawl the target profiles and collect the required attributes, then prepared the collected data before the second part (the analyzer) apply the (Supervised and Unsupervised) algorithms for the final outcome as fig. 1 shows.

IV. EXPERIMENTS AND RESULTS

In this section presets a discussion about the evaluation process of the MDFP detection's performance, along with the used algorithms, methods, and the validation metrics.

A. Dataset Description

The CRAWLER launched on three profiles collecting a total of 906, 104 were excluded as they were (irrelevant, duplicated, mutual friends, deactivated, or deactivated) profiles. 19 profiles founded fake, so they were considered in the dataset as fake profiles. For the fake profiles, we purchased a total of 250 profiles online, filtering them to 182 profiles, as some of the profiles were deactivated, irrelevant, blocked or after a while banned from Facebook. This process was ended with 982 profiles (781 re al, and 201 fake). 86 (61 real and 25 fake) out of 982 were suffering missing values in some of their attributes, a k-NN model for the data imputation was employed for handling the missing values.

Finally, manual labelling was applied to the collected dataset to addresses them as fake or real profiles for training and testing purposes.

B. Performance Metrics

A group of common metrics are applied in the validation process:

- Accuracy: Measure the performance of the detection model

Accuracy = (correct predictions) / (total examples).

- Recall: true positive rate

Recall = (true positive predictions) / (positive examples),

- Precision: Measure the probability that the positive predication is correct

Precision = (true positive predictions) / (positive predictions).

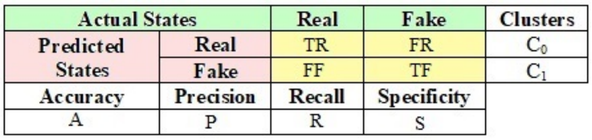
- Specificity: true negative rates

Specificity = (true negative predictions) / (negative examples).

C. The Experiments

The mining techniques (supervised and unsupervised) of this work are tested against two experiments, in the first experiment the complete dataset including missing values profiles is considered, while in the second, the profiles with missing values are excluded. finally, metrics for the validation process are calculated.

As a baseline in each experiment, a cross-validation method with 10 folds is used, and k=2 for the clustering algorithms as there are two class labels (Real and Fake). A discussion in more details will be presented in the next the subsections.



Actual States		Real	Fake	Clusters
Predicted States	Real	TR	FR	C_0
	Fake	FF	TF	C_1
Accuracy	Precision	Recall	Specificity	
A	P	R	S	

Figure 3. Notations

Before discussing the experiments and the obtained results, it is important to explain a few notations appeared in this section’s tables. As shown in Fig. 3, the green fields (*Actual States: Real | Fake*) represents the actual class cases, while the red (*Predicted States: Real | Fake*) represents the predicted class cases. The yellow area gives a visual representation of the model performance, where the row cells represents the case in a predicted class while each column represents cases in the actual class. The notations are:

- TR: True Real (No. of the real profiles that correctly predicted as real class).
- TF: True Fake (No. of the fake profiles that correctly predicted as fake class).

- FF: False Fake (No. of the real profiles that wrongly predicted as fake profiles).
- FR: False Real (No. of the fake profiles that wrongly predicted as real profiles).

The white area (C*lusters: C_0 | C_1)* represents unsupervised clustering cases, where C_0 indicates the cluster that contain the profiles of the real class, while C_1 for the fake class profiles.

The remaining fields represents the values of the performance measurements of the applied algorithms as follow:

- A: algorithm's Accuracy.
- P: algorithm's Precision.
- R: algorithm's Recall.
- S: algorithm's Specificity.

C1. Supervised Experiments: Where two experimnts are conducted:

a) Handling the Missing Values using the k-NN Estimator:

In this case, the algorithms are applied with the complete 982 (including the missing attributes profiles). However, the k-NN estimator is utilized for data imputation where the missing value is estimated based on the k most similar record in the dataset [20].

As mentioned, 10-fold cross validation method is used for the performance evaluation process.

- The ID3 decision tree is the first algorithm to assessed. Table 4 shows the confusion and the performance metrics of this algorithm, SVM achieved an accuracy of 0.9776.

TABLE 4. ID3 WITH THE K-NN ESTIMATOR

Actual States		Real	Fake
Predicted States	Real	769	10
	Fake	12	191
Accuracy	Precision	Recall	Specificity
0.9776	0.9872	0.9846	0.9502

- The second algorithm applied is the SVM. Table 5 shows performance of this algorithm.

TABLE 5. SVM WITH THE K-NN ESTIMATOR

Actual States		Real	Fake
Predicted States	Real	756	17
	Fake	25	184
Accuracy	Precision	Recall	Specificity
0.9572	0.9780	0.9680	0.9154

- Last supervised algorithm employed in this experiment is the k-NN, applied with k=3, the accuracy and other evaluation metrics are shown in table 6.

TABLE 6. K-NN WITH THE K-NN ESTIMATOR

Actual States		Real	Fake
Predicted States	Real	734	37
	Fake	47	164
Accuracy	Precision	Recall	Specificity
0.9145	0.9520	0.9398	0.8159

b) Removing the Missing Values Profiles:

In the second case, the profiles with missing attributes are excluded. As a result, a total of 86 profiles are removed, leaving 896 to be considered. The main purpose of this experiment is to eliminate any factor that could affect the model's performance, because the data imputation for the missing values is an estimation process by the k-NN.

All the algorithms are applied by following the same methods in the first experiment, the results showed tables 7, 8, and 9.

TABLE 7. ID3 WITH FILTERING OPERATOR

Actual States		Real	Fake
Predicted States	Real	709	10
	Fake	11	166
Accuracy	Precision	Recall	Specificity
0.9766	0.9861	0.9847	0.9432

TABLE 8. SVM WITH FILTERING OPERATOR

Actual States		Real	Fake
Predicted States	Real	**693**	**15**
	Fake	27	161
Accuracy	Precision	Recall	Specificity
0.9531	0.9788	0.9625	0.9148

TABLE 9. K-NN WITH FILTERING OPERATOR

Actual States		Real	Fake
Predicted States	Real	674	34
	Fake	46	142
Accuracy	Precision	Recall	Specificity
0.9107	0.9520	0.9361	0.8068

C2. Unsupervised Experiments: Following the same vein, the unsupervised algorithms applied t based on the two cases as follows:

a) Handling the Missing Values Using the k-NN Estimator:

Within As mentioned earlier, k-Means and k-Medoids employed in our model as unsupervised (clustering) techniques. It important to mention here that the training data for the unsupervised learning in the MDFP model is unlabeled dataset, this makes evaluation problematic because there is nothing to which the model's results can be meaningfully compared. So, there is no straightforward way to evaluate the accuracy of the applied algorithm.

To evaluate the clustering techniques, we form an evaluating model using RapidMiner's special operators that could be exploited in flexible ways, such as 'Map Clustering on Labels', this operator maps between clustering and prediction processes by adjusting the given clusters with class labels, this allows us to adjust the dataset and evaluate our model. Both algorithms are applied with (k=2) or two clusters C0, C1. Where, C0 represents the real profiles, while C1 represents the fake profiles. However, the k-Means algorithm partitioned the dataset and showed the accuracy of 0.6731, and the other performance metrics illustrated in table 10.

TABLE 10. K-MEANS WITH THE K-NN ESTIMATOR

Actual States		**Real**	**Fake**	**Clusters**
Predicted States	**Real**	661	201	C_0
	Fake	120	0	C_1
Accuracy	**Precision**	**Recall**	**Specificity**	
0.6731	0.7668	0.8464	0.0000	

Similar to k-Means, k-Medoids followed the same vein. And showed the accuracy of 0.6701 , as shown in table 11 with remining performance metrics.

TABLE 11. K-MEDOIDS WITH THE K-NN ESTIMATOR

Actual States		**Real**	**Fake**	**Clusters**
Predicted States	**Real**	526	69	C_0
	Fake	255	132	C_1
Accuracy	**Precision**	**Recall**	**Specificity**	
0.6701	0.8840	0.6735	0.6567	

b) Removing the Missing Values Profiles:

In this experiment, the same clustering algorithms, methods, and operators are used. Plus, employing a filtering operator to remove the missing values profiles. The both k-Means and k-Medoids applied with (k=2). Table 12 and 13 shows the obtained results of this experiment.

TABLE 12. K-MEAN WITH FILTERING OPERATOR

Actual States		**Real**	**Fake**	**Clusters**
Predicted States	**Real**	610	176	C_0
	Fake	110	0	C_1
Accuracy	**Precision**	**Recall**	**Specificity**	
0.6808	0.7761	0.8472	0.0000	

TABLE 13. K-MEDOIDS WITH FILTERING OPERATOR

Actual States		**Real**	**Fake**	**Clusters**
Predicted States	**Real**	485	66	C_0
	Fake	235	110	C_1
Accuracy	**Precision**	**Recall**	**Specificity**	
0.6641	0.8802	0.6736	0.6250	

Table 14 summarizes the obtained results of the both experiments.

TABLE 14. RESULTS SUMMARY

	Algorithms	**Accuracy**	**True Positive**	**True Negative**	**False Positive**	**False Negative**
Experiment 1	ID3	0.9776	769	191	10	12
	SVM	0.9572	765	184	17	25
	k-NN	0.9145	734	164	37	47
	k-Means	0.6731	661	0	201	120
	k-Medoids	0.6701	526	132	69	255
Experiment 2	ID3	0.9766	709	166	10	11
	SVM	0.9531	693	161	15	27
	k-NN	0.9107	674	142	34	46
	k-Means	0.6808	610	0	176	110
	k-Medoids	0.6641	485	110	66	235

V. DISCUSSION

The experiments showed that the supervised outperformed the unsupervised. Before we justified these results, we have to mention some important points:

- The model depends on the informative attributes to make the decision, fig. 4 illustrates the most informative attributes. While the mutual friends' attributes are the most informative one followed by the profile picture, the introduction attribute was the least informative attribute.

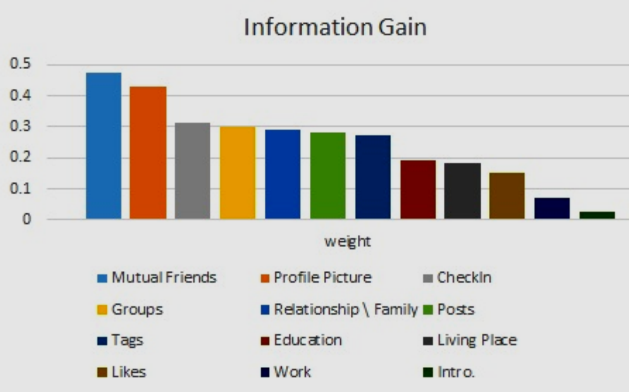
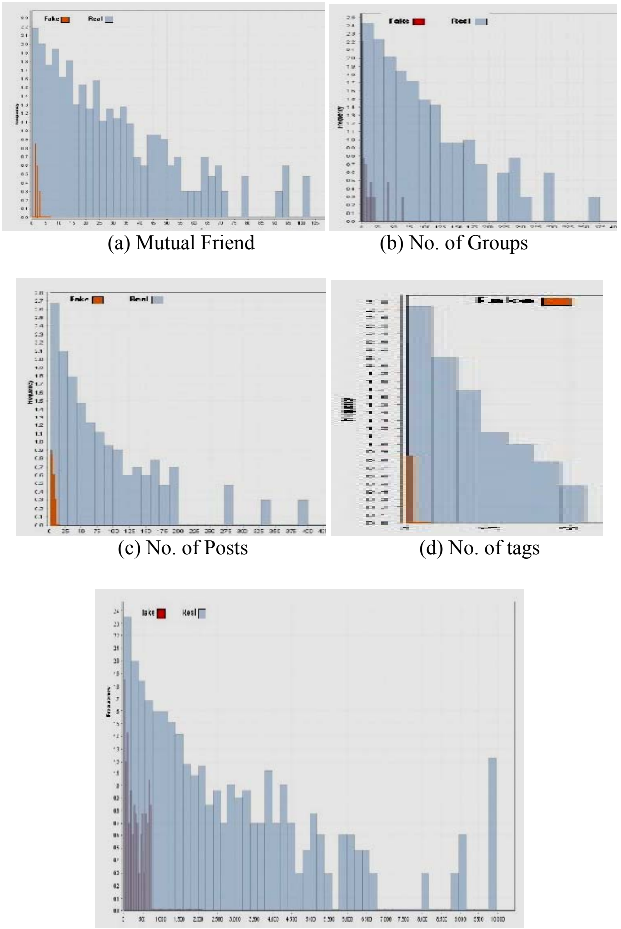


Figure 4. Attributes' Information Gain

- These attributes are usually interfered in both real and fake profiles, especially for the numeric attributes (e.g. No. of Pages Liked, No. of Tags, and No. of Groups Joined). For example, the fake profiles typically had zero tags, zero posts, and high liking activity. However, in real life many real profiles had the same behavior (zero tags, zero posts, and high liking activity), which made these profiles partially interfered and misleading the classification techniques. Fig. 5 (a - e) illustrates the histogram charts for the interfered attributes with respect to the two class labels (Fake and Real). For example, fake and real profiles may have the same mutual friends value, the red line shows the frequency of fake profiles that have the same value of mutual friends with the real profiles, the remaining charts included in fig. 5 shows the histogram distribution of the other attributes.
- The algorithm that is capable to handle the interfered attributes correctly will make the most accurate decision. Next, we will justify the performance of the supervised and unsupervised mining techniques by explaining how each technique resolves the interfered attributes.



(a) Mutual Friend

(b) No. of Groups

(c) No. of Posts

(d) No. of tags

(e) No. of Likes

Figure 5. Attributes' Frequency Distribution

A. Performance of the Supervised Vs. Unsupervised techniques:

As mentioned, the supervised outperformed the unsupervised algorithms. In case of the supervised algorithms, the training set with previously known labels play a vital role in the detection process, where the model studies the profiles in the training set and analyze each attribute in respect with the class label in order to generate the classification rules. This process minimizes the interfering factor in the attributes and made the model gain the necessary experience in the detection process.

On the other hand, the unsupervised techniques dealt with all profiles (dataset) as a single unit with no class label to separate the dataset, in this case, the interference factor is more severe, as the attributes interfered not just between the fake and real profiles, but between the profiles in the same class also, this case made the detection model distracted and grouped the profiles with similar attributes in clusters, avoiding any profile with interfered attributes.

B. The outstanding performance of the ID3 algorithm:

The ID3 decision tree is an unpruned algorithm constructed in a top-down divide-and-conquer recursive manner, “Pruning is a technique in the decision tree reducing branches in the tree by eliminating sections that provide little power to the classification process” [23].

ID3 registered highest detection rates among the employed algorithms in both experiments , as showed in table 14, the strength point is that, the ID3 algorithm performed in divide-and-conquer approach splitting the decision tree by employing the highest information gain attribute which is (Mutual Friends) as a root node and generate the classification rules in correlation between the root (Mutual Friends) and the other attributes with respect to the class label, analyzing each case without eliminating any attribute in the profiles dataset, this process generate a solid classification rules by considering the high information gain of the mutual friends attributes correlated with other attributes in respect with class label, which help decreasing the interference factor of the other attributes like posts, groups, tags…etc.

C. k-NN the Estimator and k-NN the Supervised Algorithm:

The k-NN estimator proved its efficiency for handling the missing values, as in the second experiment the model exhibits a stable performance with nearly identical accuracy for the all algorithms (supervised and unsupervised) as shown in table (14). But the supervised k-NN algorithm exhibit relatively lower detection rates compared to the other supervised techniques. As we mentioned in Section 3, the concept of the k-NN estimator for data imputation is to find the k most similar profile to the missing value in the dataset based on other available variables.

It is worth to mention here, that most of the missing values are in the (Groups and Likes) attributes, these attributes had the most interference factor as showed in fig. 5 (b and e). So, in case of the estimator schema, the k-NN exclude these attributes from the distance calculation process, which improve the efficiency of the k-NN estimator to find the nearest profile for the data imputation process. While in the supervised k-NN these attributes are employed in the model, which affect the accuracy rates of this algorithm in the MDFP model.

D. The Unsupervised Algorithms Performance:

The k-Means and k-Medoids are partitioning techniques, which they group the similar profiles in one cluster. When these techniques are applied in our model, they showed low accuracy rates because of the following two reasons:

1. These clustering techniques handle the dataset as a single unit, then they group the profiles with the similar attributes in one cluster. The problem is a raised with the informative attributes as these techniques cannot cluster

them into different clusters therefore, these techniques cannot correctly cluster the profiles into fake and real.

2. The k-Means and k-Medoids algorithms process numerical attributes only. In our case, we have 5 numerical attributes out of 12. Thus, only these 5 attributes are mainly contributed by the k-Means and k-Medoids algorithms in the detection process. However, when these algorithms consider the remaining attributes (non-numerical) the accuracy of the k-Medoids is improved by less than 2% while the k-Means accuracy is remained as is, as showed in tables 15 and 16.

.TABLE 15. K-MEAN WITH NUMERICAL ATTRIBUTES

Actual States		**Real**	**Fake**	Clusters
Predicted States	**Real**	661	201	C_0
	Fake	120	0	C_1
Accuracy	**Precision**	**Recall**	**Specificity**	
0.6731	0.7668	0.8464	0.0000	

TABLE 16. K-MEDOIDS WITH NUMERICAL ATTRIBUTES

Actual States		**Real**	**Fake**	**Clusters**
Predicted States	**Real**	529	88	C_0
	Fake	252	113	C_1
Accuracy	**Precision**	**Recall**	**Specificity**	
65380.	0.8574	0.6773	0.5622	

VI. CONCLUSION

In this work, a machine learning model (Fake Facebook Profiles Detection MDFP) is proposed to solve the problem of detecting the fake profiles on Facebook. The data mining techniques (Supervised and Unsupervised) are utilized here with three supervised algorithms (ID3 decision tree, SVM, and k-NN) and two unsupervised algorithms (k-Means and k-Medoids). These algorithms are implemented using the RapidMiner studio 8.0.1 as a mining tool. Two experiments are conducted for evaluating the accuracy of our model. In the first experiment, the profiles with missing values were handled using the k-NN model for the data imputation. While in the second experiment, these profiles were removed using filtering operator. The dataset that used in these experiments is of size 982 profiles (781 real, and 201 fake), collected by launching the developed CRAWLER on three profiles and profiles online purchasing. In the both experiments the supervised algorithms outperformed the unsupervised, showing high and promising accuracy rates, specially the ID3 decision tree, in which exhibit the highest accuracy among all algorithms. In contrast, the unsupervised algorithms showed a relatively low accuracy.

ACKNOWLEDGMENT

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

REFERENCES

[1] Romero, Daniel M., Wojciech Galuba, Sitaram Asur, and Bernardo A. Huberman. "Influence and passivity in social media." In Proceedings of the 20th international conference companion on World Wide Web, ACM, pp. 113-114,2011.

[2] Obar, Jonathan A., and Steven S. Wildman. "Social media definition and the governance challenge: An introduction to the special issue." 2015.

[3] O'Keeffe, Gwenn Schurgin, and Kathleen Clarke-Pearson. "The impact of social media on children, adolescents, and families." Pediatrics 127, no. 4: 800-804, 2011.

[4] Hajirnis, Aditi. "Social media networking: Parent guidance required." The Brown University Child and Adolescent Behavior Letter 31, no. 12: 1-7, 2015.

[5] Tang, Qian, Bin Gu, and Andrew B. Whinston. "Content contribution for revenue sharing and reputation in social media: A dynamic structural model." Journal of Management Information Systems 29, no. 2: 41-76, 2012.

[6] https://newsroom.fb.com/company-info/ (4th October 2018).

[7] Wani, Mudasir Ahmad, Suraiya Jabin, and Nehaluddin Ahmad. "A sneak into the Devil's Colony-Fake Profiles in Online Social Networks." arXiv preprint arXiv:1705.09929 ,2017.

[8] https://www.facebook.com/legal/terms (18th august 2018).

[9] Kumar, Nitesh, and Ranabothu Nithin Reddy. "Automatic detection of fake profiles in online social networks." Ph.D. diss., 2012.

[10] Gupta, Aditi, and Rishabh Kaushal. "Towards detecting fake user accounts in Facebook." In Asia Security and Privacy (ISEASP), 2017 ISEA, pp. 1-6. IEEE, 2017.

[11] Ahmed, Faraz, and Muhammad Abulaish. "A generic statistical approach for spam detection in Online Social Networks." Computer Communications 36, no. 10: 1120-1129, 2013.

[12] Fire, Michael, Dima Kagan, Aviad Elyashar, and Yuval Elovici. "Friend or foe? Fake profile identification in online social networks." Social Network Analysis and Mining 4, no. 1 (2014): 194.

[13] Nazir, Atif, Saqib Raza, Chen-Nee Chuah, Burkhard Schipper, and C. A. Davis. "Ghostbusting Facebook: Detecting and Characterizing Phantom Profiles in Online Social Gaming Applications." In WOSN. 2010.

[14] Bhat, Sajid Yousuf, and Muhammad Abulaish. "Community-based features for identifying spammers in online social networks." In Advances in Social Networks Analysis and Mining (ASONAM), 2013 IEEE/ACM International Conference on, pp. 100-107. IEEE, 2013.

[15] Mccord, Michael, and M. Chuah. "Spam detection on twitter using traditional classifiers." In international conference on Autonomic and trusted computing, pp. 175-186. Springer, Berlin, Heidelberg, 2011.

[16] Adikari, Shalinda, and Kaushik Dutta. "Identifying Fake Profiles in LinkedIn." In PACIS, p. 278. 2014.

[17] Wang, Xiaoyun, Chun-Ming Lai, Yunfeng Hong, Cho-Jui Hsieh, and S. Felix Wu. "Multiple Accounts Detection on Facebook Using Semi-Supervised Learning on Graphs." arXiv preprint arXiv:1801.09838 (2018).

[18] Viswanath, Bimal, Muhammad Ahmad Bashir, Mark Crovella, Saikat Guha, Krishna P. Gummadi, Balachander Krishnamurthy, and Alan Mislove. "Towards Detecting Anomalous User Behavior in Online Social Networks." In USENIX Security Symposium, pp. 223-238, 2014.

[19] Wang, Alex Hai. "Detecting Spam Bots in Online Social Networking Sites: A Machine Learning Approach." DBSec 10: 335-342, 2010.

[20] Hron, Karel, Matthias Templ, and Peter Filzmoser. "Imputation of missing values for compositional data using classical and robust methods." Computational Statistics & Data Analysis 54, no. 12: 3095-3107: (2010).