ID3 and k-means based methodology for Internet of Things device classification

Joan Nolla Suárez
ITAM
Mexico, Mexico City
E-mail: joan.nolla@itam.mx

Ante Salcedo
Department of Digital Systems,ITAM
Mexico, Mexico City
E-mail: ante.salcedo@itam.mx

***Abstract*—The Internet of Things (IoT) brings the issue of connecting an immense amount of diverse devices. This vast diversity will present a challenge for communications, since it is not expected that all devices will follow the same rules and standards to communicate back and forth, due to the difficulty and inefficiency of developing a unique set rules and standards for each device. A classification of devices is needed, so rules and protocols of communications could be established among the different device categories, to deal with the diversity of the things to be interconnected. In this paper, a classification methodology using a clustering algorithm like k-means is proposed; as well as, a way to establish rules of classification using a decision tree implemented with the ID3 algorithm.**

***Keywords*—Internet of Things, IoT, classification, decision tree, k-means, ID3**

I. Introduction

The Internet of Things (*IoT*) is at the core of the revolution already under way that is seeing a growing number of Internet enabled devices that communicate with each other, and with other web-enabled gadgets, creating device to device networks meant to make everyday life easier, without requiring any human direct intervention [2]. In recent years, the number of connected devices has grown dramatically, and it is expected to continue growing until the number of connected devices reaches, by the year 2020, the amount of approximately 50 billion. That is, more than six devices per person in the world. Figure 1 illustrates the expected growth according to Cisco Systems, Inc. [3]

Not only the amount of connected things will increase dramatically, but also the diversity of products, services, and classes of devices, with specific features and applications. The Internet of Things (*IoT*) will drive an accelerated adoption rate, changing in the very near future the way technology works, and establishing absolutely new paradigms about the way that people interact with the things around them, and the way things interact among themselves [4]. In such context, it is hard to believe that the huge diversity of expected commercial devices, within the complex and rapidly changing environment of interconnected things (including devices as diverse as microwave ovens, wheelchairs, robots, phones, cars, or even intelligent key chains, to mention few) will all have the same properties, performance, types of connectivity, priorities, levels of security, standards, and/or rules of operation.

Of course, it would also be very difficult (and probably inefficient too) to try to develop independent rules and standards for each of the thousands of products that there are expected to be. This is why it is very important to start thinking about alternatives to classify all the possible devices and other connected things that could emerge and be in operation in the near future [5]. Such classification could be determinant to understand the dynamics of the future technology; to establish the rules for the interaction and communications between connected things; and to resolve some of the many future issues that could derive from the *IoT* adoption.

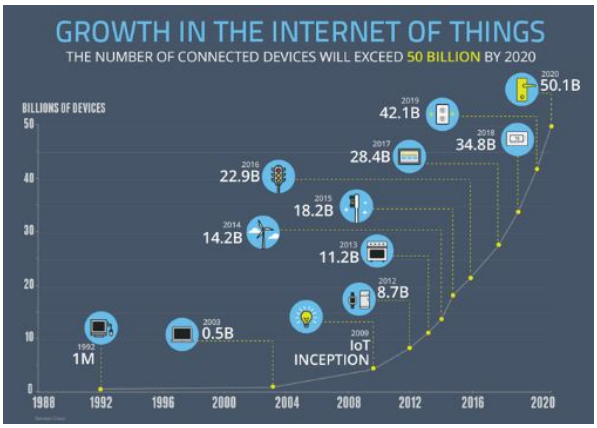

Fig. 1. Internet of Things *IoT* expected growth [4]

This paper proposes the use of clustering techniques to achieve automatic classification of devices, and presents the results obtained for a limited set of connected things, applying a k-means clustering technique. The used clustering approach considered the most relevant features that identify each of the considered connected devices (such as its mobility, battery capacity, bandwidth, object size, etc.), which were registered and summarized in a CSV file. An ID3 algorithm was used to generate a decision tree classification ruler that allows to classify any object, according to its own relevant features. The obtained decision tree determines the most important attributes that need to be considered for each classification category.

II. Approach proposal for device classification

The classification approach that is proposed and has been tested, considers four main steps: 1) the definition of the relevant features that are necessary to classify the desired objects; 2) the iterative clustering of the considered devices

978-1-5386-2532-3/17 $31.00 © 2017 IEEE
DOI 10.1109/ICMEAE.2017.10

into different numbers of clusters, until logical classes are obtained, with the k-means algorithm; 3) the verification, by means of the "elbow rule" heuristic, that the chosen clusters properly describe the universe of considered things; and 4) the integration of a decision tree ruler to continue classifying things in terms of the identified categories.

A. Definition of relevant classification features

A *features table* was integrated identifying the characteristics of 28 devices that have high probabilities to be present in the near future environment of *IoT*. Some of the chosen devices are: a smartphone, a laptop, a smart TV, a fridge, a smart key chain, and a security camera. To select the relevant features that are necessary to classify the selected objects a comparative analysis was made, until selecting the following twelve features:

- Size
- Mobility
- Battery capacity
- Memory
- Bandwidth requirements
- Internet gateway
- Source of information capacities
- Information receiver capacities
- WiFi enabled
- Blue-tooth enabled
- 4G/3G enabled
- Wired Ethernet availability

Category values were assigned to each of the selected features, in order to evaluate each device. In such way, the "size" feature considered the category values: a) small, b) medium, or c) large. In a similar way, the Internet gateway feature was considered to be either "yes", or "no". The structure of the table of features that was integrated, is shown in figure 2.

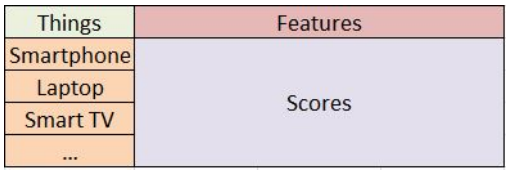

Fig. 2. Structure of the integrated table of features.

B. Iterative definition of clusters

An iterative clustering of the considered devices, until logical classes were obtained, was implemented with the k-means algorithm; which is an analytical technique that for a chosen value of k identifies k clusters of objects, based on their proximity to the center of each group [6]. The k-means is a greedy algorithm, meaning that the resulting clusters reach a local optimum, for each of the centroids where the clusters were initialized.

The k-means algorithm follows the next four steps [7]:

1) Choose the number of clusters (k) and an initial guesses for each of the k centroids.
2) Compute the distance from each data point to each centroid, to get each point assigned to its closest centroid until the k clusters are established.
3) Compute the centroid (center of mass) for each of the clusters defined at step 2, by calculating the average of the data points in each cluster.
4) Repeat steps 2 and 3, until the clusters' centroids don't change; meaning that the algorithm has converged to an answer.

To calculate the distances between the data points and the clusters' centroids, the category attributes of the considered features were assigned with numerical values. In that way, if a device is WiFi enabled it was assigned a value of 1, and if it was not WiFi enabled, a value of 0. Analogously, a small device was assigned an attribute value of 1, while a large device a value of 3. Also, a strategy to achieve a faster convergence of the algorithm was taken. In order to initialize the first centroid, an initial random guess was taken from the points of the data set. The following centroids were established to be the farthest away from the first centroid, or set of already settled centroids (also from the available points within the data set). In figure 3 the flow chart of implemented k-means algorithmic approach is presented.

It's important to remark that the k-means algorithm requires the number of clusters as an input, so the number of classification categories has to be carefully chosen. A small number of clusters would make the classification extremely general, and a large one would make it unnecessarily specific and complex. There's no unique rule to choose the number of clusters for a data set, but there are heuristics to approach an appropriate number of clusters. For instance, in this paper the number of clusters is chosen by contrasting the results of running the k-means algorithm with different numbers of clusters, and choosing the one that brings the most logical and coherent object classification. Then, the obtained number is confirmed by using the heuristic procedure known as the "elbow rule".

C. Elbow Rule

The "elbow rule" is an heuristic method to determine if a number of clusters is appropriate to classify a specific data set [7]. It is based on the fact that as more clusters are made, the centroids are closer to the data points, so the *Within Sum of Squares* (WSS) is going to be smaller. The WSS is a way to weight the error within each cluster, and represents the sum of the squared differences of the data points of a cluster, to its centroid.

The WSS formula is:

$$WSS = \sum_{i=1}^{n} (y_i - \bar{y}), \tag{1}$$

where:

n = Number of data points in each cluster;
$\bar{y}$ = Cluster centroid;
y_i = i-th data point of the cluster.

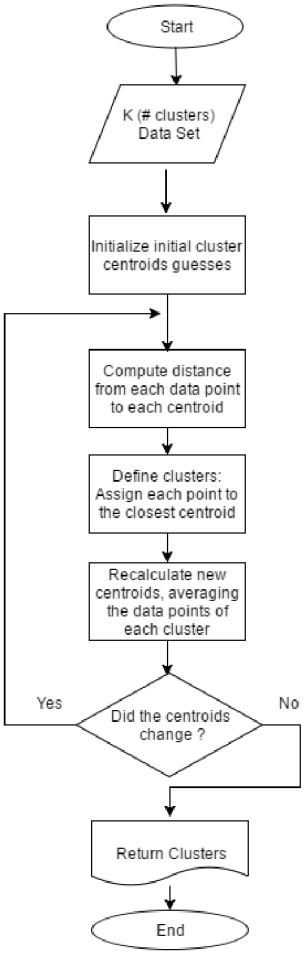

Fig. 3. Implemented k-means algorithm flow chart

The largest amount of error (WSS) is found when only one cluster is considered, because in such case all the data points are associated to one single centroid. As the number of cluster increases the WSS is reduced until reaching zero, when the number of clusters equals the number of points in the data set (i.e., the centroids of the clusters match the data points). The "elbow rule" heuristic determines when (at what number of clusters) the WSS stops reducing significantly, by slowing down its decreasing rate (see figure 4). In other words, the complexity cost of having more clusters starts to have diminishing marginal returns in the matter of reducing the WSS. The "elbow point" can be clearly visualized when the WSS is plotted against the considered number of clusters (k). The point where the curve in the graphic is cut abruptly, by rapidly changing its slope, is know as the "elbow"; as shown in the figure. The "elbow point" represents a fair estimate for the appropriate number of clusters, corresponding to a given data set.

D. Decision Tree Classification Ruler

A decision tree encompasses a structure of features and attributes, to specify decision sequences and consequences. Given a certain input, the goal is to predict a response or

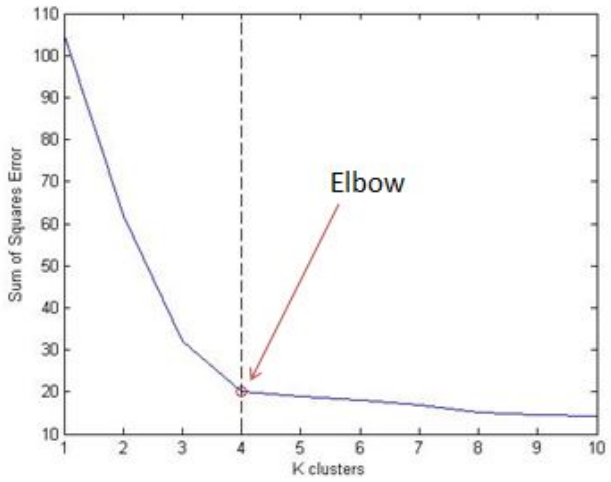

Fig. 4. Illustrative "elbow point" heuristic plot.

output. A prediction is achieved by following the decision tree, with test points and branches. At each test point, a decision is made to pick a specific branch (feature and attribute), before continuing traversing along the tree. When an end point is reached a prediction has been made [8]. In this paper, the decision tree is used as the classification ruler, where some of the most important features of an unclassified device are questioned at each test point, until reaching an end point, that results in the device is classification into a given category.

Several algorithms are available to develop a decision tree. One of the most popular, because of its fast convergence, its clever branch sorting approach, and its easiness for comprehension, is the ID3 algorithm. Created in 1979, by John Ross Quinlan [9], the ID3 algorithm uses the Sharon's information theory, accounting for three different inputs:

1) **Output variable.** It's going to determine the end points of the three, in this case the categories of the classification.
2) **Input variables.** These are going to determine the test points of the tree and conform the branches. The input variables for the experiment are all the considered features.
3) **Training set (S).** The data set that will be used to create the tree. The features table will be the training set.

The ID3 algorithm uses the concepts of entropy and information gain to built the decision tree. Entropy (H) is a measure of uncertainty, or in this case, the purity of the data set. The information gain (IG), on the other hand, is a measure of the purity; or more specifically, is an indicator of the attributes that give the highest classification certainty [8]. The formulas for the entropy and information gain are:

$$H\left(S\right) = -\sum_{x \in X} p\left(x\right) \cdot log_2(p(x)); \tag{2}$$

and

$$IG(A, S) = H(S) - \sum_{y \in Y} p(y) \cdot H(y). \tag{3}$$

In equations (2) and (3) H is the entropy; S the data set; x the set of attributes in S; $p(x)$ the probability of x against all the elements in S; IG the information gain; and Y the subsets of data obtained when splitting S by attribute A.

The top levels of the decision tree will correspond to the attributes with the higher information gain. Then, the classification process will consider greater levels of entropy as it advances along the tree branches. This means that the more important features to determine a specific category are at the top of the tree; and, as the tree is extended, the following features loose importance as they move away from the starting point. As it can be appreciated, there is an inverse relation between entropy and information gain. A summarized version of the ID3 algorithm is described below:

1) Calculate the entropy of all the attributes (features) in the training set (S);
2) Split the training set (S) into subsets using the attribute with the maximum information gain;
3) Make a test point containing that attribute;
4) Run the algorithm again on the remaining attributes' subsets.

III. Results

Following the proposed classification approach for 28 selected devices, the id3 and k-means algorithms were implemented using Python. The k-means algorithm was tested for 3, 4 and 5 clusters, to find the appropriate number of categories that made the most logical and coherent classification. The algorithm with 3 clusters grouped devices of very different nature (like a smartphone and an intelligent key chain) within the same category, just because both of them are small. The scenario with 5 clusters, on the other hand, became so specific that a smartwatch had its own category, which was considered too granular for the current exercise. Perhaps in the future, when the portables market grows, to have a category that would only account for portables (like a smartwatch) would be useful, but for now, the role of a smartwatch in a network is very similar to the role of other devices, like that of a smartphone. The best considered number of clusters that was obtained was four. The different cluster sets obtained from the k-means algorithm represent the classification categories, and could be described in the following manner:

1) Mobile Orchestrators: These devices are mobile, have small to medium sizes, are Internet Gateways, can use WiFi/bluetooth/3G/4G, and have medium to large bandwidth requirements among other things. This devices can participate in different *IoT* scenarios like and office or a house, and are going to lead the communications dynamics. The best example of a device in this category is a smartphone.
2) Fixed Orchestrators: These devices are similar to the Mobile Ochestrators, but they're not mobile, have large sizes and have large to unlimited battery. This devices can only participate in one *IoT* scenario and also will lead communications dynamics. Some examples of a devices in this category are a PC or a smart TV.
3) Fixed Followers: The devices in this category have unlimited battery and are information resources. They will follow the dynamic established by the Orchestrators, and may establish connection with the Dummy Followers if requested. Some examples of this category are an alarm system or a fridge.
4) Dummy Followers: These devices are small sized, have small battery power and overall don't have large bandwidth requirements. The Dummy Followers job in the communications dynamics is follow what the Orchestrator leads. Sensors are the main type of device that conforms this category.

The devices included in each proposed category are represented in the figure 5.

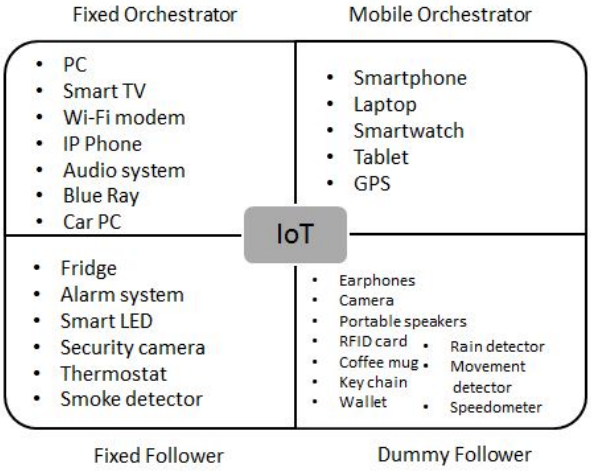

Fig. 5. Classification categories with 4 clusters

The next step in the 28 devices classification experiment, was to verify the found appropriate number of clusters, by using the "elbow rule" heuristic. Again, the WSS was calculated by means of an algorithm implemented with Python. Figure 6 presents the graphic of the obtained WSS against the number of considered clusters. As it can be appreciated, the elbow rule heuristic confirms that the best number of clusters should be four, which is in agreement with the number chosen before.

The last step of the classification exercise was to create the classification tree ruler. For this, the output variable of the ID3 algorithm was the classification categories, the input variables were all of the attributes in the feature table and the training set was the data in the features table. The resulting decision tree after running the ID3 algorithm is presented in figure 7.

The decision tree does not have as many branches as the number of features (twelve) considered at the beginning of the exercise. A possible explanation is that the attributes that are key factors to determine a category are just a few. Some other possible reasons for this are:

- Many of the features are correlated, for example a device with large size probably is going to have no mobility, or the devices with more battery capacity have bigger bandwidth requirements.
- Some features just can't be used to describe a category.

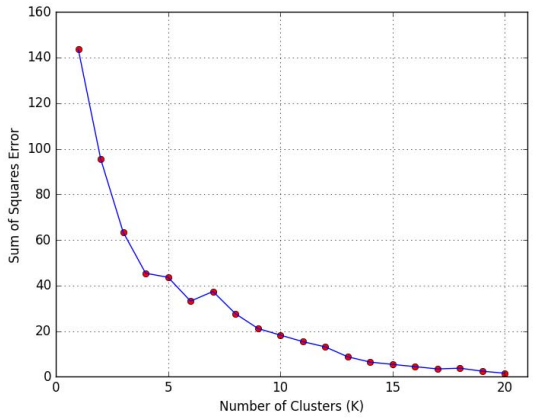

Fig. 6. WSS against the number of cluster

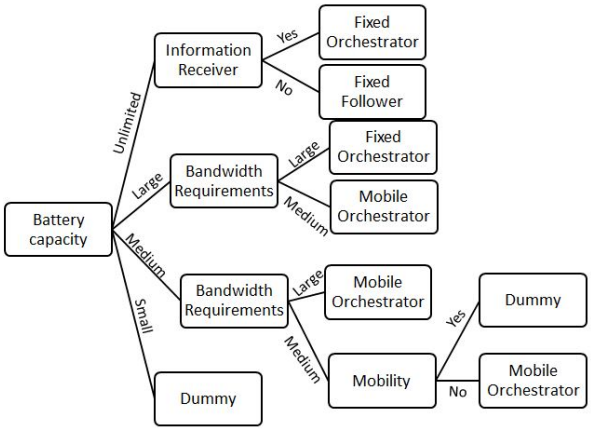

Fig. 7. WSS against the number of cluster

- The way the features were scored was just too general to create more variability among the branches.

As expected from the very beginning, the decision tree brought important information about the data set. For example, the most important feature to determine a category is the battery capacity, and the second one is the bandwidth requirements. Features like memory were simply not relevant for the classification ruler. It's important to say that the classification categories with their classification rules are not, by any means, a universal solution. Several results can be obtained with different data sets, by means of different clustering techniques, and considering very different application scenarios. The automatized and computer based classification methodology approach proposed, is just one of the many alternatives that can be applied to different sets of device classification situations and scenarios.

IV. Final remarks

The vast diversity of devices that the Internet of Things (*IoT*) will bring into our environment in the very near future opens a window of necessity and opportunity to classify them into classes that facilitate their best administration and utilization. Therefore, the development and implementation of classification methodologies to structure catalogs of devices and "connected things" is becoming more, and more, relevant everyday. For instance, in order to implement communications protocols and operation standards for emerging wireless device area networks (WDAN) [2], it is necessary to define categories in order to attend, in an efficient and easy manner, the enormous diversity of different issues that the *IoT* presents.

The classification approach that was presented along this paper to create device categories with similar features, proved to be an easy alternative to understand and structure a set of devices that are very different in nature. The results that were presented are logical in the semantics level, and the algorithms used could be easily implemented and applied to many other sets of data, for a variety of scenarios and classification objectives. Also, the classification ruler that was obtained, proved to be a good and efficient solution to implement categories that are not obvious from an initial definition of a "features table", and a appropriate means to provide useful information about the analyzed data set.

The described classification methodology approach that was presented will be used by the authors at further studies on the communications dynamics that different device categories could experiment. This may include requirements and rules of communication for different classes of connected devices. The Internet of Things *IoT* presents exciting and motivating challenges for the future; nevertheless, the accomplishment of having a more connected society have never been so expected and close.

Acknowledgment

This work has been partially supported by the *Asociasición Mexicana de Cultura A.C.*

References

[1] Technology Strategy Board - IoT Special Interest Group, *"Internet of Things (IoT) and Machine to Machine Communications (M2M) Challenges and opportunities: Final paper May 2013,"* ICT KTN.London, UK. May, 2013.

[2] Salcedo, A., Villa, F. (2016). *"Possible application of the 30 to 60 GHz spectrum band to implement IoT Wireless Device Area Networks (WDAN)"*. In Mechatronics, Electronics and Automotive Engineering (ICMEAE), 2016 International Conference. IEEE.

[3] D. Evans, *"The Internet of Things:How the Next Evolution of the Internet Is Changing Everything,"* Cisco IBSG. April, 2011.

[4] J. Twining, *"Behind the numbers: growth in the Internet of Things,"* Platform with information from Cisco IBSG. March, 2015.

[5] Ramirez-Mireles, F., Salcedo, A., Alvarez, G. (2009). *"Personal communications using an UEAN: concept, example and measurements"*. IEEE Transactions on Consumer Electronics. IEEE.

[6] J. MacQueen, *"Some Methods for Classification and Analysis of Multivariate Observations,"* in Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, CA, 1967.

[7] EMC, *"Advanced Analytical Theory and Methods: Clustering,"* in Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data, 1st ed.Indianapolis: Wiley, 2015, pp. 118-130.

[8] EMC, *"Advanced Analytical Theory and Methods: Classification,"* in Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data, 1st ed.Indianapolis: Wiley, 2015, pp. 191-206.

[9] J.K. Quinlan, *"Induction of Decision Trees,"* Machine Learning, vol. 1, no. 1.Netherlands, 1986, pp. 87-92.