IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE)
e-ISSN: 2278-1676, p-ISSN: 2320-3331, Volume 16, Issue 4 Ser. I (Jul. – Aug. 2021), PP 30-37
www.iosrjournals.org

Edge-Fog-Cloud Computing for IoT Knowledge Creation

Bel G. Raggad
Seidenberg School of CSIS, New York, USA

Hedi F. Raggad
Faculte' des Sciences, Tunis, Tunisia

Abstract: *IoT, itself, is of no consequential value unless it produces business value generation capabilities, often in a feature-driven manner, as knowledge comes in features of interest to IoT owners. We propose a feature-driven IoT lifecycle and its knowledge creation and propagation process. Knowledge is edge-computed at IoT nodes to create edge knowledge needed by operational management, cloud-computed at IoT clouds to create cloud knowedge needed by strategic management, and fog-computed half-way at IoT virtual gateways to create fog knowledge, often needed by functional management.*
Knowledge is represented as belief structures given features preselected by IoT owners, in a Dempster and Safer manner, and propagated using the Dempster rule of combination of evidence generated at IoT nodes, IoT virtual gateways, and IoT clouds. We gave examples where the reader can apply numerical examples to demonstrate the working of the IoT feature-driven knowledge creation process.

Keyword: *IoT, Knowledge creation, Edge computing, Fog computing, Cloud computing, Dempster and Shafer Theory, Belief structure, belief basic assignment*

Date of Submission: 10-07-2021 Date of Acceptance: 26-07-2021

I. Introduction

The Internet of Things is a novel paradigm shift in IT arena. The phrase "Internet of Things" which is also shortly well-known as IoT is coined from the two words i.e. the first word is "Internet" and the second word is "Things". Just when billions of users are served by TCP/IP and other communication protocols, now billions more of things are also connected, physically or virtually, in an IoT. When Kevin Ashton, in 1999, found useful an Internet-based information service architecture [1, 7], he never thought of an Internet of Things will take the prodigious spread and impact we now see everywhere . Initially, the term refered to Internet-enabled objects, of any type, like sensors, actuators or mobile phones, interacting with each other and cooperating to achieve specific goals [1, 7].

Powerful data centers were initially deployed anywhere secure but now virtual servers enhanced local and cloud computing and allowed virtual LAN and virtual WAN computing that is intergrated with cloud computing to make good use of data anywhere it is on the IoTs and anyhow it is generated by IoT things, whether these are devices, people, sensors, animals, trees, appliances, tools, etc [1, 7].

Lately, IoTs started attracting industrial computing where virtually networked machinery and physical objects are enabled anytime and anywhere to share data and information to assure business continuity and jointly created the capability of knowledge discovery that the rest of the industry benefit from it in their supply chains and operations management. [C. W. Axelrod. Enforcing security, safety and privacy for the internet of things. In Systems, Applications and Technology Conference (LISAT), 2015 IEEE Long Island, pages 1–6. IEEE, 2015.]

This paper will study the knowledge creation capabilities on IoTs, through eduge computing on IoT members, fog computing on virtual gateways, and machine learning on big data managed in cloud computing.

The Dempster and Shafer Theory

The Dempster–Shafer theory (DST) provides a mathematical framework for uncertainty management where all analysts use the same frame of discernment in studying a finite set of mutually exclusive outcomes about their decision domain. This framework is capable of combining evidence from different sources and produces a degree of belief, as a belief function, that takes into account all the available evidence.

As in Denoeux, the theory of belief functions is not a theory of imprecise probability, and it does not represent uncertainty using sets of probability measures. It instead extends probability theory by allowing some imprecision, using a multi-valued mapping in the case of belief functions. This is what you can see in this article when we go from precise labeling data to learning set labeling where a case label is imprecisely known.

The Dempster-Shafer (DS) theory started with Dempster in 1968 as statistical inference, but has been later formalized by Shafer, in 1976, as a theory of evidence. Later after the 1980's Smets reshaped it in his Transferable Belief Model before it started to see growing development in diverse AI applications in most domains.

When presented with the same decision domain information, Dempster and Shafer theory should produce the same decision support as in Bayesian reasoning, but it is capable of a superior expressive power when information is incomplete or data is not of good quality.

In order to model a belief structure for a decision domain with a frame of discernment Ω, we let the power set 2^{Ω} contain every mutually exclusive subset of the frame of discernment Ω. A basic probability assignment m is used to allocate a belief value in [0, 1] for every hypothesis defined by the subset in the frame of the discernment, as follows:

$m: 2^{\Omega} \rightarrow [0, 1]$

$m(\emptyset)=0$

$m(A)\geq 0$ for any A in 2^{Ω}

$\sum_{A \subseteq \Omega} m(A) =1.$

If x is an unknown quantity with possible values in our frame of discernment Ω, we can add a piece of evidence about x using a mass function m on Ω. Any subset A of Ω with a mass greater than zero is called a focal set of m. You can see that this is different than in Bayesian theory where probability distributions only have singleton focal sets. When we have no evidence on x, we use the vacuous mass function, defined by $m_{\Omega}(x) = 1$, which represents a completely uninformative piece of evidence.

Upper and lower probability can be obtained which will enclose the precise traditional probability the analyst is seeking. This analyst's target is then bounded by two non-additive continuous measures that DST refers to as belief and plausibility. The belief for subset of interest A is the sum of all the masses of the subsets x residing in A; and the plausibility of a subset A is, on the other hand, the sum of all the masses of the subsets x intersecting A.

A great contribution by Dempster and many others who expanded the Dempster and Shafer theory is the combination of evidence obtained from multiple sources and the modeling of conflict. Often, bodies of evidence come in small pieces obtained from different independent sources. While these bodies of evidences are included in the decision process using belief functions, the totality of the evidence is computed by combining the belief functions using Dempster Rule and its extensions. This rule consists of a mapping that considers multiple sources and produces a composite source that represents the combined impact of sources as one combined measure of belief. Given two independent sources of evidence defined on the same frame of discernment Ω and with basic probability assignments m_1 and m_2, we combine evidence as follows:

$m_{\Omega}(A) = \sum_{B \cap C=A} m_1(B)m_2(C)/(1-K)$; for $A \neq \emptyset$

Where $K= \sum_{B \cap C=\emptyset} m_1(B)m_2(C)$ and $m_{\Omega}(\emptyset)=0$

The parameter K represents the basic probability mass associated with the conflict between m_1 and m_2. It is computed as the sum of the products of the basic probability masses of all the disjoint sets from the tow sources of evidence.

Now that we explained our motivation for the resolution of data inconsistencies in big data, whether it is in an IoT environment or in any other computing environment, we are proposing a analytical model for resolving the data inconsistency by imprecising learning sets associated with data sources involved in a decision process. Once data is cleaned of inconsistencies, a DST model is adopted to generate belief functions that can be transformed using a TBM model to produce pegnistic probabilities that can be employed in managing uncertainty in the decision process. The traditional decision theoretic model can then be applied on the pegnistic probabilities as in Bayesian reasoning.

The IoT lifecycle

The IoT technology is only useful if business value generation capabilities come from it so that owners can create a competitive advantage to assure business continuity and a lasting lucrative or social wealth. Those capabilities cannot stay invariant and have to align with changing missions, values, and goals of owners. So, talking about a lifecycle for IoT make great sense, but has to be modelled and studied in terms changing features that owners need to configure as needs arise to study system efficiencies and reliabilities.

The IoT lifecycle should therefore be feature driven. For, the IoT should be configured in a cluster manner, and virtual LANs should be configured for current features and reconfigures when new features are added. The data collection for a while until the sought feature-driven knowledge for the IoT is created and acted upon. The knowledge will become relatively obsolete for the current features because IoT owners have just adopted new features aligned with the new mission and goals. That is, the IoT lifecycle should be studied in terms of the current features as defined by IoT owners. Figure 1 depicts the IoT lifecycle as we just presented.

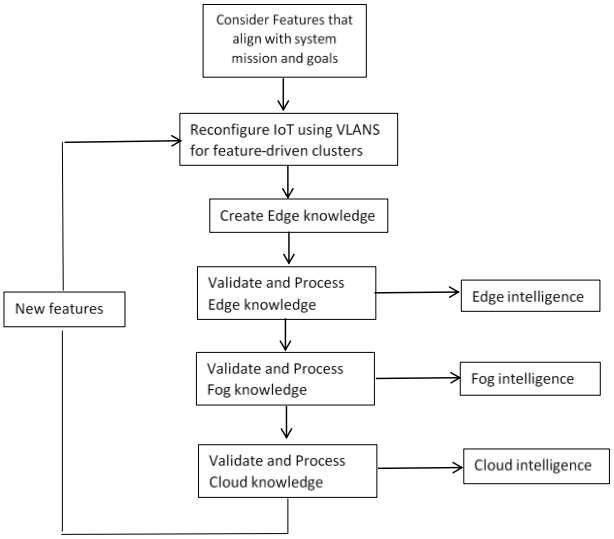

Figure 1: IoT Lifecycle

IoT feature-driven edge-fog-cloud knowledge management

In order to introduce our IoT feature-driven knowledge management model, we consider an open and stable IoT reorganized according to a finite number of virtual subnet of things (SoT), {si}, i=1,|S|, where selected members are residing in with members

IoT knowledge is organized into feature-driven nets of knowledge segments parallel to the IOT virtual subnetwork made of virtual LANs containing feature-based IoT nodes, as shown in Figure 2.

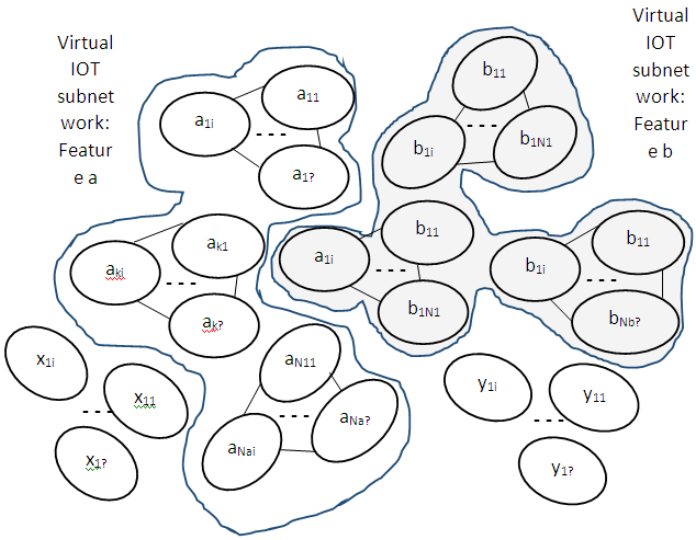

Figure 2: IoT virtual subnetworking

IoT knowledge creation and propagation

The IoT knowledge discovery and propagation process starts at IoT nodes clustered according to features defined by owners. Each cluster is configured in a virtual LAN where knowledge K_{ij} is created at every node n_{ij} of the VLAN j and where cluster knowledge is created, in a Dempster and Shafer manner, at the VLAN gateway. That is, granular data d_{ij} are born at IoT nodes n_{ij} for a prescribed feature fj, j=1,M and data is filtered in an iterative manner until a small big data D_{ij} is created and a belief structure m_{ij} is constructed for the feature-driven virtual LAN where IoT nodes reside. At every VLAN gateway, a small big data is then generated for its feature. Once converged, the VLAN knowledge sources are then validated, and will then be fused to create the IoT knowledge for the prescribed feature. Knowledge for the IoT is represented as belief structures for the features in question and is validated using the Dempster Rule of Combination (DRC). Figure 3 depicts the knowledge creation and propagation process, according to the IoT life cycle presented earlier in the article.

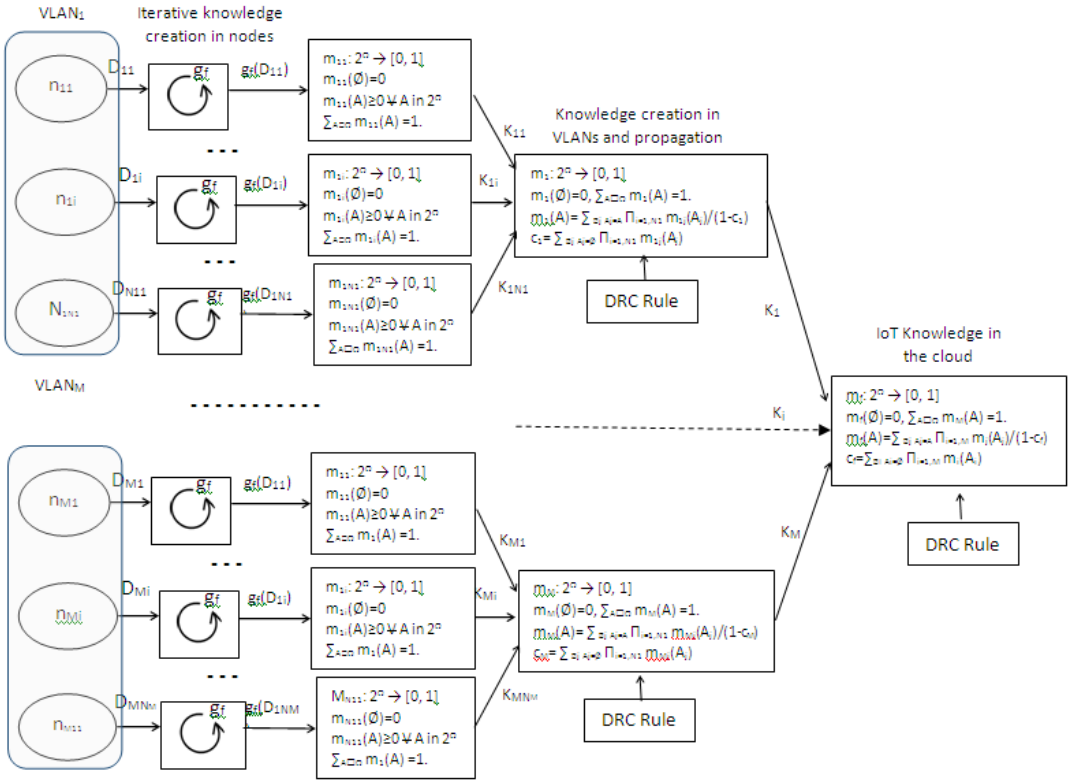

Figure 3: knowledge creation and propagation process

IoT edge knowledge creation

Let us beforehand consider an IoT node that belongs to the feature-driven virtual LAN selected for the sought knowledge discovery, like for example, impact of electrical surge on a factory machinery line, or the effect of some security threats on the disaster recovery in a computing environment or its business continuity. Lets consider an IoT node a_{ij} and a feature vector a. As in Raggad [8, 9], we start by constructing a belief structure on the feature-based data D collected in the IoT node a_{ij}. D is made of M tuples containing the N feature-based attributes {Xi}, i=1,N. We then construct the power sets 2^{Xi}, i=1,N and construct the frame of discernment $F=2^{X1}x \ldots x\ 2^{XN}$. Let us then consider a hypertuple e, $e=\{e^1, \ldots, e^N\}$ where e^k is a subset of X_k. Also let Δ_α be a partial order relation on all the data sets on hand. If x and y are elements of a set E, we say that $x\Delta_\alpha y$ if and only if $|x\cap y|/|x| \geq \alpha$. The intersection defines the amount of support x provides to y, or alternatively, the amount of α-compatibility between x and y (i.e., a compatibility with level α).

We define the evidence support $s_D^{\ \alpha}$ (e) of x in D as the set of y in D such that $y\Delta_\alpha e$. That is, $s_D(e)$ = {yεD, such that $y\Delta_\alpha e$}. The subset D is a poset with respect to the partial order relation Δ_α and it may hence have elements that are related to e (α-compatible with) and others that are not related to e (not α-compatible). Only the compatible elements y in D such that $y\ \Delta_\alpha\ e$ are accepted to support e.

Let F, defined above, be our frame of discernment. The belief structure for D in F is defined as follows:

$m_D^{\ \alpha}$: F → [0 1]

$m_D^{\ \alpha}$ (e) = $|s_D^{\ \alpha}$ (e)|/ $|s_D^{\ \alpha}$ (F)|

where $s_D^{\ \alpha}$ (F) = {{yεD such that $y\Delta_\alpha e$}, eεF}

It is sometimes useful, for simplicity, to denote as follows:

$|s_D^{\ \alpha}$ (e)| = $|e\Delta_\alpha D|$ = Cardinal of {yεD, such that $y\Delta_\alpha e$}.

We then have the following:

$m_D^{\ \alpha}$ (e) = $|e\Delta_\alpha D|/|F\Delta_\alpha D|$,

$m_D^{\ \alpha}$ (F) = $|F\Delta_\alpha D|/|F\Delta_\alpha D|$=1.

That is, if we consider the feature attributes $\{X_i, i=1,N\}$, we can produce knowledge in terms of probability distributions of $\{X_i, i=1,N\}$ in order to be able to make a decision. The probability distributions are a good Bayesian model to manage uncertainty in this decision process. We are however not sure that these probabilities even exist given the types of data we collect at the IoT node and given the ambiguities associated with them. Smets [] showed that we can use pignistic probabilities to approximate these Bayesian probabilities and adopt a Transferred Belief Model instead of Bayesian. In order to do so, we need to construct belief structures on extracted subsets that contain $\{X_i, i=1,N\}$, as shown above, and induce basic probability

assignments [1, 8, 9]. This is however only possible after envisaging the product of all the power sets on domains of single feature attributes X_i, i=1,N. That is, we obtain a table of hypertuples made of the respective attributes' domains in 2^{X1} x … x 2^{XN}. We denote F=2^{X1} x … x 2^{XN} that we earlier called frame of discernment of the belief structure that we intend to construct.

As in Raggad [8, 9], given our frame of discernment F defined above, the belief structure for D in F is defined as follows:

m_D^{α}: F → [0 1]
m_D^{α} (e) = $|s_D^{\alpha}$ (e)|/ $|s_D^{\alpha}$ (F)|
where s_D^{α} (F) = {{yεD such that $y\Delta_{\alpha}e$}, eεF}
m_D^{α} (e) = $|e\Delta_{\alpha}$ D|/|FΔ_{α}D|,
m_D^{α} (F) = |FΔ_{α} D|/|FΔ_{α}D|=1.

The knowledge creator g applies an iterative process, as in the algorithm in Figure 2 , where at any stage k, we randomly select feature-based data and process it to produce its belief structure. This basic probability assignments at stage k is compared to basic probability assignment in the previous stage k-1. While there are many other distance definitions you can use, we decided to use, due to their easier computations, one of the following distances:

Bhattacharyya Distance (m^k, m^{k-1}) = -Ln[$\sum_{wi\leq F} m^k(w_i).m^{k-1}(w_i)$].
Hellinger Distance (m^k, m^{k-1}) = 2 sqrt [1 - $\sum_{wi\leq F} m^k(w_i).m^{k-1}(w_i)$].

The idea here is that whenever this distance becomes smaller than an error factor prescribed by node owners and stays steady smaller for many iterations to go despite the observed variability in the big data, it is now good time to stop and accept the current belief structure we sought to have. This is the knowledge construct created by the generator.

At this point of time, no matter how much data we can still collect on the X_i's and no mater how much variability we can observe in the feature-based data collected at the IoT node, the basic probability assignments output will still stay the same. We say that the data subset randomly extracted from the IoT node give a stable representation of the entire data generated at the node with respect to the variable Xi's.

Algorithm:
Step 1: Randomly select feature-based data from data generated at the IoT node.
Step 2: k=1; I_0=0; I_1=Value prescribed by IoT node owners.
Step 3: As long as I_0<I_1, do:
Begin Step 3:
Step 3.1: Extract the data subset D_k containing the M tuples {d^k_1, …, d^k_M}.
Step 3.2: Construct the hypertuples in the cartesian product of power sets of domains of X_i, i=1,N; that is, the frame of discernment F= 2^{X1} x … x 2^{XN}.
Step 3.3: Construct the α-compatibility scores expressing the support of the extracted subset D_k to the hypertuples in the frame of discernment F.

We have: For any e in F: s_{Dk}^{α} (e) = |e Δ_{α} D_k|
Step 3.4: Contruct the basic belief assignment m_k as follows:
m_{Dk}^{α}: F → [0 1]
m_{Dk}^{α} (x) = $|s_{Dk}^{\alpha}$ (x)|/ $|s_{Dk}^{\alpha}$ (F)|
where s_{Dk}^{α} (F) = {{yεD_k such that $y\Delta_{\alpha}x$}, xεF}
$x\Delta_{\alpha}y$ if and only if $|x\cap y|/|x| \geq \alpha$
Step 3.5: If k>1:

Step 3.5.1: Apply Dempster rule to fuse the two basic probability assignments: m_{Dk}^{α} and $m^{\alpha}_{D1, \ldots, k-1}$.

$m_{1, \ldots, k}(z) = m_k \oplus m_{1, \ldots, k-1}(z)$
$= 1/(1-L_{k,k-1}) \sum_{X\cap Y= Z} m_k(X)m_{1, \ldots, k-1}(Y)$
where $L_{k,k-1} = \sum_{X\cap Y=\varnothing} m_k(X)m_{1, \ldots, k-1}(Y)$
Step 3.5.2: Compute ∂_k distance between m^k and m^{k-1}.

Step 3.5.3: If $|\partial_k|<\eta$ then $I_0=I_0+1$ else I_0=0.
Step 3.5.4: k=k+1.
End Step 3.

Step 4: The k extracted subsets $\{D_i\}_{i=1,k}$ are a good representation of the big data with respect to the feature-based attributes $\{X_i\}_{i=1,k}$. The basic probability assignments m(w) for w in F are a good representation for the knowledge representing the features sought at the IoT node.

IoT fog knowledge discovery

At this point in the IoT knowledge process, edge knowledge has been transferred to and accumulated the VLAN gateways waiting to be input in the fog knowledge generator to produce fog knowledge. We are now half way between the IoT edges and its cloud and IoT owners are now in a position when they can act upon fog knowledge instead of waiting until clouds are populated and hefty big data analytics executed.

Fog knowledge is represented in terms of belief structures at the virtual gateways where belief structures on edge knowledge created in IoT nodes of the VLANs. At each gateway, we apply Dempster Rule of combination of evidence on belief structures representing edge knowledge, as follows:

Given, for j=1,M, $m_j = \{2^{\Omega} \rightarrow [0, 1]; m_{ji}(\emptyset)=0; m_{ji}(A) \geq 0 \; \forall A \text{ in } 2^{\Omega}; \sum_{A \square \Omega} m_{ji}(A) =1\}$, we have:

$M_j: 2^{\Omega} \rightarrow [0, 1]$

$m_j(\emptyset)=0, \sum_{A \square \Omega} m_j(A) =1.$

$m_j(A)= \sum_{\mathbb{I} Ai=A} \Pi_{i=1,Nj} m_{ji}(A_i)/(1-c_j)$

$c_j= \sum_{\mathbb{I} Ai= \emptyset} \Pi_{i=1,Nj} m_{ji}(A_i)$

IoT cloud knowledge discovery

IoT cloud is only valid for the features selected in defining the virtual subnetworks. Feature-driven data collected at IoT nodes members of the the virtual LANs was processed to create edge knowledge that is in turn processed to obtain fog knowledge. Edge decisions are made using edge knowledge for problems associated with the features that owners selected for their decision process. Fog knowledge is concerned with decisions that concern the virtual LANs. The cloud knowledge covers decisions related to the entire IoT bust limited to features extracted with the data population process at the IoT nodes.

At the IoT cloud, we have accumulated fog knowledge, at every virtual gateway, which is represented in Belief stuctures. The IoT knowledge is obtaing by applying Demspter Rule of combination of evidence to the available fog knowledge. his type of knowledge will represeis point in the IoT knowledge process, edge knowledge has been transferred to and accumulated the VLAN gateways waiting to be input in the fog knowledge generator to produce fog knowledge. We are now half way between the IoT edges and its cloud and IoT owners are now in a position when they can act upon fog knowledge instead of waiting until clouds are populated and hefty big data analytics executed.

Cloud knowledge is represented in terms of belief structures fusing the fog knowledge created at the virtual gateways. We apply Dempster Rule of combination of evidence on belief structures representing fog knowledge, as follows:

Given, for j=1,M, $m_j = \{2^{\Omega} \rightarrow [0, 1]; m_j(\emptyset)=0; m_j(A) \geq 0 \; \forall A \text{ in } 2^{\Omega}; \sum_{A \square \Omega} m_j(A) =1\}$, we have:

$m: 2^{\Omega} \rightarrow [0, 1]$

$m(\emptyset)=0, \sum_{A \square \Omega} m(A) =1.$

$m(A)= \sum_{\mathbb{j} Aj=A} \Pi_{i=1,M} m_j(A_j)/(1-c)$

$c= \sum_{\mathbb{j} Aj= \emptyset} \Pi_{i=1,M} m_j(A_j)$

Some examples

There are many examples that come to mind where IoT knowledge is needed to conduct edge processing on the state of machines, fog computing on the VLANs connecting the machines on different stations, and where cloud computing is conducted to apply data analytic on big data in the cloud to create and maintain IoT knowledge on different features of interest to IoT owners.

Example 1, in Figure 4, depicts a print factory that consists of four main stations: Storage, Logistics, Warehouse, and Delivery. All the stations have multiple sensors generating information on features indicating the states of machine connected to the station. The state produce data that is used to produce machine state knowledge in terms of belief structure. At every station belief structures on the states of machines in the station are processed at the gateway of the VLAN containing the station to produce knowledge on the entire station. Knowledge produces at the stations are combined to produce general knowledge on the IoT on the print factory. The reader needs to follow the computation process explained in details above in the paper. Example 2, in Figure 5, shows how virtual subnets are designed and how knowledge creation is created throughout.

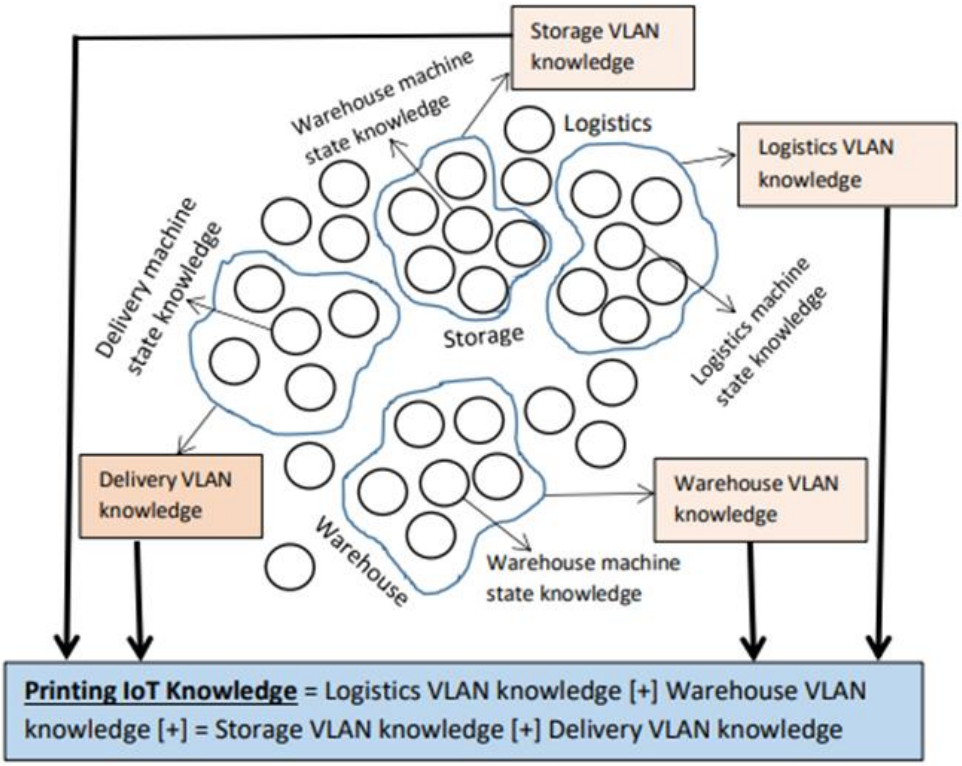

Figure 4: Example 1: A printing factory

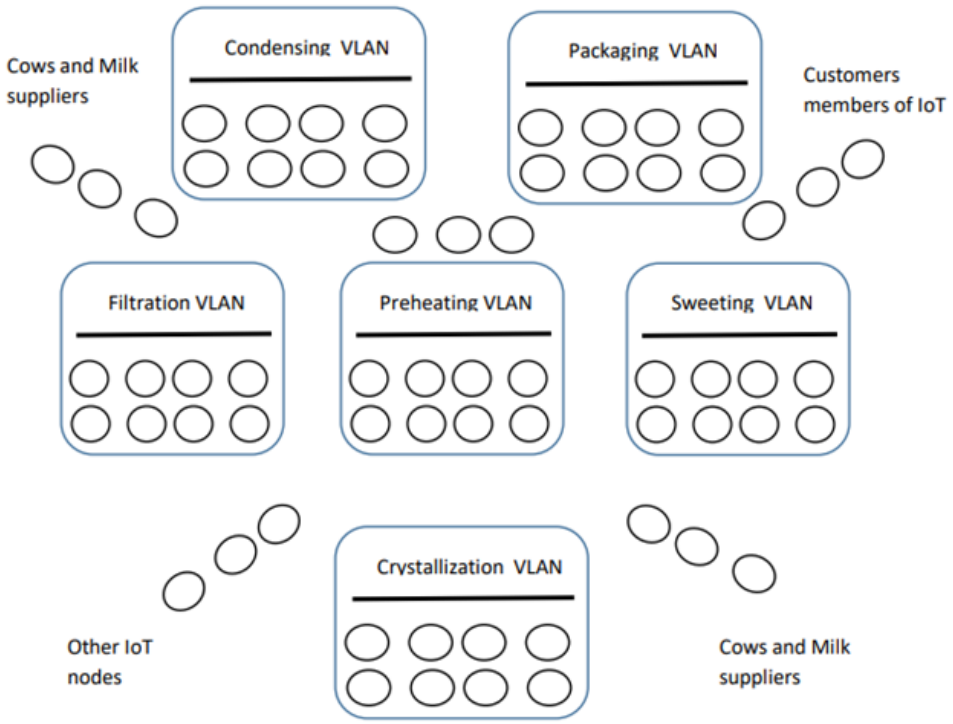

Figure 5: IoT Business example of a milk factory

II. Conclusion

The paper discussed feature-driven IoT knowledge creation for the purpose of establishing a lasting business value generation capability. IoT, itself, is of no consequential value unless it produces business value generation capabilities, often in a feature-driven manner, as knowledge comes in features of interest to IoT owners. We proposed a feature-driven IoT lifecycle and its knowledge creation and propagation process. Knowledge is edge-computed at IoT nodes to create edge knowledge needed by operational management, cloud-

computed at IoT clouds to create cloud knowedge needed by strategic management, and fog-computed half-way at IoT virtual gateways to create fog knowledge, often needed by functional management.

We represented knowledge as belief structures given features preselected by IoT owners, in a Dempster and Safer manner, and propagated using the Dempster rule of combination of evidence generated at IoT nodes, IoT virtual gateways, and IoT clouds. We showed some examples that the reader can treat using fictitious numbers on their own to demonstrate the working of the IoT feature-driven knowledge creation process.

References

[1]. Aguirre et al., (2013), Construction of Belief Functions From Statistical Data About Reliability Under Epistemic Uncertainty, IEEE Transactions on Reliability, 62(3).
[2]. Ashton. K., (2009), That internet of things thing. RFiD Journal, 22(7):97–114.
[3]. Dempster, A. P. (1969). Upper and lower probability inferences for families of hypotheses with monotone density ratios. Ann. Math. Statist. 40(3), 953–969.
[4]. Axelrod. C.W., (2015), Enforcing security, safety and privacy for the internet of things. In Systems, Applications and Technology Conference (LISAT), 2015 IEEE Long Island, pages 1–6. IEEE.
[5]. Dempster, A. P. (2008). Dempster–Shafer calculus for statisticians. *International Journal* of *Approximate* Reasoning, 48, 265–277.
[6]. Denoeux, T. (2006). Constructing belief functions from sample data using multinomial confidence regions. *International Journal* of *Approximate* Reasoning, 42, 228–252.
[7]. Giusto, D. et al., (2010), The Internet of Things 20th Tyrrhenian Workshop on Digital Communications. Springer New York Dordrecht Heidelberg London.
[8]. Raggad, B. and S. Mansouri, (2017), Evidential Modeling for Telemedicine Continual Security, International Journal of Computer Science and Network Security, ijcsn.org/publications.html.
[9]. Raggad, B. and S. Mansouri, (2017), Risk of Telemedicine Infeasibility: An Evidential Reasoning Approach, International Journal of Medical Research & Health Sciences.
[10]. Shafer, G. (1976). A Mathematical Theory of Evidence, Princeton Univ. Press, Princeton, NJ.
[11]. Smets, P. and R. Kennes, (1994), The transferable belief model, Artificial Intelligence, 66. 191-234.
[12]. Smets P., (1990), The combination of evidence in the transferable belief model. IEEE-Pauern analysis and Machine Intelligence, 12, 447-458.
[13]. Tedeschi et al., (2019), A Design Approach to IoT Endpoint Security for Production Machinery Monitoring, Sensors 2019, 19(10).

Bel G. Raggad, et. al. "Edge-Fog-Cloud Computing for IoT Knowledge Creation." *IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE)*, 16(4), (2021): pp. 30-37.