Future Technologies Conference (FTC) 2017
29-30 November 2017| Vancouver, Canada

Knowledge Management Metamodel from Social Analysis of Lessons Learned Recorded in the Cloud

Lopez-Quintero JF'
Researcher Groups
jflopezq@hotmail.com

Medina V.H.
Universidad Distrital
Department of Systems Engineering
vmedina@udistrital.edu.co

Montenegro-Marín C.
Universidad Distrital
Department of Systems Engineering
ingcarlosmontenegro@gmail.com

Nieto, Y.V.
CUN, AXON Group
Department of Systems Engineering
Yuri_nieto@cun.edu.co

***Abstract*—This article describes the development of a prototype of a metamodel for Personal Knowledge Management (GCP), which is defined and implemented based on the "Lessons Learned", registered on a social network for mass use. The functional architecture is applied in the implementation of a registration system and personal lessons learned in the cloud, through a social network: Facebook. The process begins with the acquisition of data from the connection to a non-relational database (NoSQL) in which it has set up a complementary analysis algorithm for the semantic analysis of the recorded information, on the lessons learned and thus study the generation of Organizational Knowledge Management (GCO) from the GCP. The end result is the actual implementation of a functional architecture to integrate a web 2.0 application and an algorithm of semantic analysis from unstructured information using techniques of machine learning and to demonstrate a way to make management of organizational knowledge through personal knowledge management.**

Keywords—Knowledge management; personnel knowledge management; lessons learned; semantic analysis; cloud computing; social networks; machine learning

I. Introduction

One of the main influences to knowledge management (KM) that has gained traction in recent years is personal knowledge management (PKM), which is supported as a process previous to managing organizational knowledge. This work is focused on the design and implementation of a functional architecture for knowledge management as a basic tool for integrating systems supported in cloud computing through social networking systems. The work aims to demonstrate the possibility of proposing a knowledge management metamodel (KMM) through a prototype implemented in the social network Facebook application, which demonstrates the possibility of managing organizational knowledge from personal knowledge management; the latter is supported on the lessons-learned concept.

The development of the technical aspects that define the architecture are shown in the first section, which begins with the conceptualization of aspects of Metamodels [1] and the phases that should be accounted for its construction. In the next section, the definitions of knowledge management (KM) and personal knowledge management (PKM) are elaborated on. This work deepens on and shares the new research trend that supports that PKM is the basis for achieving a real KM, based on flexible scenarios that are required to support the knowledge generated by each person or individual [2]. At the end of this conceptualization phase similar works are shown, in which other developed metamodel and some discussions have been held on the subject in recent years, which are similar to the lessons-learned concept [3] since it is the type of knowledge involved in this research.

The second part is focused on describing in detail the design, implementation and testing of the prototype for the Facebook social network, on the functionality of each component used methods and mainly on interfaces. In addition, the implementation is shown in the cloud using a non-relational database, which supports the actual record of a permanent random number of lessons learned, which applies as evidence to the concept of flexibility scenarios in PKM and to the functionalities of the proposed architecture.

The last section shows the second component of the model applied, which refers to the ontological analysis system, as a tool and technique of Big Data that verifies the real possibility of producing an ontological model of knowledge (OMK) from personal knowledge management (PKM). Examples implemented in the prototype display the use of generated strengths in socializations on social networks in recent years, the ability to combine structured and unstructured knowledge for OMK, and ultimately they demonstrate a way to generate organizational knowledge.

II. Context

A. Personal Knowledge Management and Lessons Learned

Tacit knowledge as a first state of knowledge [4], has distinctive features that have been characterized to define management strategies. This knowledge can be divided into knowledge that has not yet been formalized and knowledge that cannot be formalized [5]. The knowledge that can be formalized and explicitly described is particularly characterized as the "know-how"; it is also called "Tacitus Cognitive" knowledge and when specified through any tangible medium becomes the so-called "Explicit Knowledge". From this concept, Knowledge Management (KM) theories focus on the mechanisms that maintain knowledge within organizations [6]. In relation to all these theories, there is a tendency for one last job called personal knowledge management (PKM), which according to [7], [8], is a trend that complements and rethinks the dynamics of research and formalization of the KM at the organizational level. This trend summarizes the fact that, the *"person"* is the center of knowledge generation and that there is a need to facilitate the environment so that it can record, organize and collaborate with the generation of new knowledge.

To support the principle of "the adequacy of the environment", necessary for knowledge management, PKM is based on the Web 2.0 concept through a set of tools that allow people to create, encode, organize and share knowledge, but also to socialize, expand personal networks, collaborate in the organization and to create new knowledge [7], [8]. These authors express characteristics that influence a system of personal knowledge management (PKM) using Web 2.0 resources as a mechanism of communication and socialization online. This last interaction rests on the basis that there is a prospect of creation related to the Entity-Expert [4]; which defines collective knowledge as a build-up of individual knowledge. This build-up cannot be generalized as a sum of elements in a linear fashion, for its development creates a synergy which states that the collective knowledge is a more complex process involving structures, dynamics and relations [9].

On the other hand, knowledge sharing has been explained using the social capital framework, known as transfer of knowledge in organizations [3]. One such mechanism is called lessons-learned, that can be defined as a kind of explicit knowledge resulting from errors or strengths that were obtained during the implementation of a process of knowledge, or the possibility of innovation in a given context [10].

Lessons-learned can also be defined as a kind of knowledge that comes from experience, through complex processes, systemic, asynchronous and individual reflection [11]. For knowledge transfer to meet the needs of organizations, it requires lessons- learned to be presented in a specific moment and context, that is, to ensure the principle of opportunity. Thus, knowledge generated can be reused [12].

B. Analysis of Social Behavior in Knowledge Management

The analysis of social behavior using semantic techniques is considered a new paradigm in knowledge management in organizations. Recently, the use of data and information extraction from structured sources such as Web 2.0 applications is gaining ground in the study of the social web [13]. There have been reports of interest and publications in the field of integration of social networks and their analysis. This new semantic approach allows dynamic change towards a semantic social networking and establishment of knowledge management models from people in organizations.

Modern organizations have never had new needs and opportunities to leverage their knowledge, in a faster and efficient way, from implementations supported by semantic analysis applications. Building sophisticated knowledge bases, decision support system and other intelligent systems often takes considerable time and [14] economic resources. In order to extract the semantics of the underlying social structure of behavior, preferences and trends of people, studies such as those made by [15] have implemented various web mining techniques. Even though it is important to analyze existing online social networks, the process of extracting data and information related to user profiles, supported by social web applications from structured sources, causes inevitably a loss of the actual semantics of the social system.

C. Similar Studies

A metamodel for knowledge management abstracts and describes, the essential entities related to knowledge management (KM) and the connection between these, as a concrete model represented in an architecture or in a graphic display. A number of authors have discussed what has been stated above.

One of the most studied metamodels for knowledge management and from which others can be derived from is that posed by [16]. This model is supported in six entities which expand its scope and in turn allows you to add other essential entities of a domain of interest, as well as the relationships among them.

In [17], authors suggest a system based on a framework of model-driven architecture (MDA). This allows mitigating and reducing the effect of the loss of relationships between requirements and behavior of this relationship, against elements of the multidimensional models metamodel conceptual and data source in the process.

In most metamodelling environments, domain models cannot be uploaded properly in the modeling tool without a corresponding metamodel. To do this [18] worked in a metamodel recovery system using grammatical inference algorithms aiming to infer a new one from a collection of instance models. The motivation behind the problem was to focus itself on the derivative metamodel, which happens when the domain models in a repository are separated from the definition of the problem.

Other authors like [19] propose systems to extract information from databases of business using a model of formal ontologies. The system uses this information to build a knowledge base in RDF format; it is based on the transformation between different models: SQL ontology engineering techniques using models (MDE).

In a complementary manner, the opportunity to use lessons-learned also depends on the systematic aspects that integrate, manage, support and streamline knowledge management. On these aspects, it can be said that an effective system should be: 1) interested in staff; 2) requiring topic generation and consultations; 3) related to experts; supporting interaction and management flow [20].

In addition, the conclusions of the related works recommend that any process of generating lessons-learned can be supported by information systems with databases that allow including diversity of models and objects of knowledge [11]. This aims to facilitate fast and accurate location information query required for knowledge, in order to distribute, subsequently, and access time information of all stakeholders involved, in the context or situation in which you are working [21].

III. Metamodel of Knowledge

A. Proposed Architecture

Metamodel KM (Knowledge Management) is applied and derived from two references. The first metamodel developed by [16] describes its six (6) structure entities: People, processes, documents, topics, tacit knowledge and explicit knowledge. The second metamodel is "Collaborative Model-based software process engineering metamodel: a metamodel for Processes collaborative". It is defined as CMSPEM [22], which describes the process of collaboration in software projects engineering. This is assuming that the implementation of metamodel CMSPEM facilitates collaborative support, thanks to the combination of information on process models with services offered by other tools that support collaboration; this principle is reflected in paragraph three where the results of the implementation of the proposed metamodel prototype called QUIRISYA are discussed.

Advanced processing techniques to be applied metadata through techniques '*Big data'* are *included.* Fig. 1 shows the flow or relationship between elements or entities in the metamodel exposed: data-information-knowledge. The metamodel proposal integrates entities, which are shown in the following figure.

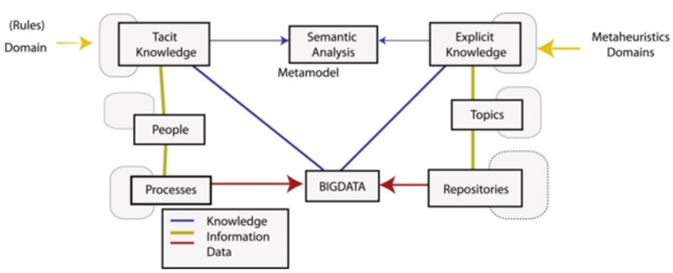


Fig. 1. Global metamodel architecture of KM (own source).

The above architecture is structured so that each of the components are interrelated to be transmitted within the organizational string comprised by data, information and knowledge. This is understood through:

– *Processes:* The definition of the necessary processes for interaction of people and resources or platforms that make up the system.

– *People:* The metamodel should be based on interaction of tacit knowledge from lessons-learned from each participant.

– *Repositories:* The opportunity to develop the prototype model and the necessary documents and evidence of these lessons and of the model itself.

– *Topics:* These are depicted on the possibility of defining categories and themes for knowledge generation, from different people profiles whom interact with the system.

– *Tacit knowledge:* It is described and shown in every human being, through lessons- learned that exist inside them.

– *Explicit knowledge:* It is reflected through the addressing of lessons-learned into new forms of knowledge, by using advanced processing techniques.

– *Semantic Analysis:* The process of semantic analysis is applied on an unstructured basis i.e. on a set of terms in a particular domain in textual format. Each analyzed data behaves as a taxonomy: the process is responsible for identifying key terms and classifies containing vocabulary terms within the database. This is done in order to for the system to perform an easier search. Taxonomy groups several terms around a set of concepts, then maps and fractions them through flow of text mining implemented in 'KNIME'. Thus, the process of semantic analysis facilitates the inference of the concepts that the user is interested in, even if those concepts are not explicitly between the terms of the user.

Equally important as the structure, the modeling of relationships between components is found [23]. This relationship is defined by the lines shown in the architecture, which use a sequence of data, information and knowledge. These three elements circulate through each of the model entities. In the top tier, a running semantic analysis of sources fed with tacit and explicit knowledge can be evidenced. Both, tacit and explicit knowledge collect information from people and their job issues, respectively. These, in turn, receive and process information repositories, while simultaneously providing data to be processed by big data techniques. Next, in a more detailed way, the relationships of metamodel components are described:

– *People - Processes:* A relationship that describes the activation of different processes leading, naturally, any individual to generate data, information or knowledge.

– *Processes - big data - Repositories:* This relationship is based on the flow that leads to enabling information analysis algorithms reflected by big data techniques.

– *People - tacit knowledge:* This is the natural process of knowledge management generated when individuals take their lessons according to different scenarios and categories that have been set.

– *Big data - Explicit Knowledge:* This is the relationship based, once applied techniques big data (semantic analysis) explicit knowledge is generated in two stages, the first when the person or individual records in the system lessons learned and the second, when the system transforms and presents the analysis of the same from an organizational point of view.

– *Repository - Topics - Explicit Knowledge:* This relationship describes the design to be done naturally, but consistently; it includes the organization of lessons-learned topics and forms of organization, in the unstructured database.

Therefore, features they want to take from lessons-learned for this metamodel are: flexibility, diversity in their time of generation, and the ways they can be generated. These characteristics correspond to those previously associated with the PKM exposed. As a consequence, lessons-learned can be reaffirmed as a type of knowledge that represents the characteristics of PKM systems.

B. General Model: QIRISYA

The prototype (QIRISYA as will be known from now on) will represent the metamodel, as a practical application itself. It is generated as a conception of the Lessons. Learned model, in a social network environment as described above in Fig. 1.

For the implementation phase, a software has been developed to record lessons-learned by each user or person defined within templates. This recording is done from a tree structure that starts with the profile, the categories and subcategories. These three levels may be configured, extended, modified and added by each individual in a personalized way.

Since the prototype is built for a social network, where a lot of contributions from lessons are always expected, a non-relational database was used to avoid the possibility of saturation of the application. Therefore, the system as seen in Fig. 2 consists of: 1) user or knowing-entity; 2) the application on the social network (Facebook): 3) the platform supports the application: non-relational database; and 4) the semantic analysis system.

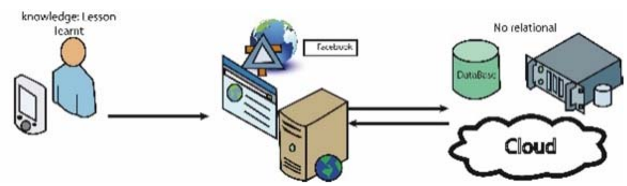


Fig. 2. Diagram of the prototype (own source).

The user records lessons-learned in an explicit knowledge form by describing them with written words. Knowledge or acquired experience is registered in a desired profile category and subcategory. These lessons don't have a word limit; they can be described as reflections, both simple and complex. The former may be very specific statements, while the latter may be analysis on project participation in a project or in another reflection on professional, personal, educational activity, etc.

To complement the prototype, an application on semantic analysis has been developed in order to identify and quantify accurately, the trends in knowledge generation. This is done by using keywords and configuring the app for group analysis in a social network (Facebook); thus it facilitates characterization of generation of knowledge in teamwork. In other words, the latter module allows carrying out personnel knowledge management in an organizational knowledge management application.

For the development of a functional lessons-learned application, it was necessary to use different libraries that allow making connections and structured designs. Among them, some libraries found are: Facebook SDK PHP [24], Fancybox API [25], GoogChart [26], jQuery API, jsDatePick [27] and AWS SDK for PHP [28].

C. Design and Semantic Analysis Algorithm of a Functional Prototype

Social analysis supported on semantics is a type of explicit specification of a shared formalization of behavior. It represents concepts, objects and other entities that are supposed to exist in an area of interest along with their relationships [13]. Semantic analysis studies the conceptual aspects of social graphs; it is set in knowledge engineering and, particularly, in the semantic web along with the text mining and web mining principles.

The framework includes six entities, a knowledge discovery process and a semantic analysis process. It is important to note that both processes have a common element in a data repository of JSON and XML/RDF formats. Advanced processing techniques to be applied as metadata through Natural Language techniques are included. Fig. 3 shows the developed framework and the relationship among the elements or entities which comprise it.

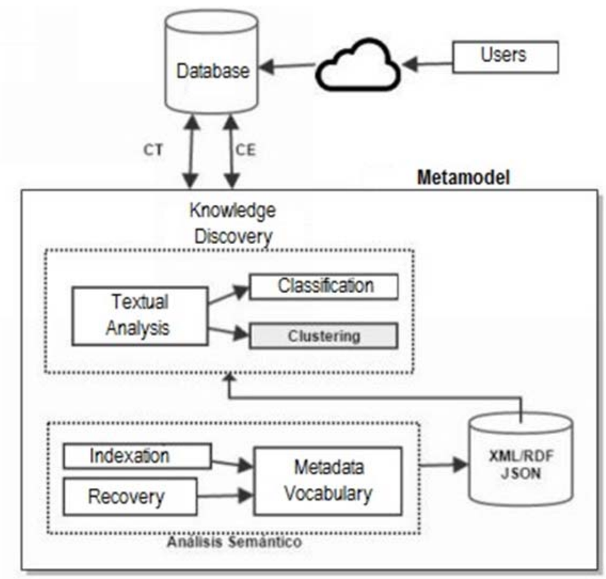


Fig. 3. Developed metamodel architecture (own source).

Below, each of the main elements of the framework along with its relationships and data flow, information and knowledge is explained:

1) Semantic Analysis

a) Recovery and Indexation

The Textalytics API performs indexation and recovery of information. This API is a component of text analysis and is capable of extracting meanings, concepts, topics and feelings of any text structure. It is also able to understand the content of any text and its main features are: topics analysis; identification of relevant keywords and concepts; thematic classification, identification of key data (dates, physical and virtual addresses and financial amounts) and content enrichment with related information (Fig. 4). For this work, without a doubt, Textalytics' most important feature is that it gives the user the ability to create custom dictionaries and other language resources.

The services offered by this API are listed below in Fig. 4:

Service	XML	Access point
Semantic labeling	<?xml version = "1.0" encoding = "UTF-8"?> <dictionary xmlns = http://textalytics.com/api/sempub/1.0/schemas/xml/dictionary-1.0-input xmlns: xsi = "http://www.w3.org/2001/XMLSchema-instance"> <Name> idi </ name> <Language> is </ language> <description> <! [CDATA [Management i + d + i]]> </ description> </ Dictionary>	https://textalytics.com/api/sempub/1.0/semantic_tagging
Proofreading		https://textalytics.com/api/sempub/1.0/check
Resource management		https://textalytics.com/api/sempub/1.0/manage

Fig. 4. Services and Access Points - Semantic Publishing AP.

The first step in the development of this metamodel was to create a dictionary, for which Resource Management API was used. The format selected for the creation of this dictionary was XML and it has the following structure: An indexed payload Python dictionary containing the following elements: "Key", it is an important parameter of the API and must be purchased from the Textalytics website; "dictionary", it indicates the object where the previously created XML dictionary is loaded and it requires the using of libraries for reading XML data in python. It can also be seen that a definition of the data input and output format can be done, with "input" and "output", respectively. In this case the XML format was used in both cases. You should specify the service address endpoint, in this case, it contains the address. Finally, if all the parameters for the service request is possible to do, this is done through the "requests.post" command of the REST architecture. The process only defines the name of the dictionary and the language it is in. However, the dictionary concepts have not yet been related so another XML has to be created, with the following structure:

payload *= { 'key': '1bbef26a43c34b453b1713039c49a513', 'dictionary': Dictionary, 'entity': entities, 'src', 'sdk-python-1.0' 'input' 'xml', 'output' 'xml' }*

endpoint *= 'http://textalytics.com/api/sempub/1.0/manage/dictionary_list/idi/entity_list'*

response *= requests.post (endpoint, data = payload)*

After creation of the XML file, it is required to load the previously defined dictionary. This is also done by a request. The following code fragment illustrates how this request should be made.

This request is very similar to the previous one with the exception of the **endpoint**. In this case, the previously created dictionary must be mentioned and a clarification of the created dictionary entities is in order. In both requests, the response is stored in **response**, if all parameters are well-declared. Assessing the status of the response, code 200 indicates the request and response were successful. Response is shown in Fig. 5.


```
<?xml version="1.0" encoding="UTF-8"?><response xmlns="http://textalytics.com/ap
i/sempub/1.0/schemas/xml/manage-1.0-output" xmlns:xsi="http://www.w3.org/2001/XM
LSchema-instance"><status><code>0</code><message>OK</message></status><id><![CDA
TA[idi]]></id></response>
```


Fig. 5. Recovery response and indexing of terms (own source).

b) Vocabulary and Metadata

For this exercise, vocabulary from the world of standardization will be taken. More relevant, it will be used in the implementation, development and improvement of management systems and project management phases. The concept of research, development and innovation management (R + D + i) has taken momentum, recently. Up to date, some sources that highlight specific models for these activities, as knowledge management systems within organizations can be found.

These models have characteristics that make them unique and characterize them; their origin comes from a focus on processes. One of the main features is that it is continually fed with data, information and existing explicit knowledge, that is, they are based on their operation on different levels of knowledge generation. Equally important, models require their strategy be based on key technological surveillance and competitive intelligence activities, so that they are really managed, worked on and planned. This allows them to respond to real needs where, often, they can be found.

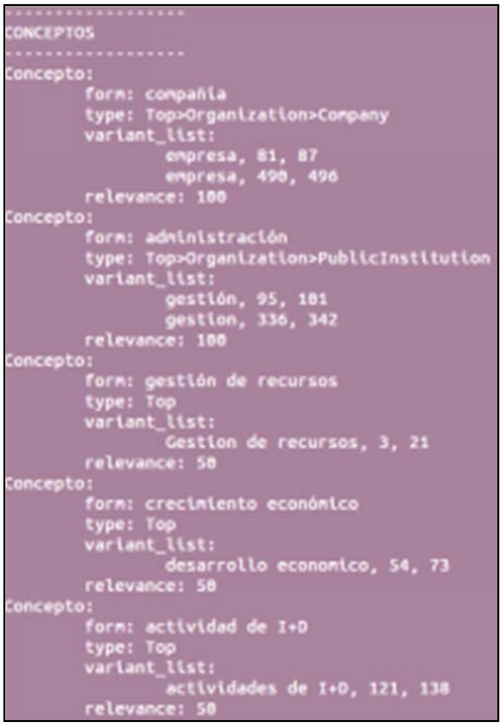
In the operational phases, these models are developed based on the concepts of risk and uncertainty developed in projects. Therefore, knowledge management reprises relevance in order to define scenarios, to establish controls and improve effectiveness of said elements. Also, these are identified because they are implemented and improved based on people's aptitudes, which reflect another form of implicit knowledge made explicit.

Last, it is worth mentioning these models show their impact from the establishment of systems or mechanisms for protection of generated knowledge. Any product, deliverable or outcome of projects should be protected or exploited so as to ensure realized knowledge management. It is for this reason that, keywords used in the development of this type of management systems have been selected for this job and that its use may become evidence of a true culture towards R + D + i, and therefore of knowledge management.

2) Repositories

The API delivers a JSON or XML file, where relevant concepts are related within the personal dictionary terms, with their respective relevance and location within the text *(variant_list)*. With the result above, it is possible to show trends in an organization, to people, when speaking about any subject, such as management of R + D + i. This may be done through a web service. Fig. 6 illustrates the entry result of the following fragment using the vocabulary created: "Resource management is very important for the economic development of the company. Management goes hand in ***hand with R + D + I***".

By stating the above, it means that when the word *"Resource Management" is shown, it* has a relevance of 100 and is located between 3 and 21 characters to label text. All this is because of the extracting of concepts (see Fig. 6).



```
-----------------
CONCEPTOS
-----------------
Concepto:
        form: compañía
        type: Top>Organization>Company
        variant_list:
                empresa, 81, 87
                empresa, 490, 496
        relevance: 100
Concepto:
        form: administración
        type: Top>Organization>PublicInstitution
        variant_list:
                gestión, 95, 101
                gestion, 336, 342
        relevance: 100
Concepto:
        form: gestión de recursos
        type: Top
        variant_list:
                Gestion de recursos, 3, 21
        relevance: 50
Concepto:
        form: crecimiento económico
        type: Top
        variant_list:
                desarrollo economico, 54, 73
        relevance: 50
Concepto:
        form: actividad de I+D
        type: Top
        variant_list:
                actividades de I+D, 121, 138
        relevance: 50
```

Fig. 6. Concepts output (own source).

3) *Knowledge Discovery*

a) Text analysis

After creating the dictionary with the R + D + i vocabulary, lessons-learned entries that are recorded in the Facebook application are labeled. The API receives a document as an input and labels it using the dictionary that was created in the previous section. The XML form necessary to place a request to the API is shown below.

Segments detection is applied; this text analysis process is examined and grouped for finding patterns and information related to semantic annotations. Most of the times, these segments are complete sentences in *"common"* texts (lessons-learned admitted to the Facebook social network). However, this case is not necessarily a pre-established structure. Because the analysis depends on the presence of punctuation, special characters and timely synchrony of terms is necessary, in order to apply natural language for structuring the text input to be analyzed. In Fig. 7, a text input to be analyzed in the API is shown. This entry corresponds to an XML format. The entry "Resource management is very important for the economic development of the company" is taken as a test set of terms for analysis.


```
<? xml version = "1.0" encoding = "UTF-8"?>
<Document xmlns = "http://textalytics.com/api/sempub/1.0/schemas/xml/document-1.0-input" xmlns: xsi = "http://www.w3.org/2001/XMLSchema-instance" xsi: schemaLocation = "http://textalytics.com/api/sempub/1.0/schemas/xml/document-1.0-input.xsd">
        <Id> 0 </ id>
        <txt> <! [CDATA [resource management is very important for the economic development of the company,]]> </ txt>
        <Language> is </ language>
        <Itf> plain </ itf>
        <Source> UNKNOWN </ source>
        <Timeref> 02.05.2013 16:08:00 GMT + 01: 00 </ timeref>
</ Document>
```


Fig. 7. Non-structured input source (own source).

r = requests.post ("http://textalytics.com/api/sempub/1.0/semantic_tagging" params = {"key":"1bbef26a43c34b453b1713039c49a513","output":"xml","input":"xml","do":Document,"filter_data":"y",'dictionary':'idi'})

b) Lemmatization

This process consists positions the direction of the semantic service, in this case: *http://textalytics.com/api/sempub/1.0/semantic_tagging*. As a configuration parameter it is necessary, as in every *request made* to the API, to place the *"key" so as to specify how to enter and get data to and from XML. Additionally, one must specify the document that was previously uploaded to the API.* Given that a dictionary aimed towards the labeling of lessons learned was created, it is necessary to place it in the request. Finally, an option that offers the TextAlytics API is to filter or not the data i.e. to perform extra lemmatization and Natural Language processes of the text to be labeled.

IV. Results and Discussion

The implementation and use of this framework work was done from several characterizations of users with different profiles and needs. The selected users have followed the procedure of linking the application and began to define their profiles and categories, while allowing to record lessons learned in the application (App).

In Fig. 8, a chronological behavior of progress is shown in the lessons learned record, in their respective categories and profiles within the framework established. It is observed as an increase in their level of interest in recording lessons learned in different categories, while the dynamic of use increases. For example, whenever a graph is done, users' lessons are recorded in six different categories without abandoning the aspects mentioned above, in less than five months. The system allows recording a great number of lessons and, by making use of various options (such as private and public records), information management from suppression lessons operations is achieved. Also, modification and display of statistics and recording of evidences, that classify lessons learned for different profiles and categories is accomplished.

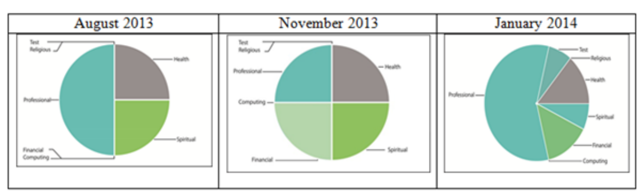


Fig. 8. Categorization diagram of lessons learned (own source).

This project has been visualized through the concept of *"Categorization diagram of lessons learned"* through a social network, as shown in Fig. 8 (Tacit knowledge→ Explicit knowledge). Therefore, it allows the profiling of knowledge generation for each person or group of people; this emphasizes or advances lessons learned in ranges or time periods defined by the person himself. Now, by applying an algorithm supported in textual techniques at a semantic level such as Latent Semantic Indexing LSI [29], an inquiry can be made about the trends and reality of knowledge generation made in a work team environment by disseminating lessons-learned from each of its members.

This research makes use of the TextAlytics API, which is a text analysis engine that is capable of extracting meanings, concepts, topics and sentiments of any text structure. One of their tools called "*Media Release*" is used in this work. This API is able to understand the content of any text; its main features are: topics analysis, identifying relevant keywords and concepts, thematic classification, identification of key data (dates, physical and virtual addresses and financial amounts) and enrichment of content with related information. But certainly, for this work, the most important feature of this API is that it gives the ability to create custom dictionaries and other language resources.

To develop the project of semantic analysis, the data structure of the lessons learned registered through the Facebook API and some Django and PostgreSQL tools are used. This allows the development of the platform to become a reality.

Derived from the *"lessons learned" structure* data models with certain variables that guarantee the correct mapping of information is created. This structure is imported and stored in the database with various formats like *CSV, TSV and XLS.* The models are synchronized with the database whenever the application is started. The database structure is as shown in Fig. 9.

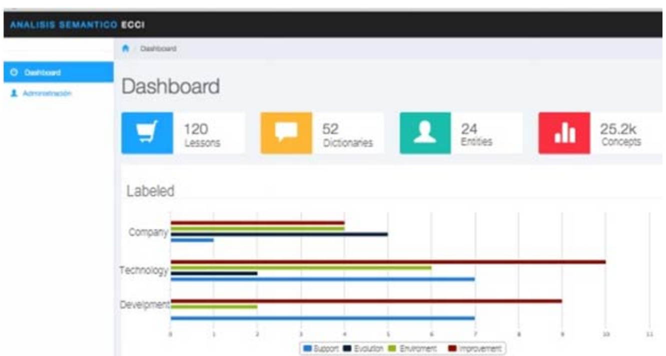


Fig. 9. Lessons learned output generated by the API (own source).

The application has an aggregate value: it provides a dashboard with various filters that help classify information, which is then displayed in graphs for observing the behavior trends on lessons learned about R + D + i.

The analysis of lessons is done by uploading the lessons to the TextAlityсs platform. It is a cloud platform that provides customers with natural language processing and semantic analysis services so they can extract: sentiment analysis; detection of irony; intentions (shopping signals); extraction of concepts; relationships between them; automatic classification; entity recognition; spelling, grammar and style checking; user profiling.

The process of sending the lessons to the API can be described as: checking of the processing state; sending of lessons learned in packets; receipt of the response on the TextAlytics App and storing of the response in a cloud repository. This repository updates the status of real-time processing and it ensures that this lesson will never be sent to the API, while also remaining latent for consultation by through the dashboard.

V. Conclusions

This work allows inferring that the development of empirical systems to implement personnel management knowledge from algorithmic techniques, supported by social semantic analysis, is a latent but real organizational tool that enables to manage and define betterment opportunities.

In particular, it is important to recognize that proposing an architecture that applies and adapts to the most used social network worldwide was successful. Also, an important achievement was that the proposal is supported by tools that facilitate its exponential growth as a database, similar to Azure's case from Microsoft.

The scope of this work is wide, both in its practical and research application. It arises from each of its functional and structural modules. For instance, from a functional viewpoint, it would be important to further deepen into the systemic use of the application in the knowledge generation routines, for each person who is involved in the process of generating lessons learned. One could also deepen into the dynamic generation of more lessons when modules are included, such as early alarms on the use of the application. From a structural point of view, it would be necessary to verify the characterization in using the database to measure efficiency in itself. This would be an indicator of how far it is from reaching a commercialization process and massive use of the prototype at the organizational level.

For large volumes of data, a semantic analysis applied in a bigdata tool allows to inquire how dynamic capabilities are being generated for a team or work group for KM, and what is the profile that develops inside it. Everything from a systematic analysis of the individual or personal profiles from each and every one of their members, placed on a timeline conveniently.

As commented by some of the referenced authors in this paper, development of the appropriate and integrated spaces should continue. This is where each person or individual can feel comfortable, thus, it will facilitate knowledge flows that allow self-recognition and, a generating capacity of translating their tacit into explicit knowledge. This influences the person to collaborate with learning and development objectives, in contexts or fields in which knowledge is developed.

Finally, this model would be the first version to reach a very powerful version of a non-probabilistic, predictive model of Semantic Social Network Analysis (SNA) acting as a basic component of the proposed metamodel in the functional architecture.

References

[1] E. Montenegro Marín, P. A. Gaona García, J. M. Cueva Lovelle y O. Sanjuan Martínez, "Application of Model-Driven Engineering (MDA) for the construction of a tool for domain -specific modeling (DSM) and the creation of modules in Learning Management Systems (LMS) platform independent", Dyna, vol. 169, pp. 43-52, 2011.

[2] S. Nasiri, F. Ansari y M. Fathi, “Dynamics of knowledge assets and change management perspectives”, de Electro/Information Technology (EIT), 2013 IEEE International Conference on, 2013.

[3] G. Widén, “Social Capital and Knowledge Sharing - Lessons Learned”, de Adaptation and Value Creating Collaborative Networks, vol. 362, L. Camarinha-Matos, A. Pereira-Klen y H. Afsarmanesh, Edits., Springer Berlin Heidelberg, 2011, pp. 48-57.

[4] Nonaka y V. Peltokorpi, “Objectivity and subjectivity in knowledge management: a review of 20 top articles”, Knowledge and Process Management, vol. 13, nº 2, pp. 73-82, Apr 2006.

[5] N. K. Kakabadse, A. Kouzmin y A. Kakabadse, “From tacit knowledge to knowledge management: leveraging invisible assets”, Knowledge and Process Management, vol. 8, nº 3, pp. 137-154, Jul 2001.

[6] M. R. Haas y M. T. Hansen, “Different knowledge, different benefits: toward a productivity perspective on knowledge sharing in organizations”, Strategic Management Journal, vol. 28, nº 11, pp. 1133-1153, Nov 2007.

[7] L. Razmerita, K. Kirchner y F. Sudzina, “Personal knowledge management: The role of Web 2.0 tools for managing knowledge at individual and organisational levels”, Online Information Review, vol. 33, nº 6, pp. 1021-1039, 2009.

[8] Pauleen, “Personal knowledge management: Putting the person back into the knowledge equation”, Online Information Review, vol. 33, nº 2, pp. 221-224, 2009.

[9] J. Spender, “Method, philosophy and empirics in KM and IC”, Journal of Intellectual Capital, vol. 7, nº 1, pp. 12-28, 2006.

[10] R. Brent, M. Meruvides y D. Ben, “A software-based lessons learned management systema: Enhancing knowledge management in organizations”, IIIE Annual Conference Procedings, pp. 787-792, 2008.

[11] S. Greer, “A lessons-learned knowledge management system for engineers: an organizational lessons-learned system facilitates the transfer of knowledge from one project team to another.(Engineering Practice)”, Chemical Engineering, 2008.

[12] M. M. Richter y R. O. Weber, “Knowledge Management”, Case-Based Reasoning, pp. 487-505, 2013.

[13] M. Schatten, “Knowledge management in semantic social networks”, Comput Math Organ Theory, vol. 19, nº 4, pp. 538-568, Dec 2013.

[14] J. Breslin y S. Decker, “The Future of Social Networks on the Internet: The Need for Semantics”, IEEE Internet Computing, vol. 11, nº 6, pp. 86-90, 2007.

[15] Cantador y P. Castells, “Multilayered semantic social network modeling by ontology-based user profiles clustering: application to collaborative filtering”, Managing Knowledge in a World of Networks, pp. 334-349, 2006.

[16] Ammann, “A meta-model for knowledge management”, de Proceedings of the 5th International Conference on Intellectual Capital, Knowledge Management and Organisational Learning, 2008.

[17] T. Maté y J. Trujillo, “A trace metamodel proposal based on the model driven architecture framework for the traceability of user requirements in data warehouses”, Information Systems, pp. 753-766, 2012.

[18] Javed, M. Mernik, J. Gray y B. Bryant, “MARS: A metamodel recovery system using grammar inference”, Information and Software Technology, pp. 948-968, 2008.

[19] Lahoud, D. Monticolo, V. Hilarie, S. Gomes y E. Bonjour, “A multi-sources knowledge management system”, IFAC Proceedings Volumes, pp. 1177-1183, 2012.

[20] D. F. Fahey y G. Burbridge, “Application of Diffusion of Innovations Models in Hospital Knowledge Management Systems: Lessons to Be Learned in Complex Organizations”, Hospital Topics, vol. 86, nº 2, pp. 21-31, Apr 2008.

[21] Cardenas y M. Spinola, “Role of Knowledge Management Systems within the expertise transfer”, de Technology Management in the IT-Driven Services (PICMET), 2013 Proceedings of PICMET '13:, 2013.

[22] Kedji, R. Lbath, B. Bounabat, K. Benali y M.-P. Gervais, “Modélisation et mise en œuvre de processus collaboratifs ad hoc”, l'Université Mohammed V-Souissi (Rabat, Maroc), Toulouse, 2013.

[23] B. Barros, M. Verdejo, T. Read y R. Mizoguchi, “Applications of a collaborative learning ontology”, MICAI 2002: Advances in artificial intelligence, pp. 103-118, 2002.

[24] Facebook, “Facebook SDK PHP”, 2015. [En línea]. Available: https://developers.facebook.com/docs/reference/php/#reference.

[25] T. FancyApps y J. Skarneli, “Fancybox API”, 2015. [En línea]. Available: http://fancyapps.com/fancybox/#license. [Último acceso: 16 Marzo 2015].

[26] Pettersson, Googchart: PHP Google Chart Class, 2008.

[27] *Arjuan y P. Walsh, “jsDatePick”, 2015. [En línea]. Available: http://javascriptcalendar.org/. [Último acceso: 16 Marzo 2015].*

[28] Amazon Web Services, AWS SDK for PHP, 2013.

[29] Suárez Barón y S. Kathleen, “An approach to semantic indexing and information retrieval”, Revista Facultad de Ingeniería Universidad de Antioquia, vol. 48, p. 174 – 187, 2009.