Advanced Methods for Complex Network Analysis

Natarajan Meghanathan
Jackson State University, USA

A volume in the Advances in Wireless Technologies and Telecommunication (AWTT) Book Series

Information Science REFERENCE
An Imprint of IGI Global

Published in the United States of America by
Information Science Reference (an imprint of IGI Global)
701 E. Chocolate Avenue
Hershey PA, USA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-global.com
Web site: http://www.igi-global.com

Library of Congress Cataloging-in-Publication Data

Names: Meghanathan, Natarajan, 1977- editor.
Title: Advanced methods for complex network analysis / Natarajan Meghanathan, editor.
Description: Hershey, PA : Information Science Reference, [2016] | Includes bibliographical references and index.
Identifiers: LCCN 2015050776| ISBN 9781466699649 (hardcover) | ISBN 9781466699656 (ebook)
Subjects: LCSH: System analysis.
Classification: LCC QA402 .A295 2016 | DDC 003/.72--dc23 LC record available at http://lccn.loc.gov/2015050776

This book is published in the IGI Global book series Advances in Wireless Technologies and Telecommunication (AWTT) (ISSN: 2327-3305; eISSN: 2327-3313)

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

Chapter 17
A Network Analysis Method for Tailoring Academic Programs

Luis Casillas
University of Guadalajara, Mexico

Thanasis Daradoumis
University of the Aegean, Greece & Open University of Catalonia, Spain

Santi Caballe
Open University of Catalonia, Spain

ABSTRACT

Producing or updating an academic program implies a significant effort: involving people, academic units, knowledge elements, regulations, institutions, industry, etc. Such effort entails a complexity related to the volume of elements involved, the diversity of the origins of contributions, the diversity of formats, the representation of information, and the required granularity. Moreover, such effort is a common task performed by humans who collaborate for long periods of time participating in frequent meetings in order to achieve agreement. New educational approaches are heading to adaptive, flexible, ubiquitous, asynchronous, collaborative, hyper-mediated, and personalized strategies based on modern Information and Communication Technologies (ICT). We propose an approach for tailoring academic programs to provide a practical and automated method to discover and organize milestones of knowledge through the use of Complex Networks Analysis (CNA) techniques. Based on indicators from CNA, the act of tailoring an academic program acquires meaning, structure and even body elements.

INTRODUCTION

As most people would agree, there is a continuous digital revolution due to advances in computational capabilities, effective communication protocols, new understanding on data managing/mining/sharing as well as distributed programming approaches; which have allowed achieving standardize new ways to carry out common computer tasks, all over the world. As a consequence, societies nowadays have specific demands on digital services, so significant amounts of knowledge elements have been digitalized. In

DOI: 10.4018/978-1-4666-9964-9.ch017

the same way, current web-infrastructure is able to support new approaches in scientific and academic developments (Isaila, 2012). In educational institutions, academic programs are the mechanism to manage and lead scientific and academic efforts. Such programs require regular updating in order to provide certainty to students, institutions, industry, and even to society.

Updating an academic program implies a significant effort, involving people, academic units, knowledge pieces, regulations, institutions, industry, and etcetera. Such effort entails a complexity related to the volume of elements involved, the diversity of the origins of contributions, the diversity of formats and representation of information, and the required granularity. Moreover, such effort is a common task performed by humans who collaborate for long periods of time participating in frequents meetings in order to achieve agreement. A preliminary work that aimed to develop an automated method for tailoring academic programs was performed in (Casillas & Castillo, 2012). Here, the authors extend this previous work, by including new formalizations and features that leaded to a more integrated and effective approach to model and customize academic programs.

Indeed, the new educational approaches are heading to adaptive, flexible, ubiquitous, asynchronous, collaborative, hyper-mediated, and personalized strategies based on modern information and communication technologies (ICT). This approach for tailoring academic programs, focuses on providing a practical and automated method to discover and organize milestones of knowledge, through the use of Complex Networks Analysis (CNA) techniques (Newman, 2003; Aldana, 2008). In the past, the authors had worked with complex interaction scenarios using neural nets (Daradoumis & Casillas, 2006); then the authors found out by 2008, that complex networks are better suited to model the dynamics and interactions among entities. From that very moment, the research efforts are mainly guided by CNA; due to its functional capabilities.

The CNA is a fresh strategy to view and understand diverse phenomena in nature, societies, physics and any other occurrence in the universe; that includes diverse elements interacting with each other. The main promoters of this so called *new science of networks* are Barabasi (2003), Newman (2003), and Watts & Strogatz (1998). The authors believe that milestones of knowledge populating the traditional syllabuses are inherently organized as scale-free networks, due to its selective binding among concepts, as established by Reka & Barabasi (2002). This proposal is oriented to define a method for building academic programs and will follow up the principles underlying complex-networks, which enables the discovery and construction of educational nuclei. Such nuclei represent the backbone of every academic program.

In current scientific context, the technological advances and the socialization of ICT (social nets, blogs, video repositories, augmented reality, etc.) constitute the basic premises for the development of modern academic programs for present and future higher-education institutions (HEI). Present proposal aims at discovering the pieces of knowledge supporting an academic program, using practical and/or automated steps. The collected knowledge represents the semantic essence of academic programs in HEI. Popolo (2010) and Martin (2003) argue that such collection provides context and trajectory in Bayesian approaches for reality. This perspective agrees with the approach to academic-corpus[1] configuration. Besides, the authors are looking for automating many of the steps from this process. Due to the natural complexity of this challenge, and believe that computers are not able to perform the whole task and complete every known step by themselves. Nevertheless, a creative use of computational capabilities would be very helpful. By modeling the problem elements as a complex network, some automated analysis could be performed on this structure, allowing humans to promptly discover the main aspects and simplifying the discussion. This team has previously dealt with this kind of problem, as described in (Casillas & Daradoumis, 2012) and (Casillas, Daradoumis & Caballé, 2013).

In an effort to discover the body of knowledge that conforms to the training core of an academic program, this study will try to unfold the essential branches of education for the analyzed programs, as well as their cognitive infrastructure. In order to use expert-system techniques for the discovery and extraction of knowledge stored in humans, a group of professors has been invited to collaborate in the study. The results produced are discussed and prove to be interesting and meaningful.

RELATED WORK

Since the 1960's, computers have been used in scientific fields. The use of computers at that time was limited to tasks involved with calculations in: iterative, recurrent, and concurrent contexts. More recently, advances in computer graphics allowed improved simulations from reality. Nowadays, computers are not the actor that uses *brute force* anymore. Of course, they will make the calculations as needed; however, the main goals have been indeed changed. One of current goals for computers, in this context, is to support collaboration among professionals or scientists, as it is shown in the field of e-science (Casillas & Daradoumis, 2012).

The report from Atkins, et al. (2003) is highly focused in the exploitation of the available cyber-infrastructure. According to them, there is a revolution in science due to advances in information and communication technologies. This revolution is based on the innovative capabilities to successfully emulate reality in the digital dimension. Specifically, these authors argue "…the classic two approaches to scientific research, theoretical / analytical and experimental / observational, have been extended to in silico [sic] simulation and modeling to explore new possibilities and to achieve new precision…"; with important achievements in Forestry, Ocean Science, Environmental Science and Engineering, Space Weather, Computer Science and Engineering, Information Science and Digital Libraries, Biology / Bioinformatics, Medicine, Physics, Astronomy, Engineering, Materials Science & Engineering, and Social & Behavioral Sciences.

Throughout the use of Topic Maps (Garshol, 2004), academic researchers, teachers, and students are truly enabled to construct learning scenarios from diverse kind of resources. This method is aimed at collecting knowledge elements from specialists, connect such elements as complex networks, and finally provide students a set of resources. The amounts of knowledge elements to be managed by this approach are defined through the capacity of humans to handle information (Miller, 1956). The resulting global map will represent the course syllabus and the tools for selecting and automatically organize the milestones that students will follow throughout their experience when following a course. Hence, diverse paths over the very same contents are enabled. Each formative nucleus (path) is highly compatible both with the specific student and the global perspective of the course syllabus. These paths are mixed up to integrate a complex network, which represents a synthetic ontology (Gruber, 1993). By the use of specific techniques, the elements in such ontology could be bound to people, and be related to the managing and/or collecting of these elements. In this sense, even Social Networks Analysis (SNA) (Wasserman & Faust, 1994) could be used as a means to analyze and represent these knowledge elements.

Syllabus design, in educational scopes, is mainly bound to language teaching or processing (Horrigan & Haag, 2008) (Taghizadeh, 2013). Thus, most of consulted sources are focused in such approach for syllabus design. Even though there are many ties from syllabus design with language courses; it is possible to admit that learners in any course are *learning a language*, e.g. the students from a Calculus course are learning the symbols (vocabulary) and rules to manage and handle such symbols (grammar), i.e. students are learning a language called *calculus*. Thus, syllabus design could be aimed for course designing. As Kang (2012) has elegantly developed a multipurpose mechanism for syllabus design. A common machinery based on CNA and SNA might imply an infrastructure for academic programs' design. The present study as such goal in the long-term.

Throughout the use of Topic Maps and Ontology, academic programs could be assembled from scratch management (Garshol, 2004; Kannan, 2010; Soori & Ghaderi, 2015; Huang & Chen, 2013), based on minimal pieces of knowledge and the discovered relationships among them. By controlling some elements as: vocabulary, concepts, relationships, and clustering; the body of knowledge acquires a meaningful topology, and eventually satisfies the demands for an academic program.

The study from Long & Crookes (1992), makes an assertion about the importance of task based analysis when syllabus are designed. These authors refer to avoid an analysis based only on words in vocabulary, and include the active perspective involving the concepts. Trough out the inclusion of concepts as: relationships, critical paths among concepts, and the notion of clustering; it will be implied an active perspective, by producing structure and functionality.

Discovering minimal pieces of knowledge, relationships, and supervising rules; becomes the main source of complexity for present effort. Dealing with this complexity is the foremost goal of this work. The authors understand and admit *complexity*, as the resultant-vector from collateral constraints such as: soundness, completeness and decidability; thus, current challenge is to satisfy these constraints while providing a general approach for tailoring academic programs. Regarding *soundness*, current efforts are oriented to provide certainty for every piece of knowledge or awareness, by setting the trust on professionals collaborating. The selection process for professionals must be carefully undertaken. Regarding *completeness*, current efforts are oriented to warrant that every piece of knowledge or awareness in every step of the process; can be followed up to its origins and/or implications along the whole model. Finally, regarding the *decidability* aspect; current efforts are oriented to have the conviction of knowing all the time, what is the following step in procedure.

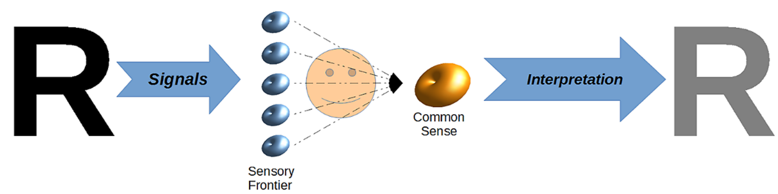
A CNA/SNA METHOD FOR TAILORING OF ACADEMIC PROGRAMS

Most of the activities developed by humans are based on sets of concepts and the relationships among such knowledge elements. The humans' understanding of the environment is defined by the concepts and ideas acquired before. These previous elements are organized as nets of perceptions in which the relationships are modeled by links of proximity. Hence, humans have a natural tendency to organize understanding as complex networks.

The authors believe that CNA could become a general algorithm to solve problems from diverse nature, even those phenomena that fall away from regular and *computable* treatment, such as social studies, economical matters, personal and/or ethical values, and etcetera. As long as a phenomenon can be modeled as a network, CNA algorithms are enabled to act over such phenomenon and produce some solution indicators. Nevertheless, and inspired by Gödel's Incompleteness[2], modeling a problem as a network cannot guarantee finding the optimal solution; although finding a representation inspired by a network may enhance the chances to solve the problem.

On the one hand, let us consider the scenario presented in Figure 1. In this Figure, it is characterized a situation when humans capture signals from reality through their sensory frontier, and the common sense (at the bottom of understanding) produces an internal image for the captured reality. As the reader

Figure 1. A representation for humans capturing and interpreting reality; the dark 'R' at left is Reality manifesting at full and bold tone. Unfortunately, humans will not capture that fullness and only will be able to interpret a constrained representation to reality: gray 'R' at right. The cause are constraints in sensory frontier and lack of expertise when applying common sense.

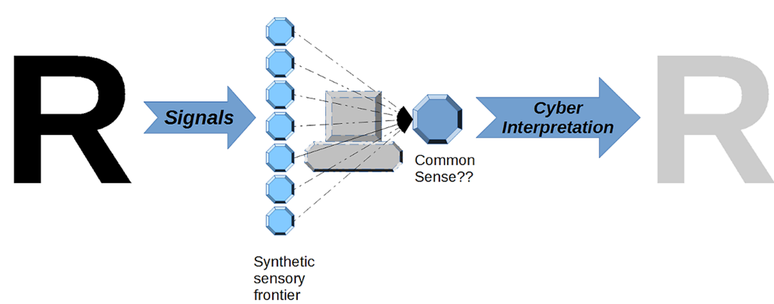


can notice in Figure 1, the interpreted reality ('R' at the right) has a lighter intensity in color; this refers to the loss of meaning and/or semantics during interpretation. This is a normal issue, caused by natural constraints in humans' sensory frontier and the induced constraints is common sense.

On the other hand, let us consider the scenario presented in Figure 2. In this Figure it is characterized a situation when computers capture signals from reality throughout a sophisticated sensory frontier; but in absence of common sense, machine produces an internal image of the captured reality by using only a synthetic version of common sense (which is usually, some sort of programs processing signals creatively). As the reader can notice in Figure 2, the interpreted reality ('R' at the right) has a much lighter intensity in color; this refers to the significant loss of meaning and/or semantics during interpretation. Although machines could include a wide range of sensors, as well as high sensibility for such inputs, constraints bound to a simulation for common sense will restrict the capability to interpret reality considering its richness.

Such frustrating loss of meaning, when machines collect and manage reality by themselves, inspired us to search for a creative approach for cyber-interpretation of meaning from reality. Considering that common sense cannot be described by a computable algorithm, machines are currently unable to perform many reality processing tasks. There are diverse efforts aiming at allowing machines to have a *human*

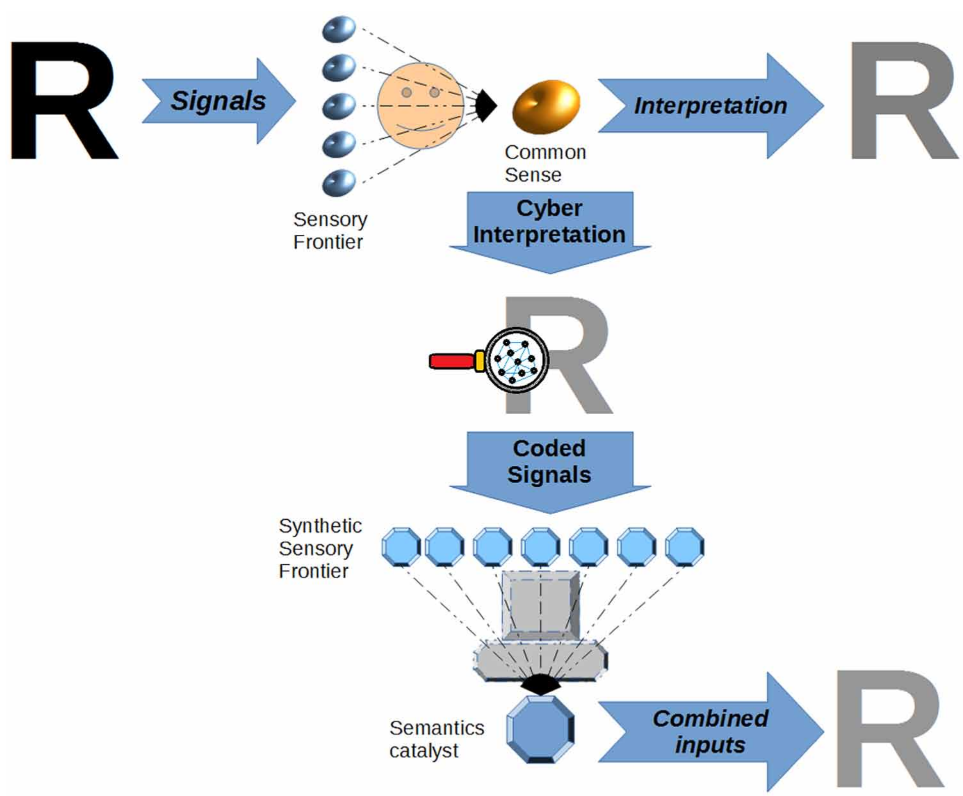
Figure 2. A representation for machines capturing and interpreting reality; the dark 'R' at left is Reality manifesting at full and bold tone. Unfortunately, computers will not capture most of that that fullness, and only will be able to interpret a very constrained representation to reality: light gray 'R' at right. Although machines could be enabled with highly sensitive frontier, the absence of common sense will break down any possible interpretation of reality.



understanding, but there are always loose ends cracking such efforts. The *von Neumannian*[3] essence in machines' operation is the main responsible for all constraints in the *humanization* of computers.

Among diverse options, one strategy to improve machine understanding over reality; is to *help* machines in the interpretation process. Humans are certainly enabled to *preprocess* reality, and produce the bases for machine interpretation. Present strategy consists in using humans' capabilities to identify and bind the main aspects of the studied reality. Figure 3 shows a scheme for this approach. Dark 'R' at upper left corner is interpreted by humans' common sense. The upper right corner in Figure 3 shows the discolored 'R', bound to humans' interpretation for reality. Under the bagel shaped common sense, there is the cyber-interpretation, made by humans. Such act produces a lighter 'R'. Cyber-interpretation made by humans produces a lighter 'R' due to the production of computable structures, which have less strength than conceptual structures in humans' brains. The human-made cyber-interpretation is fed to machines. Then machines produce an interpretation to reality just by combining inputs through a semantics catalyst. The reader can compare the 'R' produced by combining inputs in Figure 3, and the 'R' at the right of Figure 2.

Figure 3. A representation for machines capturing an already interpreted reality, and the production of improved machine interpretation; the dark 'R' at upper-left is Reality manifesting at full and bold tone. As shown in Figure 1, humans could partially interpret reality. Besides, humans could produce an attempt of interpretation for machines; grasping the most significant matters. The magnifying glass, in the middle, is showing the networked essence achieved for the cyber-interpretation.



When machines interpret reality using some of their own means, there is a significant loss of meaning; but when machines use a cyber-interpretation made by humans, they produce a lesser amount of loss of semantics. Complex-networks and their analyzing algorithms could be used to achieve this human-made cyber-interpretation. Such approach is the axis for present effort described in this work.

Inspired in such strategy; the authors based this method for tailoring academic programs, on humans' expertise. To this end, the first step consists in identifying professors and specialists who would be collaborators in the study. Authors agreed that these professionals should be individuals possessing a deep understanding of the syllabus studied, both in the pedagogical dimension, and in research or professional areas; reaching approximately 10% of faculty staff and researchers from the program studied (Figure 4). This study could be conducted through electronic means (email, forums, wikis, and etcetera), so collaborators could be in remote locations and continue to participate asynchronously through internet access.

Once the teamwork was formed, collaborators were asked to identify the main areas of knowledge related to the academic program studied, preferably a list of 7 ± 2 items (Miller, 1956), containing only the name of each branch. In this study, collaborators' effort was drove to update an already existing program; so the new knowledge branches were expected to be different from knowledge branches in the actual program, as shown in Figure 5. At this point, collaborators had to look at the program curriculum as a critical mass of knowledge that is fragmented into its parts. After finishing this challenging step, collaborators were requested to send the main branches they individually reached.

The following step consists in collecting the names of branches defined by each collaborator. These branches had some already expected differences in name, and they needed to be somehow merged, combined, unified, and refined into a common nomenclature that meets all collaborators' proposals, as sketched in Figure 6. The weight assigned to contributions is initially democratic, that is, the knowledge engineer reviews only the names of the branches without having access to the collaborators' name. Eventually, a supervisor could define some criteria regarding the assigned weight to contributions, according to every collaborator expertise. Nevertheless, such weighting policies are optional and subject to discussion. The final product from this step must be a list of 7 ± 2 elements, which represents a common ideological kernel regarding collaborators' contributions.

Figure 4. A group of distinguished professors and/or professionals has been invited to collaborate in the study. Most of the invited individuals gladly accepted. Those professionals rejecting the invitation argued they had no time or availability.

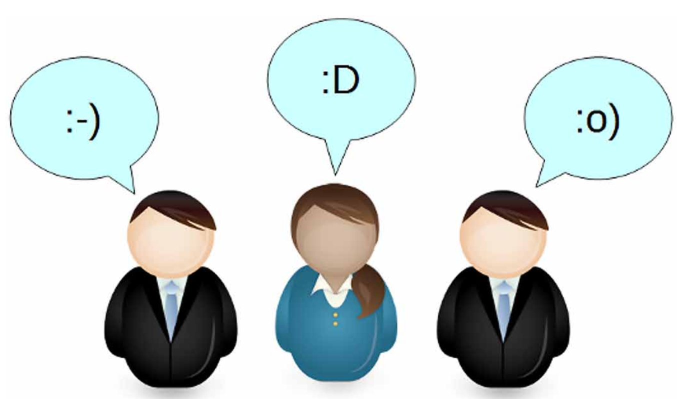


Figure 5. Every collaborator produces a list of knowledge items for the academic program in which he/she is participating. Different colors in message-bubbles are used to denote diverse perspectives or proposals from collaborators.

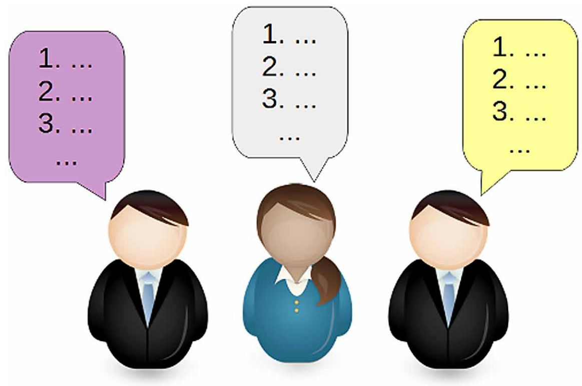
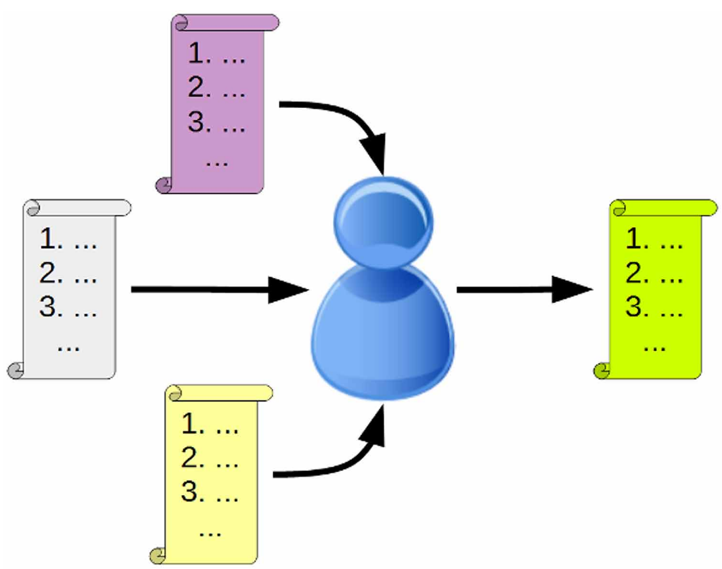


Figure 6. The knowledge engineer proceeds to collect the lists of knowledge items (branches) produced by professors. At this point, the challenge for this engineer is to produce a compact list (7 ± 2 items) which merges the branches proposed. The engineer needs to discover those elements which are common, similar or opposite of each other, without missing the valuable awareness provided by collaborators. Different colors in rolled up papers are used to denote diverse approaches.



It is important to note that these steps involving professionals crumbling huge knowledge-corpus into branches, weighting policies, and definition of a common list of names for branches; are strictly human tasks. Although some automatic support could be provided, involving natural language processing to identify and bind knowledge names; final decisions must be made by humans.

The next step consists in distributing the already consolidated list of branches among collaborators, as shown in Figure 7. Once the list is distributed and unified, collaborators have to divide each branch into sub-branches or milestones of knowledge. This is sketched in Figure 8. Again, the number of sub-branches should be around 7 ± 2 elements. So far collaborators usually perform a classical decomposition of knowledge, based on the hierarchical approach; humans make this kind of decomposition almost inherently.

The following step constitutes a major effort to combine, unify and refine the milestones (sub-branches) that have been provided by each collaborator. The final product of this effort is, once again, a common list of 7 ± 2 milestones per branch. Such list represents an ideological kernel of contributors, but now it has been made from milestones (sub-branches provided by collaborators), as shown in Figure 9.

The unified list of milestones is sent to collaborators, as presented in Figure 10. Once collaborators have received the list, they are asked to perform the last task: find out dependencies and/or influences among milestones of knowledge (sub-branches).

Every collaborator can set all the links he/she considers as relevant among sub-branches, regardless of the parent-branch to which the involved sub-branches belong. As long as influences are rational or consistent links and collaborators are people with solid experience, this should not be a cause of concern. Once the information from relationships is collected, the authors can proceed with building a directed graph containing all the information. The nodes of this graph are the sub-branches (milestones), and the edges represent the relationships of influence or dependence among them. By doing this, the authors now pass to follow a networked approach to deal with the problem (rather than a hierarchical one followed before), as shown in Figure 11.

As seen in Figure 11, a networked construction enables the search of training core items from the academic program under study, as well as the definition of critical paths in training. The output degrees (the number of arcs leaving each node) indicate the influence of a milestone on others. The input degrees

Figure 7. The knowledge engineer transmits the merged list of items (knowledge branches) to collaborating professors. They collect these elements and proceed to the next stage of the study. Different colors and order in message-bubbles are used to denote diverse perspectives or understanding from collaborators.

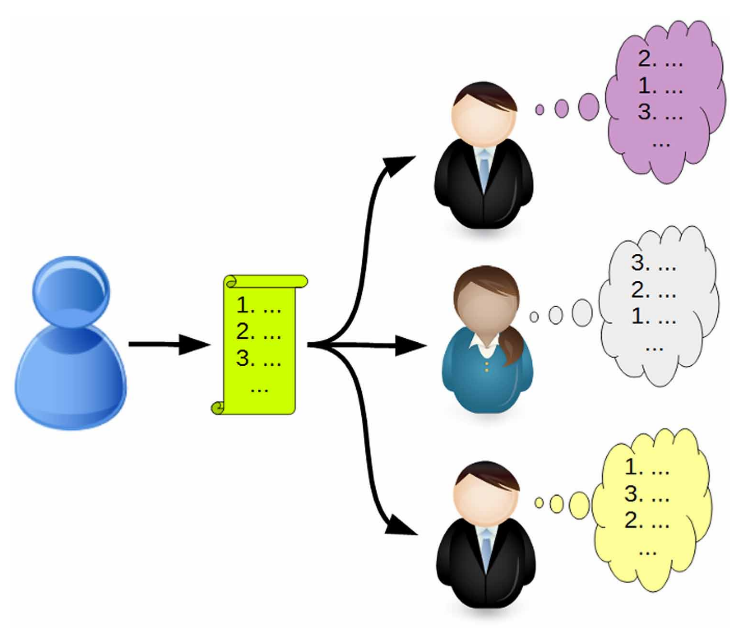


Figure 8. At this point, every collaborating professor produces coherent sub-branches for each branch received. These sub-branches are the milestones he/she considers as the individual concepts composing every knowledge branch. Different colors for message-bubbles are used to denote diverse perspectives or proposals from collaborators.

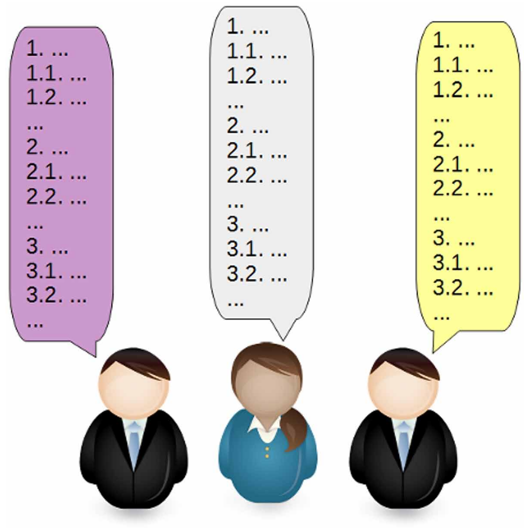


Figure 9. Once again the knowledge engineer performs a collection of lists coming from collaborating professors. These lists contain the knowledge milestones they considered as sub-branches. Now, the knowledge engineer proceeds to merge, unify and bind all the milestones produced. This task was implemented by electronic worksheets and their formulae, filters and macros. Different colors for rolled up papers are used to denote diverse approaches.

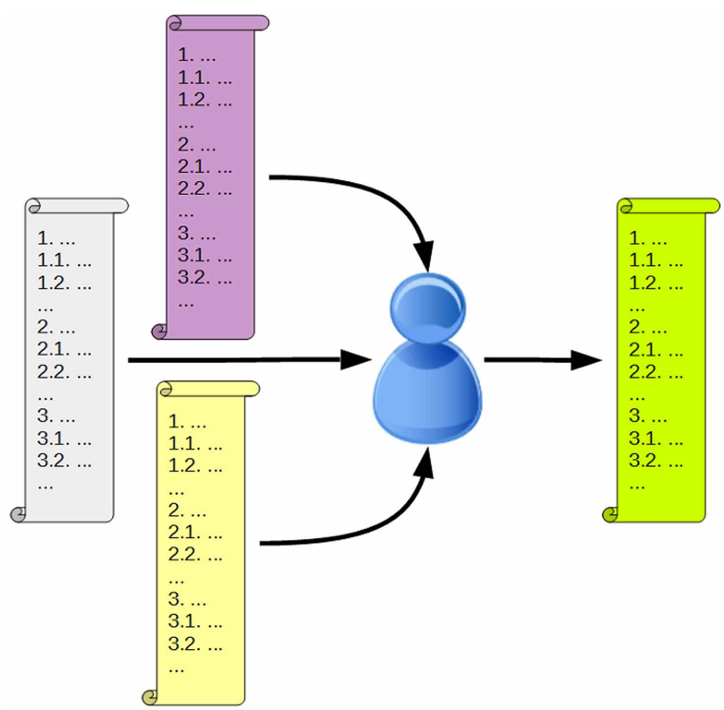


Figure 10. The resulting merged list of sub-branches is now forwarded to collaborating professors in order to start their last task. Different colors for message-bubbles are used to denote diverse perspectives or proposals from collaborators.

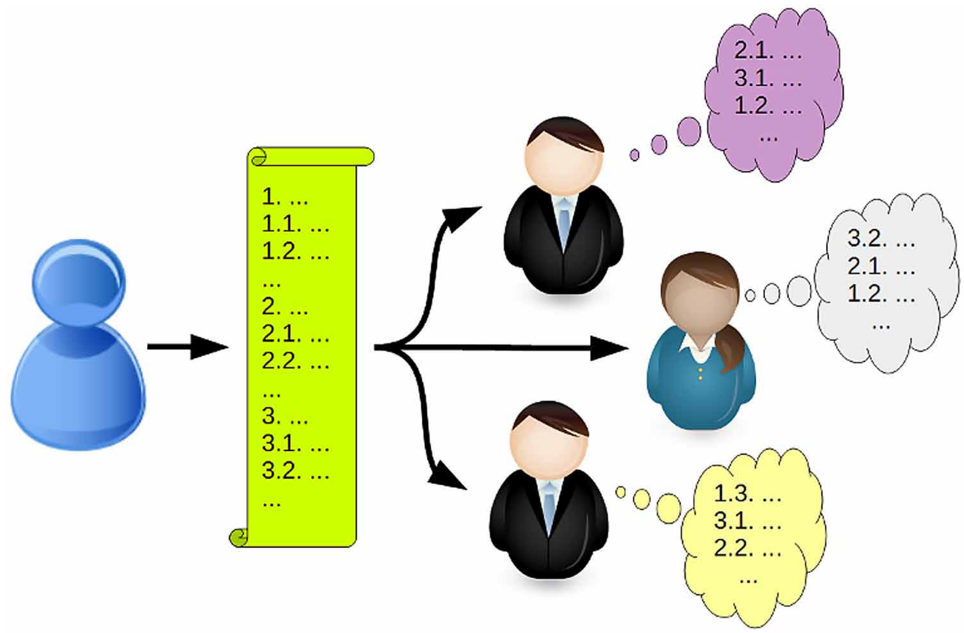
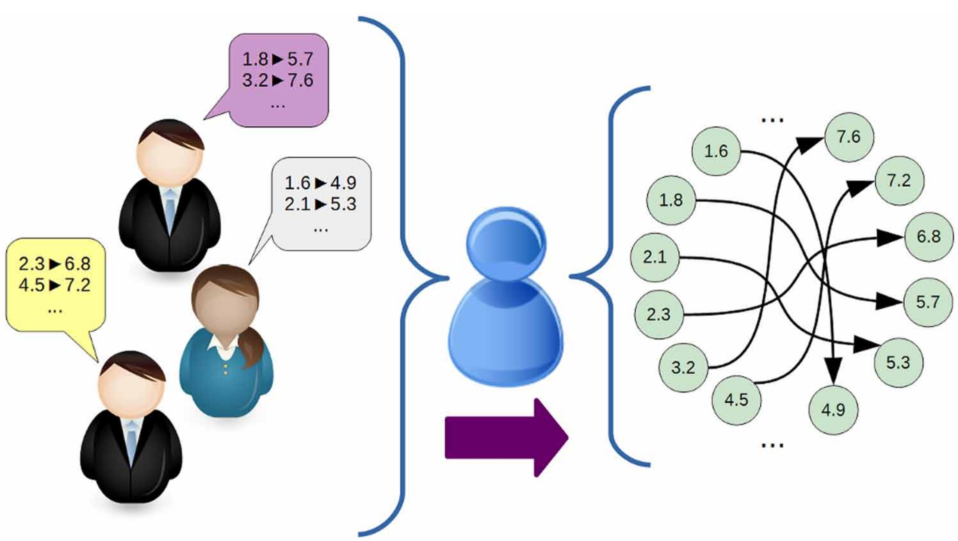


Figure 11. Every collaborating professor establishes influences and dependencies among the knowledge milestones provided in a unified list. These binary relations can be combined into a complex network structure. All the assertions provided by collaborators are used in the network. Different colors and order in message-bubbles are used to denote diverse perspectives or understanding from collaborators.



(the number of arcs entering each node) indicate the dependence from other milestones. At a first sight, the most influential milestones, as well as the less dependent ones, constitute the main aspects that shape every training core item in the academic program.

Input and output degrees may incidentally acquire magnitudes which are disconnected from the rules that explain the studied reality. In fact, this network tends to match the form of a scale-free network (Barabasi, 2003). Thus, it is possible to complement these results with the discovery of hub nodes in the graph. By discovering hub nodes, the authors found out relationships among families of nodes in the same neighborhood; which share access to common nodes. This neighborhood is called *cluster*. The clustering coefficient (Watts & Strogatz, 1998) is calculated locally (for each node), whereas the global clustering coefficient is based on the average of all the coefficients calculated for each node locally.

Based on the arguments of Watts & Strogatz (1998), the formula for clustering coefficient in directed graphs is:

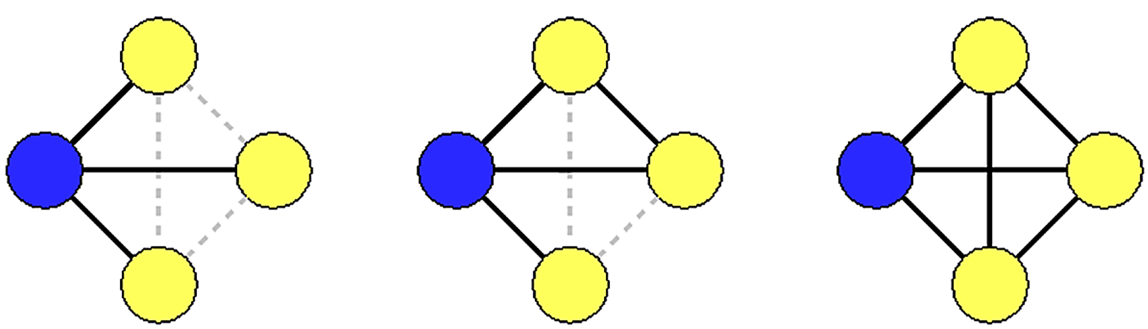
$$Cn = pn / (vn\ (vn-1)),$$

and the formula for undirected graphs is

$$Cn = pn / vn,$$

where Cn is the clustering coefficient for a node n, p is the number of pairs formed between the observed node neighbors, vn is the observed number of neighbors of node. Figure 12 presents a simple example for undirected graphs.

Figure 12. Different cases for calculating the clustering coefficient in undirected graphs; black solid lines denote actual links among the involved nodes. Light gray dotted lines denote possible, but not present links among the involved nodes. Darker circle is the observed node and clearer ones are its neighbors, which could be or not bounded among themselves. The case for (a) denotes a zero clustering coefficient, due to the absence of links among neighbors. In (b) there is only one bound between two of the neighbors, hence the clustering coefficient is 1/3. Finally, in (c) all possible links among neighbors are set; therefore the clustering coefficient is 3/3 or 1.



(a) No pairs formed among neighbors: ***C = 0/3 = 0***

(b) One pair formed among two neighbors: ***C = 1/3***

(c) All possible pairs (3) formed among neighbors: ***C = 3/3***

The clustering coefficient is measured within the interval [0, 1] Є R, and indicates the strength of the group which maintains a common node as its center. The local coefficient speaks of the capacity of a specific node to be a hub node; the global ratio refers to the cohesion of the entire structure. A good level in the overall coefficient (0.51 to 1) as well as the discovery of sub-networks involving hub nodes, lead to the threshold definition for training corpus.

The networked stages of this strategy could be supported by a tool made specifically for such goal. As concerns the experiments that were carried out with this method, the tailoring of the network structure was performed using *Cytoscape*, which is a tool made specifically for biological networks. Besides, the rest of the analysis for these networks has been performed using software tools developed, from scratch, by the authors. Such analysis consists in measuring input and output degrees, detecting functional dependencies, isolating critical paths, calculating clustering coefficients for every node and the whole network, sorting the nodes based on these coefficients, and finally defining hub-nodes. These steps in the analysis process are automatic.

The knowledge engineer must compare the information about the influences and dependencies between milestones as well as the awareness regarding hub nodes to form groups among them. This process should be carried out among the knowledge engineer and the experts.

Regarding the opportunity to automate some additional steps in the model the authors are presenting here; just as they did in (Garshol, 2004) and (Kannan, 2010), by the use of topic maps and ontology processing emerge some changes. Authors are familiar with Natural Language Processing Techniques (NLP) (Casillas & Daradoumis, 2009). Topics, milestones, branches, and sub-branches could be automatically be: unified, bound, related or unrelated throughout the use of NLP. In order to discover the semantics linkable to every element in discourse, all the branches and sub-branches provided by collaborators would be processed. Tables with synonyms and antonyms can be used over substantives in sentences. Besides, a set of verbs would be considered. All these elements allow grasping the common semantics in contributions from collaborator. Further works in this line will include the automation of such stages.

APPLYING THE METHOD ON A REAL ACADEMIC PROGRAM

In order to verify the convenience and effectiveness of this method, the authors applied it during the curricular redesign of the *Computers Engineering* academic program, following the steps explained in the previous section.

The basis of this technique depends on correct selection of participants. A survey among students and academic staff leaded to detect the preliminary list of participants. In order to achieve such list, some rules were applied: involving know-how based on time, know-how based on skills, research skills and professional recognition. Since it is hard to find individuals with 100% in every desired aspect, the list included individuals that possessed the most of the ideal skills, so participants were not required to fulfill all desirable aspects. The authors believe that this restriction is a reality that cannot ignore. Nevertheless, this networked-based method has the advantage of collecting and combining multiple expertise without being affected by such restrictions; in fact, only assertions were collected. Most of considered professionals accepted to collaborate in the study. Indeed, 15 professors, scientists and professionals accepted to participate in the study. They were trained and given support and orientation during the whole process.

The list containing the first decomposition for knowledge corpus was refined in the first interactions. The knowledge engineer was able to achieve a list of knowledge milestones.

Knowledge Milestones

1. Sciences supporting computers science,
2. Networking, Communications, Distribution, and Parallelism,
3. Intelligent Systems,
4. Base software and hardware programming,
5. Computer architecture,
6. Deployment of Information and Communication Technologies (ICT).

This list of knowledge branches was distributed among collaborators, and every participant found the sub-branches. The knowledge engineer managed to collect all collaborations and unify perspectives. The sub-branches achieved and unified after all this process are set in a list.

Branches and Sub-branches of Knowledge

1. **Sciences Supporting Computers Science:**
    - 1.1. Calculus,
    - 1.2. Mechanics,
    - 1.3. Numerical analysis,
    - 1.4. Probability and Statistics,
    - 1.5. Waves physics,
    - 1.6. Optics,
    - 1.7. Electromagnetism,
    - 1.8. Logics and Sets,
    - 1.9. Discrete mathematics,
    - 1.10. Computers theory.
2. **Networking, Communications, Distribution, and Parallelism:**
    - 2.1. Tele-informatics,
    - 2.2. Computers networks,
    - 2.3. Multitask systems,
    - 2.4. Parallel systems,
    - 2.5. Multi-core programming,
    - 2.6. Distributed systems,
    - 2.7. Pervasive computing,
    - 2.8. Redundancy and fault tolerance,
    - 2.9. Network management and networking services.
3. **Intelligent Systems:**
    - 3.1. Knowledge based systems,
    - 3.2. Inference machines,
    - 3.3. Machine learning,
    - 3.4. Artificial neural networks,
    - 3.5. Fuzzy logic,
    - 3.6. Automatic control,
    - 3.7. Evolutive computing,

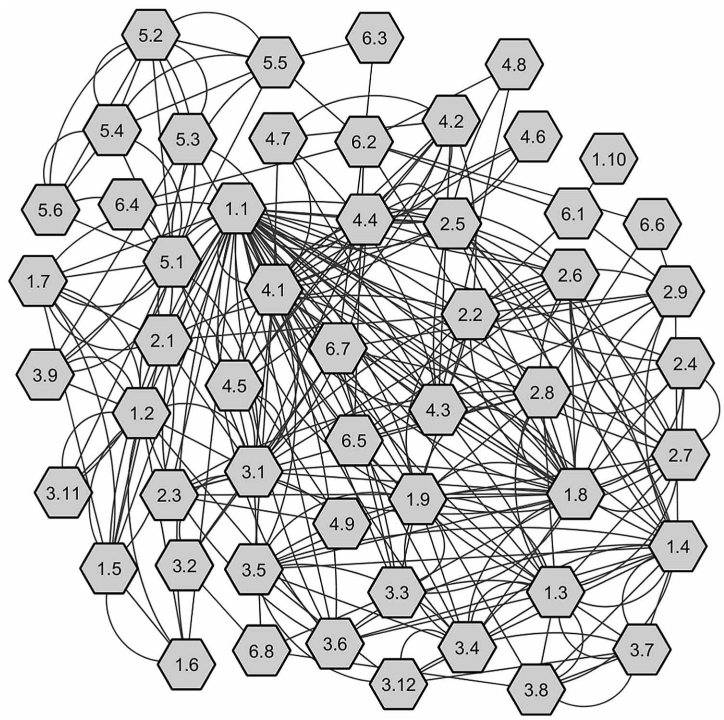
3.8. Automatic optimization,
3.9. Industrial robotics,
3.10. Mobile robotics,
3.11. Ludic robotics,
3.12. Robotics.

4. **Base Software and Hardware Programming:**
    4.1. Programming paradigms,
    4.2. Virtual machines,
    4.3. Assembly programming,
    4.4. Programming languages and tools,
    4.5. Software engineering,
    4.6. Embedded software programming,
    4.7. Language translators: assemblers, compilers, etc.,
    4.8. Operating systems: use and development,
    4.9. Projects management.
5. **Computer Architecture:**
    5.1. Digital systems,
    5.2. Hardware architectures,
    5.3. Firmware development,
    5.4. Computers organization,
    5.5. Human-computer interaction,
    5.6. Low level communications.
6. **Deployment of Information and Communication Technologies (ICT):**
    6.1. Operating systems deployment,
    6.2. Technology management,
    6.3. e-Commerce, e-Government,
    6.4. ERPs (Enterprise Resource Planning),
    6.5. Information security,
    6.6. IT services,
    6.7. Business intelligence,
    6.8. Decision support systems.

The unified sub-branches were sent to participants, so they could perform the influence-dependence definition among these sub-branches. According to his/her expertise, every collaborator made a list of ordered pairs. Every pair contained the sub-branch influencing and the sub-branch influenced. The first element in ordered pair, influencing the second one. Using the accumulated list of pairs, including all contributions and without any removals; an influence-dependence network is built. Figure 13 shows a simplified representation for the resulting complex-network.

The availability of this network is the foundation supporting this study. Different software modules collected and mixed the contributions from participants. When the network is completed; input and output degrees per node, imply the order traceable by different critical paths in the academic program. The clustering coefficient allowed the detection of hubs. These hubs represent the knowledge milestones on which the academic program is built. The clustering coefficient for the whole network presented in Figure 13 is 0.4. A specific analysis regarding clustering coefficients on every node is shown in Table 1.

Figure 13. Complex network modeling the influence-dependence relations among knowledge sub-branches from an academic program; in this case, the academic program is Computers Engineering of the University of Guadalajara. This model is presented in order to allow the reader to have a clear view of complexity achieved by modeling such situation and when including multiple sources of knowledge. The real model presents many more details, but they are not shown in order to simplify the picture of the model.



A final meeting was planned. This seminar included professors and knowledge engineers. Participants reviewed the achieved results. From these results and some additional assertions from collaborators, the structure of the academic program was finally redefined. Such structure is described by the list of academic corpuses.

Academic Corpuses Discovered

- Science supporting computing,
- Systems programming and embedded software,
- Computers architecture and organization,
- Cloud computing and networking services,
- Intelligent and robust systems.

In fact, the order presented in academic corpuses tries to represent the value and advantages of every knowledge corpus, as well as the recommended sequence for them. Both aspects (value and sequence)

Table 1. Clustering coefficient per node of the computers engineering academic program

Sub-Branch	Clustering Coefficient
1.1. Calculus	0.26
1.2. Mechanics	0.21
1.3. Numerical analysis	0.36
1.4. Probability and Statistics	0.2
1.5. Waves physics	0.7
1.6. Optics	1
1.7. Electromagnetism	0.5
1.8. Logics and Sets	0.22
1.9. Discrete mathematics	0.4
1.10. Computers theory	0.32
2.1. Tele-informatics	0.3
2.2. Computers networks	0.19
2.3. Multitask systems	0.5
2.4. Parallel systems	0.83
2.5. Multi-core programming	0.47
2.6. Distributed systems	0.25
2.7. Pervasive computing	0.52
2.8. Redundancy and fault tolerance	0.25
2.9. Network management and networking services	0.07
3.1. Knowledge based systems	0.19
3.2. Inference machines	0.33
3.3. Machine learning	0.3
3.4. Artificial neural networks	0.52
3.5. Fuzzy logic	0.4
3.6. Automatic control	0.5
3.7. Evolutive computing	0.5
3.8. Automatic optimization	0.3
3.9. Industrial robotics	0.18
3.10. Mobile robotics	0.23
3.11. Ludic robotics	0.12
3.12. Robotics	0.33
4.1. Programming paradigms	0.26
4.2. Virtual machines	0.6
4.3. Assembly programming	0.4
4.4. Programming languages and tools	0.39
4.5. Software engineering	0.83
4.6. Embedded software programming	1
4.7. Language translators: assemblers, compilers, etc.	1
4.8. Operating systems: use and development	0.15
4.9. Projects management	0.2
5.1. Digital systems	0.18
5.2. Hardware architectures	0.5
5.3. Firmware development	0.2
5.4. Computers organization	0.83
5.5. Human-computer interaction	0.35
5.6. Low level communications	1
6.1. Operating systems deployment	0.22
6.2. Technology management	0.39
6.3. e-Commerce, e-Government	0.21
6.4. ERPs (Enterprise Resource Planning)	0.2
6.5. Information security	0.2
6.6. IT services	0.33
6.7. Business intelligence	0.36
6.8. Decision support systems	0.23

result from the proposed CNA/SNA method. Consequently, the available and already known sub-branches were slightly adjusted and reorganized in order to fit the new schema of academic corpuses. The *Computers Engineering* academic program was finally completed using the results from this model. Table 2 shows the current structure for the *Computers Engineering* academic program. This structure was widely influenced by the results from this study. Building an academic program remains a human responsibility; nevertheless this method has provided significant data and information in order to achieve an effective, robust and useful academic program.

Table 2. Current structure of the computers engineering academic program

Semester	Subjects					
1	Mathematical methods I	Programming	Seminary for problems solution: Programming	Discrete mathematics	Philosophical Foundations of Computer Science	*Open Optional*	
2	Mathematical methods II	Seminary for problems solution: Mathematical methods I	Data structures I	Seminary for problems solution: Data structures I	Computers Theory	*Open Optional*	
3	Mathematical methods III	Seminary for problems solution: Mathematical methods II	Statistics and stochastic processes	Data structures II	Seminary for problems solution: Data structures II	Algorithmics	Seminary for problems solution: Algorithmics
4	Seminary for problems solution: Mathematical methods III	Databases	Seminary for problems solution: Databases	Software engineering I	Seminary for problems solution: Software engineering I	Networking and communication protocols	Seminary for problems solution: Networking and communication protocols
5	Language translators I	Seminary for problems solution: Language translators I	Computers Architecture	Seminary for problems solution: Computers Architecture	Security	*Specializing Subject*	*Specializing Subject*
6	Language translators II	Seminary for problems solution: Language translators II	**Internships**				
7	Artificial Intelligence I	Seminary for problems solution: Artificial Intelligence I	Operating systems	Seminary for problems solution: Operating systems	Fault tolerance computing	Internet Programming	
8	Artificial Intelligence II	Seminary for problems solution: Artificial Intelligence II	Networking operating systems	Seminary for problems solution: Networking operating systems	Distributed and concurrent systems	Computer simulation	

FUTURE RESEARCH DIRECTIONS

The following efforts over this experiment will be aimed at including, as mentioned in section 3 of this work, automated mechanisms with NLP; when producing common lists of branches and sub-branches from collaborators' responses. This would be possible is a controlled thesaurus, built from official definitions for the academic program. A group of professionals in the area could always recheck this thesaurus, as well as the synonyms and antonyms stored. When branches for knowledge-corpus are presented from collaborators, a NLP could work over the concepts in order discover: unions, intersections, complements and contractions.

In fact, this system will tend to become a web-based application that could be fed with diverse documents describing the knowledge field supporting the academic program that will be constructed. The list of professionals selected, as well as their contact information. The system, by itself would build the thesaurus and knowledge engineer supervises the results. When knowledge engineer approves thesaurus, emails are automatically sent to professionals. They crumble into branches the knowledge corpus, based on their expertise, and capture the branches names in the web. Now, system would use NLP to produce a common list. This common list should be reviewed and approved by knowledge engineer. Now professionals would crumble branches in common list, into sub-branches. Once again they are capture in web application. The system would arrange these sub-branches into a new common list of sub-branches. This new list should be reviewed and approved by knowledge engineer. Finally professionals will work over the common list of sub-branches, presented in web and they would define all the influences they detect. From this point system, by itself, will produce the complex network and the whole set of indicators as: input and output degrees, clustering coefficients, critical paths, and etcetera. From this point, humans must coordinate in meeting in order to analyze results and build the final definition for academic program. The authors are preparing the human resources to build the web-based automated representation for this model.

CONCLUSION

Updating an academic program remains a task demanding significant efforts. Such endeavor involves: people, academic units, knowledge fragments, regulations, institutions, industry, etcetera, and usually entails a specific form of complexity; related to the quantity of elements involved, the diversity of origins from contributions, the diversity of formats and representations of information, and the required granularity.

Along this work, the authors are offering a method for: discovering, handling, and exploiting the knowledge available in academic curricula. Due to nature of such knowledge construction, the awareness is stored in mixed deposits. Besides, persons responsible to stored or manage those deposits are not always aligned to deal with new and different approaches; when handling the expertise under their charge. Hence most of the efforts when dealing with collaborators, were aimed to persuade them to produce knowledge elements in the required terms. There was no struggle, because they wanted to cooperate. The problem came from the common misunderstanding of computers' capabilities and their real skills to capture arguments presented as inputs. Thus, the knowledge engineer has to crumble those *non-crumbled* responses; without missing the essential arguments from collaborators.

Although this proposal is not fully computerized; the automatic parts from this method, provide elements to improve the agreement when looking for consensus. When the collaborating group reviewed the numbers from input and output degrees, the clustering coefficients, and the critical paths along milestones in curricula; the defensive position in *everyone's castle*, was released. The benefit of representing phenomena as complex networks, is that all ideas are incorporated. When collaborators see the structure, they can find out their contributions in the model. Actually, in the real scenario, where the authors applied this method, the relaxed atmosphere drove situation to achieve agreement among participants with opposing positions. Every participant can see his/her contribution in the final result.

The complex network is a democratic model that maintains all contributions, with the same weight, from either collaborator. Nevertheless preemptive notions might be included, if it is required to work with weighted contributions from collaborators; based on collaborators expertise. The whole model could be easily adapted to include such approach a give a different understanding to indicators involved, so that contributions from best weighted collaborators will have a bigger influence in results.

REFERENCES

Aldana, M. (2008). *Complex Networks*. Cuernavaca, Mexico: UNAM.

Atkins, D. E., Droegemeier, K. K., Feldman, S. I., Garcia-molina, H., Klein, M. L., Messerschmitt, D. G., . . . Wright, M. H. (2003). *Revolutionizing science and engineering through cyberinfrastructure: Report of the National Science Foundation blue-ribbon advisory panel on cyberinfrastructure.* Arlington, VA: National Science Foundation, NSF.

Barabasi, A. L. (2003). *Linked: How Everything Is Connected to Everything Else and What It Means*. Cambridge, MA: Perseus Publishing.

Casillas, L., & Daradoumis, T. (2009). Constructing a Multi-agent System for Discovering the Meaning over Natural-Language Collaborative Conversations. In T. Daradoumis, S. Caballé, & J. M. Márquez (Eds.), *Intelligent Collaborative e-Learning Systems and Applications* (pp. 99–112). Berlin: Springer-Verlag. doi:10.1007/978-3-642-04001-6_7

Casillas, L., & Daradoumis, T. (2012). An ontology structure for gathering and sharing knowledge among scientists. In *Collaborative and Distributed-Research: Innovations in Technologies, Strategies and Applications* (pp. 165–179). Hershey, PA: IGI Global. doi:10.4018/978-1-4666-0125-3.ch008

Casillas, L., Daradoumis, T., & Caballé, S. (2013). A network analysis method for selecting personalized content in e-Learning programs. *Seventh International Conference on Complex, Intelligent, and Software Intensive Systems* (pp. 407-411). Taichung, Taiwan: IEEE. doi:10.1109/CISIS.2013.74

Casillas, L. A., & Castillo, A. (2012). Discovering epistemological axes for academic programs in computer science through network analysis. *ReCIBE.* Retrieved from http://recibe.cucei.udg.mx

Daradoumis, T., & Casillas, L. (2006). A Neural Approach for Modeling the Inference of Awareness in Computer-Supported Collaboration. In N. Wolfgang & K. Tochtermann (Eds.), Innovative Approaches for Learning and Knowledge Sharing (LNCS), (vol. 4227, pp. 464–469). Hannover, Germany: Springer. doi:10.1007/11876663_37

Garshol, L. M. (2004). *Metadata? Thesauri? Taxonomies? Topic Maps! Making sense of it all.* Retrieved from http://www.ontopia.net/topicmaps/materials/tm-vs-thesauri.html

Gruber, T. R. (1993). *A Translation Approach to Portable Ontology Specifications*. Palo Alto, CA: Knowledge Systems Laboratory, Stanford University.

Horrigan, M., & Haag, R. (2008). *Syllabus Design*. Oxford, UK: Oxford University Press.

Huang, T.-C., & Chen, C.-C. (2013). Animating Civic Education: Developing a Knowledge Navigation System using Blogging and Topic Map Technology. *Journal of Educational Technology & Society*, *16*(1), 79–92.

Isaila, N. (2012). The Technology in New Learning Environments. *International Journal of Academic Research in Accounting, Finance and Management Sciences, 2*(Special Issue 1), 128-131.

Kang, S. (2012). *From Curriculum to Syllabus Design: AATK Workshop*. Stanford University.

Kannan, R. (2010). Topic Map: An Ontology Framework for Information Retrieval. *National Conference on Advances in Knowledge Management (NCAKM'10)*.

Long, M. H., & Crookes, G. (1992). Three Approaches to Task-Based Syllabus Design. *TESOL Quarterly*, *26*(1), 27–56. doi:10.2307/3587368

Martin, K. (2003). Epistemic Motion. In *Quantum Searching*. Oxford, UK: Oxford University Computing Laboratory.

Miller, G. A. (1956). The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. *Psychological Review*, *101*(2), 343–352. doi:10.1037/0033-295X.101.2.343 PMID:8022966

Newman, M. (2003). The Structure and Function of Complex Networks. *Society for Industrial and Applied Mathematics: SIAM Rev*, 167–256.

Popolo, D. (2010). *A New Science of International Relations*. Lancaster, UK: Ashgate.

Reka, A., & Barabasi, A. L. (2002). *Statistical mechanics of complex networks.* (Doctoral Dissertation). University of Notre Dame, South Bend, IN.

Soori, A., & Ghaderi, M. (2015). A Topic-Based Syllabus Design for a Conversation Course. *Language in India, 15*(1), 538-546.

Taghizadeh, M. (2013). EAP Syllabus and Course Design. *International Research Journal of Applied and Basic Sciences, 4*(12), 3791-3797.

Wasserman, S., & Faust, K. (1994). *Social Network Analysis: Methods and Applications*. New York, NY: Cambridge University Press. doi:10.1017/CBO9780511815478

Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature*, *393*(6684), 440–442. doi:10.1038/30918 PMID:9623998

ENDNOTES

1. Regarding the term *corpus*: this concept refers, along this chapter, to a set of knowledge-elements bound by certain conditions.
2. Stanford Encyclopedia of Philosophy. “Gödel’s Incompleteness Theorems”. First published Nov 11, 2013; substantive revision Jan 20, 2015. [online] Retrieved April 29, 2015 from http://plato.stanford.edu/entries/goedel-incompleteness/
3. The *von Neumannian* essence refers to a collateral effect, produced by the operation of computers; which based on John von Neumann’s architecture. This effect consist in a rigid sequential operation when executing instructions and processing data, because of machine’s cycle followed up by computers with this architectural approach, i.e. nearly every computer.