Software Engineering
2016; 4(2): 41-49
http://www.sciencepublishinggroup.com/j/se
doi: 10.11648/j.se.20160402.16
ISSN: 2376-8029 (Print); ISSN: 2376-8037 (Online)

Organisational Knowledge Acquisition with Contested Collective Intelligence in the Web Environment

Gangmin Li

Department of Computer Science and Software Engineering, Xi'an Jiaotong-Liverpool University, Suzhou, Jiangsu, China

Email address:
Gangmin.li@xjtlu.edu.cn

To cite this article:
Gangmin Li. Organisational Knowledge Acquisition with Contested Collective Intelligence in the Web Environment. *Software Engineering*. Vol. 4, No. 2, 2016, pp. 41-49. doi: 10.11648/j.se.20160402.16

Received: March 25, 2016; **Accepted**: May 11, 2016; **Published**: May 12, 2016

Abstract: Knowledge acquisition (KA) is a hard problem in knowledge engineering. Big Data Analytics (BDA), aiming at derives value out of big data, sheds light on this problem. Advanced data analysing methods and computational platforms make it possible to imitate large members of communities and interactions among the community members. This paper reports the efforts on capturing organisational knowledge through a "Contested Collective Intelligence (CCI)" model in the web environment. We assume that web users are individual experts and the whole web community is a big organisation. The organizational knowledge on the web is emerged and revealed through the interactions where individual users freely express themselves and interact with others to clarify facts, argue about meaning and debate about truth through claim and counterclaims. It is a hope that by capturing those claims, the connections between claims and the final agreement on understanding of the meaning, the collective knowledge emerged on the web can be captured, stored and reused.

Keywords: Contested Collective Intelligence, Knowledge Acquisition, Knowledge Services, Knowledge Repository, Argumentation Structure, Sensemaking

1. Introduction

Knowledge Acquisition (KA) is a hard problem in knowledge engineering [1]. It refers a process to find rules, ontologies that can be used to build a knowledge based systems and to serve knowledge users [2]. It is a common view that the Web and specially the Social Networks like Twitter and Facebook contain huge amount of knowledge. Advanced technologies developed to scan text in post in public forums. Complicated algorithms are used to filter valuable signals and patterns from noise to work out what users are referring to and even with what emotional tone [3], [4]. These methods in data and text mining have been successful in capturing explicit knowledge.

However, there is a kind of knowledge on the web, often regarded as tacit knowledge. This is because the important insights and valuable knowledge are not expressed directly in explicit terms. It is often revealed and emerged after a long process of interactions. To capture this long interactive process is meaningful and valuable, which can help to understand how a meaning is formed and how a common agreement is reached. This process is also called as sensemaking [5]. Sensemaking itself is a new type of knowledge that not only including the sense made but also the process of sense making. Conventional method to capture sensemaking process is simulating a community of experts and monitoring their reactions to the environment and interactions with each other. This approach demands huge resources and considered is infeasible in the most business organisations where computational resources are considered as limited [6], [7].

Advanced Big Data Analytics (BDA) methods shed light on this problem [8]. Massive distributed architecture with parallel computing make valuable insights of huge amount of data become visible [9]. It is now possible to monitor behaviours, to pick up individual signals and interactions among members of a large community and to derive the social knowledge that otherwise is impossible [10], [11]. However lacking of modelling of interactions on the web makes it become difficult to capture.

Contested Collective Intelligence (CCI) model is one of attempts in modelling and capturing implicit organisational knowledge in a closed academic environment [12], [13], [14]. Academic document discourse, documents are mostly academic papers. The way of a paper is written has a relatively fixed style and pattern. The most important and significant sentences in a paper are claims that the author is trying to convince readers to accept. These claims are normally to state a concept, or to interpret an existing concept with own understanding, or to provide evidence to support or to argue against a statement. Interactions occurred implicitly among many publications in the community. The whole community maintains a common understanding by reading and writing many papers. In any a given time, a community will always have a stable knowledge base where commonly agreed knowledge has been maintained. New knowledge can emerge and the whole field of research is advanced by this continuous interactions.

In the web environment, all the web users together can be seen as a big community. Individual web users, facing complex even contradicted data and information, interact with others based on personal knowledge, understanding and experiences, also through claim and counterclaims in post, blogs or even in long reviews, clarifying meanings, understanding insights, and finally reaching or towards reaching a neutral and common agreement [15]. This slowly emerged common agreement, and the process that leads to the common agreement, together are valuable knowledge that we are aiming to capture, store and reuse.

The paper is organised in the following manner. Section two is the proposed CCI model. It is a foundation for a knowledge network to be constructed. Section three is formalizations of the CCI framework. It enables advanced knowledge computing methods to be deployed for knowledge services. Section four reports our pilot application to demonstrate the usage of the CCI model in a typical application. Finally section five presents our experience, lessons learnt and future works.

2. Contested Collective Intelligence Model

Research into Contested Collective Intelligence seeks to develop a conceptual foundation, which will increase our capability to make sense, and to construct a sociotechnical infrastructure, which can capture collective intelligence combining contributions from many sources in the web environment.

The core of the CCI model is its three key elements: claims, sensemaking process, and the commonly agreed knowledge called “contested collective intelligence”. A claim, in form of a text statement in which a true value is clearly stated by the claim maker, is the basic entity in CCI, which is also called a concept. Claim has properties such as author, date and the source that specify the where the supporting document is from. The sensemaking process is the interactions between users. It is represented as a chain of claims linked with certain relations, such as debate logics in the “argument structure” [16], [17], [18]. This rhetoric relation has fixed label, type, polarity, weight and directions. Contested collective intelligence is a label of commonly agree and accepted knowledge. It is represented as a subset of larger knowledge networks [19], [20].

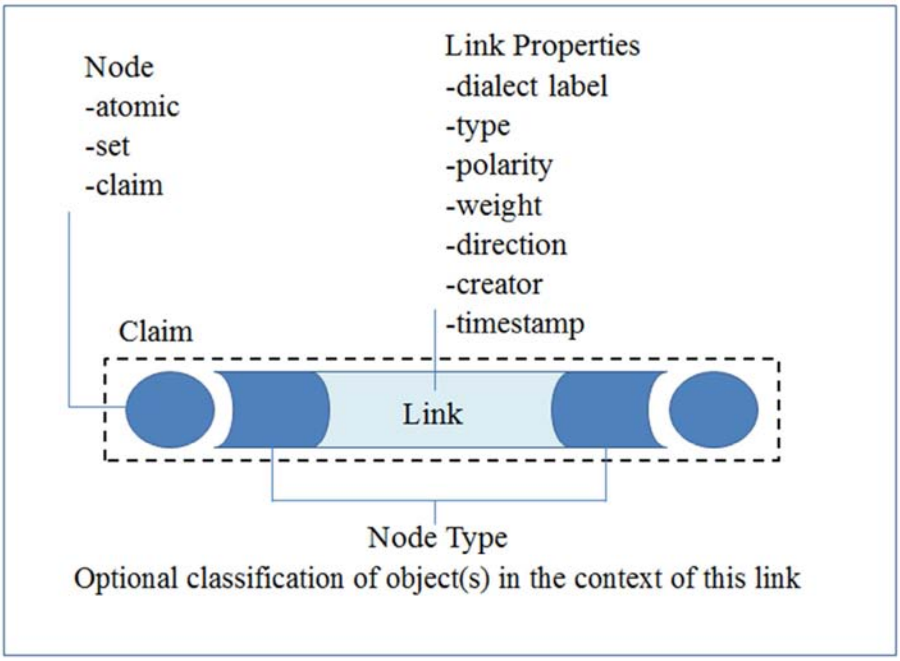

Figure 1. *Structure of a Claim in the discourse ontology.*

Figure 1 shows the original data model for claims. It comprises nodes and links. Nodes may be atomic or composite at the end user’s discretion. Atomic nodes are concepts expressed as short pieces of free text succinctly summarising a ‘contribution’. A node may optionally be assigned a type such as data, theory or evidence to express the different contexts. Two kinds of composite object can be used as the nodes in Claims. A Set is a group of objects (concepts, Sets or Claims) declared by the user to share a common theme and enabling them to be referenced by a single named node (e.g. Constructivist Theories of Learning). Claim triples themselves can also be linked from or to other atomic nodes, Sets or Claims. This nesting allows users to build complex conceptual and argument networks.

A link between two nodes is typed with a natural language label from a discipline-specific dialect, which in turn is a member of a generic, discipline-independent class such as problem-related, taxonomic, causal etc. A structure of the discourse scheme is shown schematically in Figure 2.

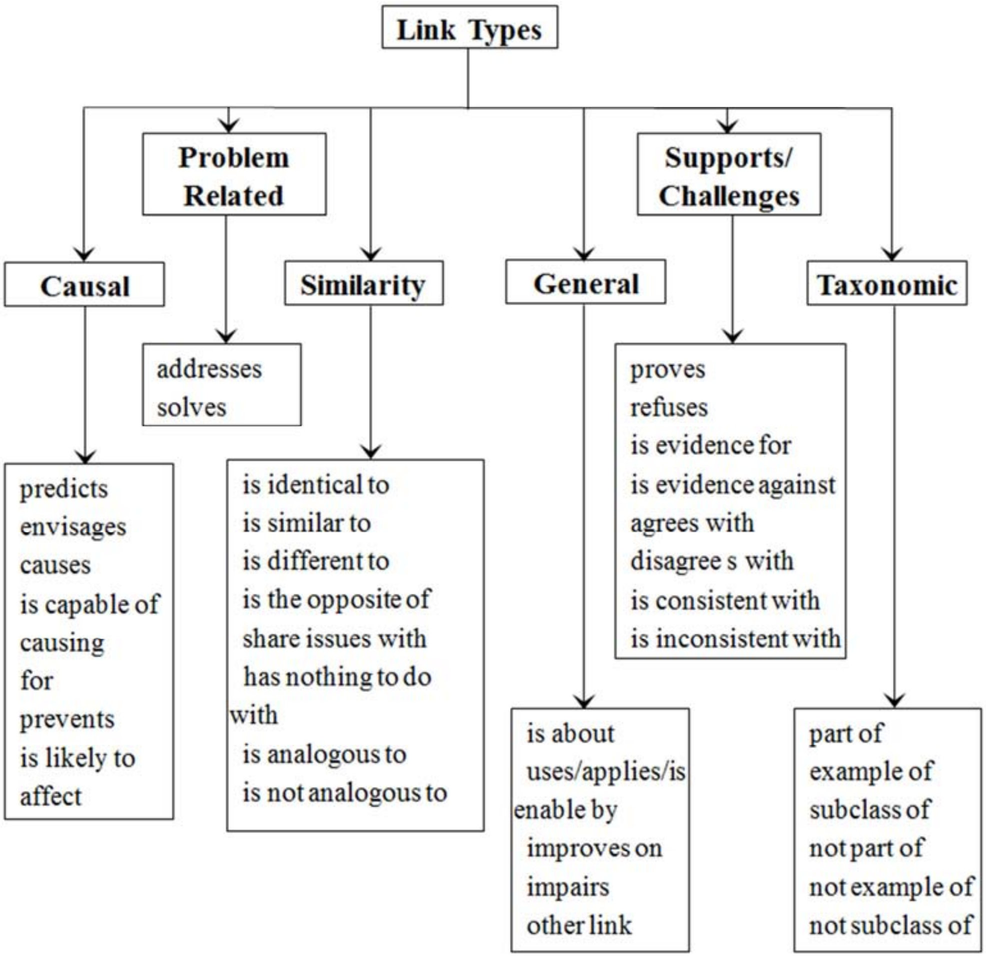

Figure 2. *Class structure of the scholarly discourse ontology.*

It demonstrates that a given research community with a dialect that will cover the most common claims that they make. There may well be exceptional kinds of contributions that fall outside the expressiveness of the vocabulary, but the generic type of "Other Link" is available for those situations.

A sensemaking process is evolved in academic discourse through a combination of theoretical and data-driven processes. The theory-driven approach derived from psycholinguistics and computational research on Cognitive Coherence Relations (CCR), combined with a semiotic perspective on representation which emphasises the interpretive act of modelling. Data-driven process, on other hand uses comprehensive parameters to express relational primitives such as addictiveness, temporality or sequentiality and causality. Each of these is then parameterised. Additiveness can be conjunctive or comparative, which also referred as similarity. Causality can be actual or hypothetical which also referred as conditionality. Both causal and additive relations can be semantic which reflect the cause and effect or pragmatic which represents argument and claims. They all can have positive or negative polarity. The order of the related units can be forward or backward.

Figure 3 is a screen capture of an example of knowledge networks using CCI model. Where, notes are concepts from multiple sources and different users. They are connected with the fixed text labels as cues to the nature of argumentation relationship. The relations also have complicated properties to reflect the connections. Many cases where users are emphasis on the same concepts or relations among concepts are enhance the part of the networks. Other cases, the extension of the networks has been created. A lot of cases connections between previously patched networks have been newly established. In the same time, some negative connections are suggested. These all reflect the arguments and the debate among knowledge users in the sensemaking process. Contested collective intelligence is the part or whole knowledge networks that satisfy a search criterion.

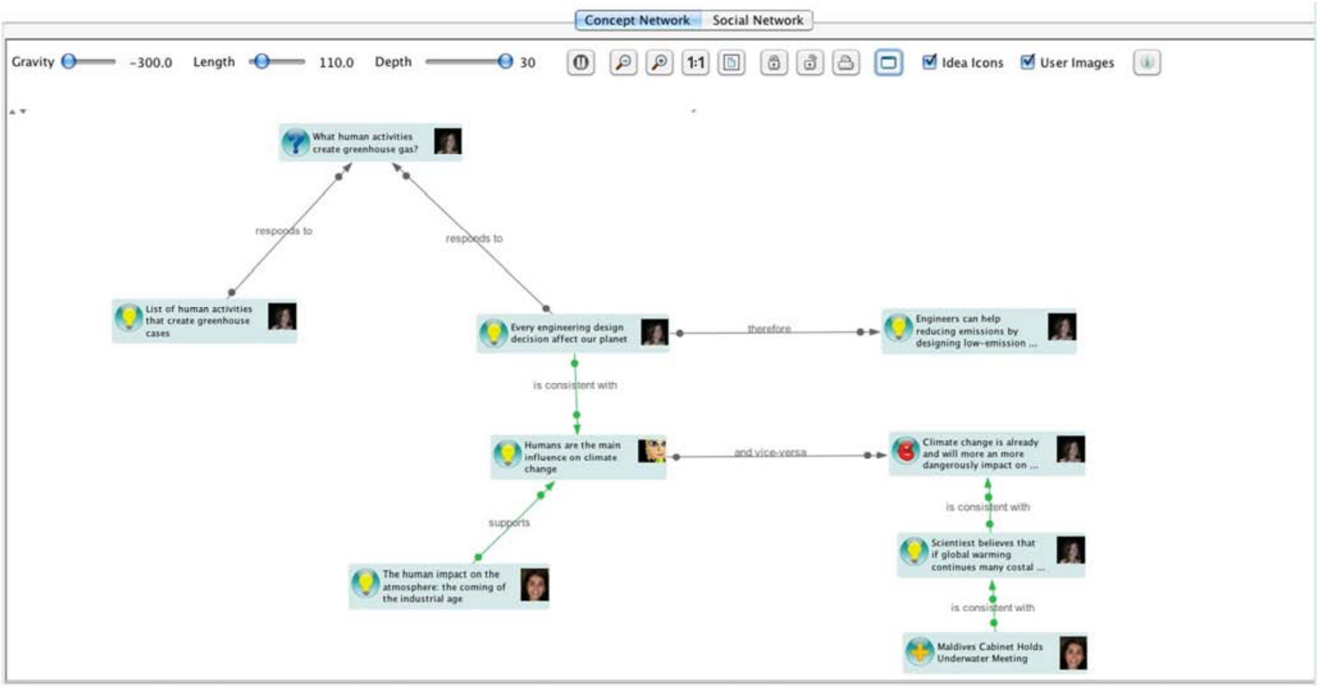

Figure 3. *Screen Capture shows Network of concepts and the relationships between them in the CCI model.*

Figure 3 is an illustration of a concept network where only a small part is displayed. It is a focused view on connections where debates have been displayed in connections. Other views including a bird's eye view of the whole networks where only network structure is showed but the details are omitted.

3. Formalization of the Contested Collective Intelligence Model

Formalizing CCI model is necessary. Only this is done, some basic calculus can be defined and complex knowledge computation and search services can be provided.

3.1. Node

A node in knowledge networks define by CCI is a claim on a concept. It is a 5-tuple.

$$C = (V, A, T, S, P) \quad (1)$$

Where, V is a concept set, A is an author set and its element is the originator of the concept, T is time when the concept is defined. S is the supporting document identifier showing where the concept originally comes from, P is a set of concept types. It is easy to find out that a claim is a basic unit in the knowledge representation in CCI. It has property of time and space. The sets associated with a claim can be regarded as mapping functions from a concept to its corresponding feature's domain. They together provide computational foundations for knowledge computations such as redundant removal, valuation, integration and inferring. Types in CCI model can be view, fact, hypothesis, question, phenomena and data etc.

3.2. Edge

Edges in knowledge networks in CCI represent relationships between connected nodes. The relationship defined in CCI is the relationship following "Argumentation Structure". It can be formalised as a 6-tuple.

$$R = (E, A, T, S, P, \omega) \quad (2)$$

Where, E is a set of directional edges. Its element is a triple (u, v, r), where $u, v \in V$ are claims, $r \in R$ is relations. It means that every pair of nodes has been assigned to one or more relations. That is to say any pair of claims is connected by one or more relations. As seen in the node definition, here A is a set of authors who is the originator of the connections; T is set of time stamps specifying when the connections are created; S is the supporting document identifier, which is used to show where the connections are derived from; P is a set of types showing the semantic type of the connection. Based on the argument structure, there are mainly two types: supportive and opposite. Under these two major types, there are more fine-gran types to support detailed categorizations on types such as causal, similar, evidence and sub-class, etc. ω is a mapping function representing weight of the relations.

3.3. Knowledge Networks

Once we have node and edge definitions, it is simple to define knowledge networks. Clearly, knowledge networks are heterogeneous networks where their nodes and edges have dimensions of time and space. It means that any part of knowledge networks can have a life-span. Their "*True*" value is always relative to its context. Furthermore, the knowledge networks always have a list of associated mapping functions and calculus. Given time sets T and space set S, node types A and edge types R, Knowledge Networks GT, S are defined in the following 8-tuple,

$$G_{T,S} = (V, E, \gamma, \rho, \theta, \tau, \varphi, \delta) \quad (3)$$

Where, V is a node set; E is an edge set; there is a list of argumentation relations triple (u, v, r), $u, v \in V$ and $r \in R$. Also,

$\gamma: V \to P_V$ is a mapping function on node set. It returns node types. In general, a node v represents a concept, therefore $\gamma(v)$ represents the types of the concept v and $\gamma(v) \in P_V$.

$\rho: V \to P_E$ is a mapping function on edge set. It returns edge types. In general, an edge e represents an argumentation relation, therefore $\rho(e)$ represents the types of the relations e and $\rho(e) \in P_E$.

$\theta: V \to 2^T$ is a time mapping function on node set. It returns specific time stamp of a node. It can be used to describe node's life span. A node generally represents a concept therefore it is used to describe a concept's life span.

$\tau: E \to 2^T$ is a time mapping function on edge set. It returns specific time stamp of an edge. It can also be used to describe an edge's life span. An edge is generally representing an argumentation relation therefore it is used to describe the life span of a specific relation in argumentation structure.

$\varphi: V \to S$ is a space mapping function on node set. It returns space information of a node. It is actually a document identifier showing the concept is derived from a specified document.

$\delta: E \to S$ is a space mapping function on edge set. It returns space information of an edge. It is actually a document identifier showing the argument relations are derived from a specified document.

3.4. Knowledge Computation

With the formal definitions on knowledge networks, its node and edges, we are able to provide knowledge networks' computation to support its construction, integration and services. Based on Marcus Kracht's argument structure [17], an argument is a triple:

$$\langle \mathrm{X}: \nabla\olessthan: \begin{bmatrix} CASE: & nom \\ NUM: & pl \end{bmatrix} \rangle \quad (4)$$

Where, X is a concept variable; ∇ and $\olessthan$ define the argument logic and computation of the concept. ∇ is the condition that X holds; $\olessthan$ is the argument operator that indicates the direction of the argument. The matrix contains the properties and the specific cases of the concept. It is a complex expression. However it follows simple first-order logic in representing the causes and the results of a given concept. In the first-order logic a basic argument structure is a pair $\langle D, I \rangle$, D is a concept set and I is a function assigning to a relation R with $\Omega(R)$ = n a subset of D^n and to a function f with $\Omega(f)$ = n a function from D^n to D. Here function $\Omega: Rel \cup Fun \to \omega$, is the first-order logic (FOL). Rel is a set of relations; Fun is a set of functions and ω denotes the set of natural numbers.

Notice that we have two special cases, namely relations of arity zero and functions of arity zero. By definition, a relation of arity zero is a subset of D^0, which we take to be $\{\emptyset\}$. Hence there exist two such relations, $\emptyset$ and $\{\emptyset\}$. A function of arity zero is by construction interpreted by a function from D^0 to D. Since $D^0 = \{\emptyset\}$, we get that the function is uniquely identified by $I(f)(\emptyset)$. This is why these functions are also called constants. We interpret that relation of arity zero is no argument relations and the functions of arity zero is a constants that can be interpreted as no ambiguity and a common consent holds. Marcus Kracht represents complicated argument relations with "λ-calculus". Although it provides a foundation for argument verification and integration it comes with a cost of the computation complexity.

We have proposed three basic calculi for knowledge computations. They are knowledge verification, knowledge integration and knowledge inferring.

3.5. Knowledge Verification

Knowledge verification happens after some concepts are captured from a document with text mining techniques. It is generally consisting of operations of check its contents, verify its text and unify its format. The purpose of the knowledge verification is to reduce noise, remove redundancy and resolve conflict. The key is to define equality "=" in a given domain. Generally, assume we have two claims *C1* and *C2* describing concepts *v1* and *v2*, equality *C1= C2* is defined as:

Definition of equality (=): *C1 = C2 iff v1 = v2*, and

$$\gamma\ (\mathrm{v1}) = \gamma\ (\mathrm{v2}),$$
$$\tau\ (\mathrm{v1}) = \tau\ (\mathrm{v2}),$$
$$\varphi\ (\mathrm{v1}) = \varphi\ (\mathrm{v2}) \quad (5)$$

This definition guarantees the same claims on the same concepts. Despite that they may have different authors (no restriction on authors), the two concepts have the same time and space restrictions and even they have a same type. This is too restrict to be useful in finding the same concepts that they may have different time stamps, come from different authors and different sources even have different types. These conditions reflect the same concepts but from different authors and have different views on it. A relaxed equality is defined as follows:

Definition of loosely equality ($\triangleq$):

$$\mathrm{C1} \triangleq \mathrm{C2} \text{ iff } \mathrm{v1} \approx \mathrm{v2} \quad (6)$$

Where, $\approx$ represents similarity. If two concepts are same, then the two claims are same too despite they may have different time stamps, come from different authors and different sources even have different types. This definition is loose but it is useful in knowledge integration. The essence of the two claims defined above is the reflection of the fact that the concepts are fundamentally talking the same thing.

They are the ones need to be integrated to avoid the redundancy.

3.6. Knowledge Integration

Knowledge integration is vital in resolve problem of patched networks. Integration includes node integration and edge integration. Node integration is to find a same node and using it as an anchor to extent existing knowledge network to include newly created networks. Edge integration has two cases in two different scenarios. One is that a new relation is created on existing two nodes. In this case, the integration of the new relation is simply adding it into the existing edge set. The other scenario is that the newly created relation already exist, in this case the integration is to increase the weight of existing edge in one unit. They are denoted respectively as follows.

For knowledge networks $G_{T,S}$, a new node v, and a new edge e then the Integration $INT\ (G_{T,S}, v)$, $INT\ (G_{T,S}, e)$ and

$$INT\ (G_{T,S}, v) \stackrel{\text{def}}{=} G_{T,S} \rightarrow G'_{T,S}: V' = V \cup v. \text{ i.e.}$$

$$G'_{T,S} = (V', E, \gamma, \rho, \theta, \tau, \varphi, \delta) \quad (7)$$

and,

$$INT\ (G_{T,S}, e) \stackrel{\text{def}}{=} G_{T,S} \rightarrow G'_{T,S}: E' = E \cup v, \text{ when } e \notin \text{E}.$$

$$INT\ (G_{T,S}, e) \stackrel{\text{def}}{=} G_{T,S} \rightarrow G'_{T,S}: \omega(e) = \omega(e) + 1, \text{ when } e \in E.$$

$$G'_{T,S} = (V, E', \gamma, \rho, \theta, \tau, \varphi, \delta) \quad (8)$$

The general integration can be done through these basic integrations.

3.7. Knowledge Inferring

Knowledge inferring is advanced operations on a given graph following argument structure under certain time and space constraints. Apart from the conventional propositional connectives: $\neg$, $\wedge$, $\vee$, and $\rightarrow$, Lambda Calculus *(λ-Calculus)* is useful but out of scope of this report.

4. Implementation of Contest Collective Intelligence Model

The challenges for knowledge acquisition in the web environment are not only on theoretical and computational models, but also on the operational methods. An implementation framework is necessary in providing guidelines on how theory can be used to solve real problems. A pilot implementation in a real application can explore the correctness and the rationale of the model and the implementation framework. Technologies that support CCI model should be coordinated for enabling and facilitating users to express ideas, understandings and interpretations so that others can reflect, build on and learn from it.

4.1. Knowledge Network Construction Framework

In our experimental framework both human and machine annotations (text extraction) are used for the sensemaking activity with the aim of reducing the cost of the process. The framework comprises four stages as illustrated in the Figure 4.

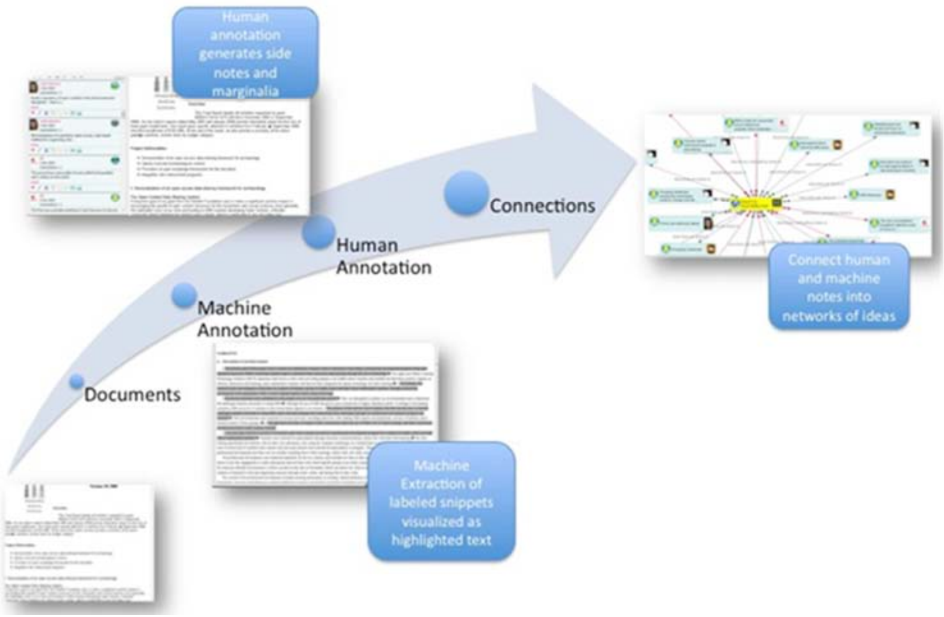

Figure 4. *Conceptual model of the CCI platform[1].*

1 The figure is quoted from De Liddo, A.; Sandor, A. and Buckingham Shum, S. (2012). Contested Collective Intelligence: rationale, technologies, and a human-machine annotation study. Computer Supported Cooperative Work (CSCW), 21(4-5), pp. 417–448.

Stage 1 begins with documents that may in different format for the analyst such as academic publications, posts in a forum, blogs in a personal space, papers, reports, diagrams, charts, etc. The knowledge workers firstly read the documents and trying to identify and extract information and knowledge which can be relevant for the issue they have to investigate. To make a sense of the document they have to read the full documents and in a lot of cases take notes and marginalia. Their annotations mark up as key issues, or evidence to an argument, or extensions to a theory, which may be relevant to a problem, or may be surprising, or contradicting the reader's expectations. Once the notes have been taken they may be used to reflect on the contents of the document, and on what they may imply for the contingent inquiry. Our model aims at assisting the analysts by proposing tools and computer software to carry out these tasks.

In Stage 2 automatic text analysis technologies are used to further retrieve from the document relevant passages conveying contested ideas in a form of claims to reflect thinking. Machine annotation produces two main kinds of output as visual artifacts: sentences and labels. Sentences represent salient contents extracted from the document, and the labels indicate the semantics of the link between the salient content and the document or part of the document.

Stage 3 is human annotation: analysts can validate some of the automatically suggested text snippets and add their interpretation, or they may highlight and comment on new snippets, and thus create further visual artifacts. If the documents are shared by a group of analysts, all the annotations can be used. Human and machine annotation can thus be combined to provide analysts with a view of the salient contents in the document.

Finally stage 4 is the process of encoding the retrieved claims to answer specific questions or making sense of a specific issue. This is a key activity to enable sensemaking or to obtain collective intelligence. It is necessary and supported by a number of specific actions in order to make connections. These actions include validation or verification, integration and duplication removal.

4.2. A Pilot Implementation

Our chosen application is a focused academic publications analysis on the web. The system we are developing is called CCIKS. It is acronym of Contested Collective Intelligence based Knowledge Service. CCIKS system has 4 basic blocks: knowledge acquisition, knowledge computing, knowledge repository and knowledge services.

Knowledge acquisition is using text mining technologies and other software agents search web pages, on-line Wikipedia, other available corpus and knowledge bases finding related knowledge components such as concepts, claims, relations and facts. Import them into local knowledge repository.

Before acquired knowledge can be stored into the knowledge base, knowledge computing in terms of verification and integration are carried out. The purposes are to unify knowledge representation, verify the trueness of the knowledge entity, and resolve conflicts, so that integrations on both concepts and relations can be achieved and patch networks can be connected.

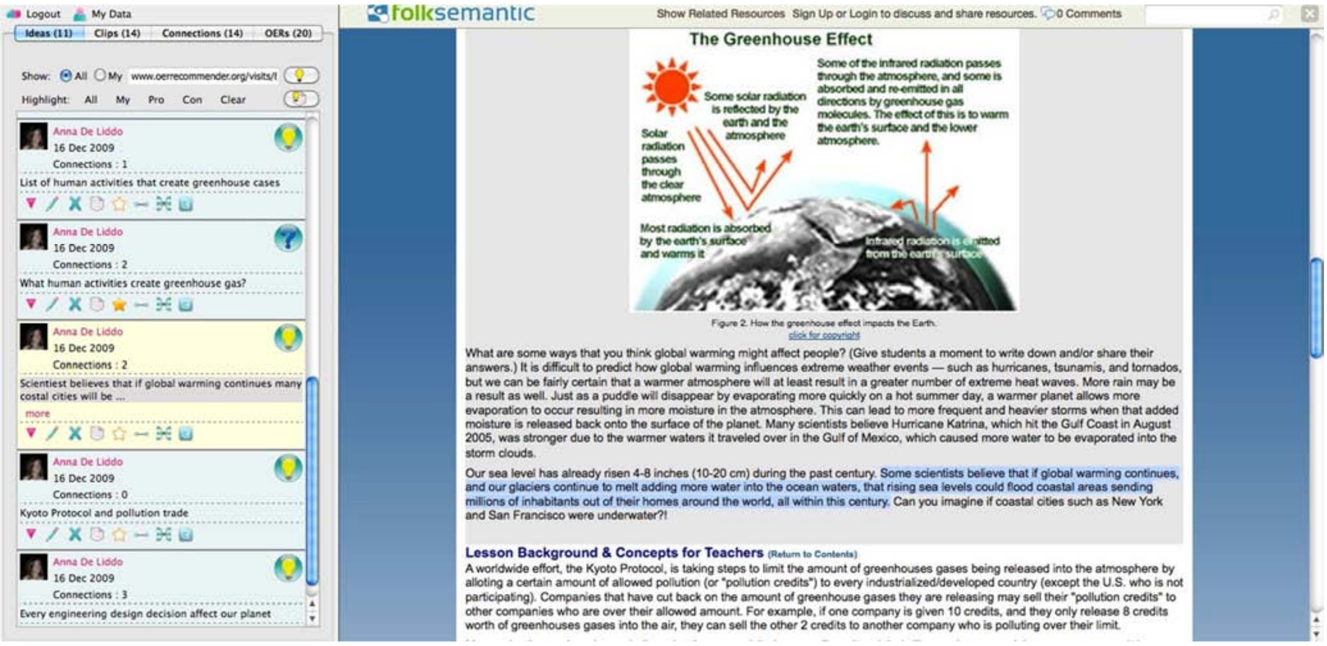

Figure 5. *Screen capture shows that a document on the right is displayed with claims on the left.*

Knowledge repository is a simple arrangement of storage. Apart from the documents and relational metadata, graph database is used to store graph and networks data.

Knowledge services are the added-value of the CCIKS systems. It not only includes keywords search on stored documents as normal document repository does and databases search on metadata as some semantic search engine does, but also advanced knowledge services such as looking for supporting evidences based on given concepts and specific cases that counter the given concepts. More advanced knowledge services like tracing research history, split point analysis, and research trend prediction.

For CCIKS system to function we have the following practical arrangements:

1) Any document stored in the CCIKS is mandated to have a list of claims created either by human or machine using text mining techniques initially as knowledge "seeds". Those claims are also connected together with simple labels reflecting argument structure in types of supportive and disagree.

2) Document display always with current existing related knowledge networks as part of annotation and tag clouds. It serves two purposes. One is visualising the key knowledge discovered from the document and the other is to stimulate users to contribute on the extension of the knowledge networks by providing new concepts and connections. Figure 5 shows a screen capture of the CCIKS. Where, the right hand panel shows the focused document and the left hand panel displays knowledge networks related with the focused document.

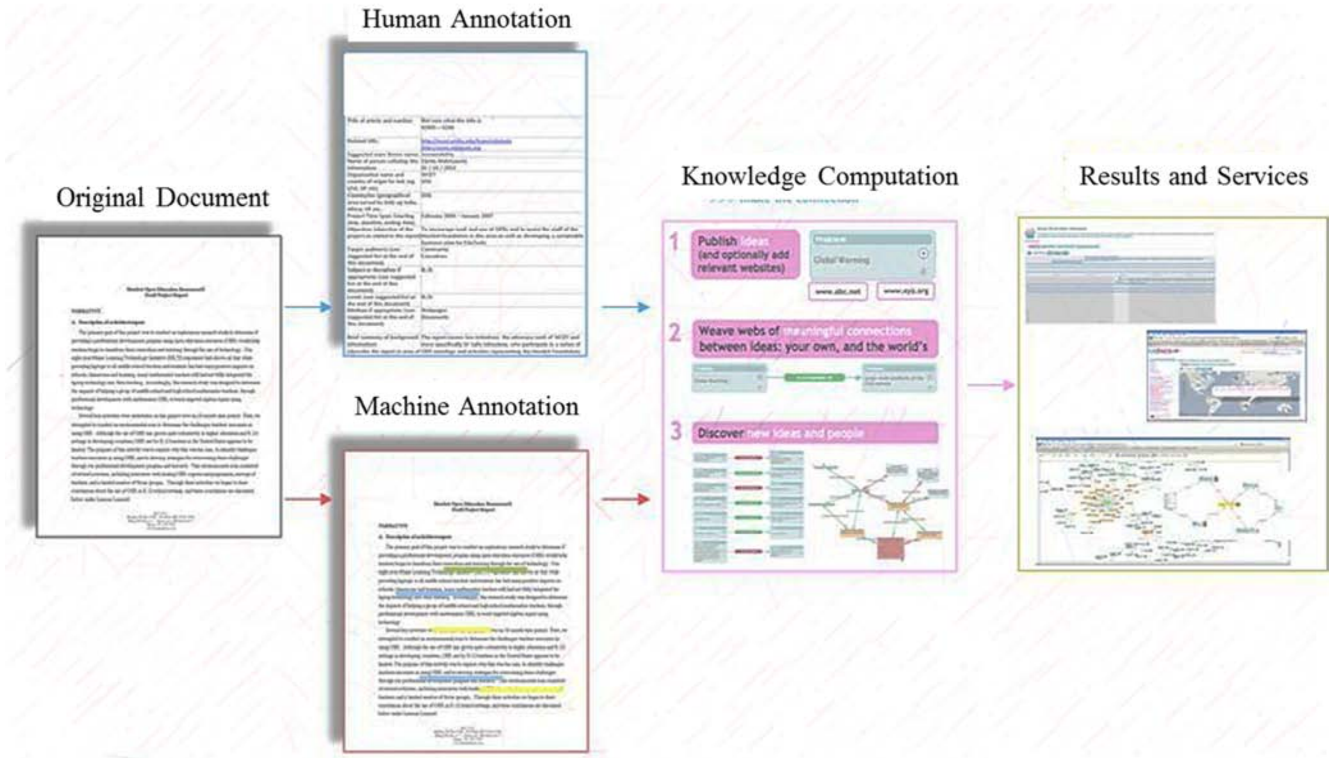

***Figure 6.** Methodology: targeted academic papers have been analysed in parallel; results of annotations have been imported and integrated into the system to generate larger knowledge networks.*

3) Computational methods in CCI model are implemented and studied in the CCIKS system. Both human and machine concepts, claims and connections creation are deployed to test machine creation and construction of knowledge networks with defined knowledge computation. Figure 6 shows the methodology adopted, with human and machine analysis of the corpus conducted independently to enable comparison on the performances and therefore verification on the theory.

5. Conclusion and Future Works

We have made the case that Contested Collective Intelligence (CCI) can be considered as a significant and distinctive method for knowledge acquisition and knowledge services. Research in sensemaking and the modelling of dialogue and debate, motivates a conceptual model of CCI and its formalization which together begins to address the hard problem of the knowledge acquisition. Facilitated by the BDA methods, both machine automatic annotation and human annotation can be deployed into one platform, where complicated computation model can be studied and compare with human manual knowledge networks construction. A pilot implementation in a real application helped us to explore the rationale of the CCI model and issues associated with the implementation.

There are a number of issues discovered. 1) Problems associated with the human creation of claims following CCI model emerged in two areas. One is lack of motivation. Knowledge engineers working on a published academic paper have no motivation to create concepts, claims and making connections since they have to interpret and annotate the whole paper in a defined format. The other one is that the knowledge workers may not be an expert in the given domain, lacking of domain knowledge make it hard for them to interpret and annotate contents in a proper level of correctness. 2) Problems associated with the machine annotation. Purely based on the syntactic markers machine annotation missed some important claims. The redundant claims created by the machine increase the workload for knowledge computation. 3) Integration by machine on both concepts and relations are not very satisfactory because define the same concept by computer program is difficult. So there are still patch networks to be manually connected.

Several strands of ongoing work seek to advance the research programme. We are studying the sentence structure and syntactic markers to improve cue spotter. More efforts will be focused on analysing the property of the same or nearest concepts not only on words, semantics but also on the structure of the networks they are in. We are also working tools to facilitate interoperability, assisting the sharing of datasets, knowledge networks across platforms. A structural search engine will enable more complex queries, and the recognition of patterns that might enable the platform to be more proactive in alerting users to similar situations, and hence, to potential resolutions. As networks grow in size, a recommendation engine will be needed, combined with better visualization interfaces.

Acknowledgements

Research and development reported in this paper are continuation of Cohere project in UK. Cohere was being developed to enable Collective Intelligence for the Open Educational Resource movement, as part of the OLnet Project (http://olnet.org).

References

[1] William Kent. Data and Reality: “A Timeless Perspective on Perceiving and Managing Information in Our Imprecise World”, 3rd Edition, ISBN-10: 1935504215, 2012.

[2] Potter, Steven. "A Survey of Knowledge Acquisition from Natural Language". Technology Maturity Assessment (TMA). Retrieved 9 July 2014.

[3] Cambria, Erik. "Affective computing and sentiment analysis" (PDF). IEEE Intelligent Systems 31 (2): 102–107. 2016.

[4] "A Twitter and web sentiment analysis tool". werfamous.com.

[5] Sensemaking Workshops, ACM Computer-Human Interaction (CHI) Conferences, 2012, 2013, 2014, 2015.

[6] Kiku Jones and Lori N. K. Leonard. , “From Tacit Knowledge to Organizational Knowledge for Successful”. KM. W.R. King (ed.), Knowledge Management and Organizational Learning, Annals of Information Systems 4, DOI 10.1007/978-1-4419-0011-1_3, 2009.

[7] Ikujiro Nonaka, Georg von Krogh, “Perspective—Tacit Knowledge and Knowledge Conversion: Controversy and Advancement in Organizational Knowledge Creation Theory”. Organization Science 20(3):635-652. 2009

[8] Xindong Wu. “Data Mining with Big Data”. International Conference on Big Data and Cloud Computing December 28-30, 2013 Xiamen, China, 2013.

[9] V. Terziya,O. Shevchenko, M. Golovianko, “An Introduction to Knowledge Computing”, 2014.

[10] De Liddo, Anna, G. Li and Buckingham Shum, Simon. “Cohere: A prototype for contested collective intelligence”. In: ACM Computer Supported Cooperative Work (CSCW 2010) - Workshop: Collective Intelligence In Organizations - Toward a Research Agenda, February 6-10, 2010, Savannah, Georgia, USA.

[11] Buckingham Shum, S., Uren, V., Li, G., Domingue, J., Motta, E., “Visualising internetworked argumentation”. Visualizing Argumentation: Software Tools for Collaborative and Educational Sense-Making. P. A. Kirschner, Buckingham Shum, S. and Carr, C. London, Springer-Verlag, 185-204. 2003.

[12] De Liddo, A., Sandor, A. & Buckingham Shum, S., “Contested Collective Intelligence: Rationale, Technologies, and a Human-Machine Annotation Study”, Computer Supported Cooperative Work: the journal of collaborative computing, vol. 21, no. 4-5, pp. 417-448, 2012.

[13] Buckingham Shum and De Liddo, Anna. “Collective intelligence for OER sustainability”. Collective Intelligence in Organizations. February 6-10, Savannah, 2010.

[14] De Liddo, A. & Buckingham Shum, S. , “The Evidence Hub: Harnessing the Collective Intelligence of Communities to Build Evidence-Based Knowledge”, Large Scale Ideation and Deliberation Workshop, 6th International Conference on Communities & Technologies, 2013.

[15] G. Li, “Contested Collective Intelligence for Knowledge Networks Construction and Services”, International conference on Computing and Technology Innovation (CTI 2015), 27th-28th May, 2015, best paper.

[16] Klein, G., Moon, B. and Hoffman, R.F. (2006). Making sense of sensemaking 1: Alternative Perspectives. IEEE Intelligent Systems, 21(4), 70-73.

[17] Knight, S., Buckingham Shum, S. & Littleton, K., “Collaborative sensemaking in learning analytics”, CSCW and Education Workshop (2013): Viewing education as a site of work practice, co-located with the 16th ACM Conference on Computer Support Cooperative Work and Social Computing, CSCW 2013.

[18] De Liddo, A. & Buckingham Shum, S., “Improving online deliberation with argument network visualization”, Digital Cities 8. 2013.

[19] Iandoli, L., Quinto, I., De Liddo, A. & Buckingham Shum, S., “A Debate Dashboard to Enhance Online Knowledge Sharing.”, VINE The Journal of Information and Knowledge Management Systems, vol. 42, no. 1, pp. 67-93, 2012.

[20] Marcus Kracht, “Agreement Morphology, Argument Structure and Syntax” 3rd Ed, Los Angeles, September 2005.