Review Article

Huajun Chen and Zhaohui Wu
Zhejiang University, China
Philippe Cudré-Mauroux
Massachusetts Institute of Technology, USA

Semantic Web Meets Computational Intelligence: State of the Art and Perspectives

I. Introduction

In the early sixties, the concept of *Semantic Network* was firstly introduced as a knowledge representation model by cognitive scientist Allan M. Collins, linguist M. Ross Quillian and psychologist Elizabeth F. Loftus [1]. In 1998, the term *Semantic Web (SW)* was coined by Web inventor Tim Berners-Lee as an extension of the current Web [2]. It was described as a giant global *semantic network* of data that is directly consumable and understandable to machines. In contrast to a *hypertext Web* that indicates texts linked to other texts in other places by hyperlinks, the Semantic Web projects a *hyperdata Web* that indicates data objects linked with other data objects across the Web through formal semantics and ontologies. It enables the formation of *a global web of data* or *open linked data* [3] that interlinks distributed data at a Web-scale. The Semantic Web is led by the World Wide Web Consortium (W3C) as an international collaborative movement [4].

Computational Intelligence (CI) [5] is a set of nature-inspired computational approaches that primarily includes Fuzzy Logic Systems (FLS) [6], Evolutionary Computation (EC) [7] and Artificial Neural Networks (ANN) [8]. Fuzzy logic was introduced as a tool to deal with vagueness and uncertainty that is common for human intelligence. Evolutionary computation could produce highly optimized processes by mimicking the population-based evolution. Neural networks is adept at modeling and learning complex relationships by mimicking the human brain.

The Semantic Web, in its intrinsic nature, creates even more sophisticated problems than hypertext Web. Firstly, the uncertain nature of the Web calls for more expressive languages capable of dealing with fuzziness and vagueness in Web semantics. Secondly, in a highly open, decentralized, and vast Web environment, more efficient computational approaches are required to reduce the computational complexity of a diverse of new problems inherent to the Semantic Web. Typical examples include Web-scale query answering and reasoning, distributed semantic storage, complex ontology alignment across multiple domain boundaries, and massive linked data analysis, etc.


PUBLIC DOMAIN PHOTOS.COM/JOY SHRADER

These insights have triggered a body of researches and innovation with a synergy of the Computational Intelligence and the Semantic Web recently [9][10][11][12][13][14]. For examples, Fuzzy Logic has inspired the design of variant fuzzy extensions of several Semantic Web languages [9][10]; Nature-inspired optimization methods such as Genetic Algorithms (GA) [11], Swarm Intelligence (SI) [12], and Artificial Immune Systems (AIS) [13] have been witnessed in optimizing query answering and reasoning over such a vast, decentralized data space; Artificial Neural Networks are equipped by many Semantic Web applications to improve the learning capability.

This article attempts to survey the state of the art of applying CI approaches into Semantic Web applications. We first identify the vital characteristics or challenges inherent to the Semantic Web including vastness, vagueness, and inconsistency in Section II. We then survey existing researches with CI methods incorporated into the Semantic Web from perspectives of these characteristics. In order to present the literature review, we collected and selected those well-established and most representative works. We classified them into three primary categories of CI methods including Fuzzy Logic, Evolutionary Computation, and Artificial Neural Network, corresponding to Section III, IV, and V respectively. We emphasize on the discussion of perspectives and potential future research directions in this arena in Section VI, followed by a conclusion in Section VII.

II. Semantic Web in a Nutshell

A. The Emergence of a Global Linked Data Space

In a nutshell, the key innovative idea of the Semantic Web is to create a ***Global***

Digital Object Identifier 10.1109/MCI.2012.2188592
Date of publication: 13 April 2012

In 1998, the term *Semantic Web (SW)* was coined by Web inventor Tim Berners-Lee as an extension of the current Web. It was described as a giant global *semantic network* of data that is directly consumable and understandable to machines.

Linked Data Space [3] through formal semantics. Tim Berners-Lee suggested four principles to the question of "how" [15]. The first is to use URIs to identify things. The second is to use HTTP URIs so that these things can be referred to and de-referenced by both people and intelligent agents. The third is to publish machine-understandable information about the things when their URIs are de-referenced, using standard Semantic Web languages such as Resource Description Framework (RDF) [16], Web Ontology Language (OWL) [16], and XML. The fourth is to include links (with semantic annotations) to other related URI-identified things in other places on the Web to improve discovery of related information.

Figure 1 illustrates the new architecture of the Internet inspired by Tim Berners-Lee. It contains three major levels of abstraction: Net, Web, and Graph. The Graph layer is thought of as the Semantic Web. It allows Web users or intelligent agents to create and explore the connections between the things or data objects without the awareness of the boundaries of Web sites.

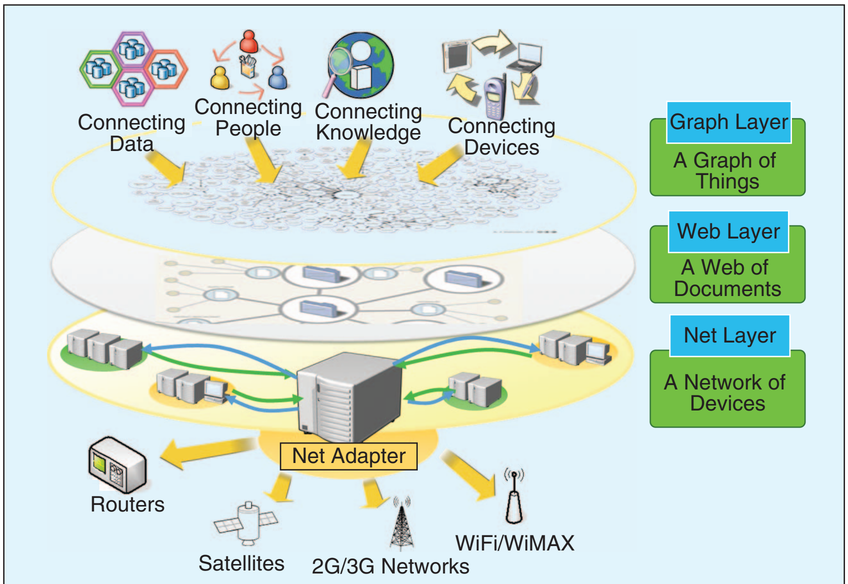

FIGURE 1 Three major levels of abstraction of the Internet: Net, Web, and Graph.

B. The Semantic Web Languages: RDF and OWL

Succinctly, RDF [16] is a data model based on the idea of making statements about Web resources in the form of triple: <*subject, predicate, object*>. The subject denotes a resource, and the predicate denotes an attribute of the resource or a relationship with other resources denoted by the object. For example, we can represent the notion *"Tom has the symptom headache"* in RDF as a triple: <*Tom, hasSymptom, Headache*>. The triple statement model provides an especially straightforward and simple way of describing arbitrary things and their relations on the Web.

OWL [16] goes beyond RDF in its more expressive ability to represent a full ontology that gives a formal, explicit specification of a shared conceptualization for a domain. An OWL ontology typically defines vocabularies for classes, properties, instances and their operations. The data described by an OWL ontology is interpreted as a set of *individuals*, a set of *property assertions* which relate these individuals to each other, with additional axioms that place constraints on classes and properties. These axioms enable systems to infer additional knowledge based on the knowledge explicitly provided. The W3C-endorsed OWL specification [17] includes three variants of OWL family languages including *OWL Lite*, *OWL DL* and *OWL Full* with different levels of expressiveness. The latest version of OWL is OWL 2 [16] that includes three tractable profiles targeted at different types of applications. For example, OWL 2 EL is particularly useful in applications that contain very large numbers of properties and/or classes, OWL 2 QL is aimed at applications that use very large volumes of instance data and require efficient query answering reasoning, and OWL 2 RL is aimed at applications that require scalable rule-based reasoning.

As content in forms of RDF or OWL can manifest itself as self-descriptive data, more accurate, complete and meaningful results can be obtained with the assistance of automated query and reasoning process.

C. Synergy of the Semantic Web and Computational Intelligence

The Semantic Web presents difficult challenges owing to its nature of being decentralized, vast, uncertain, incomplete and inconsistent. We summarize these challenges from three perspectives, on which different CI techniques have been proven to be effective and advantageous.

- **Vastness and Tractability**: The vastness of the Semantic Web is obvious. For example, the W3C Linking Open Data community project [3] has collectively gathered 295 data sets consisting of over 31 billion RDF triples, which are interlinked by around 504 million RDF links as of September 2011. Any automated query and reasoning systems will have to tackle truly huge inputs in a highly optimized and tractable way.

❏ **Vagueness and Uncertainty**: Descriptive data on the Web usually comes along with imprecise concepts like "high" or precise concept with uncertain values [18]. The vagueness and uncertainty may exist in user queries, in data content, in matching queries to data content, and in combining disparate data sources with overlapping ontologies. Both current RDF and OWL languages can only handle crispy descriptions which are definitely not satisfactory.

❏ **Divergence and Inconsistence**: There are always contradictions that inevitably arise when data from divergent sources are combined. In an open system such as the Semantic Web, the aim is to have the agents inter-operate irrespective of the conflicts between their semantics in solving a problem collectively. This requires advanced methods to deal with semantic mapping, ontology alignment, inconsistent reasoning, etc. [19].

In studies related to both CI and the Semantic Web, many attempts have been made to apply variants CI approaches to tackle these challenges. For examples, Evolutionary Computation has been demonstrated to deal with the vastness and tractability issues; Fuzzy Logic has been proven to be effective for the management of vagueness and uncertainty in Web semantics; Artificial Neural Networks have been applied in solving inconsistent issues with regards to data mapping, ontology alignments, and the like. Figure 2 establishes a connection between typical CI approaches and their applications in the Semantic Web, which will be surveyed in detail in the following sections.

III. Semantic Web Meets Fuzzy Logic

One big problem in many Semantic Web applications is to model, store, query, map and reason with fuzziness and vagueness in data semantics [18]. This section first presents several representative fuzzy Semantic Web applications, then introduces the proposed fuzzy languages and their corresponding reasoning methods.

A. Fuzzy Applications in the Semantic Web

Fuzzy concepts are useful in enhancing query or search in the Semantic Web [20][21][22]. For example, a fuzzy concept *Agile Animal* can be used to annotate and index data items about animals in a semantic search engine. A *Tiger* can thus be classified as an *Agile Animal* to a

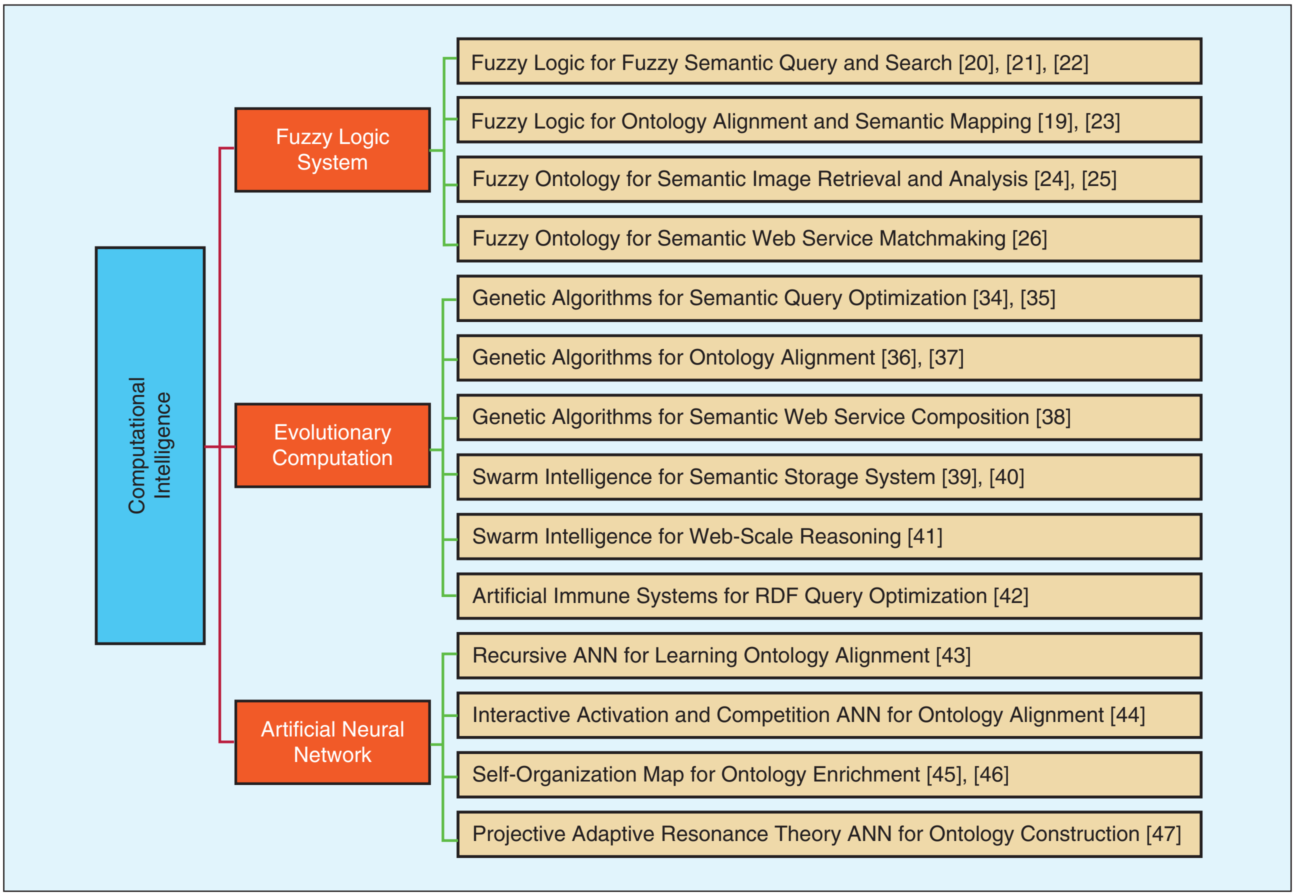

FIGURE 2 An illustration of the landscape of the applications of CI techniques in the Semantic Web.

There are today many proposals for fuzzy extensions to the Semantic Web languages such as fuzzy RDF [27], fuzzy OWL [28] and the most recent fuzzy OWL2 [29].

degree of 0.9. In this way, the degree of relevance to the concept can be determined and ranked while querying and searching based on such a fuzzy semantic index.

Fuzzy approaches have also been applied to resolving semantic conflicts between ontologies [19][23]. Fuzzy semantic mapping can state the generic similarity of two concepts of two different ontologies. We can assert that the objects modeled by the first concept can be also modeled by the second concept to a certain degree. For example, the concept of *TabletComputers* can be mapped to *MobileDevice* to a degree of 0.8. This is very useful in the Semantic Web as the boundaries of semantic mappings between data sources are usually fuzzy.

There are also schemes for using fuzzy ontology to enhance image analysis and retrieval [24][25]. For example, an object *o1* in an image could be *Red* to a degree 0.8 and *closeTo* another object *o2* to a degree 0.6. These fuzzy assertions together with ontology axioms enable agents to recognize objects contained in an image in a fuzzy way so that fuzzy semantic search and analysis can be performed against those images.

The application of fuzzy matchmaking of Semantic Web services is also found in the literature [26]. The matchmaking activity exploits a fuzzy ontology to represent multi-granular information enclosed in the semantic descriptions of a web service using the OWL-S language, an OWL-based semantic markup language for web services. The matchmaking is computed based on a fuzzy distance between user queries and the semantic profiles of matched services.

B. Fuzzy Representation Languages for the Semantic Web

There are today many proposals for fuzzy extensions to the Semantic Web languages such as fuzzy RDF [27], fuzzy OWL [28] and the most recent fuzzy OWL2 [29]. Most of these languages are based on Fuzzy Description Logics [30] [31] that are extended from standard Description Logics (DL) and Fuzzy Set Theory.

At this point we want to make clear that this article is not intended to present a comprehensive and formal specifications on these languages (See [9][10] for such purposes). As an alternative, we use several examples to present the syntax, semantics, and axiom expressions.

We commence from a running scenario in healthcare. As usual, we use $C1, C2\ldots$ to denote concepts and $o1, o2\ldots$ to denote instances. We use DL syntax [31] for simplicity and readability. We firstdefine the *TBox*, which denotes the terminological component of a knowledge base.

$$\begin{aligned} TBox = \{C1 &\equiv YoungPatient \\ &\quad \cap HeavyWeight \\ C2 &\equiv \exists\, hasSymptom.(HighFever \\ &\quad \cap \exists\, hasSign.WeakPulse)\} \end{aligned}$$

C1 denotes the concept for those young patients with heavy weight. *C2* denotes the concept for those patients who have symptoms of both high fever and weak pulse. With the fuzzy concepts defined in the *Tbox*, we can create a corresponding *ABox*, the assertion component of a knowledge base, with fuzzy assertions like the following ones (see the box at the bottom of the page).

If using t-norm (triangular norm), in order for *o1* to be an instance of *C1* it should hold that:

$$\begin{aligned} C1(o1) &= t(YoungPatient(o1), \\ &\qquad HeavyWeight(o1)) \\ &= t(0.8, 0.6) \end{aligned}$$

Depending on which *t-norm* we use, we can infer different values for *o1* being a *C1* or a *C2*.

For example, if *t* is the product *t-norm* then, *C1(o1)= 0.48*.

We may want to define fuzzy axioms in OWL such as *fuzzy subsumption, fuzzy functional role, fuzzy disjointness*, etc. [10]. For example, to define the *fuzzy subsumption* relation between the concepts of *BodyItching* and *SkinSymptom* with a degree 0.9, we use:

$$(BodyItching \sqsubseteq SkinSymptom) > 0.9$$

Last but not least, we show examples on how to concretely represent fuzzy knowledge in an RDF/XML concrete syntax so as to publish fuzzy ontologies on the Web. There are several approaches to encoding fuzzy knowledge [10][29]. One is to extend OWL construct with the elements degree and *ineqType* that takes values such as ">=". The following is a simple example.

<HighFever rdf : about="#symptom01" owlx : ineqType=">=" owlx : degree ="0.7" />

Another approach is to store fuzzy knowledge in the form of OWL annotations [29], thus avoiding the burden of extending the language. The advantage of using annotation is its strong compatibility with existing tools such as parsers or reasoning engines for non-fuzzy ontologies since these tools can simply ignore fuzzy descriptions encoded in the annotations.

C. Reasoning in a Fuzzy Semantic Web

Being similar to crispy OWL knowledge base, reasoning services in fuzzy OWL

$$ABox = \begin{array}{l} \{(o1 : YoungPatient) > 0.8, (o1 : HeavyWeight) > 0.6 \\ (o2 : HighFever) > 0.7, ((o1, o2) : hasSymptom)) > 0.9, \\ \{(o3 : WeakPulse) > 0.5, ((o2, o3) : hasSign)) > 0.8\} \end{array}$$

include: *KB satisfiability, concept n-satisfiability, concept subsumption, and entailment.* In addition, two other important reasoning problems are *the Best Truth Value Bound Problem (BTVBP)* that computes the best lower and upper truth value bounds for an axiom, and the *Best Satisfiability Bound Problem (BSBP)* that determines the maximal degree of truth that a concept may have over all individuals in the domain (See [9] for a formal definition on these problem).

Reasoning services over a fuzzy OWL ontology are normally reduced to reasoning over classical fuzzy description logic [30][31], thus one can implement fuzzy DL reasoners to support reasoning for fuzzy extensions of OWL. There exist several fuzzy reasoners particularly designed for OWL. FiRE [18] is tableaux-based fuzzy reasoner that supports a nominal and datatype-free subset of fuzzy-OWL DL. FuzzyDL[28] is a mixed integer programming fuzzy reasoner that supports fuzzy-OWL Lite. Scalability issue has also been investigated in the literatures [22][32]. Particularly, Pan et al. presents algorithms of fuzzy OWL 2 QL for a family of expressive fuzzy query languages for the Quill query engine in the TrOWL infrastructure [22]; Liu et al. report a MapReduce-based framework that enables scalable reasoning for a subset of fuzzy OWL 2 RL on a cloud infrastructure [32].

IV. Semantic Web Meets Evolutionary Computation

The Semantic Web, in entirety, is a large-scale self-organized complex system that is akin to being evolutionary [33]. In this consideration, new adaptive approaches are required to exploit the ever growing amounts of dynamic, multi-dimensional, and evolutionary semantic data at a Web-scale. This section surveys relevant applications of three typical Evolutionary Computation methods including Genetic Algorithms, Swarm Intelligence, and Artificial Immune Systems, which have been proven to be effective and efficient in reducing the complexity of the problem space with regards to web-scale query answering and reasoning.

The Semantic Web, in entirety, is a large-scale self-organized complex system that is akin to being evolutionary.

A. Genetic Algorithms for the Semantic Web

A Genetic Algorithm (GA) is a search heuristic that generates optimization solutions to problems using techniques inspired by natural evolution, such as inheritance, mutation, selection, and recombination [11]. GA methods have been investigated in optimizing semantic query processing [34][35], reducing complexity for ontology alignment [36][37], and computing optimal composition of Semantic Web services [38].

eRDF [34] is a GA-based framework for optimizing semantic query evaluation. The problem of semantic query evaluation can be formulated as finding semantic subgraphs that match the pattern of a given semantic query. eRDF firstly generates partial matched subgraphs as the *population* of candidate solutions. To propagate new candidate solutions, it then *mutates* the subgraphs by extending the graphs, or *recombines* two candidate subgraphs to generate new candidate solutions. Subgraphs that survive the fitness selection are considered to be optimal. Alexander Hogenboom et al reports similar approaches to finding optimal orders of join-paths for RDF query optimization [35].

GA methods have also been applied in reducing complexity of ontology alignment when a number of matchers are required to be combined to find optimal mappings from huge number of pairs of entities [36]. The problem consists of finding a best combination of weights for different ontology matchers such as those based on string normalization, string similarity, data type comparison, and linguistic methods, etc. The advantages of using GA methods lay in its capability of learning the best combination of weights for optimum alignment functions without requiring human intervention.

Tizzo, N.P. [37] reports the use of Asynchronous Teams algorithm with genetic agents to optimize the composition of Semantic Web services. There are agents responsible for creating new composition patterns. There are other agents performing the crossover and mutation over these patterns based on genetic algorithms.

B. Swarm Intelligence for the Semantic Web

Swarm Intelligence (SI) is the collective behavior of decentralized, self-organized systems [12]. SI systems are typically made up of a population of simple agents interacting locally with each other and leading to the emergence of "intelligent" global behavior. SI methods have inspired researchers to implement self-organized storage systems for large-scale semantic data sets [39][40] and optimize the decentralized Web-scale reasoning process [41].

Scalable query and reasoning over decentralized semantic storage requires both smart strategies of storing semantic data and highly optimized query and reasoning approaches. Ant-inspired swarm approaches have been used to implement distributed storage and retrieval system for large-scale RDF data sets [39][40]. The idea is to model storage operations on RDF triples as ants moving virtual network of nodes. While writing, the ants move from nodes to nodes until finding a number of RDF triples sufficiently similar to the one to be stored, and store it based on similarity clustering over the triples. While reading, the ants regard RDF triples as food and forage from nodes to nodes until finding the similarity clusters containing the particular result. Successful operations trace back the paths they took and maintain virtual pheromones for each node to node connection. Subsequent

The most representative advantage of CI methods for the Semantic Web is their capability to tackle difficult problems in a highly dynamic and decentralized setting.

operations can make use of these pheromones as heuristic rules to evaluate the possibility of finding results if selecting that particular connection. Results show that this approach can generate optimized storage strategy to improve the overall efficiency of semantic storage system.

Swarm approaches are also feasible for optimizing Web-scale reasoning particularly geared towards a decentralized setting [41]. The idea is that reasoning tasks can be decomposed by distributing inference rules to individuals of a self-organized swarm. All individuals "walk" on the triples of a RDF graph locally to expand their knowledge in a parallel way, thereby optimizing the whole reasoning processes significantly owning to the large number of swarms. One advantage of the approach is that triples can be added and deleted to the store at any time without affecting the inference process due to the random behavior and large number of individuals. This is an important feature because of the highly dynamic nature of the Semantic Web.

C. Artificial Immune System for the Semantic Web

Artificial Immune Systems (AIS) are a type of EC systems inspired by the principles and processes of the vertebrate immune system [13]. It typically exploits the immune system's characteristics of learning and memory to solve a problem. AIS has inspired solutions to optimize query answering in the Semantic Web [42]. The idea is to develop an analogy between semantic queries and antibodies of immune systems. Successful antibodies that are activated by an infection are to be cloned and mutated, thus generating a number of similar antibodies better suited to tackle the infection. Analogously, successful queries that produce relevant results are to be cloned and modified to give rise to various similar queries, each of which may be an improvement and mutation on the original query.

V. Semantic Web Meets Artificial Neural Network

An Artificial Neural Network (ANN) is a computational model inspired by the structure and functional aspects of biological neural networks [7]. They are useful to learn complex relationships or patterns hidden in large scale semantic data. Researchers have used ANN to enhance ontology alignment [43][44], ontology enrichment [45], concept mining [46], automatic ontology construction [47], etc. This section surveys and classifies the usage of ANNs in two typical categories: unsupervised ANN for ontology learning and supervised ANN for ontology alignment. For each category, we select two most representative works from the literature.

A. Supervised ANN for Ontology Alignment

In the literature, supervised ANNs are widely applied in learning semantic mappings between heterogeneous ontologies.

Recursive Neural Network model (RNN) [48], designed to process structured data efficiently, is suitable for use with ontologies which are in a structured data representation too. RNN has been used to model automatic ontology alignment [43]. The idea is to use concept structures and instance relations in an ontology as input to a neural network to learn classifiers, which will be used for aligning concepts of other ontologies.

One problem concerning ontology alignment is to find an optimal configuration that can best satisfy ontology constraints, such as "if a concept *c*1 maps to another concept *c*2 is *true*, then concept *c*3 maps to concept c4 is false." The Interactive Activation and Competition (IAC) neural network [49], designed to solve constraints satisfaction problems in word perception, is used to search for a global optimal solution satisfying as many ontology constraints as possible [44]. The idea is to use a node in the IAC neural network to represent an element mapping hypothesis. The connections between nodes represent constraints between hypotheses. If two hypotheses supports or are against each other, the connection between them is positive or negative respectively. A learning process can thus be applied over the network to learn the optimal solution for constraints satisfaction.

B. Unsupervised ANN for Ontology Learning

In the literature, unsupervised ANNs are found to be used to learn new concepts and instances from domain corpus, in order to enrich or automatically construct an ontology.

Self-Organization Map (SOM) [50] is a type of unsupervised neural network that can produce a low-dimensional representation of the input space of the training samples, called a map. SOM has been used to enrich domain ontologies with concepts and instances extracted from a domain text corpus [45][46]. The idea is to first convert an ontology into a neural representation as an initial state of a self-organization map. Next, the terms representing concepts or instances that are to be added into the ontology are extracted from a domain text corpus by text mining process. The actual ontology enrichment takes place via an unsupervised training of the neural network by exposing the initialized SOM to the terms and their contextual information extracted from the domain corpus based on certain types of similarity metrics.

Another type of unsupervised ANN, called the Projective Adaptive Resonance Theory Neural Network (PART) [51], has also been employed to support automatic ontology construction from web pages [47]. The PART is trained to cluster the collected web pages for the sake of looking for representative terms of each cluster of web pages. The representative terms are input to a Bayesian network to complete the hierarchy of the ontology.

VI. Perspectives and Potential Research Directions

The idea of using CI techniques to the Semantic Web has been successfully implemented by many researchers. This section points out the most representative advantages and disadvantages of CI methods for the Semantic Web, and proposes several potential directions for this research area.

A. Advantages and Disadvantages of CI methods for the Semantic Web

The most representative advantage of CI methods for the Semantic Web is their capability to tackle difficult problems in a highly dynamic and decentralized setting. Enriching the Web with semantics and enabling Web intelligence are nontrivial missions due to the decentralized nature of the Web. The fact that no central components are on duty imposes autonomous, dynamic, uncertain, and random behaviors on its constituents. Autonomy, uncertainty, and randomness have been well studied in nature-inspired approaches and the CI communities have accumulated a wealth of well-established approaches. These approaches are particularly adept in addressing the challenges of dealing with autonomy, uncertainty, randomness and chaos.

One disadvantage of currently available CI methods is that they are usually targeted at only one or two specific aspects of a problem, and can only resolve problems separately. However, one reality of the Semantic Web is that the problems of vagueness, autonomy, randomness and inconsistence are present simultaneously for many applications. This requires a hybrid and integration of different CI methods, enabling them to work collectively. We extend the discussion in more details in Subsection *D*.

B. Broader Use of Nature-inspired Methods for the Semantic Web

It is noticed that the "marriage" of the two fields is still new, and only a small portion of this vast wealth have been explored in the Semantic Web community. Therefore, it may be interesting to try a broader variant of CI techniques such as ant colony optimization, particle swarm optimization, harmony search, memetic algorithm, multi-valued logic, chaos theory, and many more, into the Semantic Web. For example, Harmony Search (HS) may overcome the drawback of GA's building block theory which works well only if the relationship among variables is carefully considered. It is thus promising to explore HS methods to evaluate more complex semantic queries that may involve a number of variables. Chaos theory is promising to facility content discovery in the Semantic Web. The chaos theory can be used to create useful content focal points from the chaotic mess of semantic data resources distributed across the Web. Instead of looking for repeatable semantic patterns, it looks for high-level correlations or trends by exploring the chaotic nature of the Web.

On the other hand, from the perspective of the Semantic Web, it may also be interesting to investigate how to use different CI techniques to resolve broader problems in the Semantic Web such as ontology evolution, query rewriting, provenance tracking, linked data mining, semantic routing optimization, etc. For examples, linked data mining is a promising field where CI methods can be particularly useful [52], given the astonishing size of the linked data generated so far and the decentralized nature of the data. Provenance tracking is another interesting field that remains unexplored. Provenance tracking requires tracing, recording and querying the paths of data production from one site to another site. Such path relations can be extremely complex in the context of the Semantic Web. Ant-inspired method may be helpful when dealing with such a complexity.

C. Research on Emergent Semantics and Self-Organizing Semantic Web

Considering the complexity of the Semantic Web, it is not far-fetched to regard it as a complex system where the chaotic and loosely-defined nature of the Web need to be tamed by novel approaches. In such a system, global structures embodying global semantic agreement may probably emerge from a multiplicity of pair-wise, local interactions between data sources and agents, generating eventually a self-organizing semantic infrastructure. These statements project the nature-inspired view of "emergent semantics in the Web and evolutionary Semantic Web" [33] that imposes a complex system perspective on the problem of dealing with semantics and intelligence on the Web.

We state that these open new fields study how semantics can emerge and semiotic relations can originate, spread, and evolve over time in a social Web by combining recent advances in a number of nature-inspired methods such as evolutionary computation, chaos theory, and self-organization systems.

D. Hybrid Approaches of EC, NN, and FL for Semantic Web Applications

The reality that the problems of vastness, autonomy, vagueness, randomness, and inconsistence exist simultaneously poses more difficulties for many Semantic Web applications. For example, mapping data across the Web induces the problem that the space of mapping possibilities among a multitude of data sources is especially rich. Meanwhile, resolving the mapping heterogeneity imposes the problem that the meaning of mapping is usually fuzzy. The algorithms need to take care of both uncertainty and scalability issues in many applications. The situation also exists in large-scale query routing and integrative reasoning among intelligent agents that may hold fuzzy knowledge on their status.

The way out of such conundrum may lay in a combination of variant CI techniques. For example, Neural Fuzzy System (NFS) [53] refers to the synthesis of neural network and fuzzy logic. The neuro-fuzzy combination results in a hybrid approach that fuzzy reasoning style is integrated with the learning and connectionist structure of neural networks. For another example, the synthesis of genetic algorithm with fuzzy logic extends its capability of dealing with fuzzy information in complexity reduction and optimization [54] [55]. We believe it could be a considerable research direction to take in these hybrid approaches of EC-FL [54][55], NN-FL[53], or EC-NN [56] to design systematic solution to tackle the hybrid challenges faced by the Semantic Web community.

VII. Conclusions

The Semantic Web, as a decentralized complex system, is akin to be fuzzy and evolutionary. In this article, we have provided a comprehensive survey on the applications of variant Computational Intelligence methods to enhance a variety of Semantic Web applications. The survey consists of three aspects: fuzzy logic to deal with vagueness and uncertainty in Web semantics; evolutionary computations to deal with the vastness and tractability issues in storing, querying, reasoning and mapping semantic data; artificial neural network to improve the learning capability of the Semantic Web. Based on the survey of the existing approaches in the literature, some potential future research directions in this area have also been discussed and proposed.

Acknowledgments

We thank all of our colleagues including Dr. Jeff Pan, Dr. Tong Yu, and Dr. Chunyin Zhou, for the fruitful discussion. The work of the authors is funded by NSFC61070156, and national 863 program *China Cloud Initiative*.

References

[1] A. M. Collins and M. R. Quillian, "Retrieval time from semantic memory," *J. Verbal Learn. Verbal Behav.*, vol. 8, no. 2, pp. 240–247, 1960.

[2] T. Berners-Lee, J. Hendler, and O. Lassila, "The semantic web," *Scientif. Amer. Mag.*, 2001.

[3] M. Fischett, "The web turns 20: Linked data gives people power—Part 1 of 4," *Scientif. Amer.*, 2010.

[4] World Wide Web Consortium (W3C). W3C semantic web activity [Online]. Available: http://www.w3.org/2001/sw/

[5] M. J. Er and R. J. Oentaryo, "Computational intelligence: Methods and techniques [Book Review]," *IEEE Comput. Intell. Mag.*, vol. 6, no. 4, pp. 76–78, 2011.

[6] J. Mendel, L. Zadeh, E. Trillas, et al., "What computing with words means to me," *IEEE Comput. Intell. Mag.*, vol. 5, no. 1, pp. 20–26, 2010.

[7] C. A. Coello, "Evolutionary multi-objective optimization: A historical view of the field," *IEEE Comput. Intell. Mag.*, vol. 1, no. 1, pp. 28–36, 2006.

[8] S. Haykin, *Neural Networks: A Comprehensive Foundation*. Englewood Cliffs, NJ: Prentice-Hall, 1998.

[9] T. Lukasiewicz and U. Straccia, "Managing uncertainty and vagueness in description logics for the Semantic Web," *J. Web Semantics*, vol. 6, no. 2, pp. 291–308, 2008.

[10] G. Stoilosa, G. Stamoub, and J. Z. Pan, "Fuzzy extensions of OWL: Logical properties and reduction to fuzzy description logics," *Int. J. Approx. Reason.*, vol. 51, no. 6, pp. 656–679, July 2010.

[11] D. E. Goldberg, *Genetic Algorithms in Search Optimization and Machine Learning*. Reading, MA: Addison-Wesley, 1989.

[12] C. Blum and D. Merkle, *Swarm Intelligence: Introduction and Applications*. New York: Springer-Verlag, 2008.

[13] D. Dasgupta, "Advances in artificial immune systems," *IEEE Comput. Intell. Mag.*, vol. 1, no. 4, pp. 40–49, 2006.

[14] G. Acampora, V. Loia, and M. Gaeta, "Exploring e-learning knowledge through ontological memetic agents," *IEEE Comput. Intell. Mag.*, vol. 5, no. 2, pp. 66–77, 2010.

[15] T. Berners-Lee. Linked data web architecture note, W3C [Online]. Available: http://www.w3.org/DesignIssues/LinkedData.html

[16] I. Horrocks, "From SHIQ and RDF to OWL: The making of a web ontology language," *J. Web Semantics: Sci. Services Agents World Wide Web*, vol. 1, no. 1, pp. 7–26, 2003.

[17] W3C Standard Specification. OWL 2 Web ontology language document overview [Online]. Available: http://www.w3.org/TR/owl2-overview/

[18] G. Stoilos, N. Simou, G. Stamou, and S. Kollias, "Uncertainty and the semantic web," *IEEE Intell. Syst.*, vol. 21, no. 5, pp. 84–87, 2006.

[19] A. Ferrara, D. Lorusso, G. Stamou, et al., "Resolution of conflicts among ontology mappings: A fuzzy approach," in *Proc. 3rd Int. Workshop Ontology Matching (OM),* Karlsruhe, Germany, 2008, pp. 121–130.

[20] M. Holi and E. Hyvonen, "Fuzzy view-based semantic search," in *Proc. Asian Semantic Web Conf.*, 2006, pp. 123–135.

[21] L. Zhang, Y. Yu, J. Zhou, et al., "An enhanced model for searching in semantic portals," in *Proc. Int. World Wide Web Conf.* ACM Press, 2005, pp. 453–462.

[22] J. Pan, G. Stamou, G. Stoilos, and E. Thomas, "Scalable querying services over fuzzy ontologies," in *Proc. Int. World Wide Web Conf.*, Beijing, 2008, pp. 340–348.

[23] A. Bahri, R. Bouaziz, and F. Gargouri, "Dealing with similarity relations in fuzzy ontologies," in *Proc. 16th IEEE Int. Conf. Fuzzy Systems*, 2007, pp. 1836–1841.

[24] N. Simou, T. Athanasiadis, G. Stoilos, and S. Kollias, "Image indexing and retrieval using expressive fuzzy description logics," *Signal, Image Video Process.*, vol. 2, no. 4, pp. 321–335, 2008.

[25] M. d'Aquin, J. Lieber, and A. Napoli, "Towards a semantic portal for oncology using a description logic with fuzzy concrete domains," in *Fuzzy Logic and the Semantic Web, Capturing Intelligence*, E. Sanchez, Ed. New York: Elsevier, 2006, pp. 379–393.

[26] G. Fenza, V. Loia, and S. Senatore, "A hybrid approach to Semantic Web services matchmaking," *Int. J. Approx. Reason.*, vol. 4, no. 8, pp. 808–828, 2008.

[27] V. Vaneková, J. Bella, P. Gurský, and T. Horváth, "Fuzzy RDF in the semantic web: Deduction and induction," in *Proc. Workshop Data Analysis*, 2005, pp. 16–29.

[28] G. Stoilos, G. Stamou, V. Tzouvaras, et al., "Fuzzy owl: Uncertainty and the Semantic Web," in *Proc. Int. Workshop OWL: Experiences and Directions*, 2005, pp. 230–242.

[29] B. Fernando and U. Straccia, "Fuzzy ontology representation using OWL 2," *Int. J. Approx. Reason.*, vol. 52, no. 7, pp. 1073–1094.

[30] U. Straccia, "Reasoning within fuzzy description logics," *J. Artif. Intell. Res.*, no. 14, pp. 137–166, 2007.

[31] G. Stoilos, G. Stamou, V. Tzouvaras, et al., "Reasoning with very expressive fuzzy description logics," *J. Artif. Intell. Res.*, vol. 30, no. 5, pp. 273–320, 2007.

[32] C. Liu, G. Qi, H. Wang, and Y. Yu, "Large scale fuzzy pD*reasoning using MapReduce," in *Proc. Int. Semantic Web Conf.*, 2011, pp. 405–420.

[33] S. Staab, "Emergent semantics," *IEEE Intell. Syst.*, vol. 17, no. 1, pp. 78–86, 2002.

[34] E. Oren, C. Guéret, and S. Schlobach, "Anytime query answering in RDF through evolutionary algorithms," *Lect. Notes Comput. Sci.*, vol. 5318, pp. 98–113.

[35] A. Hogenboom, V. L. Milea, F. Frasincar, and U. Kaymak, "Genetic algorithms for RDF query path optimization," in *Proc. Int. Workshop Nature Inspired Reasoning for the Semantic Web (CEUR Workshop Proc.*, vol. 419).

[36] J. Martinez-Gil and E. Alba, "Optimizing ontology alignments by using genetic algorithms," in *Proc. Int. Workshop Nature Inspired Reasoning for the Semantic Web (CEUR Workshop Proc.*, vol. 419).

[37] J. Wang, Z. Ding, and C. Jiang, "GAOM: Genetic algorithm based ontology matching," in *Proc. IEEE Asia-Pacific Conf. Services Computing*, 2006, pp. 617–620.

[38] N. P. Tizzo and J. M. A. Cardozo, "Automatic composition of semantic web services using A-Teams with genetic agents," in *Proc. IEEE Congr. Evolutionary Computation*, June 2011, pp. 370–377.

[39] H. Mühleisen, A. Augustin, T. Walther, et al., "A self-organized semantic storage service," in *Proc. 12th Int. Conf. Information Integration and Web-Based Applications and Services,* Paris, France, Nov. 2010, pp. 357–364.

[40] R. Tolksdorf and R. Menezes, "Using swarm intelligence in linda systems," in *Proc. 4th Int. Workshop Engineering Societies in the Agents World*, 2003, pp. 2004–2014.

[41] K. Dentler, C. Guéret, and S. Schlobach, "Semantic web reasoning by swarm intelligence," in *Proc. 5th Int. Workshop Scalable Semantic Web Knowledge Base Systems*, 2009.

[42] R. Kashif Ali and S. Cayzer, "AIS and semantic query," in *The Semantic Web: Research and Applications (Lecture Notes in Computer Science*, vol. 3532), pp. 1–13, 2005.

[43] A. Chortaras, G. B. Stamou, and A. Stafylopatis, "Learning ontology alignments using recursive neural networks," in *Proc. Int. Conf. Neural Networks (Lecture Notes in Computer Science*, vol. 3697), Poland. Berlin: Springer-Springer-Verlag, 2005, pp. 811–816.

[44] M. Mao, Y. Pengb, and M. Springc, "An adaptive ontology mapping approach with neural network based constraint satisfaction," *J. Web Semantics: Sci., Services Agents World Wide Web*, vol. 8, no. 1, pp. 14–25, Mar. 2010.

[45] S. C. Emil and A. L. Ioan, *Self-organizing Maps in Web Mining and Semantic Web*. Self-Organizing Maps, InTech, 2010.

[46] T. Honkela and M. Pöllä, "Concept mining with self-organizing maps for the semantic web," in *Advances in Self-Organizing Maps* (Lecture Notes in Computer Science, vol. 5629), pp. 98–106, 2009.

[47] R. C. Chen and C. H. Chuang, "Automating construction of a domain ontology using a projective adaptive resonance theory neural network and Bayesian network," *Expert Syst.*, vol. 25, no. 4, pp. 414–430.

[48] D. Mandic and J. Chambers, *Recurrent Neural Networks for Prediction: Learning Algorithms, Architectures and Stability*. New York: Wiley, 2001.

[49] J. McClelland and D. Rumelhart, *Explorations in Parallel Distributed Processing: A Handbook of Models, Programs, and Exercises*. Cambridge, MA: MIT Press, 1988.

[50] S. Haykin, "Self-organizing maps," in *Neural Networks—A Comprehensive Foundation,* 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1999.

[51] Y. Cao and J. Wu, "Dynamics of projective adaptive resonance theory model: The foundation of PART algorithm. *IEEE Trans. Neural Networks*," vol. 15, no. 2, pp. 245–260.

[52] D. S. Martino and A. Sperduti, "Mining structured data," *IEEE Comput. Intell. Mag.*, vol. 5, no. 1, pp. 42–49, 2010.

[53] C. T. Lin and C. S. G. Lee, *Neural Fuzzy Systems: A Neuro-Fuzzy Synergism to Intelligent Systems*. Upper Saddle River, NJ: Prentice Hall, 1996.

[54] O. Cordon, F. Herrera, F. Hoffmann, et al., *Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases*. Singapore: World Scientific, 2002.

[55] A. P. Chen and Y. C. Hsu, "Dynamic physical behavior analysis for financial trading decision support," *IEEE Comput. Intell. Mag.*, vol. 5, no. 4, pp. 19–23, 2011.

[56] P. Durr, C. Mattiussi, and D. Floreano, "Genetic representation and evolvability of modular neural controllers," *IEEE Comput. Intell. Mag.*, vol. 5, no. 3, pp. 10–19, 2010.