A Bigraph Representation Model and Directional Search Mechanism for Debates

Diep Thi-Ngoc Nguyen
Graduate School of Media and Governance
Keio University, Japan
Email: chupi@sfc.keio.ac.jp

Yasushi Kiyoki
Graduate School of Media and Governance
Keio University, Japan
Email: kiyoki@sfc.keio.ac.jp

***Abstract*—The beauty of thinking resides not only in finding answers but also in challenging already existing ones and creating intelligible networks of thoughts and ideas of many different perspectives. Understanding that the essence of critical thinking presents in the flows of questions and answers and aiming at a tool of augmenting it, we propose a new knowledge representation model based on bipartite graphs and a set of functions aided to explore those graphs. A bipartite graph has two disjoint vertex sets that are question and answer sets. Every edge connects a question to an answer holds an evidence for the answer and every edge connects an answer to a question holds an argument raised from the answer. The directional search mechanism is particularly designed to reuse reasoning flows in the debate graphs by a projecting function between the question and answer sets. We also introduce a framework placing the model and the search mechanism altogether in collaborative applications. Many philosophical debates are collected to demonstrate the advantages of the bigraph model besides its simplicity. The analysis and experiments on optimizing semantic search show a sufficient performance for a real-time application.**

Keywords*—*critical thinking augmentation; graph-based data model; directional search; collaborative system.

I. Introduction

There are two main observations combined to motivate our search for a new knowledge representation model. First, the linked data is powerful to represent and discover knowledge but the existing graph representation models are suffering from their complex schemas and varying information descriptions. Second, debates and dialogues are important for critical thinking but there is not yet any successful attempt to systematically organize them so that a querying method can be effectively implemented to augment learning or thinking processes.

The linked and structured data models have always been effective formats for knowledge storing, representation, and discovery. Based on their structure of organizing information, we classify the existing models into three types:

- Wiki style. This type is used in question-answer systems such as Quora[1], Mind the book[2], Yahoo Answers[3], Stackoverflow[4], Linkedln Answers[5], Answers[6], and so on.
- Entity-link style. This type is used in known knowledge graphs such as Dbpedia[7], Freebase[8], Google's Knowledge Graph [9], and others.
- Argument web. Some typical systems for this type are Idebate[10], Truthmapping[11], Debate graph[12], Rationale[13], AIFdb[14], and others. These systems provides argument visualization tools.

The wiki style QA systems are the most conventional way to browse information, but scarce in interrelationships between topics. The argument webs collect and visualize arguments but pay too much attention to the details that lead to complex schemas of argument types, and lack of functionality to expand a topic to others. On the other hand, entity-link models store simple entities as vertices and relations between those entities as edges. When anything can be an entity and and any relation can be an edge, a system of those graphs needs a very complex system of ontologies to keep track of descriptions for information. Moreover, both entities and relations stored in those graphs are often so simple that make us question if they are more computer-friendly than human-friendly. Not yet considering the truth of information that, for example, $\{$a person A – *influenced by* – a person $B\}$, knowing this information may not give us much in improving our thinking skill. Rather we are interested in what ideas the person A, B has proposed and what problems await them since the connections, the flows between ideas seem to be more attractive and useful.

We realize that critical thinking is neither all about being argumentative nor remembering facts, but rather being cooperative and constructive for the sake of understanding and improvement. In other words, our research is not aiming at a tool of persuasion strategies like [12] but a tool to keep all the ideas at their most openness and the connections between them. Moreover, we emphasize the use of information technology to augment human intelligence and thinking skill with a belief

Many thanks for the support of the H27 Keio University Doctorate Student Grant-in-Aid Program and the Taikichiro Mori Memorial Research Fund 2015.

[1]https://www.quora.com/
[2]http://mindthebook.com/
[3]https://answers.yahoo.com/
[4]http://stackoverflow.com/
[5]https://www.linkedin.com/help/ linkedin?lang=en
[6]http://www.answers.com/
[7]http://wiki.dbpedia.org/
[8]http://wiki.freebase.com
[9]https://www.google.com/intl/bn/insidesearch/features/search/knowledge.html
[10]http://idebate.org/debatabase
[11]https://www.truthmapping.com/
[12]http://debategraph.org
[13]https://www.rationaleonline.com/
[14]http://www.aifdb.org/search

978-1-5090-1640-2/16/$31.00 ©2016 IEEE

that IA (Intelligence Amplification) as a contrasting field to AI (Artificial Intelligence) is worth more consideration.

In this paper, we propose a graph model of debate data for representing the diversity of perspectives that links ideas in a form of questions and answers together, and argue its potentials in constructing a collaborative environment for a study of idea across culture and timeline. The table I shows a comparison of the three conventional graph models with our QA bigraph model. Contrasting to other representation models, our bigraph representation model is simple in schema as there are only two types of vertices and two types of edges. We allow the nodes' content to be flexible and preferably informative. There is no need of any ontology system to keep the description of nodes and relations.

Our contributions in this paper are:

- A new knowledge representation model using bipartite graphs aiming at a novel way to organize ideas and the flows between them,
- A set of functions for debate mining, particularly including a directional search mechanism with optimization of semantic similarity calculation for a realtime application,
- An overview of a framework for collaborative applications using the proposed model and search mechanism.

In following sections, we discuss the design, advantages, analysis and evaluation of the model and search mechanism in details.

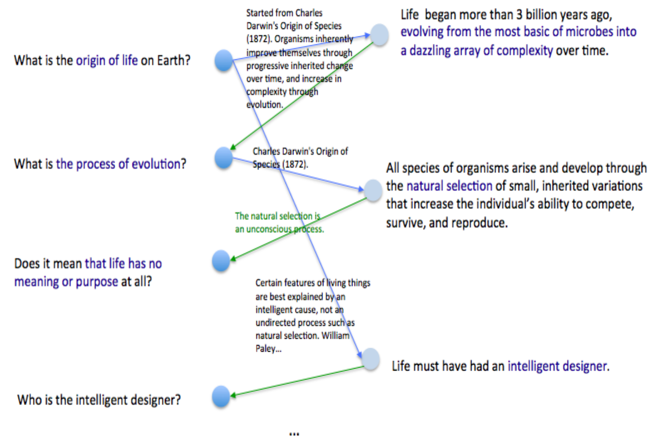

Fig. 1: An example of a QA bigraph of evolution theory debate. The nodes at left side are questions and ones at right side are answers.

II. Bipartite graph representation model

While the content of ideas exchanged during debating discussions vary according to topics and participants' perspectives, they can be classified into four types based on their functionality: *stating an issue, concluding a decision, reasoning or considering supporting information*, and *redirecting the discussion* to a new problem for further investigation. This classification enables a simple graph model that uses the two later types of information to link two former types together, in order to represent information flowing in any discussion. Respectively, the four types are named as **question, answer, evidence,** and **argument**.

In this section, we describe the bipartite graph representation model (or QA bigraph model) and its advantages as well as its potential for information discovery.

A. QA bigraph as data structure

A QA bigraph is a directed bipartite graph consisting of two disjoint vertex sets that are question (Q) and answer (A) sets. The collection of edges of a QA bigraph consists of directed edges from Q to A and A to Q. Every edge connects a question to an answer holds one (or more) evidence for the answer and every edge connects an answer to a question holds an argument raised from the answer.

The figure 1 shows an example of using QA bigraph model to represent a debate relating many evolution theories.

The starting node is the question "What is the origin of life on Earth?", which has two (at least in this example) answers: one is based on Charles Darwin's evolutionary theory and one is based on a necessity of intelligent cause. The interesting and also prominent attribute featured in this kind of QA bigraph is that answers are not ending destinations, but rather starting points of inquiry. In other words, QA graphs promote the motto of "Not take anything for granted". For example, as shown in the figure, if "natural selection" is the answer for "the process of evolution" and even it is widely accepted among society, it should still be challenged by a question of whether life has meaning or not (because the natural selection is an unconscious process). Or, as another example, if someone believes that "we" need an intelligent designer to make life meaningful as it is, then who is the designer?

B. Why questions and answers?

First, the question-answer format is a good format to stimulate thinking process. In the most declarative form, every statement is an answer to a question [10], hence, information can be rewritten in an interrogative mode. Therefore, thinking is driven by questions, not answers. A problem starts with a question, depending on a responder's answer, some criticism will raise. Most criticisms can be expressed as questions, helping to link debates together and to create flows during a discussion.

Secondly, the QA bigraph model can be the most economical model that can represent even complicated discussions, since it has only two kinds of vertices and two kinds of edges is to be considered the one[15].

C. Advantages and applicability

Besides its simplicity, the QA bigraph model has several advantages such as reusable, collaborative, and applicative properties. As to be discussed in next sections, we can combine relating bigraphs to make a larger one (collaboration) or use a part of a bigraph to complete another graph (reuse).

[15]This is inspired by the Occam's razor thinking tool: "Do not multiply entities beyond necessity."

TABLE I: Comparison of QA bigraph model and other graph models

	Wiki style	Entity-link style	Argument web	QA bigraph
Pros	• Entry indexing • Large communities • Traditional way of browsing information	• Structured data • Simple entities and links	• Linking argument data • Interactive displaying and editing	• Structured debates • Simple schema • Interactive displaying, editing, and querying
Cons	• Relations between topics are mostly unorganized • Difficult to search without knowing keywords to query	• Complex schemas and ontologies	• Complex argument types • Difficult to link or expand topics	• Depending on media processing capacity such as natural language processing
	Human-friendly	Computer-friendly	Human-friendly	Human-friendly

Regarding applicability, the QA bigraph model has a natural feature which is suitable for comparing and contrasting ideas. The evidence edges from the question set to the answer set can hold information such as timestamp or culture for timeline and cross-culture contrasting visualization. The figure 2 and 3 show examples of visualization to contrast many answers by timeline and culture, respectively.

In figure 2, the answers for the same question of 'What is the world made of' are varying by a timeline. From an answer based merely on reason, then on observation, then by thinking and experimenting, human knowledge evolves. In figure 3, we see how different Western and Japanese cultures define and find beauty. These two examples show potentials of the QA bigraphs in putting together many aspects at the same place to consider and appreciate.

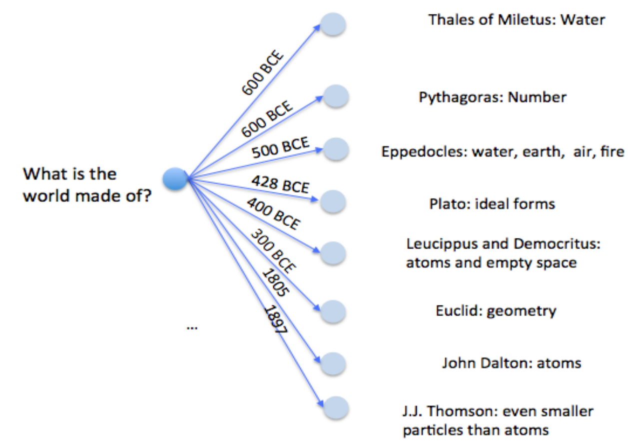

Fig. 2: How our knowledge about the world has changed across a timeline

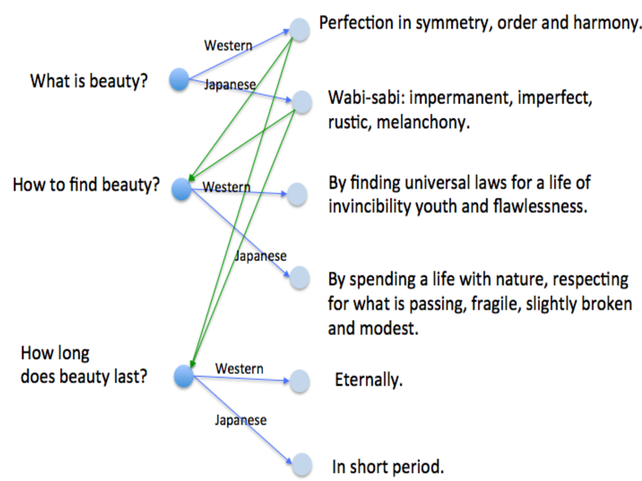

Fig. 3: How different Western and Japanese cultures define and appreciate beauty

III. Debate Mining

In this section, we describe some algorithms can be equipped to the QA bigraph model to aid its functionality. There are two modes that can be observed in critical thinking: a broad mode and a deep mode. In the broad mode, one investigates a problem at different perspectives in order to evaluate the situation at its largest context. In the deep mode, on the other hand, one continuously judges any attempting idea by raising new questions as a means to perfect a solution.

A. *Basic functions*

We can interpret the meaning of some common graph operations that can be applied for mining debates as follows:

- A DFS (depth-first search) as a deep analysis process of a problem.
- A BFS (breadth-first search) at a node as a broad observation for the node.
- Degree (in-degree or out-degree) of a node as information for the diversity.
- A path finding as a dialogue automatic generation function.
- A graph homomorphism as a debate comparison function.

It is to note that the list is suggestive functions for the QA bigraph model but it is not restricted. One with her own imagination can make many other interpretations for other graph operations.

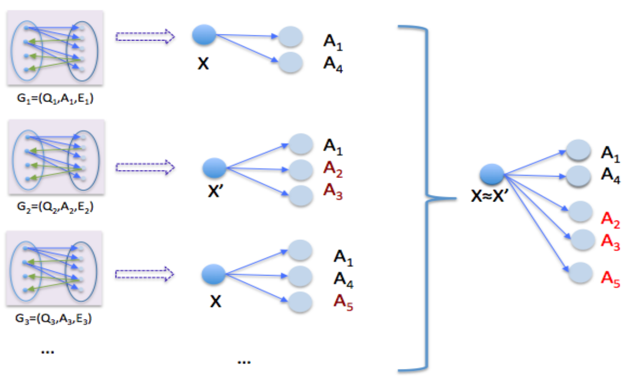

Fig. 4: A demonstration of using bigraphs of others to update one's bigraph

B. *Directional search mechanism*

In this section we describe a 'on-the-way' search method that updates one's graph by using graphs of others as demonstrated in figure 4.

In figure 4, there are three bigraphs G_1, G_2, G_3 that have three subgraphs rooted at X and X'. As X and X' are semantically similar, we want to update our graph G_1 at node X with corresponding edges to three nodes A_2, A_3, A_5 from graphs G_2, G_3.

1) Query generalization: We define a panunion of a collection of sets is the non-distinct set of all elements in the collection. A database is a collection of one or more QA bigraphs. A bigraph database is a data structure consisting of two domains and two one-to-many mapping functions from one to another, which are the panunions of collections of two vertex sets (question, answer), and two edge sets (evidence, argument) of all QA bigraphs, respectively.

2) Directional search: For each point, there are two directions to perform a search: (1) what might be reached from here and (2) what could reach here. The directional search method plays a role of projecting a set of similar instances in one domain to the other domain. The algorithm 1 describes four steps in detail. The function $semanticsimilarity(s, v)$ is a semantic similarity calculation function, varies according to applications. For example, if the instances in search domains are images, $semanticsimilarity$ can be a function calculating how similar two images are. If the instances are texts, $semanticsimilarity$ can be a function calculating how similar two texts are. The section V-B shows an implementation of a $semanticsimilarity$ function for text data.

IV. A FRAMEWORK FOR COLLABORATIVE APPLICATIONS

We describe a framework for collaborative applications using the server-client model as shown in figure 5. At the client side, each user will have her own session to create, edit and view their QA bigraphs. She also queries information from all QA bigraphs stored at a system database then uses the results to update her own bigraphs. By this way of interactive querying, the user is supposed to grasp a broad view regarding her current point of interest, thus she can always learn new ideas 'on-the-way'.

Algorithm 1 Directional search algorithm on a bigraph database

Input:

- A database $DB = (D_1, D_2, F_{12}, F_{21})$ where D_i is a set of instances in the domain i, $F_{i,j}$ is the one-to-many mapping from D_i to D_j.
- A query condition set $\{s, k, f\}$ where s is an instance of domain D_k, and f is the direction of mapping that is either of the type 'in' (*into s*) or 'out' (*out of s*).
- A threshold T for semantic similarity score

Output: A set of instances which is the image of instances similar to s projected in the other domain.

[*Step 1: Get all instances in the same domain*] $V \leftarrow D_k$
[*Step 2: Get similar instances in the same domain*]
$similars \leftarrow \{v \in V | semanticsimilarity(s, v) \geq T\}$
[*Step 3: Get directing function*]
if $f =$ 'in' **then**
  $F \leftarrow F_{\{1,2\}\setminus k,k}$
else if $f =$ 'out' **then**
  $F \leftarrow F_{k,\{1,2\}\setminus k}$
end if
[*Step 4: Return the mapping instances in the other domain*]
return $\{F(v) | v \in similars\}$

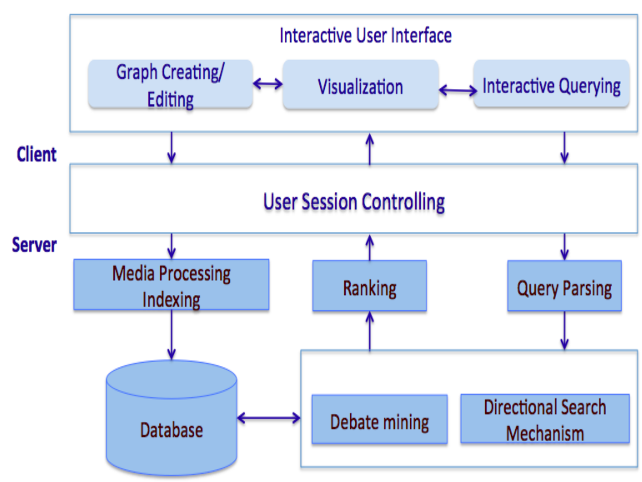

Fig. 5: The system architecture of a web application implementing the proposed model and search mechanism

All computing functions are also implement at the server side. They include database management, query parsing, debate mining, directional search and ranking modules. The parsing and ranking modules depends on actual applications and what is convenient for users to input data for query. In our prototype system, we provide two options: keyword input and click input.

V. OPTIMIZATION AND EVALUATION

A. *Datasets*

In order to evaluate how effective the QA bigraph model can present complicated debates, we chose philosophical discussions to create an experimental database. Those debates are from several philosophy books: “The philosophy book” [1], “The philosophy gym” [3], “What is this thing called knowledge” [11], “The geography of thought” [8] and from Standford Encyclopedia of Philosophy[16] and other online lectures. We manually created 33 QA bigraphs. The total number of nodes (a node is either a questions or an answer) is 617 and the total number of edges is 507. The average number of nodes in a bigraph is 18. The largest bigraph has 101 nodes in total. All content of nodes and edges are English texts.

B. Optimization for semantic similarity calculation

Implementing algorithm 1 requires a concrete implementation of the $semanticsimilarity$ function. For text data, the criteria to choose such an algorithm are: (1) it computes similarity of two texts based on semantic measure, rather than simple vector-based measure, and (2) it responds in a short time (expected less than one second). Malik et al. [4] proposed a good candidate to compute the semantic similarity between two sentences S_1, S_2:

$$score(S_1, S_2) = \frac{\sum_{w \in S_1} maxS(w, S_2) + \sum_{w \in S_2} maxS(w, S_1)}{|S_1| + |S_2|} \tag{1}$$

where, $maxS(w, S_i)$ is the highest semantic similarity of the word w to a word in S_i. The semantic similarity of two words $wordsimilarity$ (in algorithm 2) is calculated by the highest concept-to-concept similarity drawn from semantic networks. There are six metrics to calculate similarity between two concepts which are *path length, Leacock & Chodorow, Wu & Palmer, Resnik, Jiang & Conrath, and Lin*as described in [6].

The implementation of function 1 is straightforward. However, a direct implementation of this semantic similarity function performs very badly as shown in the first result of table III, in which it took averagely 74 seconds to return top 10 similar segments among 617 text segments to one input segment.

1) Semantic similarity calculation optimization using memoization: In order to speed up the $semanticsimilarity$ function based on formula (1) we propose an optimization using a memoization technique as described in algorithm 2. The $memo$ dictionary can be generated by the time of a new computation, or it can be loaded from a file which is the dictionary of some previously loaded sessions. We recommend the later since it shows a better performance as in table III.

2) Optimization using open socket session: Natural language processing often includes loading many heavy trained datasets in order to process a new text segment and this can be a bottleneck for a realtime text retrieval application. NLTK[17] is a great platform to work with text data, but loading some libraries themselves are costly as shown in table II. The table shows running time of four simple python programs, one imports the WordNetLemmatizer class, one imports the PerceptronTagger class, one imports those classes and process a sentence “what is the process of evolution”, and the last one does the same with the third but only processing time is recorded. We can see that the PerceptronTagger class takes averagely almost two seconds to be imported. Processing a sentence without preloaded modules costs mostly six seconds

[16] http://plato.stanford.edu/

[17] http://www.nltk.org/

Algorithm 2 Semantic similarity calculation algorithm with memoization

Input: Two word vectors of two texts $S_1 = [w_{11}, w_{12}, \dots, w_{1m}]$,
and $S_2 = [w_{21}, w_{22}, \dots, w_{2n}]$
Output: Similarity score as a float number

[*Initialize* n*-by-*m *word similarity matrix*] $M \leftarrow 0.0$
[*Initialize memo dictionary*] $memo \leftarrow \{\}$
for i **from** 1 **to** m **do**
  for j **from** 1 **to** n **do**
    if $w_{1i} > w_{2j}$ **then**
      $k \leftarrow (w_{1i}, w_{2j})$
    else
      $k \leftarrow (w_{2j}, w_{1i})$
    end if
    if not k **in** $memo$ **then**
      $score \leftarrow wordsimilarity(w_2, w_1)$
      $memo[k] \leftarrow score$
    end if
    $M[i][j] = memo[k]$
  end for
end for
[*Sum of maximum scores in every column*] $sum_{12} =$ sum([max(M[i][:]) for i in n])
[*Sum of maximum scores in every row*] $sum_{21} =$ sum([max(M[:][i]) for i in m])
return $1/2 * (sum_{12} + sum_{21})/(m + n)$

TABLE II: Loading time of some NTLK classes and processing time of a short text segment using them

Program	Running time (second)
Loading WordNetLemmatizer	0.051
Loading PerceptronTagger	1.831
Processing a sentence	5.708
Processing a sentence with preloaded modules	**0.002**

while the actually processing time costs only about two milliseconds.

Based on this observation, we suggest to use a open socket session aided to actual applications. This session plays as a server which preloads all necessary modules, and implements all algorithms. The actual processing program will call its functions and get the return via its open port.

C. Performance evaluation

This section shows how performance of an interactive search can be improved using our optimization. First, we created two search databases, one database ($DB111$) has only 7 bigraphs with 111 nodes in total and one database ($DB617$) includes the whole dataset with 617 nodes.

Then we implemented four search programs: a direct implementation based on the equation 1, an implementation using memoization as in algorithm 2, an implementation using memoization based on the same algorithm but the memo dictionary is saved and reloaded, and an implementation with two sessions: a server session that is similar to the third one

TABLE III: Average running time of four interactive search programs. With optimization, the response time is significantly improved.

Implementation	Running time (second)	
	$DB111$	$DB617$
Brute force	20.7536	74.4222
Using memoization	8.478	20.1012
Using saved memoization data	2.8158	2.8694
Using saved memoization data and open socket	**0.0416**	**0.0498**

and a client session that makes requests to the server and get the return.

We run each program and recorded their running time in a Macbook computer that has a 1.7GHz Intel Core i7 processor and 8 GB DDR3 memory.

Table III shows the average running time. The program implemented using both optimization as we have introduced responds in shortest time, about 0.05 second in case working with the whole dataset. This is also a reasonable response time as of a realtime interactive query function.

VI. Related works

There are many researches on question-answering systems using graphs such as [7], [2], [9], however, their goals are to make entity-link style knowledge graphs. As discussed in section I, the proposed QA bigraph model has a different goal that is to capture complex ideas and the flows between them in a more human-friendly, rather than to organize simple entities.

Other question answering systems and collections like [4], [13] attempt to use natural language text to organize and search for information. First, our proposed model differs from them in the focus on the continuing of flows between not only question to answer but also answer to question: what questions can be raised from an answer. Secondly, we pay attention to the diversity of answers for a question and organizing them in bigraph format. We assume the essence of critical thinking resides in this kind of format.

Regarding analyzing and visualizing philosophical arguments, McAlister et al. in their Digging by Debating project attempted to visualize arguments in philosophical debates related to animal psychology [5]. They described their choice of philosophical debates as these debates contain a rich vein of arguments and argued the advantages of an argument mapping tool for non-expert with limited or no domain knowledge of philosophy. We agree with McAlister and his coauthors in the choice of philosophical debates. Additionally, we emphasize the usefulness of philosophy in critical thinking like Law stated "an advantage of a little philosophical training is that it can provide the skills needed to think independently and question what others might take for granted" [3]. Nonetheless, our approach to philosophy differs from theirs since we draw more attention in the flows between arguments and evidences in QA bigraphs. By focusing on this point, we pay a special attention on the potential of reusing ideas, concretely, how to raise a good question and how to retrieve different perspectives for every question.

VII. Conclusions

In this paper, a new knowledge representation model and a set of mining and search functions for realtime collaborative applications are proposed. We have described with attentive analysis the advantages and novelty of this model over other models which also use graph for representing information. The QA bigraph model is supposed to be more human-friendly and augments human critical thinking skills since it focuses on not facts but the flows between ideas and their diversity.

Philosophy is always considered a difficult discipline to study, but the simplicity of the QA bigraph model demonstrates the hidden poise of many philosophical discussions. Those difficult debates can be shown in simple bigraphs so that the ideas and their connections become clear and intelligible. Above all, thinking is similar to making networks connecting thoughts and ideas regarding various perspectives. The appreciation of perspectives, across timelines, regarding cultures is the skill a person should develop besides the skill of finding solutions. The bipartite knowledge presentation model coupled with the directional search mechanism is yet simple but effectively augments that skill.

One challenge to the QA bigraph model is in the data creation process itself. May our next challenge be a development of a tool that can automatic collect and create good bigraphs.

References

[1] W. Buckingham. *The Philosophy Book*. London, New York, DK Pub, 2011.

[2] B. Hixon, P. Clark, and H. Hajishirzi. Learning knowledge graphs for question answering through conversational dialog. In *Proceedings of the the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA*, 2015.

[3] S. Law. *The Philosophy Gym: 25 short adventures in thinking*. Macmillan, 2003.

[4] R. Malik, L. V. Subramaniam, and S. Kaushik. Automatically selecting answer templates to respond to customer emails. In *IJCAI*, volume 7, pages 1659–1664, 2007.

[5] S. McAlister, C. Allen, A. Ravenscroft, C. Reed, D. Bourget, J. Lawrence, K. Börner, and R. Light. From big data to argument analysis. *Intelligence*, page 27, 2014.

[6] R. Mihalcea, C. Corley, and C. Strapparava. Corpus-based and knowledge-based measures of text semantic similarity. In *AAAI*, volume 6, pages 775–780, 2006.

[7] D. Mollá. Learning of graph-based question answering rules. In *Proceedings of the First Workshop on Graph Based Methods for Natural Language Processing*, pages 37–44. Association for Computational Linguistics, 2006.

[8] R. Nisbett. *The Geography of Thought: How Asians and Westerners Think Differently... and*. Simon and Schuster, 2010.

[9] A. K. Pappu and A. I. Rudnicky. Learning situated knowledge bases through dialog. *Proceedings of Interspeech*, 2014.

[10] R. W. Paul and L. Elder. Critical thinking handbook: Basic theory and instructional structures. *Retrieved November*, 3:2005, 2000.

[11] D. Pritchard. *What is this thing called knowledge?* Routledge, 2013.

[12] C. Tan, V. Niculae, C. Danescu-Niculescu-Mizil, and L. Lee. Winning arguments: Interaction dynamics and persuasion strategies in good-faith online discussions. *arXiv preprint arXiv:1602.01103*, 2016.

[13] E. M. Voorhees and D. M. Tice. Building a question answering test collection. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 200–207. ACM, 2000.