The 7th International Conference on Semantics, Knowledge and Grids, Beijing, China, Oct 24-26, 2011.

The Textual Semantic Lens

B.Xu[1, 2] and H.Zhuge[1]

[1] *China Knowledge Grid Research Group, Key Lab of Intelligent Information Processing*
Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100080, China
[2] *Graduate School of the Chinese Academy of Sciences*
xubei@kg.ict.ac.cn
zhuge@ict.ac.cn

***Abstract*—People often feel the limitation of time to read the continuously increasing articles they need to read. It is a grand challenge to handle the explosion of articles. To understand how humans read articles and get the meaning is the basis of improving the efficiency of reading articles. The underlying semantic links between language units of different granularities reflect some basic semantics. Sentence is the basic language unit for accurately indicating semantics. By defining concepts and the dependent set of sentences, and constructing the semantic link network of dependent sentences and semantic link network of concepts, this paper proposes the textual semantic lens with a set of functions for helping people comprehend articles. Integrating with the semantic link networks of articles, the semantic lens can help people efficiently read large-scale articles.**

I. INTRODUCTION

Continuously increasing articles limit people efficiently getting useful information.

The first solution is automatic abstracting. It focuses on how to extract important information from text [3], but it does not concern multi-facet contents and neither reflects users' personal needs as users' requirements may change. For instance, different users may want to read different facets of an article.

The second solution is faceted navigation. It mostly deals with text-annotated data [9], RDF data [11] or images [10], but it seldom considers plain text and the communities in the content of an article in deep comprehension.

This paper proposes the textual semantic lens to help users to extract, browse and analyse texts. It is based on reading ability of humans. Human reading ability extracts and organizes concepts and relations. People generally comprehend a concept in an article from the following four aspects:

What are the concepts? How do they come? What can we know from the concepts? What are the relations between the concepts and the other concepts?

The basic reading ability concerns the following four basic functions.

1. *Identify Concept*. People identify concepts in text. An author can use multiple words or sentences to describe a concept.
2. *Background generation*. The background of a sentence is formed by previous sentences that describe concepts in the sentence. People can generate the background of every sentence in an article while reading. All the backgrounds will merge into a whole background of understanding.
3. *Comprehension.* A sentence can be comprehended if all of its concepts have been comprehended.
4. *Emerge semantic image.* People emerge semantic images in mind while reading [6][12].

The Semantic Link Network (SLN) uses various relations and relational reasoning to support preliminary intelligence such as guided browsing and answer relational queries [5]. The basic form of a self-organized Semantic Link Network is $SLN=<N, L, Rules>$, where N is a set of semantic nodes, anything can be a semantic node; L is a set of semantic links, a semantic link takes the following form: $n \text{—}\alpha\rightarrow n'$, indicating a relation α between two semantic nodes n and n'; and, *Rules* is a set of rules such that new semantic links can be appropriately added or the implicit semantic links can be found as the effect of reasoning. If there are two connected links $n \text{—}\alpha\rightarrow n'$, $n' \text{—}\beta\rightarrow n'' \in L$, and a rule $n \text{—}\alpha\rightarrow n'$, $n' \text{—}\beta\rightarrow n'' \Rightarrow n \text{—}\gamma\rightarrow n''$, then $n \text{—}\gamma\rightarrow n''$ can be added to L. Different from the semantic net [2], SLN focuses on the dynamicity, complex reasoning and emerging semantics [13].

II. IDENTIFY CONCEPT

People usually indicate a concept with one or several words. For example, the concept "*hamlet*" in http://en.wikipedia.org/wiki/Hamlet can be represented as {*hamlet*, *prince*}, where any word can remind people the concept "*hamlet*". Simple concepts can form a complex concept. So, in this paper, a concept is described by a set of associated keywords:

$$C = \{k_i \mid i=1, 2, \ldots\} \quad (2.1)$$

Where C denotes a concept, *and* k_i denotes the ith keyword in C. In one article, two associated keywords indicate the same semantics if they are the same or synonymous.

A sentence indicates a concept if the sentence includes the concept's name or any keyword in the concept has the same keyword within the sentence.

Synonyms are usually used to indicate the same concept. WordNet[4] and FrameNet[1] are helpful to judge wether two words are synonyms and users can define their own synonymous keywords lists.

The idea of semantic lens was first proposed in [6][12]. Focusing on texts, this paper proposes the textual semantic lens, which is expected to recommend some concepts extracted from an article to users by using techniques of automatic abstracting [3]. Authors or readers can recommend representative concepts as incidental files for an article. The textual semantic lens can search the concept files of an article on the Internet. One article may correspond to several concept files provided by different people. For an article, a user can select a suitable concept file before reading, can modify or change concept file while reading, and can share personal concept file on the Internet for others' use after reading. The file represents concepts as form of equation (1).

III. Semantic Link Network of Dependent Sentences

A. Some definitions

A concept's meaning is indicated or enriched when it is indicated by a sentence. To comprehend a concept, a user should comprehend former sentences that indicate it. So, a sentence indicating concepts depends on previous sentences indicating the same concepts.

Definition 1. (*Depend*) *Within an article, a sentence s depends on a set of sentences SS on a concept c, denoted as* $s \xrightarrow{C} SS$, *if*

1. *s indicates c*;
2. *Every sentence in SS indicates c*;
3. *Every sentence in SS occurs before s*; *and*,
4. *All sentences occurred before s indicating c are in SS.*

If $s \xrightarrow{C_1} SS$ and $s \xrightarrow{C_2} SS$, then it is denoted as $s \xrightarrow{C_1,C_2} SS$.

Understanding a concept in an article is a linear increasingly process.

Definition 2. (*Background*) *Assume a sentence s, the background of s is a sentence set. A sentence k is in the background of s if*

1. *k is before s*; *and*,
2. *k and s indicate a common concept.*

A sentence *k* is comprehensible if all sentences before *k* sharing common concept with *k* are in the background.

Definition 3. (*Dependence chain*) *For a concept c in a given article, all the sentences indicate c form a dependence chain DC, where a sentence in DC depends on a subset of DC that includes sentences before the sentence.*

Figure 1 is an example of a dependent chain, where S_k (k=1, 2, 3, 4) indicates *c*. *SS* includes S_1, S_2, and S_3. *SS* is the background of S_4, i.e., S_4 cannot be correctly comprehended without *SS*.

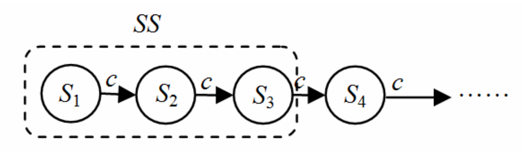


Figure 1. An example of a dependent chain.

Given different set of concepts of an article, different dependent chains may emerge. Figure 2 is an example.

Given an article, a semantic link network of dependent sentences (SLN-DS) can be formed by merging common sentences of dependence chains. An SLN-DS will evolve with adding new concepts while reading. SLN-DS has a characteristic.

No-Loop. *There is no loop in SLN-DS.*

If there is a loop, then in the loop, the background of any sentence indicates itself. It is inconsistent with definition 3.

(1) Interactive semantics = interactive system+ semantic image. (2) Interactive semantics consists of the following two parts that influence each other. (3) Open, self-organized and evolving interactive system. (4) The system consists of individuals, objects, channels, and rules for interaction, reasoning and classification. (5) Individuals can add themselves or objects to the system at any time. (6) All individuals in the system guarantee interaction. (7) Communities are formed and evolve with constant interaction. (8) Semantic image. (9) The semantic image of the interactive system records the images of individuals, objects, relations, classes and interactions within the system. (10) Individuals build their own semantic images based on semantic worldview while leaving tracks in the semantic image of the system. (11) A community in the system has a semantic image that reflects the structure of interactions and consensus on recognizing object, relation and class. (12) Semantic images evolve with constant interaction. [6]

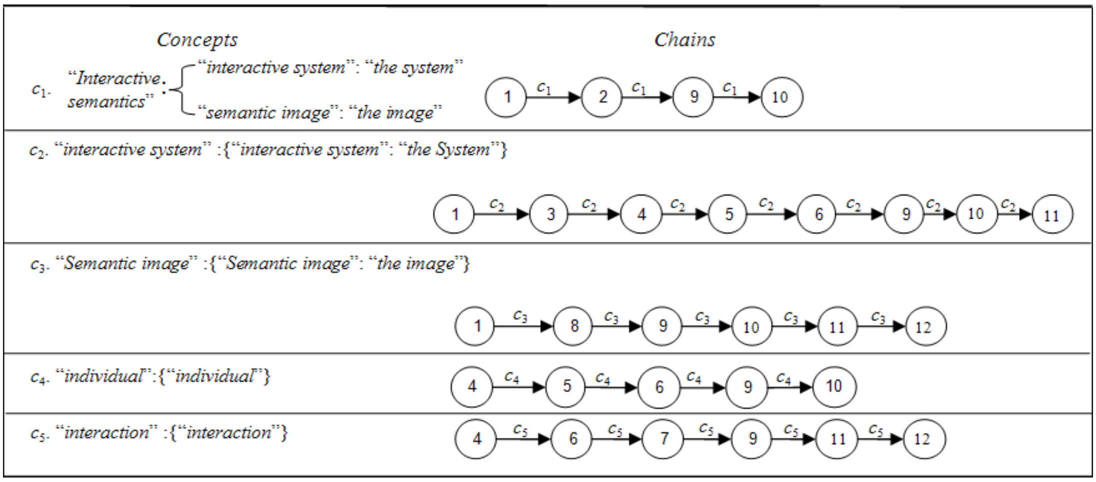


Figure 2. Five dependent chains are extracted from the text by giving five concepts. Each node in chains corresponds to a sentence.

B. Basic operations on SLN-DS

Semantic lens needs some basic operations to operate SLN-DS. The operations correspond to basic functions of reading ability.

Extract operation. *Extract the dependence chains of concepts from a given article.*

People do the operation according to the concept identity function of reading ability.

Merge operation. *Merge the common sentences between dependence chains or SLN-DSs and return a whole SLN-DS.*

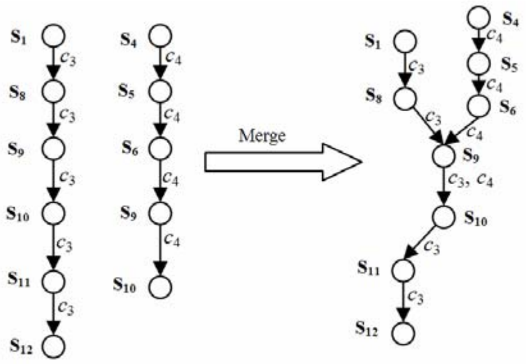


Figure 3. Merge two dependence chains.

Users may generate new concepts in mind anytime while reading. If the concepts are picked out by users, new dependence chains will be generated which will be merged into SLN-DS. Figure 3 is an example of merge operation. The two chains are the third and forth chains in figure 2 which have two common sentences: S_9 and S_{10}. The two dependence chains are merged into one SLN-DS.

Figure 4 shows an SLN-DS of the text in figure 2 by using merge operation.

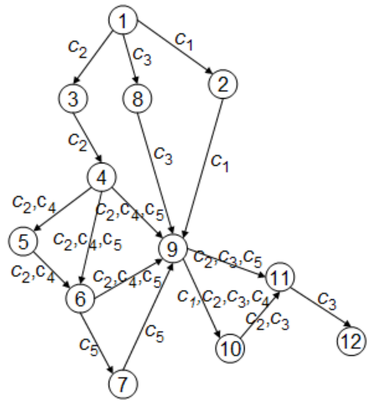


Figure 4. The SLN-DS of the text in Figure 2.

Splitting-up operation. *For a given SLN-DS, a sentence s or a concept c in a sentence, split a subgraph from the SLN-DS as the background of s or c.*

Algorithm *Splitting-up*

Input: An SLN-DS, a sentence *s or* a concept *c* in a sentence.

Output: A sub-graph SLN-DS which is the background of *s or c*.

Steps:

1. Set an empty SLN-DS *Q*.
2. If the input is a sentence *s*,
   Then add *s* into *Q*.
   Else
      If the input is a concept *c* in a sentence,
      Then add the first former sentence indicates *c* into *Q*.
3. If there is a dependence relation from a sentence *k* to any sentence in *Q*.
   Then, add *k* and the dependence relation into *Q*.
4. If *Q* changes, goto step 3; else return *Q*.
5. Algorithm ends.

According to the No-Loop characteristic, the algorithm will not be in endless loop. Figure 5 shows the output of the splitting-up operation on sentence 6 in Figure 4.

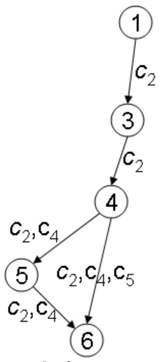


Figure 5. The background of sentence 6 in Fig. 4. To understand sentence 6, a user needs to understand sentence 1,3,4,5 first.

Splitting-down operation. *For a given article, its SLN-DS and a sentence, return a subgraph of the SLN-DS. Sentences in the subgraph are comprehensible based on the given sentence.*

Algorithm *Split-down*
Input: An SLN-DS *M* and a sentence *q*.
Output: A sub-graph SLN-DS in which sentences are comprehended based on *q*.
Steps:

1. Set an empty SLN-DS *Q*, and add *q* to *Q*.
2. If a sentence *s* in *M* satisfies the group of conditions, then add *s* and the dependence relation points at *s* into *Q*.
   (1) There is a sentence in *Q* that *s* depends on; and
   (2) No sentence outside *Q* that *s* depends on; and,
3. If *Q* changes, goto step 2; else return *Q*.
4. Algorithm ends.

If we choose sentence 3 in Figure 4 as background, then we can obtain Figure 6.

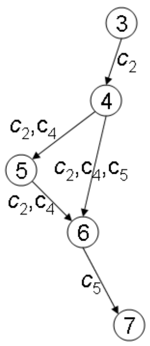


Figure 6. The result of Splitting-Down operation on sentence 3 of Figure 4.The sentences in Figure 6 represent the concept “interactive system” in the text of Figure 2.

IV. Semantic Link Network of Concepts

A. *Definition of semantic link network of concepts* (*SLN-C*)

Concepts are organised to form a network by readers or writers when reading or writing. The semantic link network of concepts reflects the tightness of concepts.

Definition 4. (1-*step relation*) *For two given concepts, if there is a sentence indicating the two concepts, then the two concepts have 1-step relation. The number of sentences that indicate the two concepts is the weight of the* 1-*step relation.*

Definition 5. (*n*-*step relation*) *For two given concepts* c_a *and* c_b, c_a *and* c_b *have n-step relation if there is a concept set* $CS = \{c_1, c_2 ... c_n\}$ *which satisfies:*

1. c_a *and* c_1 *have* 1-*step relation*; c_i *and* c_{i+1} (i=1,…,n-1) *have 1-step relation*; c_n *and* c_b *have 1-step relation.*
2. *There is not a subset of CS set satisfies condition* 1.

The number of concept sets that satisfy the two conditions is the weight of the n-step relation.

Definition 6. (*n-step SLN-C*) *For an article, its n-step SLN-C is a network where the nodes are concepts, the relation between two concepts is a set that consists of all i-step relations* (i<=n) *between the two concepts.*

We choose text from the plot of http://en.wikipedia.org/wiki/Hamlet and obtain Figure 7. Roles in the article are picked out to be concepts, for instance, “*hamlet*”: {“*hamlet*”: “*the prince*”}.

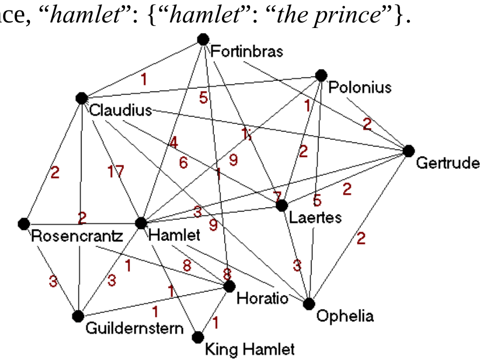


Figure 7. An example of SLN-C. The SLN-C is a 1-step SLN-C. Each node is a concept of the drama. The numbers in the figure are the weight of 1-step relations.

B. *The integrated weight of SLN-C’s relations and importance of SLN-C’s concepts*

Every relation is given an integrated weight *W* reflects the tightness between its two concepts.

$$W = w_1 \times \alpha_1 + w_2 \times \alpha_2 + \ldots + w_n \times \alpha_n = \sum_{i=1}^{n} w_i \times \alpha_i \qquad (2)$$

Where w_i denotes the weight of i-step relation. α_i denotes the ratio of adding the weight of i-step relation. Experience shows that setting α_i as $1/i$ may be suitable.

The most important and tight relation of an article is with the highest integrated weight. As shown in Figure 7, the relation between "*hamlet*" who is the prince and "*Claudius*" who murders the old king is the most important. "*Hamlet*" is the closest concept to "*Laertes*" because "*Laertes*" has relation with the highest integrated value to "*Hamlet*".

Different concepts have different importance to an article. A concept's importance is influenced by its neighbour's importance because a concept slightly influences an article if the concept's neighbours are not important.

For a given SLN-C, the concepts' importance values can be defined as follows:

$$\vec{V} = \lim_{t\to\infty}(\vec{V_0} \times (E)^t) \quad (3)$$

$\vec{V}$ denotes an N-dimension vector, each dimension denotes a concept's importance. N denotes the total number of concepts. E denotes a matrix. The element in the ith row and jth column is: *integrated weight of relation between ith concept and jth concept / the sum of integrated weights of relations related to ith concept*. $\vec{V_0}$ denotes the initialization importance of $\vec{V}$.

Based on the theory of Markov Chain of Stochastic Process, regarding the $N \times N$ matrix as transition probability matrix, it can be proved that no matter the initialization value of $\vec{V_0}$ is, the value of $\vec{V}$ will be convergent.

By using equation (3) on Figure 7, the concepts' importance is shown in table I, where every concept's initialization importance is regarded as 1.

TABLE I
THE IMPORTANCE OF CONCEPTS IN FIGURE 7.

Concepts	Importance	Concepts	Importance
Hamlet	4.38	Ophelia	1.33
Claudius	2.60	Horatio	0.76
King Hamlet	0.13	Rosencrantz	0.57
Gertrude	1.21	Guildenstern	0.57
Polonius	1.46	Fortinbras	0.57
Laertes	1.46		

V. SIGNIFICANCE OF SENTENCE

Different sentences of an article have different significance for different users. The picked concepts represent a user's interested content. Therefore, different SLN-DSs bring different significance to a sentence within an article.

For a given article and its SLN-DS, the article's sentences are separated into two parts: sentences in SLN-DS and sentences outside SLN-DS. Therefore, there are two kinds of sentence significances. In each part, two sentences are compared according to their significances. The sentences in SLN-DS are always more significant than sentences outside SLN-DS because of having concepts.

1. *Significance of sentences outside SLN-DS*

The smaller the distance between a sentence outside SLN-DS and the SLN-DS is, the bigger the significance of the sentence is. The distance between a sentence k outside SLN-DS and the SLN-DS can be represented by the distance between k and the closest sentence of k in SLN-DS. Therefore, the significance of k outside SLN-DS can be measured as follows:

$$Sig_k = \frac{1}{Min(D_{ik})} \quad (i=1, 2, 3\ldots) \quad (4)$$

Sig_k denotes the significance of k. D_{ik} denotes the distance between the ith sentence in SLN-DS and k. Function $Min()$ returns the minimal element of the input.

In equation (4), the distance between the ith sentence and k is as follows:

$$D_{ik} = \frac{L_{ik}+1}{N(S_i \cap S)} \quad (5)$$

D_{ik} denotes the distance between the ith sentence and k. L_{ik} denotes the number of sentences between the two sentences in the article. S_i and S denote the concept set of the ith sentence and k. Function $N()$ gets the number of a set's elements.

2. *Significance of sentences in SLN-DS*

The significance of a sentence k in SLN-DS is influenced by the following elements:

1. The sum of concepts in k.
   It reflects the information content of k.
2. The locations of k in different dependence chains.
   If a sentence is the first three sentences of a concept's dependence chain, it probably defines the concept.

Therefore, the significance of sentences in SLN-DS is measured as follows:

$$Sig_k = \alpha_1 c_1 + \alpha_2 c_2 + \ldots + \alpha_n c_n \quad (6)$$

Sig_k denotes the significance of k. n denotes the number of concepts in k. c_i denotes the ith concept's importance in k. α_i equals to 10 if k is the first three sentences in c_i's dependence chain, else α_i equals to 1.

VI. FUNCTIONS OF SEMANTIC LENS

A. Function based on SLN-DS

Background generation. *The function generates the background of an input which consists of concepts or sentences*.

People may forget concepts' meaning while reading, especially while reading complex scientific papers. This function helps readers look back concepts or sentences' meaning. The simple way of looking back a concept is to find all sentences indicate the concept. The simple way usually does not satisfy the user's need. The function not only finds sentences that indicate the given concept, but also finds the background of the concept. Take figure 4 for example, to find the background of c_5—"interaction" in sentence 7, the semantic lens finds sentence 6, to comprehend sentence 6, the semantic lens further finds sentence 5 and sentence 4 because sentence 6 has c_2 and c_4, and so on. Splitting-up operation is for each element of input and the return results of splitting-up operation form an integrated background by the merge operation. If a user wants to look back a sentence's meaning, the semantic lens will look back all concepts within the sentence.

Comprehension text generation. *For a given article and a background, the function selects all comprehensible sentences based on the background.*

Based on different backgrounds, readers may comprehend different contents in an article. This function helps users to find the influence of a background or compare two backgrounds' influence. Splitting-down operation will be called for each given sentence, and the return results of the splitting-down operation forms a whole text according to the order of the original article's sentences.

Focus. *For a given article and a facet of the article which is represented as a group of selected concepts, the function picks up sub-content from the article which describes the facet.*

Users sometimes do not want to read the full text of an article, but only several interested facets. Then, they can focus on one facet and rotate the multiple facets to focus on another facet with the *focus* operation as shown in figure 8.

For a given article and a facet, each concept in the facet corresponds to a dependence chain. All dependence chains generated from the facet form a subgraph of the SLN-DS by the merge operation. The facet's content includes all the sentences in the subgraph of the SLN-DS. It is called the basic content of the facet. Sometimes an user may not satisfy the content or want to further get brief content. Therefore, the function provides operations for users to extend or reduces the basic content by setting thresholds on the significance of sentences.

If a user want to extend a facet's basic content, a threshold of significance on sentences outside SLN-DS—T_{out} is set. All sentences outside the SLN-DS with significance bigger than T_{out} will be added to the content. If a user wants to reduce the basic content, a threshold of significance on a sentence in SLN-DS—T_{in} is set and sentences in SLN-DS with significance lower than T_{in} will be removed from the content.

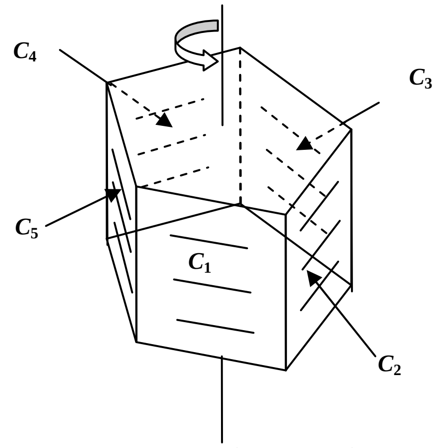


Figure 8. An article has several facets. Each facet is represented by a concept set. The number of facets depends on the number of facets picked out by users.

Semantic lens gives users a controller to adjust T_{out} and T_{in} as shown in Figure 9. Different facets correspond to different T_{out} and T_{in}.

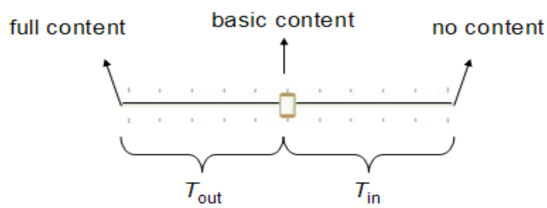


Figure 9. The controller for adjusting T_{out} and T_{in}. We assume that v_1 is the maximum significance of sentences outside SLN-DS, v_2 is the minimum significance of sentences in SLN-DS, v_3 is the maximum significance of sentences in SLN-DS. T_{out} corresponds to the left part of the controller. If we slide the button from leftmost to middle, the threshold of significance is from 0 to v_1 and the selected content in the window is from full text to the basic content. T_{in} corresponds to the right part of the controller. If we slide the button from middle to rightmost, the threshold of significance is from v_2 to v_3+1 and the selected content in the window is from the basic content to nothing. The middle point of the controller can denotes v_1 or v_2 because there is no sentence whose significance is between v_1 and v_2.

Facets' percentage graph. *This function shows one or several facets' percentage graph.*

In an article, a facet may be emphasized in an area of text, and be briefly described in another area. A facet's percentage graph shows the percentage change process of the facet. A certain area of text is called "*a window*". The percentage of a facet in a window is $\frac{\text{Number of facet's sentences in window}}{\text{Window size}} \times 100\%$. A facet's sentences are from the basic content of the facet and can be extended or reduced. When sliding a window, some facets' sentences will increase or decrease.

We choose text from the plot of http://en.wikipedia.org/wiki/Romeo_%2B_Juliet. Two facets are picked out, one is facet *Romeo* which includes concept: *Romeo*, *Balthasar*, *Mercutio*, *Tybalt*, *Benvolio*; and, the other is facet *Juliet* which includes concept: *Ted*, *Fulgencio*, *Gloria*, *Juliet*, *Lawrence*, *Paris*. Facets' sentences are from the basic content. Figure 10 and Figure 11 are two percentage graphs.

In Figure 10, the writer increasingly emphasizes facet *Romeo* until about 30th window. From about 30th window to 45th window, the content of facet *Romeo* becomes brief and it is a bit emphasized at the end.

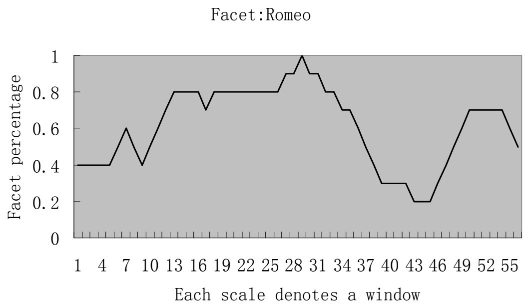


Figure 10. The *Romeo*'s percentage graph. The window size includes 10 sentences. There are totally 55 windows. Each window corresponds to a percentage value.

Several facets' percentage graphs can be set in one graph to be compared. Users may get some information from the comparison which is shown in Figure 11. We can see that from the beginning to about 20th window, facets *Romeo* and *Juliet* are described with almost the same percentage because the part of the article tells the whole background of *Romeo*, *Juliet* and their family. Facet *Romeo*'s percentage increases and facet *Juliet*'s percentage decreases from about 20th window to 36th window because the part mostly tells *Romeo* and his family. Facet *Juliet*'s percentage increases and facet *Romeo*'s percentage decreases from about 37th window to 50th window because the part tells *Juliet* and her family. At the end, the two facets have almost the same percentage again because the part tells the plot on both *Romeo* and *Juliet*.

A controller is given to users to adjust a facet's sentence area like Figure 9. Different thresholds will bring different percentage graphs to a facet.

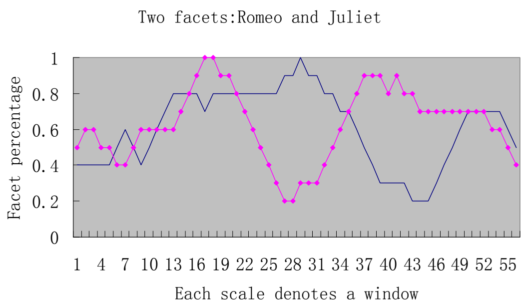


Figure 11. The *Romeo* and *Juliet*'s percentage graphs. The window size includes 10 sentences. The red function is on facet *Juliet*, and the other one is on facet *Romeo*.

Full text zoom. *For a given article, the function extracts important sentences to help a user quickly grasp the content.*

The function sets all concepts as a facet and calls function *focus* to focus on the facet. Then, the unimportant sentences will be removed. Users can adjust thresholds T_{out} *and* T_{in} to choose suitable result.

B. Functions based on SLN-C

Find communities of concepts. *This function helps users find communities of concepts.*

Finding communities of concepts is usually needed for further comprehension. The Semantic Lens can automatically find communities by using *GN* algorithm [8], where an edge with the highest betweenness will be removed. An edge's betweenness is the number of the shortest paths cross the edge. The edges of networks in GN algorithm have no weight, while the SLN-C's edges have weights. So, we redefine the concept of the shortest path in SLN-C. Assuming a path *P* with *n* edges, the weights of *P*'s edges are w_1, w_2,…, and w_n. The length of *P* (L_P) is defined as follows:

$$L_P = \frac{1}{w_1} + \frac{1}{w_2} + \ldots + \frac{1}{w_n} \tag{7}$$

A path is shorter than another path because its length is smaller.

Figure 12 is the 1-step SLN-C of text from the plot of http://en.wikipedia.org/wiki/Romeo_%2B_Juliet. Roles in the article are picked out to be concepts.

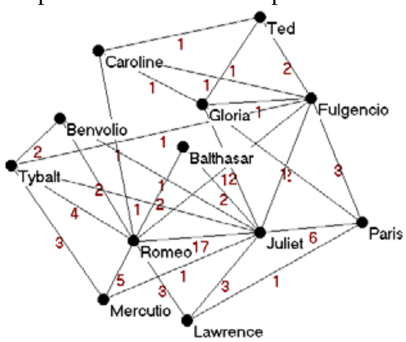
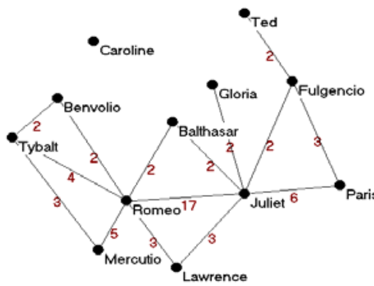
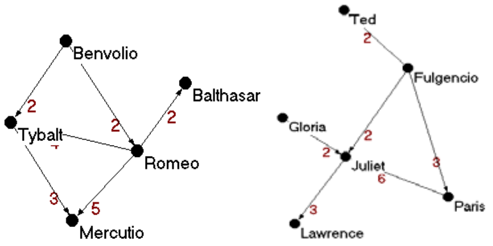


Figure 12. The 1-step SLN-C of plot of "*Romeo* + *Juliet*". Each node is a concept. The numbers are the weight of 1-step relations.

The tightness of two concepts is low if their edges' importance values equal to 1. It is a noise of finding communities. Therefore, this function will remove those edges first. The process of finding communities is shown in figure 13.



(a)



(b) (c)

Figure 13. (a) is the figure after removing edges with the importance value equal to 1 from figure 12. The function automatically picks out two communities from the SLN-C which are (b) and (c). (b) represents people close to *Romeo*. (c) represents people close to *Juliet*.

Users can also define needed communities by selecting concepts and edges from SLN-C.

Concepts' importance comparison. *This function helps users compare the importance between groups of concepts.*

A group of concepts' whole importance is the sum of each concept's importance. It can be useful for readers to judge which parts of concepts are more important. It is more useful for a writer to balance different aspects of an article. For instance, a writer may want to avoid focusing on one aspect too much especially in a multi-thread story. The writer can pick out several facets and check the facets' whole importance which will update synchronously. If a facet's importance is too high, the writer should consider whether to keep writing the facet or not. A writer can also adjust the ratio of unimportance part to emphasize the key points by the function.

VII. INTEGRATING WITH SLN OF ARTICLES

Establishing SLNs of articles can represent relations between articles. Integrating SLN of articles and the SLN of sentences has the following advantages:

- An article can be understood from the origin of the topic and previous works on the topic by tracing the explicit or implicit citation link between articles.
- Communities in the SLN of articles provide richer ground for people to understand and raise the efficiency of understanding.
- The SLN of articles can extend the scope of knowledge of an article's reader, and can inspire thinking by given rich semantic links within the area or cross-areas.

Figure 14 shows the general architecture of the textual semantic lens. Other models such as text mining and clustering methods can help establish the semantic links.

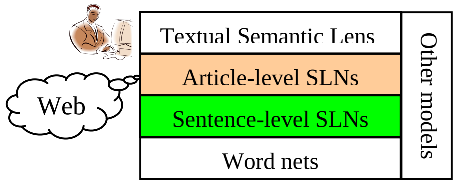


Figure 14. Integrating SLNs of different levels.

VIII. CONCLUSION

By defining SLN-DS and SLN-C, this paper proposes the textual semantic lens for helping people comprehend and analyse articles. It not only provides a new way to efficiently read articles but also an approach to represent dynamic informal knowledge in text. Integrating with the semantic images of articles, the semantic lens can help people efficiently obtain knowledge from large-scale articles. It is very useful in Web 2.0 where readers and writers can give concepts (tags) on articles according to their own opinions.

ACKNOWLEDGEMENT

This work was supported by National Science Foundation of China (61075074), and Chinese Academy of Sciences Visiting Scholarship for Senior International Scientists.

REFERENCES

[1] C. F. Baker, C. J. Fillmore, and J. B. Lowe, "The Berkeley FrameNet Project," *COLING/ACL'98*, pp. 86-90, 1998.

[2] A. Collins, M. Quillian, "Retrieval Time from Semantic Memory," *Journal of Verbal Learning and Verbal Behavior*, 8(2):240-248, 1969.

[3] H. P. Luhn, "The automatic Creation of Literature Abstracts," *IBM Journal of Research and Development*, 2(2):159-165, 1958.

[4] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross and K. J. Miller, "Introduction to wordnet: An On-line Lexical Database," *Journal of Lexicography*, 3(4):235-244, 1990.

[5] H. Zhuge, Knowledge Grid, World Scientific, 2004 (Chapter 2. Semantic Link Network).

[6] H. Zhuge, "Interactive semantics," *Artificial Intelligence*, 174(2):190-204, 2010.

[7] C. Fellbaum, editor, WordNet: An Electronic Lexical Database. The MIT Press, Cambridge, MA, 1998.

[8] M. Girvan, M. E. J. Newman, "Community structure in social and biological network," *Proc. Natl. Acad. Sci*. 99:7821-7826, 2001.

[9] W.Dakka, P.G.Ipeirotis, and K.R.Wood, "Automatic construction of multifaceted browsing interfaces," in *CIKM'05*, 2005.

[10] KP. Yee, K. Swearingen, K. Li, and M. Hearst, "Faceted metadata for image search and browsing," in *CHI'03*, 2003.

[11] E. Oren, R. Delbru, S. Decker, "Extending faceted navigation for RDF data," *ISWC'06*, 2006.

[12] H. Zhuge, "Semantic linking through spaces for cyber-physical-socio intelligence: A methodology," *Artificial Intelligence*, 175:988-1019, 2001.

[13] H. Zhuge, "Communities and Emerging Semantics in Semantic Link Network: Discovery and Learning," *IEEE Transactions on Knowledge and Data Engineering*, 21(6): 785-799, 2009.