Fusion of Simplified Entity Networks from Unstructured Text

Chan Ho Keong
DSO National Laboratories,
20 Science Park Drive,
Singapore 118230
chokeong@dso.org.sg

Zhang Xing Hu
DSO National Laboratories,
20 Science Park Drive,
Singapore 118230
zxinghu@dso.org.sg

Lee Hian Beng
DSO National Laboratories,
20 Science Park Drive,
Singapore 118230
lhianben@dso.org.sg

Abstract - To explore the fusion of unstructured text, the concept of entity networks has been proposed in many systems to facilitate analysis of entity relationship. Using appropriate visualization, entity networks could illustrate the relations of extracted entities. In this paper, we shall present an approach for combining multiple entity networks. The entity networks are derived from documents and hence, it is also a scheme for document-to-document fusion. Using an entity and relation extractor, an entity network could be built. However, such a system typically generates many entities and links from a single document, and some of them are trivial. With complicated entity networks, it is difficult and less effective to perform fusion. As such, a simplification algorithm is introduced to construct a simplified version for each entity network. Incrementally, by using our fusion algorithm, the entity network could be updated by adding a new network. A combined entity network is constructed, in which one could get the key concepts and their relationship. It is a summary of the multiple documents. The examples in this paper demonstrate the purport and effectiveness of the simplification and fusion scheme.

Keywords: entity network, entity relationship analysis, unstructured text data fusion, summarization, web mining.

1 Introduction

Increasing volumes of electronic text have given rise to a need for new methods of collecting, organizing, storing and presenting unstructured text data. Entity extraction is a method introduced to extract information from unstructured text documents. Entity network, an end state of the entity extraction process, is a graphical representation of an unstructured document. It illustrates the relations of the extracted entities. Using appropriate visualization tools, all the extracted entities and their relationship are presented using nodes and links, where entities are jointed to form an entity-relationship diagram, which is an entity network.

In year 2004, DSO National Laboratories developed a NLP Toolkit for entity extraction [4]. It is a set of software tools that efficiently analyzes large amount of unstructured text, identifies entities and their relationships. The generated entity network includes nodes of different types such as Person, Organization, Location and Date and links between nodes. They are constructed from sentences in the document and there could be many links to the same nodes extracted from different sentences. Even for a short document, there are many nodes and links among them. It is not easy to extract meaningful and concise attributes as well as entity relationship. The entity network gets even more complicated for multiple documents.

To resolve this issue, we developed a scheme to simplify an entity network [2]. This simplification scheme defines an approach to simplify and summarize an entity network constructed from unstructured text. By doing so, concise relations are extracted; the trivial and redundant relationships are pruned from the network.

Moving forward, there is a need for a method to fuse multiple simplified entity networks, which is the intent of this paper. Based on this inspiration that a simplified network can summaries the key idea of one document, we continued and proposed an approach to combine multiple simplified entity networks incrementally. By doing so, the basic concepts behind multiple documents could be captured and visualized as a fused entity networks.

Figure 1: Motivation of combining entity networks
This paper is organized as follows. In Section 2, a scheme to convert an entity network into a much simplified version is explained. In Section 3, we will discuss an approach for combining a pair of simplified entity network, on the same topic. And in Section 4, we demonstrate and analyze the effectiveness of the simplification and fusion scheme based on test results. Finally, we conclude the paper.

2 Simplification of Entity Network

An entity extraction system generates many nodes and links for each document, and some of them are trivial. As such, it is difficult to extract the key points of the text. Also, it is less effective to combine entity networks with too many trivial nodes and links. As such, simplification of entity network is necessary prior to entity network fusion.

There are two main tasks in the network simplification process. One is to segment an entity network into a number of localized smaller entity networks, with each segment representing a key event. Another is to trim an entire entity network by maintaining the primary nodes and skeletal structure. With a simplified network, an overall structure representing a document is defined, key points and their relations are also surfaced. An entity graph like this brings out the intents of the documents without the needs to read the document.

Before we proceed, it is important to understand the fundamental concepts on how a simplified entity network is derived. In this section, the mathematical concepts and algorithms will be briefly described. This include shortest-path magnitude (SPM) algorithm, centrality score (CS) degree, and shortest-path tracing (SPT) algorithm. The details of these formulations are described in [2]. At the end of this section, a procedure on network simplification will be explained.

2.1 Shortest-path magnitude algorithm

The magnitudes of shortest paths are the basis for definition of centrality score, algorithm for shortest-path tracing, etc. We introduced a basic and efficient algorithm, called the shortest-path magnitude (SPM) algorithm. It is designed for fast computing the magnitudes of shortest paths between all pairs of nodes. Using the shortest distance algorithm, we obtained (1) a shortest distance matrix \( P \) that represents the magnitudes of all shortest paths in the network; (2) a count matrix \( P_{cnt} \) that represents the number of shortest paths exist between any pair of nodes accordingly. For example, \( P(2,9)=4 \) and \( P_{cnt}(2,9)=1 \), that means there is one shortest path between node-2 to node-9 with length 4.

2.2 Centrality score (CS) method

Given an entity network \( N \), by using the above SPM algorithm, we could obtain the shortest distance matrix, \( P \), of it. Each row vector in matrix \( P \) represents the index of the shortest distances from one particular node to all other nodes. If a node has a high frequency of distance 1 in the vector, this means it has a large number of neighboring nodes.

Generally, if a node is located in a (local) central of a network, then it must have large number of neighboring nodes. Hence, we could define the Centrality Score (CS) for each node (entity) by summing the distance value with more weight assigned to the shorter distance and less weight to the larger distance.

2.3 Shortest-path tracing (SPT) algorithm

Given an entity network \( N \), by using the above SPM algorithm, we could only obtain the shortest distance matrix, \( P \), of it. However, we still do not know the actual shortest path from one node to another. To attain this path, we developed a shortest-path tracing (SPT) algorithm to trace a specific shortest path from any one node to another.

2.4 Primary node determination

The main purpose of a set of primary nodes is to choose a set of prominent nodes to serve as the initial centroids of all segments. This is determined by the two features of CS degrees, as well as the distribution of entity networks.

To obtain a set of primary nodes, first we find the node with the largest CS degree, and put it into a primary node list; then continue to search for a node with the 2nd largest CS degree, with a constraint to avoid selecting a node that is close to the earlier primary nodes; the procedure continued for the remaining nodes, until CS degrees for the next selected nodes is less than a given threshold. The selection process will also terminate when the predefined number of primary nodes are met.

2.5 Simplification scheme for entity networks

There are two objectives for simplification of entity network. One objective is to cluster an entity network into different segments, with each segment expressing an event. Another one is to trim an entity network into a simplified network, which presents a basic structural topology by just maintaining the primary nodes and their links.

The procedure of simplification scheme for entity networks are briefly described as follow:

Step 1: Computing the shortest distance matrix \( P \)
Using the SPM algorithm, obtain the shortest distance matrix \( P \) and its count matrix \( P_{cnt} \).

Step 2: Obtaining the CS degree for all nodes
Based on the matrix \( P \) and proper weightage assignment, compute the CS degrees for all nodes.

Step 3: Selecting the primary nodes
Based on the CS degrees and the distribution of all nodes in the entity network, determine a set of primary nodes by using the method at Section 2.4

Step 4: Performing segments clustering
Each primary node is set as the initial centroid of a segment. Then, to cluster all other nodes into different segments based on the Linkage Strength, which is a score associating two adjacent nodes.

Step 5: Constructing the primary frame
For any two primary nodes, use the SPT algorithm and trace a shortest path between them. Going through all the pairs resultantly link all the primary nodes to form a simplified entity network, which is termed as the primary frame of the entity network.

Step 6: Forming the structural topology
Identify the longest shortest paths (termed as trunk paths for short) by referring the distance matrix P, and trace the tracks of selected paths by using the SPT algorithm. The traced path tracks sketch the structural topology of the entity network for its frame and connection.

Step 7: Creating a simplified entity network
Lastly, combine the primary frame and structural topology of the entity network together, we obtained a simplified entity network with less nodes and connections, but maintaining the basic purport and general structural topology.

3 Fusion of Simplified Entity Networks

Fusion of entity networks requires the following important steps: correlation for a pair of entity networks, merging of the entity networks, determination of CS degrees of a fused entity network, as well as simplification of a fused entity networks. In the following subsections, we will discuss these issues accordingly.

3.1 Correlation for a pair of entity networks

In this research, an assumption we used for combining a pair of simplified entity networks is that both the entity networks should be about the same topic. Documents addressing different topics should not be combined since it is likely to result in two networks.

Given a set of documents, the process will filter the unrelated documents and merge only related documents into a single entity networks incrementally which provides the gist of these documents without the need for the reader to go through them. In order to achieve this, the algorithm compares all the nodes between the two entity networks and determines a similarity measure of the common nodes. The steps include (1) lowering the case and word extraction, (2) finding lemma of each word, (3) matching the words, (4) compute the similarity score base on number of similar words over number of total words. A pair of nodes is considering similar if the similarity score is higher than a predefined threshold. Upon completion, a list of common nodes is identified.

Next, the system proceeds to compute the correlation for the pair of entity networks. The correlation degree is determined by calculating the total centrality score for the common nodes over the total centrality score for all the nodes in the two entity networks. Suppose there are two entity networks (EN1, EN2), defined as

\[
\text{EN1} = \{ e_1^1 \} \\
\text{EN2} = \{ e_2^1 \}
\]

then the correlation degree of EN1 and EN2 is defined as

\[
\text{Corr}(\text{EN}_1, \text{EN}_2) = \frac{\sum_i \text{CS}(e_i^1) \text{CS}(e_i^2)}{\left(\sum_i \text{CS}(e_i^1)^2\right)\left(\sum_j \text{CS}(e_j^2)^2\right)^{1/2}}
\]

where

\[
\text{CS}(e_i^1) \text{ is the centrality score of common node } e_i^1 \text{ in } \text{EN}_1 \\
\text{CS}(e_i^2) \text{ is the centrality score of common node } e_i^2 \text{ in } \text{EN}_2 \\
\text{CS}(e_j^1) \text{ is the centrality score of a node } e_j^1 \text{ in } \text{EN}_1 \\
\text{CS}(e_j^2) \text{ is the centrality score of a node } e_j^2 \text{ in } \text{EN}_2
\]

The fusion of two networks will happen only if the correlation score between the pair of entity networks is above a user defined threshold.

3.2 Merging of simplified entity networks

The strategy for combining a pair of entity network is straightforward. Upon verifying that the two entity networks are of the same topic by computing the correlation score of the two networks, the system extracts all the nodes and links from the two entity networks, merges each pair of common nodes into a single node and updates its CS degree (refer to section 3.3). Consider the entity networks of first document as the base entity network, common nodes from the second entity network are added onto it. Subsequently, append the remaining nodes and links into the seed entity network. After fusing the two networks, primary nodes are determined again and another round of entity network simplification is required. The fused entity network becomes the base entity network for further update when a new document is added. See figure 2.

Suppose there are two entity networks, defined as

\[
\text{EN}_1 = \{ e_1^1 \} \\
\text{EN}_2 = \{ e_2^2 \}
\]
then the merging of $EN_1$ and $EN_2$ is $EN = \{ e_k \}$, which is the collection of $EN_1$ and $EN_2$, and entity nodes $e_{k1}$ and $e_{k2}$ in $EN$ are linked if and only if $e_{k1}$ and $e_{k2}$ both exist in $EN_1$ and linked in $EN_1$, or both exist in $EN_2$ and linked in $EN_2$.

![Figure 2: Integration process for a pair of Entity Networks](image)

### 3.3 Centrality Score for an fused entity network

The Centrality Score of a node, $e_k$, in the fused entity network $EN$ is updated as

$$CS(e_k) = \alpha \cdot CS_1(e_k) \oplus (1-\alpha) \cdot CS_2(e_k)$$

where

- $CS_1(.)$ is the centrality score in $EN_1$
- $CS_2(.)$ is the centrality score in $EN_2$

When $e_k$ is only in $EN_1$, but not in $EN_2$, $CS_2(e_k)$ is 0. In practice, the $\alpha$ is chosen as $(t-1)/t$ when $1 \leq t \leq 10$ (first 10 steps); the $\alpha$ is chosen as a constant value 9/10 for $t > 10$.

### 3.4 Simplification of fused entity network

To simplify a fused entity network, we need to go through a similar process: determining a set of primary nodes, forming a minimum network skeleton that links all the primary nodes as given by the simplification steps described in section 2.5.

In order to enhance the stability of primary nodes and the structure of the network across the steps, we introduced two measures in the process of re-defining the set of primary nodes. (1) If $e_i$ is a primary node in base network or current network, then in the comparison of centrality scores in the combined entity network, we use the higher values $CS(e_i)^{1.5}$, instead of original values, to determine the new primary entities. (2) Similarly, if $e_i$ is a primary node in base (previous) network and current network, then in the comparison of centrality scores in the combined entity network, we use the higher values $CS(e_i)^{1.2}$ instead of original values, to determine the new primary entities. In effect, the algorithms prefer existing primary nodes for a more stable fused network.

### 4 Test results and analysis

In this section, an experimental result will be presented to demonstrate the approach for fusing the entity networks. First, two documents about the same topic of the bird flu outbreak are selected.

The first document talks about “Bird flu outbreak in Southern Vietnam”. Using the NLP entity extraction engine, an entity network is generated. This network then goes through the network simplification process and output the result as shown in figure 3. Primary nodes are the keywords of the document and are red in color. Each primary node has several supporting nodes of the same color. They form one grouping (or segment) that is related to the primary node.

![Figure 3: Simplified network for document on “Bird flu outbreaks in southern Vietnam”](image)

With the simplified entity network, we could grasp several key points from the news report about Bird flu outbreak in southern Vietnam. Through observing the primary nodes, it highlights the key concepts of the document on (1) bird flu outbreak, (2) H5N1 virus, and (3) animal health authorities. By linking the remaining nodes, the key messages of the document are:

1. The bird flu killing the ducks from the farm owner
2. There are hundreds of millions of unvaccinated birds
(3) Officials warning farmers of more outbreaks due to the weather
(4) Agriculture minister urging the authorities to act
(5) World Health Organisation reported 330 known cases
(6) H5N1 virus killed people in the southern Vietnam

The second document is about “Bird flu outbreak in China Guangzhou”. Similarly, to generate the simplified network, this document need to go through the same process as mentioned above. In this document, it has been observed that the system generates two entity networks as shown in Figure 4 and Figure 5.

Figure 4: Simplified network 1 for document on “Bird flu outbreaks in Guangzhou China”

Figure 5: Simplified network 2 for document on “Bird flu outbreaks in Guangzhou China”

Nevertheless, the basic ideas of the document are reflected in the primary nodes: (1) China, (2) Guangzhou, (3) bird flu outbreak, (4) authorities and (5) ducks. The key messages are Guangzhou of China destroys 150,000 poultry, extermination of ducks in the bird flu outbreak, three kilometers from the site of outbreak that affecting nine villages, the worry of the authorities, etc.

After simplifying the second documents, the system computes the correlation score to determine whether to merge with the first entity network. If the score is above a predefined threshold, the system merges the two entity networks as shown in Figure 6.

To evaluate the effectiveness of the scheme for fusing a pair of entity network, we will need to analyze the fused entity network based on its primary nodes and the supporting nodes.

During an integration process, primary nodes with keywords such as “bird flu outbreak” and “authorities” are combined. These words exist in both documents, thus they have a higher CS degree. As such, both of these key words will be combined as a single node and remain as the primary nodes for the fused network. Beside this, keyword such as “China”, which has a very high frequency of appearance in the second document, has a very high CS degree. Hence, it also remains as the primary node in the fused network. Looking at the three primary nodes in the fused network, the gist of the two documents is captured.

Figure 6: Fusion of two documents about bird flu

For the non primary nodes, it can be seen that some nodes are discarded. The remaining nodes of the fused network contain the key messages (the main ideas) of the two documents. To analyze this in details,

(1) Observe that the nodes from the first document maintain in the fused network. Comparing the fused network against the entity network from the first document, we are able to identify similar messages such as hundreds of millions of unvaccinated birds, officials warning farmer of more outbreaks due to the weather, Agriculture Minister urging the authorities to act.

(2) Next, we want to determine the content of the second document that maintains in the fused network. To make it simple, we take the outcome from document 1 as the base entity work. Nodes from the second document are either merged or appended to the existing nodes of the base entity network. The
combined nodes are “local authorities”, “bird flu outbreak”, “9,830 ducks”, “134,384 ducks” and “150,000 poultry”. While, the appended nodes are “a subtype of H5N1 bird flu strain”, “china” and “the Panyu district government”. From this result, the messages enforced by the second document are the kills of ducks due to the bird flu outbreak, there are response from local authorities. With added messages such as a subtype H5N1 bird flu strain in China, which involved the Panyu district government.

From the above analysis, we have seen that the fused entity network preserves the gist of the two documents and it provides a summary of the two documents. It is noteworthy that the fusion combines the two entity networks generated from the second document (shown in figure 4 and 5).

And last but not least, the approach can be used for more than two documents. Entity network derived from new document can be combined with the base entity network incrementally. Continued from the above output, two more documents on the topics of bird flu are injected and fused incrementally. The fused entity network is presented in Figure 7.

Observing the displayed result, it highlighted (1) Bird flu outbreak, (2) the authorities and (3) the efforts. The first two primary nodes are the basic concepts from the earlier documents. And the third primary node is new concepts presented. The new message is UNICEF field officer said the efforts of the people on Avian Influenza Control in Panda Development Area of Nasarawa State. Other new messages include the CFIA oversee the cleaning, the CFIA/the authorities to compensate the producers, to limit any potential virus spread with restriction on poultry product within three kilometers of the infected premises. These are the valid key points and messages from the two new documents, and have been correctly identified by the system.

To validate the effectiveness, more experiments are conducted with larger documents size using different configuration setting, such as the percentage of primary nodes, the maximum and minimum number of nodes in a network, etc. In one experiment, four entity networks on the topic of ‘international relief work after the Tsunami disaster’ are generated and then fused incrementally. After four rounds of fusion processes, the summarized

![Figure 7: Fusion of four documents about bird flu](image)

![Figure 8: A fused entity network from four articles on Tsunami.](image)
result is presented in Figure 8. Once again, we can see that the fused entity network is concise and readable. The list of primary nodes maintained by the system includes: (1) tsunami, (2) Indonesia/Sri Lanka, (3) emergency aid, (4) food and water, (5) stranded survivors. All are relevant and correctly represent the main ideas of the fused documents. Through analyzing the fused entity network, reader should be able to grasp the general concepts without the need to read all the four documents.

5 Conclusion

In this paper, an approach to fuse simplified entity networks has been presented. Through such simplification and incremental fusion of multiple documents, it reduces the complexity of the fused entity network. The combined network is concise and easy to understand. A summary of the multiple documents is thus provided. Using several examples, this paper demonstrates the effectiveness of our fusion scheme in merging multiple entity networks of similar topic into a simplified network through a graph to graph matching approach. At the same time, the key concepts of the set of documents are preserved.

References


