Visualizing context words of entity pairs from corpus

Md. Rashadul Hasan Rakib, B00598853, Winter 2013

Abstract: The goal of our system is to visualize significant context words within the entity pairs from a corpus. Context words play an imperative role to obtain associations between the entities. The input of our system is a query entity and a corpus; then it finds other entities that are associated with the given entity and visualize the significant context words between that given entity and the associated entities. A radial visualization has been developed that consists of a circle at the center position which is surrounded by several arcs. The circle represents the query entity (e.g., Barack Obama) given by the user and the arcs represent the associated entities (e.g., George Bush and U.S.A) that are extracted using the query entity. Each arc consists of significant context words (e.g., associations) between the query entity and an associated entity. The query entity and associated entities form entity pairs such as Barack Obama-George Bush and Barack Obama-U.S.A. We evaluate our proposed visualization with previous systems and give an idea how our system can be improved in future.

1 Introduction: Nowadays corpus becomes a valuable resource to analyze data. It is not a trivial job for the users to analyze a corpus manually in order to obtain the desired information. This is because the data mining tasks are becoming popular day by day. Moreover the visualization of the extracted data in a meaningful way is more important. In this connection, we have developed a visualization system by which a user can easily find the associations between the query entity and associated entities from a corpus. When the user clicks on an arc that represents an associated entity, a new radial shape will be created automatically and explore more entities and relations. Each radial contour is considered as a node and there is an edge between two radial contours. Nodes and edges are created by Force-Directed Graph algorithm (Di et al., 1999) which is already implemented in Data-Driven Documents (D3).

Generally, the words that appear between two entities in a document are called context words and the discernible objects in this real world are taken into account as entities. The context words between two entities collectively form a context. Table 1 shows entity pairs, contexts and associations between the entities. Our aim is to not only determine the associations through analyzing the context words of two entities but also visualize them in an evocative way in order to let people make sense how the entities are related with each other.

Table 1: Entity pairs and contexts

	Query Entity	Context	Associated Entity	Association
Entity pair1	Barack Obama	is the president of	U.S.A	president
		is the citizen of		citizen
Entity pair2	Barack Obama	is the opponent of	George Bush	opponent

New York Times (NTY) corpus is used as our dataset that consists of news article over ten years since 1991 to 2000. Our system acts as an entity-context search engine that uses the Stanford Named Entity Tagger (Finkel and Manning, 2009) to extract the entities and contexts from the corpus. Then it performs different levels of data processing and visualizes the processed data.

We want to summarize our contribution in the following ways:

i. Given a query entity and a corpus.
ii. Extract the associated entities and context inside an entity pair.
iii. Recognize the associations between the entities from context data.
iv. Visualize the associations in the radial layouts.

All the above steps will be discussed in detail throughout the rest of this report. In addition, we give an overview for the further advancement of our system.

2 Related work: A news article visualizing system called Contexter (Grobelnik and Mladenic, 2004) that allows a user to select an entity from an entity-list obtained from news articles. If the user selects an entity then it shows a list of associated entities that appear together with the selected entity in some documents. It permits user to click on an entity in the associated entity-list to visualize the context within the news articles where the selected and associated entity appear. Jigsaw (Stasko et al., 2008), a visual analytic system that represents the documents and their entities visually so as to help the analysts to investigate criminal activities and relations between them. Nevertheless, these visualizations are not very interactive. The relationship miner (Lohmann et al., 2010) mines relations and entities that are related to the given entities using RDF (RDF, 2004) data of DBpedia (Christian et al., 2009); then visualizes them as a graph. If a user clicks on an entity, it highlights the paths between given entities via the clicked entity. However, it does not show entities as groups. FP-Viz (Keim et al., 2005), a visualization system of frequent itemsets, which is similar with our system in terms of radial shape. It explores data within the same layout where as the proposed visualization explores data using more radial layouts and it is more interactive in terms of dragging facilities of the nodes (e.g., radial layouts) within any place of the user interface.

Lots of works have been done to find associations between the entities using context. There is a fully automated system (Merhav et al., 2012) that finds single association for an entity pair from blog data. However multiple significant associations may exist for an entity pair which is not identified by this system since it considers only sentence as context where two entities appear together and clusters contexts of different entity pairs in order to assign single relation to all of them. This is because we extract contexts of an entity pair in document level given in figure 1. Then cluster contexts of the entity pair to expose more relations between two entities.

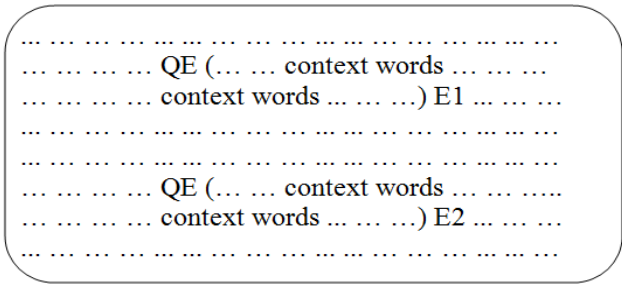

Figure 1: Context words of the entity pairs (e.g., QE-E1, QE-E2) inside a document. QE represents the query entity and E1, E2 are the associated entities of QE.

In (Jinxiu et al., 2005), context words are extracted based on five different window sizes. For example, the window size “2-5-2" means that the intervening words between an entity pair should not exceed 5 words and these intervening words together with the two words before the first entity and two words following the second entity constitute the context of an entity pair. However if the sixth word within the pair is more significant than the other words in terms of representing relation, it will be discarded. We also cluster contexts of an entity pair to obtain multiple associations within two entities. Moreover the entity pairs that have multiple relations are clustered using their contexts to visualize them as groups. Precisely, we perform two levels of clustering.

3 Proposed system: The proposed system consists of mainly two components: data preprocessing and data visualization component. Figure 2 depicts the system architecture where the extracted context data is preprocessed and stored in database and visualization of the data is provided based on the user request. These two components are considered as off-line (e.g., data preprocessing) and on-line (e.g., data visualization) components in our system.

3.1 Data preprocessing: To process data we use two external tools which are Lucene search engine (Seeley, 2007) and Stanford Named Entity Tagger (Finkel and Manning, 2009). Lucene is an open source search engine that facilitates user to search data within a corpus.

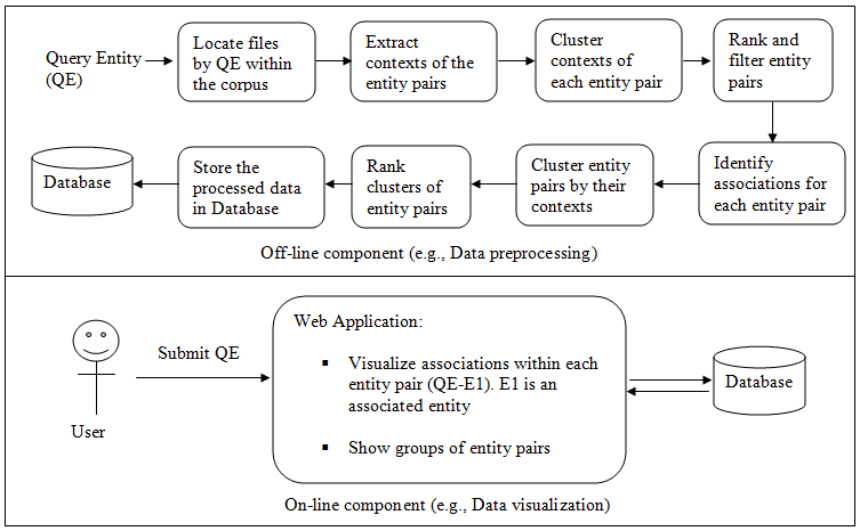

Figure 2: Proposed system architecture.

The Stanford Named Entity Tagger recognizes the entities along with their types inside a document. At this stage, we are not using the entity type as a factor of data processing; later it will be used for further enhancement of our system. The steps of data preprocessing is given in figure 2. The off-line component steps are performed for each query entity.

3.1.1 Locate files by query entity: We build index on the NYT corpus to facilitate searching for different query entities. The index has been built only once by Lucene. The off-line component takes a query entity as input; then it locates the files within the corpus where the query entity appears. We consider only the first 1000 files returned by Lucene for context processing.

3.1.2 Extract contexts of the entity pairs: The Stanford Named Entity Tagger is applied on the located files to recognize the entities. We extract contexts from each located file that consists of the query entity and an associated entity. The appearance order of the two entities is independent which implies that the query entity can become the first or second entity of the entity pair. The contexts are extracted following the context definition given in figure 1.

3.1.3 Cluster contexts of each entity pair: The stop words are removed from the contexts. We assign weight for the context words by tf-idf (term frequency-inverse document frequency) (Manning et al., 2009). Each context of an entity pair is represented as a context vector; then the Vector Space Model (VSM) (Manning et al., 2009) has been constructed for each entity pair using its context vectors. For each entity pair a context similarity matrix is prepared through computing the cosine similarity between the context vectors of that entity pair. Then we cluster contexts of each entity pair into minimum two clusters by Hierarchical Agglomerative Clustering (HAC) algorithm (Manning et al., 2009) using single link distance.

3.1.4 Rank and filter entity pairs: We obtain a lot of entity pairs for a query entity; however all of them are not interested. We calculate the average of the context similarities (avgConSim) of an entity pair and select the top t pairs according to the descending order of the average values. The value of t varies depending on the number of characters of all context words. For example: we consider entity pairs until the total number of characters of the context words are more than 220. Moreover the entity pairs that have avgConSim equal to zero or less than two contexts are ignored. We perform this filtering for convenient visualization. The higher context similarity refers to more overlap between two contexts. Every entity pair contains the query entity; so the higher avgConSim implies that the associated entity is closely related to the query entity. Suppose there are two entity pairs: first entity pair has three and second entity pair has ten contexts. In the first entity pair, the words in every context

are almost same. In the second entity pair, the words in every context are completely different from each other. Therefore it is statistically impossible to identify a particular association between the entities of second entity pair though it might have some interesting words. On the other side, we can effortlessly recognize a particular association between the entities of the first entity pair using the co-occurrence of words.

Table 2 illustrates how the average of the context similarities of an entity pair is calculated. In the following table we consider an entity pair with four contexts (C1, C2, C3 and C4). From these four contexts we get six combinations of the context pairs. For each context pair, we compute the cosine similarity and finally calculate the average of the six similarity values.

Table 2: Average context similarity of the contexts of an entity pair

Context pair		Similarity	Average of context similarity
C1	C2	Sim1	avgConSim=(Sim1+Sim2+ Sim3+ Sim4+ Sim5+ Sim6)/6
C1	C3	Sim2	
C1	C4	Sim3	
C2	C3	Sim4	
C2	C4	Sim5	
C3	C4	Sim6	

3.1.5 Identify associations of each entity pair: The contexts of each entity pair are clustered into minimum two clusters. Then we compute the centroids of the clusters. The words in the centroid (Manning et al., 2009) of each context cluster are considered as the associations for a particular entity pair.

3.1.6 Cluster entity pairs: The entity pairs in the same cluster give an intuition that they are closely related with each other. We perform two levels of clustering: context clustering in section 3.1.3 and entity pair clustering. There is an advantage of this approach. After clustering contexts of each pair, we filter the unexpected pairs. Then perform second level of clustering that clusters only the selected entity pairs using their contexts which helps us to play with small volume of data. On the contrary if we perform the reverse order of clustering; it costs more time and space than our approach. The entity pairs in a cluster or group are assigned same group number.

3.1.7 Rank clusters of entity pairs: The algorithm to rank clusters of entity pairs is illustrated in figure 3. It computes the averages of the avgConSim values for each cluster. Then sort clusters using those average values.

```
Algorithm: Rank clusters of entity pairs
//Input: Clusters of Entity pairs; Output: Sorted clusters of entity pairs
        For each cluster of the entity pairs
                Sum=0
                For each entity pair of a cluster
                        Sum=Sum + avgConSim of the entity pair
                End For
                avgSimWithinCluster = Sum/Number of entity pairs in the cluster
                Store the avgSimWithinCluster and cluster index in a list
        End For
        Sort the list in descending order based on the avgSimWithinCluster values
End Algorithm
```

Figure 3: Rank clusters of entity pairs

3.1.8 Store the processed data: To make our visualization faster we store the processed data in the MySql database (Schwartz et al., 2008). Table 4 shows the database table structure and field

description. We store data of fifteen query entities where each of them has associated entities and contexts.

3.2 Data visualization: Figure 4 shows a radial visualization which is generated by submitting a query entity from the user interface. It contains a circle at the center position surrounded by several arcs. The circle represents the query entity (e.g., Barack Obama) and the arcs represent the associations between query entity and associated entities. For example: the associations between Barack Obama and Illinois

Table 4: Database table to store processed data

Field name	Filed description
FirstEntity	Query entity (e.g., Barack Obama)
SecondEntity	Associated entity (e.g., U.S.A)
FirstEntityType	Query entity type (e.g., PERSON)
SecondEntityType	Associated entity type (e.g., LOCATION)
CommonContextWord	Centroids of the cluster of entity pairs using the contexts
ContextWord	Centroids of the cluster of contexts of each entity pair
GroupNo	Group no of an entity pair which is obtained after entity pair clustering

are Senatorial, Candidate and Democrat. The query entity and associated entities form entity pairs like Barack Obama-Bush, Barack Obama-Senate and so on. The groups having more than one entity pair show their centroids. In figure 4, the centroid Senator is shown using the contexts of the three entity pairs Barack Obama-Sharpton, Barack Obama-Chicago and Barack Obama-Illinois.

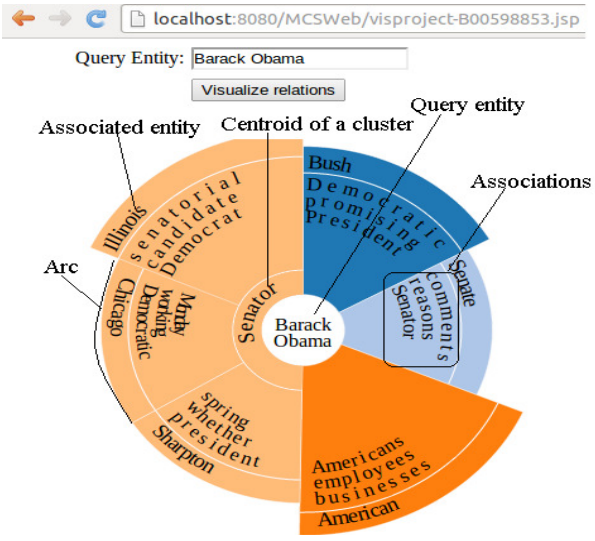

Figure 4: Radial visualization of associated entities along with the associations between the query entity and associated entities. Query entity = Barack Obama; Associated entities = (Bush, Senate, Illinois …); Associations = (Democratic, President, Senator …); Entity pairs = (Barack Obama-Bush, Barack Obama-Senate …).

3.2.1 Construction of Radial visualization: We use an open source visualization toolkit Data Driven Documents (D3) to construct the radial visualization. The radial shape consists of two types of geometric objects which are circles and arcs. The circle and arcs jointly form a node. The construction of circle, arc and node are discussed below.

3.2.2 Construction of Circle: If the query entity contains more than one word; then the words are placed in different line. The words of the query entity constitute a bounding box using D3. The radius

of circle is calculated by adding an offset (e.g., 10) with the half of the width of the bounding box. The query entity is placed at the center position of the circle.

3.2.3 Construction of Arc: The inner radius of an arc is equal to the radius of circle and the outer radius is calculated by multiplying the total number of characters of the associations with a multiplication factor for instance 5. The angle of an arc is measured using the ratio of the number of characters inside it and the total number of characters within the whole layout. The associations are aligned horizontally with respect to an arc shown in figure 5. A thin white border is drawn inside the arc to separate the associated entity and associations. The preprocessed associations are obtained from the database and placed inside an arc according to the descending order of their lengths. If the association length is greater than ten or less than three; it will be discarded. We select at best three associations for an entity pair following the above rules. The texts inside an arc are aligned anti-clock wise, if the arc is placed at the bottom part of the radial layout.

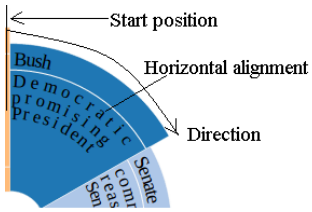

Figure 5: Horizontal alignment of context words with respect to an arc. It shows the initial position and direction of entity placement in radial layout.

3.2.4 Construction of Node: Figure 4 depicts a node which is constructed from a query entity and its associated entities using a circle and some arcs. The arcs that represent the entity pairs of same cluster are given same color and placed adjacently. The color of an arc is chosen from the D3 color range using the group number of the entity pair. If the group contains more than one entity pair; then the centroid from the contexts of those entity pairs is shown inside a small arc attached with the query entity given in figure 4. If there is more than one word in the centroid; then the word with the smallest size is picked up for visualization since all the words in centroid are common in every context of a cluster.

The initial position and direction of associated entity placement are illustrated in figure 5. We start from this position to place the associated entities in clock wise direction. Each associated entity is a part of an entity pair and each entity pair is within a particular cluster. The clusters of entity pairs are placed following the rank of clusters. The cluster ranking algorithm is illustrated in figure 3. According to the algorithm the associated entities in the top ranked cluster are placed first, then the entities of other clusters and so on. The entities in every cluster are placed according to the descending order of their individual avgConSim values. In figure 4, the lowest ranked cluster comprises three entities Sharpton, Chicago and Illionois which indicates the lowest level of association between this group of entities and Barack Obama.

3.2.5 Interaction with node: Mainly two types of user interactions are provided in our visualization system, which are entity-pair exploration and node dragging.

i. Entity-pair exploration: If the user clicks on an entity (e.g., Bush) inside an arc, then it will be considered as a new query entity. The Same kind of radial contour will be created where Bush will be placed at the center position and surrounded by other entities that are related to it. Figure 6 depicts a visualization of entity-pair exploration.

ii. Node dragging: A user can easily drag any radial node in any location of the user interface. Nodes and edges are created by Force-Directed Graph algorithm (Di et al., 1999). Force directed graph algorithm has been adopted in our visualization system to facilitate smart usability to the user.

iii. Other user interactions: If the user hovers the mouse on an edge, a tool-tip will be displayed

showing that which nodes are connected by this edge.

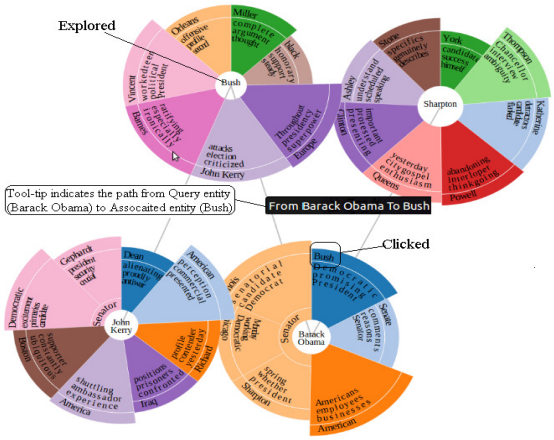

Figure 6: Entity pair exploration by clicking an entity (e.g., Bush).

4 Evaluation: To the best of our knowledge, the proposed system is a new approach to visualize associations between entities. The extracted associated entities with associations are quite meaningful. The background data preprocessing task is computationally less expensive as it filters unexpected entity pairs before clustering them. Our proposed system reveals groups among the entity pairs which is not provided by the relationship miner (Lohmann et al., 2010). Moreover the exploration of radial layout is more interactive than FP-Viz (Keim et al., 2005).

5 Limitation: The associated entities as well as associations for a particular query entity are not properly identified. It clusters contexts by Hierarchical Clustering Algorithm using only single link distance where as the average link or complete link distance can give better result. It processes only the first 1000 files for a particular query entity; so the result is not up to the mark. If the number of characters inside an arc increases it cannot handle properly. There is a reasonable size of empty space in every arc.

7 Conclusion: The proposed visualization system shows associations between the query entity and the entities that are highly related to it. The higher the similarity between the contexts of two entities stands for the higher relatedness between them. The entity pairs are clustered using their contexts to visualize them in groups. Though our system does not properly recognize the associations; it grows a mental model regarding the associations between two entities. In future, we provide the following amenities in the proposed visualization system. If an association is clicked, then the similar associations will be highlighted in other arcs. Two associations are same if they are identical or they have common root word in Word-Net (Miller et al., 1990). Identify the sentiment of associations. For example: the associations President and Opponent indicate the positive and negative relation between Barack Obama and Bush.

References:

Christian, B., Lehmannb, J., Kobilarova, G., Auerb, S., Beckera, C., Cyganiakc, R. and Hellmannb, S., 2009. DBpedia - A crystallization point for the Web of Data. Web Semantics: Science, Services and Agents on the World Wide Web, 7 (3):154–165.

D3. Data-Driven Documents. http://d3js.org/

Di, B. G., Peter, E., Tamassia, R. and Tollis, G., 1999. Graph Drawing: Algorithms for the Visualization of Graphs. Prentice Hall.

Finkel, J. R. and Manning, C. D., 2009. Nested named entity recognition. In: Proceedings of EMNLP, Singapore, p. 141-150.

Grobelnik, M. and Mladenic, D., 2004. Visualization of news articles. Informatica, 28 (4).

Jinxiu, C., Donghong, J., Lim, T. C. and Zhengyu, N., 2005. Automatic relation extraction with model order selection and discriminative label identification. In: Proceedings of the Second international joint conference on Natural Language Processing, Republic of Korea, p. 390-401.

Keim, A. D., Schneidewind, J. and Sips, M., 2005. FP-Viz: Visual Frequent Pattern Mining, in IEEE Symposium on Information Visualization. Poster, Minneapolis, MN USA.

Lohmann, S., Heim, P., Stegemann, T. and Ziegler, J., 2010. The RelFinder user interface: interactive exploration of relationships between objects of interest. In: Proceedings of the 15th international conference on Intelligent user interfaces, New York, USA, p. 421-422.

Miller, G. A., Beckwith, R., Fellbaum, C. D., Gross, D. and Miller, K., 1990. WordNet: An online Lexical database. International Journal of Lexicography. 3(1): 235-244.

Manning, C. D., Raghavan, P. and Schtze, H., 2009. An introduction to Information Retrieval. Cambridge University Press, Cambridge, England.

Merhav, Y., Mesquita, F., Barbosa, D., Yee, W. G. and Frieder, O., 2012. Extracting information networks from the blogosphere. ACM Transactions on the Web, 6(3): 11:1-11:33.

RDF, 2004. Resource Description Framework. http://www.w3.org/RDF/

Seeley. Y, 2007. Full-Text Search with Lucene. http://lucene.apache.org/core/

Schwartz, B., Zaitsev P., Tkachenko V., Zawodny, J. D., Lentz, A. and Balling, D. J., 2008. High Performance MySQL, O'Reilly Media.

Stasko, J., Grg, C. and Liu, Z., 2008. Jigsaw: supporting investigative analysis through interactive visualization. Information Visualization archive, 7 (2): 118-132.