TUBE (Text-cUBE) for Discovering Documentary Evidence of Associations among Entities

Hady W. Lauw
Nanyang Technological University
hadylauw@pmail.ntu.edu.sg

Ee-Peng Lim
Nanyang Technological University
aseplim@ntu.edu.sg

HweeHwa Pang
Singapore Management University
hhpang@smu.edu.sg

ABSTRACT

User-driven discovery of associations among entities, and documents that provide evidence for these associations, is an important search task conducted by researchers and domain information specialists. Entities here refer to real or abstract objects such as people, organizations, ideologies, etc. Associations are the inter-relationships among entities. Most current works in query-driven document retrieval and finding representative subgraphs are ill-suited for the task as they lack an awareness of entity types as well as an intuitive representation of associations. We propose the TUBE model, a text cube approach for discovering associations and documentary evidence of these associations. The model consists of a multi-dimensional view of document data, a flexible representation of multi-document summaries, and a set of operations for data manipulation. We conduct a case study on real-life data to illustrate its applicability to the above task and compare it with the non-TUBE approach.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; H.1 [**Models and Principles**]

Keywords

association discovery, interactive IR

1. INTRODUCTION

In this paper, we address the task of finding associations among entities of various types from a document collection. Examples of entity types include person or organization. Knowing how entities are connected is useful in many areas. Auditors may conduct an investigation by sifting through a company's documents to find associations of interest. Law enforcement officials may investigate criminal links by going through reports or news articles. A similar task has been attempted manually by Krebs [8], who mapped a network of 911 terrorists from publicly released news articles.

In addition to associations, we are interested in the documentary evidence, or the subset of documents that support these associations. The documentary evidence would help in validating the discovered associations, as well as provide more contextual information about the semantics of these associations. Moreover, keeping track of evidence of associations is useful, as this evidence may change over time.

As a running example, we adopt the task of learning how *Al-Qaeda*, an international terrorist organization originating in Afghanistan, is related to *Abu Sayyaf*, a separatist group in the Philippines. We apply this task to terrorism incident files maintained by Terrorism Knowledge Base (TKB)[1].

Query-driven retrieval systems [4] are ill-fitted for this task. To discover associations between two given entities, a user would pose the entities as queries, read the top document returned, extract more entities from this document, and again pose new queries formed with the newly discovered entities. This requires a lot of manual, repetitive work. There is neither representation of association nor awareness of entity types, making it hard to specify that a query is for association or that only certain entity types are of interest.

Another approach may be to first construct the association graph of all entities in the document collection and then to automatically derive a subgraph summarizing how two given entities are related[5]. This approach does not address the fact that different users may be interested in different subgraphs based on their unique information need. On the other hand, a user could not possibly construct her own subgraph from the huge graph without any assistance.

Instead, a more effective tool to tackle this task should have the following features: (1) awareness of entity types, (2) intuitive representation of association, and (3) user-driven discovery of associations. In this paper, we propose a model called *Text-cUBE* or *TUBE* with these features. Hereinafter, we use TUBE to refer to the model, and tube to refer to a particular instance. This model adopts a concept similar to data cube or OLAP [1] designed for relational databases and applies it onto textual data.

Figure 1 gives an example of what a tube is like. It is represented as a multidimensional table with entity types as dimensions. In this case, we have the entity type *Organization* as the dimension on both axes. There are four entities as dimension values. Each cell represents an association between two entities, and corresponds to the set of documents that support this association. For illustration, we shade a cell if there is at least one supporting document. The content of a cell is its cell summary, giving more information about this association. Examples of cell summary include the list of documents supporting this association or

[1]http://www.tkb.org

SAC'07 March 11-15, 2007, Seoul, Korea

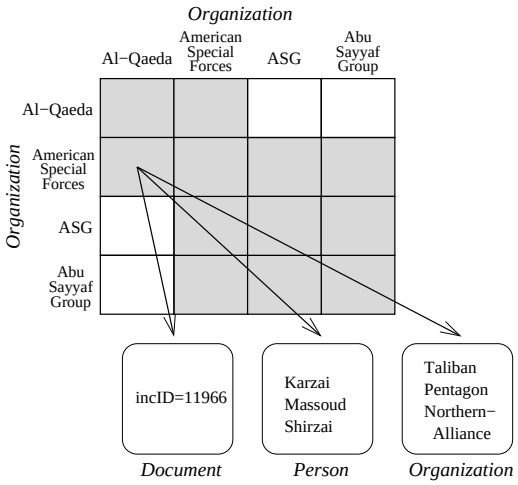

Figure 1: Example Tube

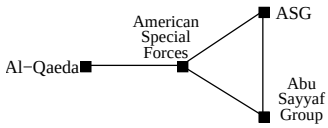

Figure 2: Network Representation

the list of other entities also related to this association. Such summaries help a user in deciding whether to read the documents or which associations to further explore. Figure 1 resembles a graph adjacency matrix. The associations can be drawn in graph form as in Figure 2. The nodes are the entities (dimension values), while each edge corresponds to a non-empty (shaded) cell. Note that TUBE is more than just a tabular representation of graphs. It helps characterize the associations with cell summaries and documentary evidence, is aware of entity types, and is supported by a set of operations (to be introduced later) to manipulate a tube.

We list the following as our contributions. Firstly, we introduce the task of user-driven discovery of associations and their supporting documentary evidence. Secondly, we propose the concept of a text cube, adapting the cube concept to text collections. Thirdly, we develop a set of TUBE operations for flexible navigation of various associations and describe several possible summarizations of nominal attributes to characterize associations. Finally, we demonstrate the application of TUBE on a real-life data.

The paper is organized as follows. In Section 2, we review some related work. The TUBE framework and operations are described in Section 3. We discuss a case study on a real-life data in Section 4. We conclude the paper in Section 5.

2. RELATED WORK

The model and operations of TUBE bear a resemblance to those found in data cube [1] designed for relational databases. However, as TUBE is designed for textual data, there are critical differences distinguishing it from data cube. Firstly, the concept of attributes such as in relational tuples is not inherent in text documents. Secondly, relational cubes are oriented primarily around numerical attributes, while for text collections, clearly we need to give greater consideration for non-numerical (e.g., nominal) attributes and the possible summarizations that can be defined for them.

In discovering associations among entities, our work is related to discovery of social networks. Social network links may be discovered from various sources, such as email exchanges [10], newsgroup postings [2], as well as co-authorship [7]. However, these works usually focus on automatic discovery of associations. In contrast, our approach allows user-driven discovery of associations and documentary evidence. This gives users additional control over exploring an association network of entities embedded in a document collection.

Several approaches to improve retrieval performance may also be relevant to our model. The ranking approach orders retrieval results by relevance, such as done by Web search engines [4]. The clustering approach groups documents into several clusters [6, 9], assuming that the user would be primarily interested in only one or two of these groups. Ranking and clustering are orthogonal concepts and can be incorporated into the TUBE model. The cell summary in a tube could be a ranked list or a clustering of documents.

3. TUBE

In this section, we provide a high-level description of the TUBE framework and operations.

3.1 Framework

Dimension and Entity Dimensions (or entity types) are specific classes of words upon which we want to draw associations. Entities are instances of these dimensions. For a document collection on a particular topic, an ontology of that collection would be helpful in identifying dimensions. We assume that we can identify a meaningful set of dimensions and the respective entities for each collection. For instance, we may extract entities from documents using natural language processing techniques or maintain a database of all entities of interest. In this paper, for simplicity, we consider two common dimensions: *Person* and *Organization*.

Document Collection A document is any delimited textual passage. It could be a file, a paragraph, or a sentence. A document collection is a set of documents. However, TUBE may not be suitable for any mixture of documents. We believe that TUBE should be applied to a document collection of a specific topic. This way, dimensions would have consistent meanings across all documents. In this paper, we work with a document collection on the topic of terrorism.

Association An association among two or more entities exists if there is documentary evidence supporting that association. There are various ways to define whether a document supports an association of entities. In this work, we use basic co-occurrence as basis for association. A document supports an association among a set of entities if all the entities co-occur within this document. The set of all such documents form the documentary evidence of the association. A stronger form of association is semantic associations, which provide awareness of subject, object, and action. Such forms of association may be considered for future work.

Cell Summary Each cell corresponds to a combination made up of one entity from each axis of a tube. We say that this cell is *indexed* by the entities in the combination. A cell represents a potential association among the indexing entities. Cell summary is the content of a cell, containing a summarization of the documentary evidence of the association. Figure 1 gives several example summaries for the

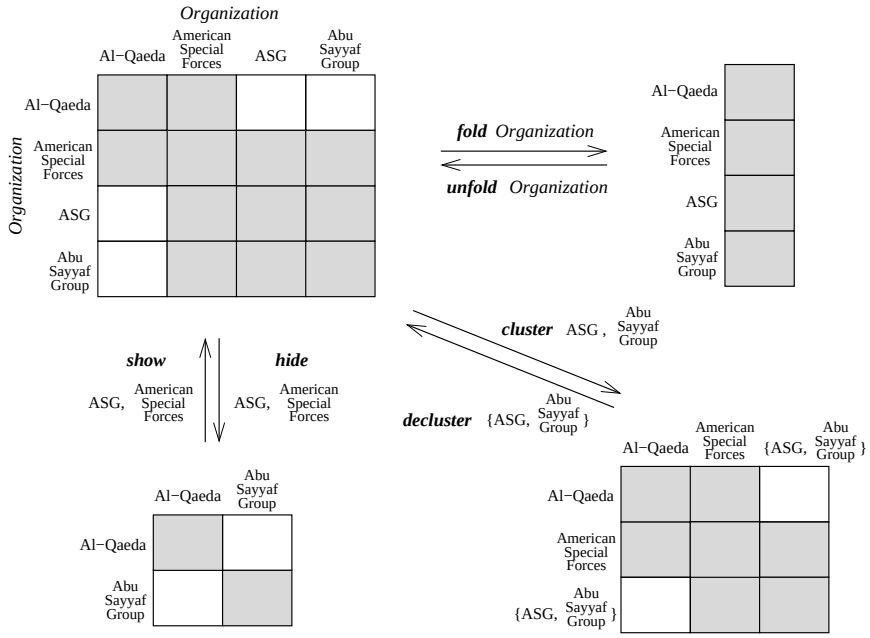

Figure 3: TUBE Operations

American Special Forces–Al-Qaeda cell. It could be the set of supporting documents, as well as the *Person* or *Organization* entities occurring within these documents. These hint at the semantics of the association and help identify other entities on which to draw associations.

3.2 Operations

A user interacts with a tube through a set of operations. These operations provide users with the following options: (1) to select dimensions (entity types) of interest, (2) to select entities to draw associations on, (3) to group together entities considered to be equivalent or closely related, (4) to select which cell summarization to use, and (5) to select a subset of interest from the document collection.

To illustrate the various operations, we introduce Figure 3. The upper left-hand tube in this figure is identical to that in Figure 1, whereas the other tubes illustrate the effects of some of the following operations.

1. **fold** operation removes one axis of a specified dimension from a given tube, while **unfold** introduces a new axis of a specified dimension to a given tube.
2. **show** operation ensures that a user-selected entity appears as a dimension value, whereas **hide** ensures the non-appearance of the specified entity.
3. **cluster** operation groups together a specified set of entities of the same dimension. A cell indexed by a cluster corresponds to documents containing any entity in the cluster. **decluster** splits a specified cluster.
4. **summarization** refers to the group of cell summarization functions. Each function yields a summary of the documentary evidence in a tube.
5. **filter** operation allows a user to impose filtering condition such that only those documents meeting the condition would remain in a tube.

Note that a tube consists of two distinct components: *view*, made up of the selection of dimensions and axial entities, and *data*, made up of the underlying collection of documents. The same *view* applied on different *data* (vice versa) would produce a tube with different cell contents. **fold/unfold, show/hide, cluster/decluster, summarization** are *view* operations and **filter** is a *data* operation.

4. CASE STUDY

We conduct a case study to investigate the applicability of TUBE on a real-life data. In addition to illustrating the workings of TUBE operations, we also seek to compare the TUBE approach against another approach, *Serial* approach, which simulates a user on a query-driven retrieval system. The main criterion for comparison is how quickly the two approaches complete a common task to discover associations between specified source and target entities. In addition, we will also compare the total number of documents covered. Moreover, since the target entity is known by several names, we will also compare how many such 'synonyms' of the target can be discovered by each approach.

4.1 Data

Of the incident files maintained by TKB's site as on April 24, 2006, we carve out the dataset *year*2002, containing 2649 incidents occurring in 2002. In addition to details such as incident date, each incident file also includes a 'Description' field, which gives a descriptive textual passage. We extract only this field from each incident file, and treat it as a document identifiable by the incID (incident ID) of the original incident. Subsequently, named entities are extracted from each document using the BBN's IdentiFinder[3] tool. Two types of entities are extracted: *Person* and *Organization*.

4.2 Task

As mentioned in Section 1, the task is to find associations between *Al-Qaeda* and *Abu Sayyaf*. We assume the user begins from the source (*Al-Qaeda*) and works her way to the target (*Abu Sayyaf*). There are other modes of exploration

```
1: entity ← source
2: add entity as tube dimension value
3: while not end of discovery do
4:    if entity = target then
5:       entity ← next most interesting entity from Al-Qaeda cell
6:    else
7:       entity ← next most interesting entity from all cells
8:    end if
9:    add entity as tube dimension value
10: end while
```

Figure 4: Pseudocode of TUBE Approach

such as starting from both entities and working our way to the "middle". As no document directly links the source and the target, any mode would require several exploration steps, which will be used for comparison of the two approaches.

Interestingness In practice, a tube user will subjectively pick dimension values according to her information need. While it is not our intent to develop an ideal interestingness measure, for comparison, we will adopt an interestingness measure to be used in both the TUBE and Serial approaches, such that a user will pick the same entity when given the same choices in both approaches. This measure is dependent on the target entity. We attach to each entity an interestingness value as follows. If we represent the set of documents containing an entity a by its capital letter A, then the strength of a link between two entities (a, b), denoted by $link(a, b)$, is given by Equation 1. Extending this definition to a path, the strength of a path from a to d made up of the links $\{(a, b), \ldots, (c, d)\}$, denoted by $path(a, d)$, is given by Equation 2. Finally, for a given node a, its interestingness with respect to a target z is the sum of the strengths of all paths originating from a and leading to z.

$$link(a, b) = \frac{|A \cap B|}{|A \cup B|} \tag{1}$$

$$path(a, d) = link(a, b) \times \ldots \times link(c, d) \tag{2}$$

The interestingness measure has several desirable properties. It assigns higher interestingness to a node (an entity) if: (1) the node has more paths leading to the target, (2) the paths to the target are shorter, (3) the links forming those paths are supported by more documents, and (4) the paths pass through nodes contained in fewer documents.

We only consider paths not longer than 3 links leading to the target *Abu Sayyaf*. In addition, we also exclude paths passing through entities occurring only within one document, since such entities cannot help to pick other entities in other documents or to pick additional documents.

4.3 Comparing TUBE vs. Serial Approach

Here, we compare the TUBE approach and the alternative Serial approach in terms of the criteria mentioned above.

4.3.1 TUBE Approach

We apply the TUBE approach on the *year*2002 dataset, which can be derived using the **filter** operation selecting only 2002 incidents. Starting with the source *Al-Qaeda*, we proceed according to the pseudocode in Figure 4. We count each execution of the while loop as an *exploration step*.

Steps 1 & 2 The user begins with a tube of only one dimension value (*Al-Qaeda*). The cell summary is set using the **summarization** operation, which takes as input a cell's supporting documents and outputs a cell summary. In this

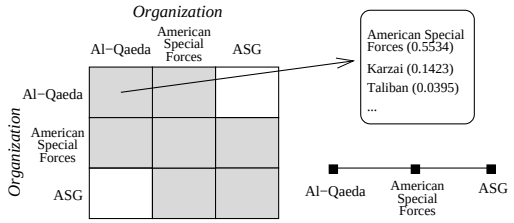

Figure 5: *year*2002: TUBE after Steps 1 & 2

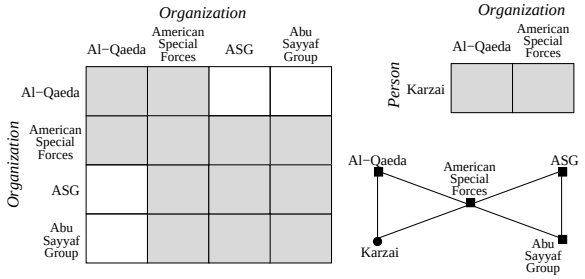

Figure 6: *year*2002: TUBE after Steps 3 & 4

case, the summary is a list of entities found in those documents, sorted in descending order of interestingness value. As shown in Figure 5, *American Special Forces* has the highest interestingness (0.5534) among entities in the *Al-Qaeda–Al-Qaeda* cell. In Step 1, we add *American Special Forces* as a dimension value (**show** operation). The cell summary of *American Special Forces–American Special Forces* cell contains *ASG* (an acronym for Abu Sayyaf Group). As the most interesting entity in any existing cell, *ASG* is added as a dimension value (**show** operation) in Step 2. Figure 5 shows the tube and the network representation after two steps, expanded into a two-dimensional tube (**unfold** operation). In two steps, the target has been reached.

Steps 3 & 4 Although the first path associating the source and the target has been discovered in Step 2, we continue the exploration in order to discover more associations involving other entities. However, as the interestingness is target-dependent, subsequent entities picked are not necessarily near the source. Consequently, once the target has been reached, we start over from the source (*Al-Qaeda–Al-Qaeda* cell). *Karzai*, the next most interesting entity in this cell, is picked in Step 3. As *Karzai* is of *Person* dimension, the addition of *Karzai* requires the user to open up a *Person-Organization* tube. In Step 4, *Abu Sayyaf Group* is picked as the most interesting entity from all cells. Though semantically equivalent, *ASG* and *Abu Sayyaf Group* have been extracted as different entities. Figure 6 shows the tubes after four steps and the corresponding network representation. We use round bullet point for *Person* entity (*Karzai*) and square bullet points for *Organization* entities.

We reiterate that each shaded tube cell is supported by one or more documents, which may be inspected to learn more about the association. For instance, upon inspection of the supporting documents, the associations in Figure 6 can be concisely explained as follows. *Karzai* was protected by his *American Special Forces* guard from an assassination attempt blamed on *Al-Qaeda*. *Abu Sayyaf Group* took eight hostages in the Jolo Island shortly after *American Special Forces* ended a sixth-month counter-terrorism exercise.

```
1: entity ← source
2: while not end of discovery do
3:    use entity as query term to retrieve most relevant document
4:    if entity = target then
5:       entity ← next most interesting entity from Al-Qaeda's docs
6:    else
7:       entity ← next most interesting entity from all docs seen
8:    end if
9: end while
```

Figure 7: Pseudocode of Serial Approach

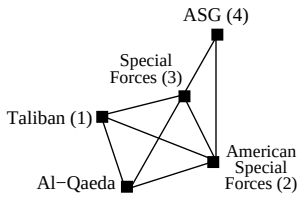

Figure 8: *year*2002: Serial after 4 Steps

4.3.2 Serial Approach

The actions under the Serial approach follow the pseudocode in Figure 7. We count each execution of the while loop as an *exploration step*. We now apply the Serial approach on the same *year*2002 dataset.

The user begins with *Al-Qaeda* as query to find the most "relevant" document. Relevance is defined as containing the most occurrences of the query entity, with any tie broken in favor of the more recent incident date. In sequence, the following documents and entities are selected at each step:

Step 1 *incID=13910* (document) and *Taliban* (entity)
Step 2 *incID=11966* and *American Special Forces*
Step 3 *incID=13618* and *Special Forces*
Step 4 *incID=13386* and *ASG*

Figure 8 shows the network representation. The number next to an entity refers to the step when the entity is picked. Although there is a direct association between *American Special Forces* and *ASG*, the document supporting this association (*incID=13386*) is not discovered until Step 4. In contrast, the TUBE approach discovers the same in Step 2.

4.3.3 Comparison

In Table 1, we give a summary comparison between the TUBE and Serial approaches after four steps. The TUBE approach first reaches the target *ASG* in two steps, while the Serial approach requires four steps. Along the way, the Serial user retrieves exactly 4 documents, while TUBE covers a total of 24 documents, containing additional information. Finally, after four steps, the TUBE discovers a second synonym of the target. This underlines the utility of TUBE in quickly discovering associations supported by a wider range of documentary evidence.

4.4 Organizing a TUBE

If we continue with the TUBE approach, the tubes are likely to get more complex. It is a useful exercise to organize and simplify the tubes. One option is to remove one or more dimension values that are not of interest, using the **hide** operation. Another option is to cluster entities that we wish to consider as a single entity. For example, *ASG*, *Abu Sayyaf*, and *Abu Sayyaf Group* all refer to the same organization. We then use the **cluster** operation on them to form the cluster {*ASG, Abu Sayyaf Group, Abu Sayyaf*}. The resulting tubes after six steps, followed by this clustering, are given in Figure 9. It is simpler than before clustering but still conveys the same amount of, if not more, information.

Table 1: *year*2002: TUBE vs. Serial after 4 Steps

	TUBE	Serial
no. of steps to first reach target	2	4
total no. of documents covered	24	4
no. of target synonyms reached	2	1

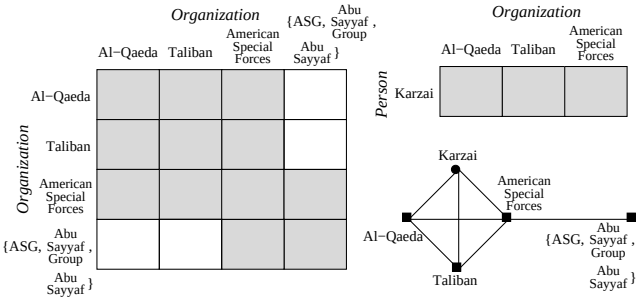

Figure 9: *year*2002: TUBE after 6 Steps

5. CONCLUSION

In this paper, we propose TUBE, or a text cube approach to discover associations of entities from a document collection. Its features include support for entity types and cell representation of association. After outlining the TUBE framework and operations, we study its applicability on a real-life data. We observe that TUBE allows for quick discovery of associations and continual organization of the discovered associations. Future work may include providing guidance on the user's next moves, such as which entities to pick or to cluster. Richer definitions of association with more semantic structure or meaning may also be explored.

6. REFERENCES

[1] R. Agrawal, A. Gupta, and S. Sarawagi. Modeling multidimensional databases. In *ICDE*, pages 232–243, 1997.

[2] R. Agrawal, S. Rajagopalan, R. Srikant, and Y. Xu. Mining newsgroups using networks arising from social behavior. In *WWW*, pages 688–703, 2003.

[3] D. M. Bikel, R. L. Schwartz, and R. M. Weischedel. An algorithm that learns what's in a name. *Machine Learning*, 34(1-3):211–231, 1999.

[4] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *WWW*, pages 107–117, 1998.

[5] C. Faloutsos, K. S. McCurley, and A. Tomkins. Fast discovery of connection subgraphs. In *SIGKDD*, pages 118–127, 2004.

[6] M. A. Hearst and J. O. Pedersen. Reexamining the cluster hypothesis: Scatter/gather on retrieval results. In *SIGIR*, pages 76–84, 1996.

[7] D. Kempe, J. Kleinberg, and E. Tardos. Maximizing the spread of influence through a social network. In *SIGKDD*, pages 137–146, 2003.

[8] V. E. Krebs. Mapping networks of terrorist cells. *Connections*, 24(3):43–52, 2002.

[9] A. Leuski. Evaluating document clustering for interactive information retrieval. In *CIKM*, pages 33–40, 2001.

[10] M. F. Schwartz and D. C. M. Wood. Discovering shared interests using graph analysis. *Communications of the ACM*, 36(8):78–89, 1993.