Available online at www.sciencedirect.com

ScienceDirect

Journal of Informetrics 2 (2008) 107–127

Journal of INFORMETRICS An International Journal

www.elsevier.com/locate/joi

Finding and tracking subjects within an ongoing debate

Rudy Prabowo*, Mike Thelwall

School of Computing and Information Technology, University of Wolverhampton, Wulfruna Street, WV1 1SB Wolverhampton, UK

Received 7 May 2007; received in revised form 20 August 2007; accepted 4 December 2007

Abstract

This paper describes a new algorithm for finding and tracking different subjects within an ongoing debate. The algorithm finds blocks of co-occurring terms, representing subjects, including blocks for which the term co-occurrence pattern forms a ring topology. We used short online debate forum data and longer summary bulletins to assess the extent to which the algorithm could correctly detect subjects, according to the judgements of human evaluators. The results show that it could normally detect subject-shifting and track different subjects over time in online debate forums and with adjustments could find subjects in bulletins, but could not track the subjects in the bulletins because the interlinking between subjects was too dense in the longer documents.

Keywords: Feature selection; Subject tracking

1. Introduction

The availability of timestamped online data, such as from discussion forums and blogs, has led a number of researchers to carry out either time-based text or link analysis. Time-based text analysis was initially developed before the web with newswire feeds and includes topic and event detection and tracking (Allan, Carbonell, Doddington, Yamron, & Yang, 1998; Allan, Papka, & Lavrenko, 1998; Schultz & Liberman, 1999; Yang, Pierce, & Carbonell, 1998), burst analysis (Kleinberg, 2003), the generation of overview timelines (Chieu & Lee, 2004; Smith, 2002; Swan & Allan, 2000), time-series analysis (Glance, Hurst, & Tomokiyo, 2004; Gruhl, Guha, Liben-Nowell, & Tomkins, 2004; Thelwall, Prabowo, & Fairclough, 2006), and evolution theme pattern discovery (Mei & Zhai, 2005). Time-based link analysis focuses on using hyperlinks or citation data to generate a graph and find changing characteristics of the graph over time (Kumar, Novak, Raghavan, & Tomkins, 2003; Leskovec & Faloutsos, 2006; Leskovec, Kleinberg, & Faloutsos, 2005).

In this paper we tackle a new but related type of problem: developing an algorithm to find and track subjects within an ongoing debate. The research objective is different from topic detection and tracking because the task is not to find and track individual topics within a general corpus of text, but to find and track all the subjects discussed within a broad topic. In terms of similar previous work, but using static collections of texts, Mei and Zhai (2005) used a mixture-model to find the evolution of themes, and Corman, Kuhn, McPhee, and Dooley (2002) used a graph-based model and betweenness centrality to find significant terms in a graph of connected terms. In contrast, we use a graph-

* Corresponding author.
E-mail addresses: rudy.prabowo@wlv.ac.uk (R. Prabowo), m.thelwall@wlv.ac.uk (M. Thelwall).

1751-1577/$ – see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.joi.2007.12.002

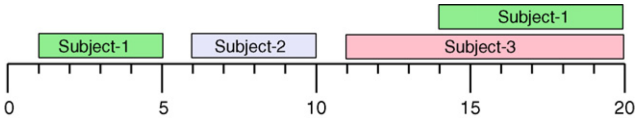

Fig. 1. A diagram illustrating the life span of the three subjects listed above.

based algorithm on a dynamic collection of texts to find and track subjects within a debate with regard to a broad topic.

For example, suppose that a number of people debate 'embryonic stem cell research' in an online forum. Fig. 1 represents this debate, which consists of 20 separate postings (represented by tick marks) within the broad theme of embryonic stem cell research but with three separate subjects listed below. Our research objective for this data would be to automatically detect these three subjects and to find their life spans.

- Subject 1: Can embryonic stem cell research legitimise abortion?
- Subject 2: Is the use of umbilical stem cells morally justifiable?
- Subject 3: Should in vitro fertilization (IVF) treatment, which can kill embryos, be allowed?

In Fig. 1 subject 1 and subject 2 appear consecutively at $1 \leq t \leq 5$ and at $6 \leq t \leq 10$. This illustrates a simple scenario of two disjoint subjects. In contrast, subject 1 and subject 3 overlap in time at $14 \leq t \leq 20$. In some cases two related subjects may also share a common issue.

In an online forum, a new subject typically appears because a participant introduces a proposition for which there is no general agreement within the forum that it is either true or false. This proposition may perhaps relate to a news story, and old issue being revisited, or a previously discussed subject. The new proposition then gives discussants a new opportunity to argue with each other and defend their own beliefs, perhaps extending a previous related debate. Facts and counter-arguments may then be presented to support each point of view until the issue is resolved or participants agree to differ, or tire and give up, or move on to another subject.

The following two concepts are used to achieve the objective of identifying and tracking subjects within a debate:

1. Blocks as dynamic objects representing subjects within a debate. First, terms (both nouns and noun phrases) are extracted from the debate. The terms are then used to form graphs with the terms being nodes and edges existing between pairs of terms that are strongly associated within the debate (defined more precisely below). From these graphs blocks of terms are extracted, which are subgraphs that have no cut vertex or bridge, and these are the dynamic objects. A cut vertex is a vertex whose deletion increases the number of components, and a bridge is an edge whose deletion increases the number of components (West, 1996).

   Blocks defined as above are used to represent subjects because our inspection of a number of debates suggested that this was the typical term pattern for subjects within debates.
2. The algorithm for finding and tracking blocks within a dynamic collection. Let us use the following example to explain the way the algorithm handles every new posting. Assume the first posting A is posted at $t = 1$. Given the posting A, a graph of connected terms, G_1 is created, using the set of term pairs with a strength of association above a threshold value (= a significant term pair set). Then, the algorithm finds blocks within the graph, G_1.

   Now, assume a new posting B is posted at time $t = 2$. From the posting B, a new significant term pair set is generated. All the new term pairs found at $t = 1$ are excluded because there are not needed. Then, the algorithm integrates the new term pair set into the existing graph, G_1, and uses the modified graph, G_2 to either create a new block or find the existing block into which the new term pairs should be incorporated.

   Hence, the algorithm can chronologically find and track the structural changes of blocks with respect to a set of significant term pairs representing a new incoming posting. Section 4 explains the algorithm in detail. To keep records of the emergence or the structural changes of each block found, each block is attached with five attributes, as listed in Section 4.3. These finding and tracking tasks are time-consuming because the algorithm needs to determine whether the existing blocks have to undergo a structural change for each new incoming term pair set, but it avoids chaining effects.

Our research hypothesis is that different subjects found in timestamped postings can be detected by finding blocks of associated terms using the method described above and explained in detail below, with each block representing a subject. The hypothesis relies on the assumption that discussants tend to use one subset of terms to discuss one subject, and another (possibly overlapping) subset of terms to discuss another subject so that different subjects can normally be detected by analysing the co-occurrence patterns of the terms used.

The motivation for identifying and tracking subjects within a broad debate is (1) to help understand public opinions on controversial science-related topics, such as climate change, embryonic stem cell research and nuclear energy power, and to (2) track debates which can potentially explode into a major issue. It is especially important for politicians, scientists, and policy makers who want to see the life span of a particular subject – from the time the subject first appears until it is no longer discussed – and want to know if and when a subject becomes a major issue.

2. Related work

This section reviews relevant research within the following topic areas: strength of association measures, theories about concept formation and organisation and graph analysis.

2.1. Strength of association measures

Measuring the strength of association between two variables, e.g., between two terms, terms and dates or terms and categories is a pre-processing task that can significantly influence the effectiveness of a text processing system. Various measures, such as Mutual Information (MI), Information Gain (*I*), χ^2, and log-likelihood, have been used for different tasks. Each measure has drawbacks and advantages, and may prove useful in one corpus, but problematic in another.

Mutual Information (MI), a measure which is based on the probability distribution of terms, has been used to measure the strength of association between two terms, to discover, for example, associations between word classes (Church & Hanks, 1989), between company and person names (Conrad & Utt, 1994), between medical terminologies (Wren, 2004), and between terms and categories (Sebastiani, 2002; Yang & Pedersen, 1997). MI has been critised for not taking into account the absence of terms, and tends to over-fit (Yang & Pedersen, 1997). When a corpus is relatively small and free of noisy terms, MI is quite modest in terms of feature reduction, however.

In time-based analysis, χ^2 or log-likelihood can be used to measure the strength of association between terms (or sentences) and dates, as a means to generate overview timelines (Chieu & Lee, 2004; Smith, 2002; Swan, & Allan, 2000). For the χ^2-test, in order to reliably accept or reject H_0, the expected values should be ¿5. Otherwise, it tends to underestimate small probabilities, which incorrectly results in accepting H_1 (Cochran, 1954). In addition, Rayson, Berridge, and Francis (2004) find that there is a need to extend the Cochran rule to reliably accept H_0, and the log-likelihood ratio test (Dunning, 1993) is more accurate than χ^2. As a ranking function, the log-likelihood ratio is therefore a better measure than χ^2 for handling rare events. The larger a log-likelihood value is, the greater the association strength between term$_i$ and term$_j$, where $-2\log\lambda$ must be greater than 3.84 ($\theta_{-2\log\lambda} > 3.84$ at the 5% confidence level). Despite the use of a Yates continuity correction, these two statistical tests are still prone to error when handling rare events.

Information Gain (*I*), a measure which is based on the conditional entropy of two variables, is used to measure the strength of association between terms and categories (Yang & Pedersen, 1997), and between terms and dates (Prabowo & Thelwall, 2006). *I* is more effective than MI and χ^2 for removing noisy terms (Yang & Pedersen, 1997), but it is an asymmetric measure and one variable should be selected as the indicator in order to measure the average reduction in uncertainty for the second variable. For example, to measure the strength of association between terms and dates, $I(D;T) = H(D) - H(D|T)$. Here, the average reduction in uncertainty about *D* that results from learning the value of *T* is measured. This is the average amount of information that *T* conveys about *D*.

Statistical language modelling approaches can also be used to find the strengths of association between words and themes. Mei and Zhai (2005) use a unigram-based model and the Expectation Maximisation (EM) algorithm (Dempster, Laird, & Rubin, 1977), as described in Zhai, Velivelli, and Yu (2004). To assign words to classes, Brown, Pietra, deSouza, Lai and Mercer (1992) use an interpolated 3-gram model. This approach requires a large corpus to smooth the statistics for the *n*-grams used, however.

We use a block of connected terms to represent a subject. To reduce the number of edges (i.e., term pairs), we experimented with Mutual Information (MI), log-likelihood and ϕ^2. These three measures were chosen because they

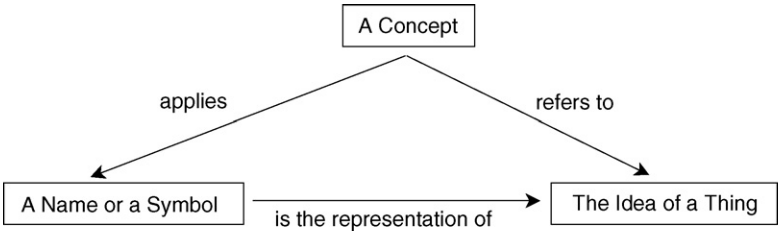

Fig. 2. The relationships between a concept, a name and the idea of a thing.

are appropriate for our task: we analyse relatively small discussions and need a symmetric measure which does not require smoothing. Section 3.2 gives details of the three measures. The experimental results show that MI and ϕ yield the same result, whilst the log-likelihood ratio test yields significantly fewer term pairs. For this reason, we used MI to select significant term pairs. The number of term pairs generated can grow exponentially when new debate elements, such as counter-arguments, proofs, references and propositions appear and trigger responses from a number of discussants. In this regard, reducing the number of edges can reduce the computational complexity of our block finding and tracking algorithm.

2.2. Concept formation, organisation and reconceptualisation

Hunt (1962) defines a concept as a statement of structure in the description of objects to which a name applies. Here, a name is a symbol that is used to refer to a set of objects, which are the instances of the concept. A name (or a symbol) can be represented by any phrase or descriptive sentence. A concept comes into existence via one kind of human inference process: abstraction. A human is able to select and abstract from many different objects. In addition, Ellis and Vasconcelos (1999) define a concept as the idea of a thing in the description of objects. Here, the idea of a thing refers to something that a human can imagine. While Ellis and Vasconcelos (1999) suggest that a concept refers to the idea of a thing, Hunt (1962) suggests that a name (or a symbol) representing a concept is used to refer to a set of objects. In other words, a name (or a symbol) is the representation of the idea of a thing to which a concept refers. Fig. 2 diagrammatically describes the relationship between a concept, a name and the idea of a thing. Maedche, Staab, Stojanovic, Studer, and Sure (2001) call this relationship ‘the meaning triangle’.

A number of researchers in the area of cognitive and educational psychology (Bourne, Dominowski, Loftus, & Healy, 1986; Conway, 1997; Dole & Sinatra, 1998; Ohlsson, 1993; Piaget, 1985; Rosch, Mervis, Gray, Johnsen, & Boyes-Braem, 1976; Smith, di Sessa, & Roschelle, 1993; Thagard, 1992) have proposed theories to explain how humans form and organise concepts. They have also investigated the way humans carry out a reconceptualisation when they encounter a new concept. The following four sections describe theories which can explain these issues, and give typical examples of systematic concept formation and organisation.

2.2.1. Three theories of concept formation

Humans are able to abstract from different objects which share common stimulus features. In this context,

- A human is acting as a concept learner.
- An object is regarded as a concept instance.
- A stimulus feature is defined as an element that a concept learner percepts, such as an object, the property of an object, the value of an object property, or an event.

On the basis of the common stimulus features, humans develop and form a concept (Bourne et al., 1986). Many cognitive psychologists regard concepts as the central constructs in cognition as they constitute units of mental representation (Carey, 1992). In this respect, three theories of concept formation, and a theory which explains the way concepts can be organised in a hierarchical model, are presented below.

Associationistic theory models how humans associate stimulus features, objects and concepts. There are three kinds of association:

1. *Stimulus response association*. A concept learner responds to novel stimulus features by attaching a conceptual response to them. A conceptual response is defined as the abstract representation of the concept instances to which the novel stimulus features refer. In this context, concept formation is described as the attachment of a single conceptual response to a set of stimulus features (Hull, 1920).
2. *Association and adaptation*. A concept learner responds to each novel stimulus feature and filters out irrelevant (or non-common) stimulus features, to identify a concept. An irrelevant (or non-common) stimulus feature is defined as a stimulus feature that a concept learner can ignore as it is inconsequential for recognising a new concept (Bourne et al., 1986).

   For example, since the instances of the concept bird can differ in size, a concept learner has to devise another feature to determine whether an object is a bird. In this example, size is regarded as an irrelevant stimulus feature.
3. *Mediational processes*. A concept learner talks to themselves about a stimulus feature at hand and how to respond to it, with respect to a number of dimensions. The concept learner changes the value of one dimension to another value in order to determine whether one object differs from the other (dimensional shifting). The term dimension refers to the property of an object (Jahnke & Nowaczyk, 1998).

   The following example illustrates dimensional shifting. Two dimensions, colour and size, are used to distinguish between a raspberry and a blueberry. If two blueberries have different sizes, a concept learner may carry out dimensional shifting to infer that though they have different size, both are blueberries (and not raspberries), as they have the same colour.

The second model, called exemplar theory, posits that unlike the associationistic theory, a concept is not characterised by a set of stimulus features. Instead, a concept is represented by a collection of exemplars (Rosch, 1973; Smith & Medin, 1981). An exemplar is defined as an object which represents a typical example of a concept. For example, instead of characterising the concept bird by using its properties, such as size and use of wings, a collection of individual remembered birds (e.g., robin, sparrow) is used to represent birds.

The third model, called ill-defined categories, posits that when a concept learner is learning a category, they subsequently

1. Analyse the novel stimulus features presented.
2. Define the ideal form for the category into which a set of objects, which are associated with the stimulus features, belong. The category must be able to deal with tolerable variability among instances of the category. Experimental results, reported in Posner, Goldsmith, and Welton (1967) and Posner and Keele (1968), suggest that when humans are learning a category, they abstract a schematic kind of representation with respect to not only the objects presented, but also previously unseen objects.

In this context, a category is regarded as an analog representation of a concept, and can subsume and represent a set of concepts. The theory also implies that there are two levels of information: category level and instance level. Categorisation of an object is mediated at category level, while recognition of an instance requires retention of instances at instance level.

2.2.2. *Semantic concepts: a theory of concept organisation*

In addition to the three theories described in the previous section, the semantic concepts model, which deals with the organisation of concepts, is described below. The theory posits that concepts are embedded in a structural context. The most common structure used for organising concepts is a hierarchical structure.

In their study, Rosch et al. (1976) found that

- There are three different levels in the hierarchical structure: superordinate, basic, and subordinate.
- Superordinate level concepts have few properties. Thus, they provide little useful information about themselves.
- Basic level concepts have a large number of informative properties, and the largest number of distinctive properties, i.e. the properties that distinguish one concept from another at the same level. They appear to be the most informative.
- Though subordinate level concepts have the largest number of properties, they do not differ much from one another.

These findings can be applied to organise a set of categories in a hierarchical model, and to reduce the complexity of the model by using the following two strategies.

1. To ensure that the hierarchical model contains as few categories as possible.
2. To ensure that each category is as rich in properties as possible.

Basic level concepts can be emphasised to gain an optimal balance of these two strategies (Jahnke & Nowaczyk, 1998). We prefer to maintain a hierarchical model which only contains few categories at superordinate level. Nevertheless, we have to define new categories and expand the existing hierarchical model, which is against the first strategy, to have categories which are rich in properties. As stated above, the basic level concepts have a large number of both informative and distinctive properties. These are useful as categories which are rich in properties without necessarily adding complexity to the existing model.

2.2.3. *Reconceptualisation*

Whilst the previous two sections describe models about how humans form and organise concepts, this section focuses on theories about the way humans change existing concepts or structures when encountering new concepts. The term structure refers to a model which represents a set of concepts and the inter-connections between those concepts.

There are two different approaches that a concept learner can use when they encounter a new concept (Piaget, 1985).

1. *Assimilation*. This is the addition of a new concept into an existing structure (Vosniadou & Brewer, 1987). A concept learner recognises where they can integrate a new concept into the existing structure without the need to restructure it.
2. *Accommodation*. This is the modification of an existing structure. The modification process results in a conceptual change, and can be carried out in two different ways: revolutionary and evolutionary.

   A change is said to be revolutionary when a structure is restructured in a radical way. The new concept is so distinct that it replaces the old one. In addition, the current structure must be restructured, such that the new concept fits into it. Thagard (1992) refers to this kind of restructuring as a radical conceptual change. In a hierarchical structure, a concept can move from one tree to another during a restructuring process. Thagard (1992) calls the relocation branch jumping or tree switching.

   A change is said to be evolutionary when no fundamental structural change occurs. A conceptual change is a gradual process of modification of the existing structure. An evolutionary change occurs when the existing structure is transformed and refined into a more sophisticated one (Smith et al., 1993). Hence, in evolutionary change the existing structure forms the basis for new conceptions (Dole & Sinatra, 1998).

2.2.4. *Examples*

Classification schemes, such as the Dewey Decimal Classification (DDC) (Online Computer Library Center, Inc., 2003), semantic nets, such as WordNet (Miller, 1995), knowledge bases, such as Cyc ontology (Lenat & Guha, 1990) and domain ontologies are important visible examples of systematic concept formation and organisation, and reconceptualisation changes. They are the product of extensive categorisation and the hierarchy-based restructuring of connected, controlled vocabularies. Domain ontologies, for example, are designed to represent the conceptualisation of a domain (Fensel, 2001) and to enable machine reasoning (Berners-Lee, 1999). These ontologies consist of a set of concepts and the relationships between the concepts which are derived from domain knowledge and controlled vocabularies. A number of tools have been developed to speed up ontology building and management (Farquhar, Fikes, & Rice, 1996; Maedche & Staab, 2000; McBride, 2002). It is important to ensure that the final outcome represents the intended ontological model, i.e., the ontological model that is expected to be the conceptualisation of the domain of interest. Human intervention is therefore required to validate the final results (Guarino, Carrara, & Giaretta, 1994; Holsapple & Joshi, 2002).

When people argue about a subject, they provide a reasons or evidence to strengthen their point of view, but also use their own existing concept structure to compose propositions against opposing views. Because counter-arguments, evidence, and propositions are part of a debate, we model the discussants as concept learners which develop and form new concepts, and organise new concepts with respect to their own existing concept structure. To maintain the coherence and consistency of their existing concept structure, they must therefore carry out reconceptualisation changes, either through assimilation or accommodation. These cognitive processess enable them to understand and respond to different subjects.

For our research objective, we aim to automatically find and capture some fragments of discussants' concept structures found in texts. This is where the idea of coherent blocks of connected terms within a graph is applied. Each block is used to capture aggregate fragments of the discussants' concept structures with respect to a particular subject, and is regarded as a partial reflection of the discussants' concept structures. Unlike DDC, WordNet, Cyc, and domain ontologies, the graph we use is unstructured, i.e., it does not conform to one particular structure, such as a hierarchical structure. The vocabularies used are uncontrolled, and no categorisation is applied. Although both domain ontologies and our graph represent sets of interconnected terms, our graph is not built to represent the conceptualisation of a domain, but to reflect the relationships between terms found in texts. Despite this difference, the blocks found in the graph can assist an ontology engineer to address the two issues in ontology building: (1) finding the context in which a concept is used and (2) extending the coverage of an existing ontology.

2.3. Graph analysis

Many graphs possess common properties, such as a power law distribution (Newman, 2005), and display small-world phenomena (Watts & Strogatz, 1998). For many natural graphs which evolve over time, Leskovec, Kleinberg, et al. (2005) showed that (1) they obey a Densification Power Law (DPL), i.e., the graphs become denser with the number of edges increasing linearly with respect to the number of nodes, and (2) the effective diameter of the graphs shrink as the number of nodes increases. In addition, a number of algorithms have been developed for different purposes by using hyperlinks or citation networks:

- Identifying communities (Flake, Lawrence, & Giles, 2000; Girvan & Newman, 2002);
- Finding hub and authority Web pages (Kleinberg, 1998) and structural patterns (Holder, Cook, & Bunke, 1992);
- Developing a simulation model that can generate a graph which mimics a real-world graph (Leskovec, Chakrabarti, Kleinberg, & Faloutsos, 2005; Leskovec, Kleinberg, et al., 2005).

In contrast, we concentrate on the generation of an unstructured graph of connected terms, as explained in Section 2.2.4. This is a type of graph that is designed for text analysis, and similar to the ones described in Danowski (1982), Corman et al. (2002), and Landauer, Laham, and Derr (2004). Danowski (1982, 1992) use a graph of connected word pairs and apply the graph for information retrieval. Two word co-occur if they appear within three word positions of each other. Corman et al. (2002) propose a Centering Resonance Analysis (CRA) graph composed as follows: given a sequence of sentences, for each sentence, each word, excluding stopwords, is connected to the next one. Corman et al. (2002) use the graph to compute the degree of importance of a word based on the notion of betweenness centrality. A visualisation of CRA graphs can be found in Brandes and Corman (2003). Landauer et al. (2004) use latent semantic analysis to compute the similarity between articles. Their graph is composed of a set of connected articles. Although our graph is similar to the ones mentioned above, the way it is constructed and used is different. The graph is composed of the connections between all possible combinations of term (word or phrase) pairs – within a sentence – which have a strength of association greater than a predefined threshold value. It is used to keep track of the structural changes of blocks representing subjects.

3. Definitions and measures used

This section gives a formal definition of a graph of connected terms, and explains the three strength of association measures used.

3.1. Formal definition

In our data sets, each item has a timestamp, indexed by t, a positive integer value starting at 1. We are interested only in the order in which the items appear and not the actual time difference between items. Assume that a strength of association measure has been chosen. Let item$_t$ be an item (e.g., a forum posting or discussion bulletin) that is posted at time t and consists of an ordered list of sentences, $S = \{s_1, \ldots, s_n\}$, where n = the number of sentences in item$_t$. For each sentence s_i, a set of terms $T = \{\text{term}_1, \ldots, \text{term}_m\}$ can be extracted and all $m(m-1)/2$ possible combinations of term pairs TP can be generated, where m = the number of terms in s_i.

Definition 3.1. Given a term extraction algorithm, an association measure and a threshold θ, the undirected graph G_{TP,s_i} of a sentence, s_i is the set of vertices T with the edges being all the term pairs which have a strength of association greater than θ.

Definition 3.2. Given a set of chronologically ordered items, where $1 \leq t \leq z$, where z is the timestamp index at which the last item was posted, let discuss$_t$ = $\cup_{I=1,\ldots,t}$item$_i$ be the total discussion until time t. The graph over time G_t is defined to be $\cup_{s_i \in \text{discuss}_t} G_{\text{TP},s_i}$.

Note that G_z is the final graph, which has evolved over time through a series of expansions from G_1. By definition, $G_1 \leq G_2 \leq \cdots \leq G_z$ but G_{t+1} is not always strictly larger (more vertices) or denser (more edges) than G_t.

To find blocks of co-occurring terms within G_t, we operate on a depth-first search (DFS) tree of G_t and find cut vertices based on the DFS tree, as explained in Tarjan (1972) and Mount (1998).

Definition 3.3. Let

- [1.] $d[v]$ be the discovery time of the vertex v as a result of traversing the DFS tree and
- [2.] Low$[u] = \min\{d[u],\ d[w] | (v, w)$ is a back edge and either $v = u$ or v is an ancestor of $u\}$.

Based on the Definition 3.3,

Lemma 3.4. *A vertex, u is a cut vertex iff either*

- [1.] *u is the root of the DFS tree and has at least two child vertices, or*
- [2.] *u is not a root and has a child vertex w such that Low*$[w] \geq d[u]$.

The proof of Lemma 3.4 can be found in Tarjan (1972), Even (1979) and Mount (1998). The associated pseudo codes are discussed in Section 4.1. To track blocks over time, we need to decompose ring topologies at the sentence level to avoid chaining effects which can result in identifying only a single giant block representing multiple different subjects. Let $\mathcal{B}_t = \{B_1, \ldots, B_k\}$ where k= the number of blocks found so far. Let $\mathcal{C}_t$ be a set of cut vertices separating each block in $\mathcal{B}_t$. We can then define $\mathcal{G}_t$ as follows:

Definition 3.5. $\mathcal{G}_t = \mathcal{C}_t \cup \mathcal{B}_t \cup \{E(u, B) | u \in \mathcal{C}_t, B \in \mathcal{B}_t, u \in V(B)\}$

For each G_{TP,s_i}, apply Lemma 3.4 to find cut vertices and carry out decomposition for a ring topology, i.e., $\mathcal{B}_t = \mathcal{B}_t \cup B_i$ if B_i is a new block found.

Lemma 3.6. *For each block pair* $(B_i, B_j) \in \mathcal{B}_t$, *either* $B_i \cap B_j = u(\in \mathcal{C}_t)$ *or* $B_i \cap B_j = \{\}$.

Proof. Let $d[B_i]$ be the discovery time of B_i. Given a DFS tree, we separate the children of the root and regard each of them as a subset of the tree. Lemma 3.4 is applied to traverse each subset to find a cut vertex, separating either between one block and one bridge or between two blocks. For each vertex, v being examined, include the children and neighbours of v to establish an exhaustive subset of $\mathcal{G}_t$ with regard to v, and keep the proper ancestor of v under examination. By using Lemma 3.4, a new block, B_j of v, if any, is separated from the current block, B_i which is associated with the proper ancestor of v, with $d[B_i] < d[B_j]$. (This is similar to how Paton (1971) recursively extracts induced subgraphs to find blocks, but unlike Paton's algorithm, recursive vertex traversing, which suffers from chaining effects, is not adopted, instead separation of blocks at execution time is used.) Since each block pair is a result of separation, each pair may have a maximum of one cut vertex. Hence, two cases are possible. $B_i \cap B_j = u \in \mathcal{C}_t$ for a pair connected by a cut vertex and $B_i \cap B_j = \{\}$ for a pair connected by a bridge. In each case, $d[B_i] < d[B_j]$. The associated pseudo codes are discussed in Section 4.2. □

Table 1
A contingency table for counts of co-occurrences of terms within a set of N sentences

	term_j	$\overline{\text{term}_j}$	
term_i	a	b	$r_1 = a + b$
$\overline{\text{term}_i}$	c	d	$r_2 = c + d$
	$c_1 = a + c$	$c_2 = b + d$	$N = a + b + c + d$

3.2. Strength of association measures used

The contingency table of a term pair is given below, where N is the total number of sentences used in a collection (Table 1). For example, b is the number of sentences containing term_i but not term_j.

As introduced above, we compared three strength of association measures: Mutual Information (MI), the phi coefficient (ϕ), and the log-likelihood ratio ($-2\ \log\lambda$) for generating sets of significant term pairs for our block finding algorithm. The definitions are given below.

$$MI = \log_2 \frac{P(\text{term}_i, \text{term}_j)}{P(\text{term}_i)P(\text{term}_j)} = \log_2 \frac{aN}{(a+b)(a+c)} \tag{1}$$

Larger MI values indicate greater association strength between term_i and term_j, when the joint probability $P(\text{term}_i, \text{term}_j)$ is greater than the product of the probabilities of $P(\text{term}_i)$ and $P(\text{term}_j)$. Two examples of the use of this method for measuring the strength of term associations can be found in Church and Hanks (1989) and Conrad and Utt (1994).

$$\phi = \frac{ad - bc}{\sqrt{(a+b)(c+d)(a+c)(b+d)}} \tag{2}$$

Instead of using ϕ^2 (Church & Gale, 1991; Conrad & Utt, 1994), which is equal to χ^2/N, we use ϕ so that negative relationships can be excluded. Again larger ϕ values indicate a greater association strength between term_i and term_j, where ϕ must be positive ($\theta_\phi > 0$).

The log-likelihood ratio is as follows, where $i = \{a, b, c, d\}$ and $j = \{c_1, c_2, r_1, r_2\}$:

$$-2\ \log\lambda = 2\left\{\sum_i O_i\ \ln\frac{O_i}{E_i}\right\}, \qquad -2\ \log\lambda = 2\left\{\sum_i i\ \ln(i) + N\ \ln N - \sum_j j\ \ln(j)\right\} \tag{3}$$

Note that although the three measures above are often used as ranking functions, here their purpose is to select a set of term pairs which have a strength of association above a threshold value, θ, to make a binary decision as to whether term_i has a significant degree of association with term_j.

4. Finding and tracking subjects

Block finding algorithms (Gabow, 2000; Paton, 1971; Tarjan, 1972) can be used to find blocks of interconnected vertices within a graph, and to find cut vertices and bridges in undirected graphs. Existing algorithms are not designed to find sets of blocks that are connected with each other in a ring topology, however. The following three figures illustrate this problem. Each block is represented by a triangle.

Existing block finding algorithms can find the cut vertex in Fig. 3 and the bridge in Fig. 4, but they are not designed to find 3 bridges in a ring topology, as in Fig. 5, and so could return the entire graph as one block. In a graph of connected terms, some terms may be used to discuss different subjects, and the connection between terms may form a ring topology. To be able to track different subjects over time, ring topologies must sometimes be broken up into separate blocks to avoid having a single giant block representing multiple subjects. We apply the Tarjan algorithm (Tarjan, 1972), as explained in Mount (1998), Tarjan (2000) and Cormen, Clifford, Leiserson, and Rivest (2001), and implement a wrapper to deal with ring topologies. Sections 4.1 and 4.2 describes the algorithm and the wrapper in detail. Section 4.3 describes the process used to find and track different blocks over time.

4.1. Block finding algorithm

The pseudo code of the Tarjan block finding algorithm (Tarjan, 1972) is shown below.

```
stack ← {}
cut_vertices ← {}
blocks ← {}
discovery_time ← 0

procedure find_blocks(G) {
   MST ← build_mst(G)
   Tree ← build_dfs_tree(G,MST)
   for each u ∈ Tree.vertices {
      u.visited ← false
      u.discovery_time ← 0
      u.low ← 0
   }
   find_blocks(Tree.root) /* goto line 18 */
   return blocks
}

procedure find_blocks(u) {
   u.visited ← true
   discovery_time ← discovery_time+1
   u.low ← u.discovery_time ← t
   $V_{adjacent}$ ← u.children ∪ u.back_vertices
   for each v ∈ $V_{adjacent}$ {
      if(v.time > u.time) {
         stack.push(edge(u,v))
      }
      if(v.visited == false) {
         find_blocks(v) /* a recursive call: goto line 18 */
         u.low ← min{u.low,v.low}
         if(u.parent == NULL) {
            number_of_root_children ← number_of_root_children+1
            if(number_of_root_children > 1) {
               cut_vertices.add(u)
            }
            store_a_new_block(edge(u,v)) /* goto line 46 */
         } else if(v.low ≥ u.discovery_time) {
            cut_vertices.add(u)
            store_a_new_block(edge(u,v)) /* goto line 46 */
         }
      } else if(v != u.parent) {
         u.low ← min{u.low,v.discovery_time}
      }
   }
}

procedure store_a_new_block(edge(u,v)) {
   Block block = new Block()
   while(stack.peek() != edge(u,v)) {
      block ← block ∪ stack.pop()
   }
   block ← block ∪ stack.pop()
   blocks ← blocks ∪ block
}
```

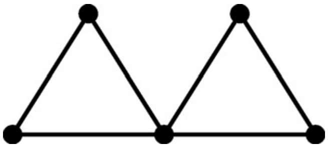

Fig. 3. Two blocks and one cut vertex.

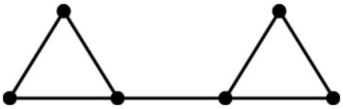

Fig. 4. Two blocks and one bridge.

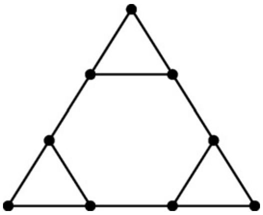

Fig. 5. Three blocks and three bridges.

The algorithm operates with a depth-first search (DFS) tree of an undirected graph, G. Prior to building the DFS tree, Kruskal's algorithm is applied to build a Minimum Spanning Tree (MST). Given the MST, a DFS tree consisting of tree-edges and back-edges is built, and all the three vertex attributes: *visited*, *discovery_time*, *low*, are initialised (lines 7–13). A *discovery_time* attribute value is a positive integer value, starting with 1, and indicates when a vertex is first discovered. A *low* value is the minimum of two values. Given a back-edge path from u to v, $u.low$ is set to $\min\{u.low, v.discovery\_time\}$. Given a tree-edge path from u to v, $u.low$ is set to $\min\{u.low, v.low\}$. By recursively updating the *low* value of a vertex, all vertices which belong to a block are assigned the same *low* value.

The algorithm traverses unvisited tree-edges recursively, (starting from the root of the tree, with $discovery\_time =$ 1), and stores the visited edges in a stack (lines 19–28). If a back-edge is found, the *low* value of u is updated (lines 40–42). When the algorithm returns from its recursive call, the *low* value of u is updated (line 29). Then, it checks two conditions (lines 32 and 36), i.e., whether the root has another child (but not its first child) or $v.low \geq u.discovery\_time$. If one of the two conditions is met, the algorithm stores u as a cut vertex (lines 33 and 37) and a new block (lines 35 and 38) by popping the stack until edge(u, v) (line 48–51). The time complexity of the algorithm is $O(|V| + |E|)$.

4.2. Block finding wrapper

We implemented a wrapper which can be used to prevent the block finding algorithm from joining several blocks connected by bridges and cut vertices in a ring topology into a single block. The pseudo code of the wrapper is shown below.

```
candidate_set ← {}
all_blocks ← {}

procedure find_blocks_incrementally(G) {
   MST ← build_mst(G)
   Tree ← build_dfs_tree(G,MST)
   for each u ∈ Tree.vertices {
      if(u.visited == false) {
         find_blocks_incrementally(u,G) /* goto line 15 */
      }
   }
   return all_blocks
}

procedure find_blocks_incrementally(u,G) {
   u.visited = true
   if(u ∉ candidate_set) {
      candidate_set ← candidate_set ∪ u
      $I_G$ ← induced_subgraph(G, candidate_set)
      blocks ← find_blocks($I_G$) /* goto line 1 in section 4.1 */
      for each block ∈ blocks {
         if(block ∉ all_blocks) {
            all_blocks ← all_blocks ∪ block
            block.is_tracked ← true
         } else if(block ∈ all_blocks && block.is_changed()) {
            block.is_tracked ← true
         }
      }
   }
   block ← get_associated_block(u)
   if(block != NULL) {
      candidate_set ← block.vertices ∪ u
   }
   V ← u.out_vertices
   for each v ∈ V {
      if(v.visited == false) {
         find_blocks_incrementally(v) /* a recursive call: goto line 15 */
      }
   }
}
```

The wrapper uses a *candidate_set* to store a set of vertices which may form blocks, and operates on a depth-first search (DFS) tree (lines 5 and 6). The tree contains not only tree-edges, but also cross-edges. The vertices of both tree-edges and cross-edges are regarded as the *out_vertices* of a vertex, *u*. The wrapper traverses each vertex, *u* in the DFS tree, and starts from the *root* of the DFS tree (lines 7–11).

At every recursive call, the wrapper checks whether *u* has not been included in the *candidate_set*. If it is the case (line 17), *u* is added into the *candidate_set*, and an induced subgraph of *G* is created, prior to applying the block finding algorithm (lines 18–20). Adding a new vertex into the *candidate_set* enables the wrapper to find the maximum size of a block. If a new block is found, then the wrapper stores the new block in *all_blocks* (lines 22 and 23). The *is_tracked* flag is used as an indicator to determine whether the histories of a block should be added, as explained in Section 4.3.

The wrapper determines the associated block of *u*, i.e. to which block *u* belongs, by looking up the predecessors and the *out_vertices* of *u*. If a block is associated with *u*, then the *candidate_set* is assigned the associated block vertices, including *u* (lines 30–33). This enables the wrapper to focus on a set of vertices which are associated with the recent new block found. In order to find the next block or to extend the size of the current block found, the wrapper recursively traverses all the *out_vertices* of *u* (lines 34–39).

In short, the wrapper incrementally traverses the DFS tree to find different blocks and their maximum sizes. The wrapper changes its focus on the next block to avoid chaining effects. These strategies enable us to deal with a ring topology, such as the one shown in Fig. 5. The time complexity of the wrapper is $O(|V|^2 + |B|)$, where $|B|$ is the number of blocks found. Although the wrapper is quadratic, it reduces the time complexity of Tarjan's algorithm to log-scale because the wrapper submits a small size of induced subgraph to Tarjan's algorithm (line 20), and reduces the size of the *candidate_set* when it changes its focus to the next block (line 32).

4.3. Subject finding and tracking algorithm

Each block is associated with the following five history lists:

- t_{hist} and timestamp$_{\text{hist}}$. These store the times at which the block undergoes a structural change, whether becoming denser or larger.
- V_{hist} and E_{hist}. These store the number of vertices and edges of the block at each point in time.
- G_{hist}. This stores the block graph at each point in time.

To find and track different blocks over time, the following algorithm is used.

```
$G_t$ ← {}
procedure find_and_track_blocks() {
    for t ← 1 to z {
        $\{G_{TP,s_1}, \ldots, G_{TP,s_n}\}$ ← build_term_pair_sets($item_t$)
        for i ← 1 to n {
            $G_t \leftarrow G_t \bigcup G_{TP,s_i}$
            all_blocks ← find_blocks_incrementally($G_t$) /* goto line 4 in section 4.2 */
        }
        for each block ∈ all_blocks {
            if(block.is_tracked == true) {
                block.insert_a_history()
                block.is_tracked ← false
            }
        }
    }
}
```

Note that in our experimental setting, the algorithm chronologically processes a fixed set of timestamped items because the number of items is known in advance (line 3). In a real-world setting, however, a first-in first-out queue buffer is needed to store the items.

For each item, item$_t$, a list of significant term pair sets is built (line 4). Each term pair set is incrementally integrated into G_t (line 6), and the wrapper (Section 4.2) is used to check whether a new block has been found or the existing blocks should be changed (line 7). This is time-consuming because structural changes for each block must be analysed for each new term pair set. The algorithm avoids chaining effects, however, and keeps track of structural changes of blocks at the sentence level. Finally, the algorithm inserts appropriate data into the five history lists for all the blocks with *is_tracked*= true, and sets the flag back to false (lines 9–14).

5. Experiment

We applied our algorithm on two sets of samples: timestamped postings and bulletins. These two data sets were used to test the algorithm on short postings and on long documents, since document length is likely to be a factor in the effectiveness of the algorithm. To test the algorithm against our research hypothesis (Section 1), we asked human assessors to judge its effectiveness for identifying subjects within broad debates. The following sections describe the samples, explain the experimental procedure used, and give the experimental and evaluation results.

5.1. Samples

We collected the following two types of samples:

- 40 postings from an online Science & Technology (ST) forum discussing global warming. Each posting separates messages as part of an ongoing debate (http://www.onlinedebate.net/).
- 9 COP3 bulletins from International Institute for Sustainable Development (IISD) regarding climate change. COP3, the third Conference of the Parties, is an international conference, which took place in Kyoto, Japan on 1–10 December 1997, and led to the creation of the Kyoto protocol for climate change. The 9 bulletins are a set of daily, timestamped bulletins which summarises daily events in chronological order (http://www.iisd.ca/process/climate_atm.htm). We selected the 9 bulletins to see whether our algorithm can also be applied to bulletins.

Both sample sets contain intense debates about particular topics, but are significantly different from each other. The ST samples are postings to a discussion forum, and the posting sizes range between 1 and 186 sentences. The COP3 bulletins are well-structured bulletins containing a summary of daily events, and the size ranges between 158 and 237 sentences. Each posting and each daily bulletin is a single timestamped item in the data sets.

5.2. Experimental procedure

For each set of chronologically ordered items, the following two steps were carried out:

1. Each sentence found within each item was processed as follows:
   (a) MontyLingua (Liu, 2004) parsed the sentences and extracted terms (i.e., nouns and noun phrases).
   (b) All noun phrases containing conjunctions (e.g., and, or, nor, but) were split into smaller units of terms.
   (c) Words were depluralised, i.e., using weak stemming only to avoid loosing word meanings (Belew, 2000).
   (d) An existing stopwords list (Fox, 1990) Was used and extended with the names of discussants and common Web-terms, such as http and www, to remove all terms consisting only of stopwords and to prune the start and end of terms.
2. All possible combinations of term pairs were generated, and the strength of association for each term pair was computed using MI (Section 3.2). Each set of term pairs with a strength of association greater than 0 was selected to generate a new graph or to extend the existing graph. We used the graph to find and track blocks (representing subjects) (Section 4.3).

5.3. Experimental results: finding and tracking subjects

Each global warming posting was assigned a discussant name and a timestamp (Table 2), and each climate change bulletin a timestamp (Table 3). To show structural changes of blocks over time and to discard noisy blocks, only blocks appearing at least twice and containing at least ≥ 3 nodes and 3 edges were used.

5.3.1. ST: global warming

The main argument in the 'Global Warming' debate was whether humans are currently the main cause of climate change. From Table 4, at $t = 9$, the discussants argued about the definition of global warming (subject 1). At $t = 17$, this subject appeared again, as one of the discussants raised this issue. Our tool could not detect the subject at an earlier point in time ($t = 2$), however, due to data sparseness. At $10 \leq t \leq 37$, the subject shifted to human influences on the environment (subject 2). This subject dominated the discussion. At $t = 19$, a new subject about the relationship between global warming and carbon dioxide emission (subject 3) was detected. Our block finding algorithm separated the third subject from the second subject. The subject appeared again at $t = \{20, 35\}$. At $t = 30$, subject 4 was found. The structural changes of the block representing the subject happened almost continuously at $32 \leq t \leq 38$. Note that one posting sometimes contained more than two subjects. Thus, at a single point in time, t, two blocks representing two different subjects may simultaneously expand or become denser. For instance, at $t = \{36, 37\}$, there were structural changes in the two blocks representing subjects 2 and 4.

Table 2
The ‘Global Warming’ posting list

t	Name	Timestamp
1	Fruitandnut	2006-02-20 11:06
2	Wannaextreme	2006-02-20 08:55
3	Wannaextreme	2006-02-20 09:01
4	Snoop	2006-02-20 09:03
5	Wannaextreme	2006-02-20 09:10
6	Snoop	2006-02-20 09:16
7	Wannaextreme	2006-02-20 09:22
8	Snoop	2006-02-20 09:27
9	Wannaextreme	2006-02-20 10:11
10	K.Browning	2006-02-20 10:40
11	Wannaextreme	2006-02-20 11:00
12	K.Browning	2006-02-20 11:04
13	Wannaextreme	2006-02-20 11:14
14	K.Browning	2006-02-20 11:22
15	Wannaextreme	2006-02-20 11:26
16	Chadn737	2006-02-20 12:25
17	Wannaextreme	2006-02-20 17:35
18	Chadn737	2006-02-20 17:44
19	Wannaextreme	2006-02-20 17:54
20	Chadn737	2006-02-20 18:10
21	Wannaextreme	2006-02-20 18:17
22	Goldphoenix	2006-02-20 18:18
23	Chadn737	2006-02-20 18:30
24	Wannaextreme	2006-02-20 18:37
25	Chadn737	2006-02-20 18:46
26	Wannaextreme	2006-02-20 18:49
27	Snoop	2006-02-20 18:50
28	Wannaextreme	2006-02-20 18:51
29	Chadn737	2006-02-20 18:57
30	Snoop	2006-02-20 18:57
31	Chadn737	2006-02-20 18:59
32	Wannaextreme	2006-02-20 19:03
33	Snoop	2006-02-20 19:07
34	Wannaextreme	2006-02-20 19:13
35	Apokalupsis	2006-02-20 22:12
36	Fruitandnut	2006-02-20 22:49
37	Chadn737	2006-02-20 23:51
38	Wannaextreme	2006-02-21 06:51
39	Snoop	2006-02-21 06:55
40	Wannaextreme	2006-02-21 07:01

Table 3
The ‘Climate Change’ posting list

t	Timestamp
1	1997-12-01
2	1997-12-02
3	1997-12-03
4	1997-12-04
5	1997-12-05
6	1997-12-06
7	1997-12-08
8	1997-12-09
9	1997-12-10

Table 4
The ‘Global Warming’ subject-time mappings

Subject-Id:Subject-Description: vertex labels	*t*
1:Definition: definition, global warming, correct definition, exact definition	9, 17
2:Human influence on environment: global warming, environment, human, influence, fact, claim, debate, statistic, carbon dioxide, result, human activity, climate, problem, weather, idea, forest, water, acre, effect, meteorology, junk science, resource, impact, global cooling, economy, inevitable change, minor shift, global warming effect, inevitable consequence, ocean level, preparation, model, scientific, economist, economic	10, 12, 13, 15, 20, 21, 23, 25, 35, 36, 37
3:The theory of global warming: definition, average global temperature, increase, atmosphere, carbon dioxide, gas, greenhouse effect, increased level, phenomena, global warming, theory, average temperature	19, 20, 35
4:Can earth explode because of global warming?: earth, negative effect, cooling, core, heat, article, proof, solid core, life, process, reactor, radiation, core reactor, nuclear fuel, radionic heat, surface, central, meltdown, solar heat, molten, inner core, segregation, condition, contribution, degree, iron, kilometer, rise, dominant factor, symptom, polar cap	30, 32, 33, 34, 36, 37, 38

5.3.2. COP3: climate change

As mentioned in Section 5.1, the way the COP3 daily bulletins were composed was different from the ST samples. A significant number of terms, such as commitment, emission, party, and country names were used to describe different subjects. This led our algorithm to return a single giant block representing different subjects (the chaining effect), when the algorithm was tracking existing blocks. For this reason, we adjusted our algorithm so that prior to processing the next daily bulletin the algorithm removed all the existing blocks, as if the algorithm had started from the beginning, but the block histories were retained. This enabled us to find blocks in each COP3 bulletin. To track blocks over time, we compared the previous and current history of each block. If there was a significant overlap between the previous and the new block found ($\geq$ 80%), in terms of the vertex labels, then the algorithm inserted a new history. Five subjects were reported more than once. Subject 1 is an argument about whether the developing and developed countries have made progress towards emission reductions. Subject 2 is about gas emission reduction proposals. Subject 3 is about the management and conservation of forests to reduce carbon dioxide. Subjects 4 and 5 are about the definition and concept of a financial budget, and the report and draft decision about Subsidiary Body for Scientific and Technological Advice (SBSTA) and Subsidiary Body for Implementation (SBI) (Table 5).

In addition, we also found 4 blocks which only occurred once, as listed in Table 6.

Table 5
The ‘Climate Change’ subject-time mappings

Subject-Id:Subject-Description: vertex labels	*t*
1:Developing countries commitments: developing country, commitment, developed country, emission, evolution clause, convention, cop, evolution of commitment, discussion, future development, binding rule, berlin mandate	1, 2, 3, 6, 8, 9
2:Gas emissions reduction: qelro, three-gas, carbon dioxide, ghg, hfc, pfc, sf6, six-gas, basket, six-ghg, bubble	1, 2, 6, 9
3:Forestry: forestry activity, management, deforestation, harvesting, reforestation, forest conservation, forest management, forestry, forestry activity, conservation, sink activity, deforestation, reforestation, afforestation	5, 7, 9
4:Budget: budget period, option, target, concept, budget, protocol, meaning	3, 5
5:SBSTA/SBI: technology, development, activity, sbsta, sbi draft decision, technology transfer, sbi, subsidiary body, implementation, technology development, sustainable development	2, 4

Table 6
The subject-time mappings of ‘Climate Change’ where each block only occurred once

Subject-Id:Subject-Description: vertex labels	*t*
6:*Differentiation on reduction rate*: differentiation, concern, country, loophole, concept, discussion.only	3
7:*Global Environment Facility (GEF)*: gef, gef replenishment, gef resource gef financing, disappointment, party, project, adoption, gef procedure, climate change, cost.	4
8:*Policies and measures*: measure, policy, paragraph, non-annex, discussion, voluntary commitment	6
9:*Financial issues*: mechanism, financial resource, finance, financial mechanism, cost, developing country fccc provision, bilateral	7

5.4. Evaluation procedure and results

Five assessors evaluated the results. The primary assessor is an insider and expert who worked for the United Nations (UN) and has in-depth knowledge about the COP3 bulletins and environmental issues. We used the primary assessor’s judgement as our gold standard, against which our results were compared. The other four assessors were used for comparison purposes.

Given 49 timestamped items: 40 postings about global warming and 9 bulletins about climate change, the algorithm generated a table containing 65 rows, each of which consists of two columns: item and subject. It also generated a list of subject-keyword mappings, each of which defines the subject to which a set of keywords belongs. 44 out of 65 rows were item-subject mappings. A total of 23 of the 44 mappings were derived from the 40 postings, and 21 of the 44 mappings from the 9 bulletins. The remaining 21 rows contained the items to which no subject was assigned.

The assessors were then asked to read the 49 timestamped items, and asked whether they agreed or disagreed with the 44 item-subject mappings. When an assessor disagreed with an item-subject mapping, they were asked to propose at least one new subject for the item. This would mean that the assessor believed that the algorithm had generated an incorrect block. For the remaining 21 rows, we also asked our assessors to assign an item at least one new subject.

We computed the averaged pairwise κ values of all the possible assessor pairs, a total of 10 pairs, to measure the level of agreement among assessors, with respect to both global warming (GW) and climate change (CC). As discussed in Eugenio and Glass (2004), there are two ways to compute pairwise κ, either after Cohen (1960) or Siegel and Castellan (1988). Both have their own advantages and disadvantages. For this reason, we computed the average pairwise κ values after both Cohen (1960) (κ_{C}) and Siegel and Castellan (1988) (κ_{SC}). The average pairwise kappa values are 75.29% for κ_{C} and 75.15% for κ_{SC}, which are greater than 67%, but less than 80% (a tentative conclusion). Detailed pairwise analysis revealed that the level of agreement between the primary assessor and the other four assessors was lower than the level of agreement among the four assessors only. By excluding the primary assessor, the average pairwise kappa values are 82.36% for κ_{C} and 82.27% for κ_{SC}, which are greater than 80% (a definite conclusion). Hence, the level of agreement among the four assessors is considered to be significant.

We used micro- and macro-averaging F1 to measure both the precision and recall, with respect to all the subjects available, including the ones proposed by the assessors, a total of 33 subjects. Micro-averaging treats each item equally, i.e., it results in averaging over a set of documents. This measurement result tends to be dominated by common classes. In contrast, macro-averaging treats each class equally, i.e. it results in averaging over a set of classes. This measurement result tends to be dominated by infrequent subjects. For these reasons, we used both. To be able to carry out a binary decision, we used item-subject mappings rather than items as our unit of measurement because one item can be assigned to more than one subject.

Table 7 shows the average precision, recall, micro- and macro-averaging F1 over 5 assessors with respect to global warming (GW) forum postings and climate change (CC) bulletins. It shows that the micro-recall level is considerably lower than the micro-precision level. The macromeasures show this low level of recall more clearly. These show that our algorithm missed many subjects found by our assessors. Table 8 shows the precision, recall, micro- and macro-averaging F1 with respect to the primary assessor. It clearly shows that the precision and recall values with respect to climate change (CC) are significantly lower than the ones shown in Table 7. This is due to the level of familiarity of the primary assessor with the COP3 bulletins. The other four assessors failed to detect some implicit subjects which could only be spotted by an insider.

Table 7
The average precision, recall and micro- and macro-averaging-F1

	Average precision (%)		Average recall (%)		Average of averaging-F1 (%)	
	GW	CC	GW	CC	GW	CC
Micro	75.42	87.62	53.32	51.13	62.26	64.33
Macro	54.90	47.67	52.32	37.42	53.32	41.85

There is no reported study that is sufficiently similar that our results can be directly compared against it and so we cannot make a specific claim of improvement over previous methods. For example, to evaluate the effectiveness of their CRA graph, Corman et al. (2002) created a matrix of the 10 most frequently occurring words in a CRA graph derived from two documents, and asked 63 undergraduate students to mark a matrix cell with a tick if they thought that a pair of words should be grouped together. The matrix contained 63 word pair associations. Whilst Corman et al. (2002) conducted their evaluation at the word level in a static collection, we focused our evaluation at the subject level in a dynamic collection. We were also faced with the problem of judgement accuracy at the level of individuals. We could not consider aggregate judgements, however, because this would overestimate the micro- and macro-averaging F1.

5.5. Discussion

Our automatic approach has a number of weaknesses. Although the MontyLingua has a high level of accuracy (97%), a small proportion of parsing errors can have a significant impact on detecting a new subject. The algorithm missed important terms which could lead to the formation of new blocks representing new subjects. Data sparseness can also lead to the same problem. Some postings contain textual data from which no relationship between significant terms was drawn. Either they contain no terms or only contain a single term. This explains the reason for detecting the first subject in the global warming debate at a late point in time ($t = 9$), and why our approach achieved a low level of recall, as shown in Table 7. We may be able to overcome this problem by generating all possible combinations of term pairs within an item (instead of within a sentence). Our experimental results, however, suggest that this is not the case because it caused our algorithm to fail to detect new subjects. As a result, the algorithm returned a single giant block representing various subjects most of the time. Another alternative is to allow human intervention to conduct conversation analysis to fill the gap (a semi-automatic approach).

Tracking the subjects found in the COP3 bulletins also led the algorithm to return a single giant block. Our assumption was that a number of discussants use one subset of terms to discuss a subject, and another subset of terms to discuss another subject. In the COP3 bulletins, this was not the case. Here, we faced with a sporadic usage of terms: a significant number of terms was used to describe different subjects. This caused the interlinkings between subjects become too dense. To find blocks in this type of documents, we need to remove all the existing blocks, prior to analysing the next document.

Despite the weaknesses, our approach proposes a novel and partially successful way for the dynamic tracking of subjects in a discussion forum. It is different from clustering. Instead of dealing with the notion of a cluster, we focus on the dynamic restructuring of a number of blocks towards regrouping. We do not focus on assigning a term to a cluster, but on finding and tracking the structural changes of blocks. It is also different from Cyc ontology and WordNet, discussed in Section 2.2. Each block does not represent a class or a synset. The relationships between terms in a block are not drawn from a predefined concept formation and organisation, but from human understanding and interpretation

Table 8
The precision, recall and micro- and macro-averaging-F1 with respect to the primary assessor

	Precision (%)		Recall (%)		Averaging-F1 (%)	
	GW	CC	GW	CC	GW	CC
Micro	78.26	71.43	51.43	36.59	62.07	48.39
Macro	42.98	36.46	46.19	33.13	44.53	34.71

about a subject at the time when a discussion takes place. Our tool can also indicate whether an intense debate has taken place and has become a hot debate by showing the subsequent points in time, let say $\{t_1, \ldots, t_n\}$, in which a block appears.

6. Conclusions

We introduced a new algorithm to detect subjects within small debates evolving over time. Tarjan's block finding algorithm was used to find blocks of co-occurring terms within a debate, representing distinct subjects. A wrapper was implemented to deal with ring topologies connecting overlapping blocks. Our algorithm was built on top of the Tarjan's block finding algorithm and the wrapper to find and track the the structural changes of the blocks over time.

The tool we developed was successfully tested on online debate forum data to find and track different subjects, with human evaluators confirming that it could detect subject-shifting and track different subjects over time. It could not track different subjects in bulletin data (i.e., long documents rather than short debates), however, because the interlinking between subjects was too dense. Due to data sparseness, our approach achieved a low level of recall. Despite this weakness, the evaluation results show that our approach achieved an acceptable level of micro-precision. As a result of subject tracking, we can see a subject which appears at subsequent points in time. This can give us an indication that an intense debate has taken place because a number of discussants continuously discuss the subject. In summary, the new algorithm appears to be suitable for identifying and tracking subjects that are discussed in relatively short documents, such as online forum postings, but a different method is needed for larger documents.

Acknowledgements

The work was supported by a European Union grant for activity code NEST-2003-Path-1. It is part of the CREEN project (Critical Events in Evolving Networks, contract 012684). We thank the assessors for evaluating the algorithm results and the reviewers for their helpful comments.

References

Allan, J., Carbonell, J., Doddington, G., Yamron, J., & Yang, Y. (1998, February 8–11). Topic detection and tracking pilot study: Final report. In *Proceedings of the DARPA broadcast news transcription and understanding workshop*

Allan, J., Papka, R., & Lavrenko, V. (1998, August 24–28). On-line new event detection and tracking. In *Proceedings of the 21st annual international ACM SIGIR conference on research and development in information retrieval* (pp. 37–45).

Belew, R. K. (2000). *Finding out about—a cognitive perspective on search engine technology and the WWW* (1st ed.). Cambridge University Press.

Berners-Lee, T. (1999). *Weaving the Web: The original design and ultimate destiny of the World Wide Web by its inventor* (1st ed.). Harper San Francisco.

Bourne, L. E., Dominowski, R. L., Loftus, E. F., & Healy, A. F. (1986). *Cognitive processes* (2nd ed.). Prentice Hall.

Brandes, U., & Corman, S. R. (2003). Visual unrolling of network evolution and the analysis of dynamic discourse. *Information Visualization*, *2*(1), 40–50.

Brown, P. F., Pietra, V. J. D., deSouza, P. V., Lai, J. C., & Mercer, R. L. (1992). Class-based n-gram models of natural language. *Computational Linguistics*, *18*(4), 467–479.

Carey, S. (1992). The origin and evolution of everyday concepts. *Cognitive Models of Science*, *15*, 89–128.

Chieu, H. L., & Lee, Y. K. (2004, July 25–29). Query based event extraction along a timeline. In *Proceedings of the 27th annual international ACM SIGIR conference on research and development in information retrieval* (pp. 425–432).

Church, K. W., & Gale, W. A. (1991, September 29–October). Concordances for parallel text. In *Proceedings of the 7th annual conference of the university of waterloo centre for the new OED and text research* (pp. 40–62).

Church, K. W., & Hanks, P. (1989, June 26–29). Word association norms, mutual information and lexicography. In *Proceedings of the 27th annual meeting of the Association for Computational Linguistics (ACL)* (pp. 76–83).

Cochran, W. G. (1954). Some methods for strengthening the common χ^2 tests. *Biometrics*, *10*, 417–451.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, *20*, 37–46.

Conrad, J. G., & Utt, M. H. (1994, July 3–6). A system for discovering relationships by feature extraction from Text Databases. In W. B. Croft & C. J. van Rijsbergen (Eds.), *Proceedings of the 17th annual international ACM SIGIR conference on research and development in information retrieval* (pp. 260–270). Dublin, Ireland.

Conway, M. A. (1997). *Cognitive models of memory* (1st ed.). Psychology Press.

Corman, S. R., Kuhn, T., McPhee, R. D., & Dooley, K. J. (2002). Studying complex discursive systems: Centering resonance analysis of communication. *Human Communication Research*, *28*(2), 157–206.

Cormen, T. H., Clifford, S., Leiserson, C. E., & Rivest, R. L. (2001). *Introduction to algorithms*. MIT Press.

Danowski, J. A. (1982). A network-based content analysis methodology for computer-mediated communication: An illustration with a computer bulletin board. *Communication Yearbook*, *6*, 904–925.

Danowski, J. A. (1992). WORDIJ: A word pair approach to information retrieval. *TREC*, 131–136.

Dempster, A., Laird, N., & Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of Royal Statistical Society, Series B*, *39*, 1–38.

Dole, J. A., & Sinatra, G. M. (1998). Reconceptualizing change in the cognitive construction. *Educational Psychologist*, *33*(2/3), 109–128.

Dunning, T. E. (1993). Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, *19*(1), 61–74.

Ellis, D., & Vasconcelos, A. (1999). Ranganathan and the net: Using facet analysis to search and organise the World Wide Web. *Aslib Proceedings*, *51*(1), 3–10.

Eugenio, B. D., & Glass, M. (2004). The kappa statistics: A second look. *Computational Linguistics*, *30*(1), 95–101.

Even, S. (1979). *Graph algorithms* (1st ed.). Computer Science Press.

Farquhar, A., Fikes, R., & Rice, J. (1996, November 9–14). The ontolingua server: A tool for collaborative ontology construction. In *Proceedings of the 10th knowledge acquisition for knowledge-based systems workshop: Workshop on distributed knowledge modeling over the internet*. Banff, Canada.

Fensel, D. (2001). *Ontologies: A silver bullet for knowledge management and electronic commerce* (1st ed.). Springer.

Flake, G. W., Lawrence, S., & Giles, C. L. (2000, August 20–23). Efficient identification of Web communities. In *Proceedings of the 6th ACM SIGKDD conference on knowledge discovery and data mining* (pp. 150–160).

Fox, C. (1990). A stop list for general text. *SIGIR FORUM*, *24*(4), 19–35.

Gabow, H. N. (2000). Path-based depth-first search for strong and biconnected components. *Information Processing Letters*, *74*, 107–114.

Girvan, M., & Newman, M. E. J. (2002). Community structure in social and biological networks. *PNAS*, *99*(12), 7821–7826.

Glance, N. S., Hurst, M., & Tomokiyo, T. (2004, May 18). BlogPulse: Automated trend discovery for weblogs. In *Proceedings of the 13th international WWW conference: Workshop on weblogging ecosystem: Aggregation, analysis and dynamics*.

Gruhl, D., Guha, R., Liben-Nowell, D., & Tomkins, A. (2004, May 17–22). Information diffusion through blogspace. In *Proceedings of the 13th international WWW conference* (pp. 491–501).

Guarino, N., Carrara, M., & Giaretta, P. (1994, July 31–August 4). Formalizing ontological commitment. In *Proceedings of the 12th national conference on artificial intelligence (AAAI 1994)* (pp. 560–567).

Holder, L. B., Cook, D. J., & Bunke, H. (1992, July 1–3). Fuzzy substructure discovery. In D. H. Sleeman & P. Edwards (Eds.), *Proceedings of the 9th international conference on machine learning (ICML 1992)* (pp. 218–223). Aberdeen, Scotland, UK.

Holsapple, C. W., & Joshi, K. D. (2002). A collaborative approach to ontology design. *Communications of the ACM*, *45*(2), 42–47.

Hull, C. L. (1920). Quantitative aspects of the evolution of concepts. *Psychological Monographs*, *28*(123).

Hunt, E. B. (1962). *Concept learning: An information processing problem* (2nd ed.). John Wiley & Sons, Inc.

Jahnke, J. C., & Nowaczyk, R. H. (1998). *Cognition* (1st ed.). Prentice Hall.

Kleinberg, J. (1998, January 25–27). Authoritative sources in a hyperlinked environment. In *Proceedings of the 9th ACM SIAM symposium on discrete algorithms (SODA 1998)* (pp. 668–677).

Kleinberg, J. (2003). Bursty and hierarchical structure in streams. *Data Mining and Knowledge Discovery*, *7*(4), 373–397.

Kumar, R., Novak, J., Raghavan, P., & Tomkins, A. (2003, May 20–24). On the bursty evolution of blogspace. In *Proceedings of the 12th international WWW conference* (pp. 568–576).

Landauer, T. K., Laham, D., & Derr, M. (2004). From paragraph to graph: Latent semantic analysis for information visualization. *PNAS*, *101*, 5214–5219.

Lenat, D. B., & Guha, R. V. (1990). *Building large knowledge based systems, representation and inference in the Cyc project* (1st ed.). Addison Wesley.

Leskovec, J., Chakrabarti, D., Kleinberg, J., & Faloutsos, C. (2005, October 3–7). Realistic, mathematically tractable graph generation and evolution, using kronecker multiplication. In *Proceedings of the 9th European conference on principles and practice of knowledge discovery in databases (PKDD 2005)* (pp. 133–145).

Leskovec, J., & Faloutsos, C. (2006, August 20–23). Sampling from large graphs. In *Proceedings of the 12th ACM SIGKDD conference on knowledge discovery and data mining* (pp. 631–636).

Leskovec, J., Kleinberg, J., & Faloutsos, C. (2005, August 21–24). Graphs over time: Densification laws, shrinking diameters and possible explanations. In *Proceedings of the 11th ACM SIGKDD conference on knowledge discovery and data mining* (pp. 177–187).

Liu, H. (2004). *MontyLingua: An end-to-end natural language processor with common sense*. Available at http://web.media.mit.edu/hugo/montylingua (accessed 1 February 2005).

Maedche, A., & Staab, S. (2000, July 6–8). Semi-automatic engineering of ontologies from text. In *Proceedings of the 12th international conference on software engineering and knowledge engineering (SEKE 2000)*

Maedche, A., Staab, S., Stojanovic, N., Studer, R., & Sure, Y. (2001, July 9–11). SEAL—a framework for developing SEmantic Web portALs. In *Proceedings of the 18th British national conference on databases (BNCOD 2001)* (pp. 1–22).

McBride, B. (2002). Jena: A Semantic Web toolkit. *IEEE Internet Computing*, *6*(6), 55–59.

Mei, Q., & Zhai, C. (2005, August 21–24). Discovering evolutionary theme patterns from text—an exploration of temporal text mining. In *Proceedings of the 11th ACM SIGKDD conference on knowledge discovery and data mining* (pp. 198–207).

Miller, G. A. (1995). WordNet: A lexical database for English. *Communications of the ACM*, *38*(11), 39–41.

Mount, D. (1998). *Articulation points and biconnected components*. Lecture Note.

Newman, M. E. J. (2005). Power laws, pareto distributions and zipf's law. *Contemporary Physics*, *46*(5), 323–351.

Ohlsson, S. (1993). Abstract schemas. *Educational Psychologist*, *28*(1), 51–66.

Online Computer Library Center, Inc. (2003). *Dewey decimal classification*. http://www.oclc.org/ (accessed 1 April 2003).

Paton, K. (1971). An algorithm for the blocks and cutnodes of a graph. *Communications of the ACM*, *14*(7), 468–475.

Piaget, J. (1985). *The equilibration of cognitive structures: The central problem of intellectual development* (1st ed.). University of Chicago Press.

Posner, M. I., Goldsmith, R., & Welton, K. E. (1967). Perceived distance and the classification of distorted patterns. *Journal of Experimental Psychology*, *72*, 28–38.

Posner, M. I., & Keele, S. (1968). On the genesis of abstract ideas. *Journal of Experimental Psychology*, *77*, 353–363.

Prabowo, R., & Thelwall, M. (2006). A comparison of feature selection methods for an evolving RSS feed corpus. *Information Processing and Management*, *42*(6), 1491–1512.

Rayson, P., Berridge, D., & Francis, B. (2004, March 10–12). Extending the Cochran rule for the comparison of word frequencies between corpora. In G. Purnelle, C. F. C, & A. Dister (Eds.), *Proceedings of the 7th international conference on statistical analysis of textual data (JADT 2004)* (pp. 926–936). Louvain-la-Neuve, Belgium.

Rosch, E. H. (1973). Natural categories. *Cognitive Psychology*, *4*(3), 328–350.

Rosch, E. H., Mervis, C. B., Gray, W. D., Johnsen, D. M., & Boyes-Braem, P. (1976). Basic objects in natural categories. *Cognitive Psychology*, *8*, 382–439.

Schultz, J. M., & Liberman, M. (1999, February 28–March 3). Topic detection and tracking using idf-weighted cosine coefficient. In *Proceedings of the DARPA broadcast news workshop* (pp. 189–192).

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys*, *34*(1), 1–47.

Siegel, S., & Castellan, J. N. J. (1988). *Nonparametric statistics for the behavioral sciences* (2nd ed.). McGraw-Hill.

Smith, D. A. (2002, August 11–15). Detecting and browsing events in unstructured text. In *Proceedings of the 25th annual international ACM SIGIR conference on research and development in information retrieval* (pp. 73–80).

Smith, E. E., & Medin, D. L. (1981). *Categories and concepts* (1st ed.). Harvard University Press.

Smith, J. P., di Sessa, A. A., & Roschelle, J. (1993). Misconceptions reconceived: A constructivist analysis of knowledge in transition. *Journal of Learning Sciences*, *3*(2), 115–163.

Swan, R., & Allan, J. (2000, July 24–28). Automatic generation of overview timelines. In E. Yannakoudakis, N. J. Belkin, M.-K. Leong, & P. Ingwersen (Eds.), *Proceedings of the 23rd annual international ACM SIGIR conference on research and development in information retrieval* (pp. 49–56). Athens, Greece.

Tarjan, R. E. (1972). Depth-first search and linear graph algorithms. *SIAM Journal on Computing*, *1*(2), 146–160.

Tarjan, R. E. (2000). *Depth-first search*. Lecture Note.

Thagard, P. (1992). *Conceptual revolutions*. Princeton University Press.

Thelwall, M., Prabowo, R., & Fairclough, R. (2006). Are raw rss feeds suitable for broad issue scanning? a science concern case study. *JASIST*, *57*(12), 1644–1654.

Vosniadou, S., & Brewer, W. F. (1987). Theories of knowledge restructuring. *Review of Educational Research*, *57*(1), 51–67.

Watts, D., & Strogatz, S. (1998). Collective dynamics of small-world networks. *Nature*, *393*, 440–442.

West, D. B. (1996). *Introduction to graph theory*. Prentice Hall.

Wren, J. D. (2004). Extending the mutual information measure to rank inferred literature relationships. *BMC Bioinformatics*, *5*, 145–157.

Yang, Y., & Pedersen, J. O. (1997, July 8–12). A comparative study on feature selection in text categorization. In *Proceedings of the 14th international conference on machine learning (ICML 1997)* (pp. 412–420).

Yang, Y., Pierce, T., & Carbonell, J. (1998, August 24–28). A study on retrospective and on-line event detection. In *Proceedings of the 21st annual international ACM SIGIR conference on research and development in information retrieval* (pp. 28–36).

Zhai, C., Velivelli, A., & Yu, B. (2004, August 22–25). A cross-collection mixture model for comparative text mining. In *Proceedings of the 10th ACM SIGKDD conference on knowledge discovery and data mining* (pp. 743–748).