AUGUR: Forecasting the Emergence of New Research Topics

Angelo A. Salatino
KMi, The Open University
Walton Hall, Milton Keynes, UK
angelo.salatino@open.ac.uk

Francesco Osborne
KMi, The Open University
Walton Hall, Milton Keynes, UK
francesco.osborne@open.ac.uk

Enrico Motta
KMi, The Open University
Walton Hall, Milton Keynes, UK
enrico.motta@open.ac.uk

ABSTRACT

Being able to rapidly recognise new research trends is strategic for many stakeholders, including universities, institutional funding bodies, academic publishers and companies. The literature presents several approaches to identifying the emergence of new research topics, which rely on the assumption that the topic is already exhibiting a certain degree of popularity and consistently referred to by a community of researchers. However, detecting the emergence of a new research area at an embryonic stage, i.e., before the topic has been consistently labelled by a community of researchers and associated with a number of publications, is still an open challenge. We address this issue by introducing *Augur*, a novel approach to the early detection of research topics. Augur analyses the diachronic relationships between research areas and is able to detect clusters of topics that exhibit dynamics correlated with the emergence of new research topics. Here we also present the *Advanced Clique Percolation Method* (ACPM), a new community detection algorithm developed specifically for supporting this task. *Augur* was evaluated on a gold standard of 1,408 debutant topics in the 2000-2011 interval and outperformed four alternative approaches in terms of both precision and recall.

CCS CONCEPTS

• **Information systems** → Digital libraries and archives; • **Computing methodologies** → Artificial intelligence; • **Information systems** → Ontologies • **Computing methodologies** → Topic modeling • **Information systems** → Clustering • **Information systems** → Network data models

KEYWORDS

Scholarly Data, Embryonic Topic, Topic Detection, Topic Trends, Semantic Technologies, Clustering Algorithms, Ontologies.

1 INTRODUCTION

The ability to promptly recognise the emergence of new research topics is an important asset for anybody involved in research, including academic publishers, researchers, institutional funding bodies and so on. Nowadays, we are experiencing a rapid growth of the number of research publications produced each year [1], and keeping up with new emerging trends is becoming progressively more challenging. In the last two decades, as very large repositories of scholarly data have become available, we have witnessed the emergence of several approaches capable of detecting novel topics and their trends [2-5]. However, these approaches are only able to detect the emergence of new topics that are already associated both with specific labels and a good number of publications. This is an important limitation since relevant stakeholders need to react as timely as possible to changes in the research landscape. For instance, academic publishers such as Springer Nature, who is funding this research, can take advantage of early intelligence to commission a pertinent book or journal. In this paper, we address this issue by introducing an innovative approach for forecasting the emergence of new research topics at a very early stage.

According to the literature, a research topic lifecycle traditionally consists of two main stages [6]. In the *initial stage*, a group of scientists agree on some basic tenets, define a conceptual framework, and begin to establish a new scientific community. Afterwards, a research area enters the *recognised phase*, one in which a substantial number of authors are active in it, producing and disseminating results. Current methods for identifying new research topics focus on these two phases.

In our recent study [7], we highlighted the existence of an *embryonic phase*, which predates the *initial stage*. In this phase, a topic has not yet been explicitly labelled and recognized by a research community, but it is already taking shape and affecting the research landscape, e.g., by fostering new collaborations between previously distant research communities. This is consistent with the well-known paradigm shift theory proposed by Kuhn [8], which states that research is pursued using a set of paradigms and when these paradigms cannot cope with certain problems, a paradigm shift can lead to the emergence of a new scientific discipline. In this phase, the relevant research communities usually start to build the foundations of the emerging new area by defining the associated challenges and paradigms, forming new collaborations, and producing seminal publications.

In our previous work [7], we showed that the emergence of new research topics is correlated with specific dynamics of already established topics, paving the way to the detection of topics at their embryonic stage. In particular, using a sample of three million papers, we compared the sections of the topic co-occurrence graphs, where new research areas are about to emerge, with a control group of subgraphs associated with established topics. The results provided evidence that the emergence of a novel research topic can be anticipated by a significant increase in the pace of collaboration between relevant research areas, which can be considered as the “ancestors” of the new topic.

JCDL '18, June 3–7, 2018, Fort Worth, TX, USA

ACM 978-1-4503-5178-2/18/06…$15.00
https://doi.org/10.1145/3197026.3197052

In this paper, we present *Augur*, a novel approach that aims to effectively detect the emergence of new research areas by analysing topic networks and identifying clusters associated with an overall increase of the pace of collaboration between research areas[1]. *Augur* operates in three steps. First, it creates *evolutionary networks* describing the collaboration between research topics over time. Then it uses a novel clustering algorithm, the *Advanced Clique Percolation Method* (ACPM), to locate areas of the network that exhibit a significant increase in the pace of collaboration. Finally, it post-processes the results, merging and filtering the resulting clusters. The output of the process are clusters of existing topics (the *ancestors* of the new topic) that are nurturing a new research area that should shortly emerge. In addition, Augur also returns, for each cluster, a number of significant papers and authors, which can provide more details about the emerging research.

The main contributions of this paper are: 1) a new framework for the detection of research topics at their embryonic stage, 2) ACPM, a community detection algorithm developed for supporting this task, and 3) a gold standard composed by 1,408 debutant topics in the period 2000-2011.

The rest of the paper is organized as follows. In Section 2, we review the literature regarding the early detection of trends in research, pointing out the existing gaps. In Section 3, we describe Augur and in Section 4 we evaluate it versus four alternative approaches. Finally, in Section 5 we summarize the main conclusions and outline future directions of research.

2 LITERATURE REVIEW

Topic detection and tracking (TDT) has attracted considerable attention in the last two decades so that we can find it applied to different domains, such as social networks [9], blogs [10], and scientific literature [2, 11-16].

One of the main tasks for TDT is to analyse how topics develop in time, paying special attention to emerging topics and trends. In literature, we can find several approaches that aim to track the development of topics as well as their emergence. A classic method for identifying emerging topics, based on the identification of their rapid growth, is the burst detection algorithm of Kleinberg [17]. This approach observes the frequencies of each word and highlights the ones that occur with higher intensity over a limited period of time. However, this burst analysis is performed on every word (including stop words) and not for specific topics, therefore it must be included in a pipeline that selects relevant keywords.

Other approaches use custom metrics relying on the number of documents [12, 18] or authors [19] associated to the topic. Some other approaches perform more complex analyses, such as determining the citation patterns between documents [4, 20]. For instance, Jo, et al. [20] developed an approach that combines distributions of terms (i.e., n-grams) with the distribution of the citation graph related to publications containing that term. In particular, the authors assume that if a term is relevant for a topic, documents containing that term will have a stronger connection than randomly selected ones. Then, their algorithm identifies the set of terms having citation patterns that exhibit synergy. Similarly, He, et al. [4] combined the citation network with Latent Dirichlet Allocation (LDA) [21]. Generally, LDA is used to extract topics from a corpus, modelling topics as a multinomial distribution over words [21]. Within their study, He, et al. [4] detect topics in independent subsets of a corpus and leverage citations to connect topics in different time frames. However, these approaches suffer from time lag, as newly published papers might need some years before being cited.

Another category of approaches focus on the co-word analysis, which studies the co-occurrence of words within documents [22, 23]. Furukawa, et al. [22] proposed a method which analyses the development of conference networks to indicate the emergence of topics. In particular, using co-word analysis, they created progressive conference networks, in which nodes represent conferences and links represent their similarity in terms of keywords extracted from the papers. Then, as indicators for emerging topics, they observe conferences that are becoming similar and thus they are collapsing over each other. Di Caro, et al. [23] designed an approach for observing how topics evolve over time. After splitting the collection of documents according to different time windows, their approach selects two consecutive slices of the corpus, extracts topics using LDA and analyses how these topics change from one time window to the other. The main assumption is that by comparing the topics generated in two adjacent time windows, it is possible to observe how topics evolve as well as capture their birth and death. However, comparing two time windows implies that the new topics must appear in at least one of them, hence they have already emerged.

Another set of approaches fall into the category of overlay mapping techniques to build maps of science and enable users to assess emerging topics [24, 25]. Although these approaches provide a global perspective, the interpretation of those maps is based on visual inspection by human experts.

In brief, many approaches are capable of both tracking the development of topics over time and acknowledge their emergence. However, they focus on recognised topics, which are already associated with a good number of publications. Detecting research topics at an embryonic stage remains an open challenge.

3 AUGUR

We devote this section to presenting Augur, which is a novel approach for the effective detection of new emerging research trends. Its workflow is depicted in Figure 1, and it consists of three main stages:

i. **Creating the evolutionary networks**. Here we create semantic enhanced topic networks from publication metadata, and then convert them to evolutionary networks, which track the pace of collaboration between research topics over the last *n* years.

[1] We use the expression “collaboration between research areas” as a shortcut for “collaboration between research communities associated with specific research areas”. The community of a research area is given by the authors who publish in the area in question.

ii. **Clustering**. Here we detect cluster of topics that exhibit a significant increase in collaboration pace.

iii. **Post-Processing**. Here we filter and further enhance the returned clusters with information regarding influential authors and papers. This information is needed to help users to make sense of the results.

In the next sections, we will describe these stages in detail.

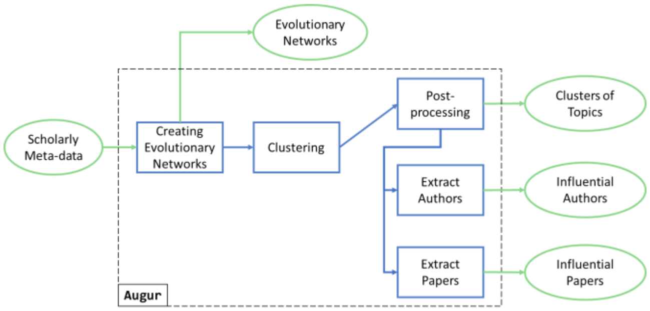


Figure 1: *Workflow of Augur. The rectangles represent the stages. The circles represent the input/output data.*

3.1 Evolutionary networks

In order to cluster the topics that exhibit an increase in collaboration pace, we first need to produce an evolutionary network for each year of the period under analysis. This is a network in which the nodes represent topics and the links represent the *pace of collaboration* (Eq. 4) of two topics in a previous time interval. In this paper, we will use a time interval of five years, since this solution was already tested successfully in Salatino, et al. [7]. Therefore, the evolutionary network of the year *t* will contain a snapshot of the interactions between topics in the year interval (*t-4*, *t*). For instance, an evolutionary network of the year 2000 will describe how the interaction between topics developed in the years between 1996 and 2000.

For generating the evolutionary networks, we first create a *semantic enhanced topic network* for each year of the time interval under consideration. Formally, each network is a fully weighted graph $G_{year} = (V_{year}, E_{year}, p_{year}, w_{year})$, in which *V* is the set of topics while *E* is the set of links representing the topic co-occurrences. The weight of a node in *p* represents the number of publications in which a topic appears in a year, while the link weights in *w* are equal to the number of publications in which two topics co-occur in the same year.

We generate the *semantic enhanced topic networks* by exploiting a dataset describing three million papers in the field of *Computer Science*, which have been classified using CSO[2], a large-scale ontology of research topics in Computer Science. CSO was originally created to model research topics in the Rexplore system [26], and is currently used by Springer Nature to classify proceedings in the field of *Computer Science* [27], such as the well-known LNCS series. CSO was automatically generated by applying the Klink-2 algorithm [28] to a corpus of 16 million scientific publications in the field of Computer Science.

We create the *semantic enhanced topic network* for a given year by selecting all the keywords from the publications in that year that also appear as concepts in CSO, and then aggregating keywords representing the same concept, i.e., keywords linked by a *relatedEquivalent* relationship in the ontology [28]. For instance, we aggregate keywords such as "semantic web", "semantic web technology" and "semantic web technologies" in a single semantic topic and we assign it to all publications associated with these keywords. The weight of nodes is the number of publications associated with the node keyword/s and the weight of a link is equivalent to the number of publications in which the keywords of the nodes co-occur.

In the current prototype of Augur, we have built fifteen topic networks representing topic co-occurrences in the 1995-2009 timeframe. We then produced an evolutionary network for each year in the 1999 to 2009 interval. The evolutionary network for a given year *t* is mathematically represented by the graph in Eq. 1.

$$G^{evol}_{year_t} = (V^{evol}_{year_t}, E^{evol}_{year_t}, p^{evol}_{year_t}, w^{evol}_{year_t}),\ \ p{:}V \to \mathbb{R}, w{:}E \to \mathbb{R} \qquad (1)$$

The evolutionary graph $G^{evol}_{year_t}$ is a fully weighted graph composed by the set of vertices $V^{evol}_{year_t}$, the set of edges $E^{evol}_{year_t}$, the weights of the vertices $p^{evol}_{year_t}$ and the weights of the edges $w^{evol}_{year_t}$.

The function that maps the five semantic enhanced topic networks (*t-4*, *t*) to an evolutionary network is showed in Eq. 2.

$$\begin{aligned} G^{evol}_{year_t} &= f(G^{topic}_{year_t}, G^{topic}_{year_{t-1}}, G^{topic}_{year_{t-2}}, G^{topic}_{year_{t-3}}, G^{topic}_{year_{t-4}}) \\ V^{evol}_{year_t} &= unique(V^{topic}_{year_t} \cup V^{topic}_{year_{t-1}} \cup V^{topic}_{year_{t-2}} \cup V^{topic}_{year_{t-3}} \cup V^{topic}_{year_{t-4}}) \qquad (2) \\ E^{evol}_{year_t} &= unique(E^{topic}_{year_t} \cup E^{topic}_{year_{t-1}} \cup E^{topic}_{year_{t-2}} \cup E^{topic}_{year_{t-3}} \cup E^{topic}_{year_{t-4}}) \end{aligned}$$

The resulting evolutionary network is composed by the unique set of topics $V^{topic}_{year_t}$ and edges $E^{topic}_{year_t}$ from the five input networks. The weight of an edge is computed as the *pace of collaboration* (Eq. 4) between two nodes while the weight of a node is computed as its *pace of growth* (Eq. 5).

In Salatino, et al. [7], we defined the pace of collaboration as the rate at which the number of publications shared by two topics changes in time and we showed that the pace of collaboration in a portion of a network is strongly correlated with the emergence of a new research area. For each edge, we first compute the *strength of collaboration* of all links in a year *t*, according to Eq. 3, and then we compute the *pace of collaboration* in a sequence of years, as showed in Eq. 4. In particular, given a link connecting node *u* and node *v*, we compute the strength of their collaboration ($\hat{w}^{topic}_{u,v}$) by normalising the weight of their link ($w^{topic}_{u,v}$) against the number of publications of both nodes (p^{topic}_{u} and p^{topic}_{v}) and then computing the harmonic mean of these two values.

$$\hat{w}^{topic}_{u,v_{year}} = HarmonicMean\left(\frac{w^{topic}_{u,v_{year}}}{p^{topic}_{v_{year}}}, \frac{w^{topic}_{u,v_{year}}}{p^{topic}_{u_{year}}}\right) \qquad (3)$$

Next, with Eq. 4, we calculate the pace of collaboration ($w^{evol}_{u,v_{year}}$) as the slope of the linear regression (computed with the least-squared method) that fits the five strengths of collaboration obtained from the same link in the five topics networks. In Eq. 4,

[2] The Computer Science Ontology http://cso.kmi.open.ac.uk

$\overline{\hat{w}_{u,v}^{topic}}$ represents the mean value of the five weights, $\overline{year}$ is the mean value of all the years $\{year_{t-4}, year_{t-3}, ..., year_t\}$ the topic networks refer to, and $year_{t-i}$, is the instance value from that set. If two topics do not have any co-occurrence in a year their strength of collaboration $\hat{w}_{u,v_{year-i}}^{topic}$ is zero.

$$w^{evol}_{u,v_{year}} = \frac{\sum_{i=0}^{4}(year_{t-i} - \overline{year})(\hat{w}^{topic}_{u,v_{year-i}} - \overline{\hat{w}^{topic}_{u,v}})}{\sum_{i=0}^{4}(year_{t-i} - \overline{year})^2} \qquad (4)$$

The node weights in $p_{year_t}^{evol}$ represent their *pace of growth* and are computed according to Eq. 5. In particular, the weight of a given *k-th* vertex ($p_{k_{year}}^{evol}$) is the slope of the line that best fits the weights of the same *k-th* vertex in the different topic networks ($p_{k_{year-i}}^{topic}$). In particular, $p_{k_{year-i}}^{topic}$ is the number of publications the *k-th* topic received in $year_{t-i}$, and $\overline{p_k^{topic}}$ is the average value of publications the same node received in that period of five years. If a topic has zero publications in a year, the weight $p_{k_{year-i}}^{topic}$ is zero.

$$p^{evol}_{k_{year}} = \frac{\sum_{i=0}^{4}(year_{t-i} - \overline{year})(p^{topic}_{k_{year-i}} - \overline{p^{topic}_{k}})}{\sum_{i=0}^{4}(year_{t-i} - \overline{year})^2} \qquad (5)$$

3.2 Advanced Clique Percolation Method

In this phase, Augur uses the Advanced Clique Percolation Method (ACPM) for detecting clusters of topics in the evolutionary networks, which exhibit an intense activity in terms of pace of collaboration, since this dynamic was shown to be linked with the eventual emergence of new topics. ACPM is an extension of the well-known Clique Percolation Method (CPM) [29] that we specifically developed to support the Augur framework. Indeed, CPM suffers from two main limitations when addressing this task. First, it does not consider the weight of edges. Secondly, since evolutionary networks tend to be very dense, it usually returns coarse-grained and large-scale communities composed by hundreds of topics. ACPM address these issues by using also the weights and by radically redefining the concept of community, with the aim of selecting fine-grained communities even in dense networks.

ACPM consists of four steps:

1. Detecting *k-cliques* within the network;
2. Measuring pace of collaboration per *k-cliques* and filtering noise;
3. Creating the k-clique adjacency graph;
4. Locating local maxima and extracting neighbourhoods.

Algorithm 1 reports the pseudocode of the ACPM. We will now report the steps of ACPM and highlight the differences with CPM.

Step 1: Detecting k-cliques within the network. Cliques and in general, k-cliques are complete sub-graphs of order *k* in which all the nodes are connected to each other. The algorithm explores the topology of the network and detects 3-cliques (therefore k=3). This step is similar to the standard CPM.

Step 2: Measuring pace of collaboration per k-cliques and filtering noise. CPM works on binary networks (i.e., undirected and without any weight). An arbitrary network can always be converted into a binary network, simply inducing the graph that contains only the links with weight higher than a threshold *w* [29]. Indeed, with CPM, the link weights can only be used to filter the links when producing the binary network upon which locating the 3-cliques. However, filtering the network with a static threshold is not the best solution. Indeed, a link having weight below *w*, can be still be used to detect 3-cliques with an intense activity of collaboration.

Therefore, inspired by Farkas, et al. [30], we filter at the clique level rather than at the link one. After detecting the k-cliques, we compute the intensity of each clique and then remove all k-cliques having an intensity below the threshold *I*. This intensity (see Eq. 6) is computed as the harmonic mean of the weights associated to its three links (w_{ab}, w_{bc}, w_{ca}), where *a*, *b*, and *c* are the nodes of the 3-clique. In particular, this value of intensity is equivalent to the pace of collaboration of the clique, analysed in our first study [7]. This strategy allows us to detect also k-cliques containing weak links (low weights) and include them in the percolation cluster, as long as they contain edges with large weights that help them to exceed the threshold *I*.

$$PaceOfCollaboration(clique) = \frac{3}{w_{ab}^{-1} + w_{bc}^{-1} + w_{ca}^{-1}} \qquad (6)$$

Step 3: Creating the k-clique adjacency graph. ACPM creates the k-clique adjacency graph $G = (V, E, W)$. *V* is the set of vertices representing the identified 3-cliques from the original graph, *E* is the set of links connecting adjacent k-cliques that share *k-1* vertices, and *W* is the set containing the node weights, i.e., the intensities of each clique computed using Eq. 6.

Step 4: Locating local maxima and extracting neighbourhoods. In this final phase, the algorithm identifies the communities within the evolutionary network. Another important difference between the standard CPM and ACPM lays in this step. CPM defines communities as connected components of the *k-clique adjacency graph*. As a result, in the presence of very dense networks, as it is the case when processing evolutionary networks, it returns very coarse-grained communities. Palla, et al. [29] suggest to monitor how communities change by trying different values of link weights *w* and tuning the value *k* for the dimension of cliques. Such analysis produces a similar effect as changing the resolution in a microscope. Increasing the threshold *w* leads the communities to shrink and fall apart, as fewer cliques will be formed. Conversely, increasing the dimension of cliques, *k*, makes the communities smaller, more cohesive and more fragmented. However, changing the values of *w* and *k* is not a good solution in this case. First, as mentioned in step 2, choosing any static threshold for *w* is not feasible. Secondly, when increasing the value of *k* (from 3 to 4 and so on), communities become too granular, preventing smaller cliques from belonging to a community.

ACPM addresses this limitation by taking advantage of the weighted *k-clique adjacency graph* produced in Step 3. It identifies the local maxima of the *k-clique adjacency graph* and then selects as communities the surrounding portion of the network.

This consists of three main steps:

1. converting the clique graph into an overlap matrix M;
2. locating local maxima in the matrix M;
3. selecting their neighbourhoods which contain all the closely related topics that show an intense collaboration.

First, ACPM converts the clique graph into an overlap matrix *M*, according to Eq. 7.

$$m_{i,j} = \begin{cases} W_j & if\ (i,j) \in E \\ W_i & if\ i = j \\ 0 & if\ (i,j) \notin E \end{cases} \qquad with\ m_{i,j} \in \mathbb{R}^{|V| \times |V|} \tag{7}$$

For each *i* and *j* clique M reports the pace of collaboration of the *j-th* clique (W_j), if a direct link between the *i-th* and *j-th* clique exists. Since a clique is always connected with itself, the main diagonal (where *j*=*i*) will report the pace of collaboration of the *i-th* clique (W_i). ACPM identifies which cliques are in a local maximum by checking if the maximum value in the corresponding row is on the main diagonal.

Then, ACPM extracts the neighbourhood of a local maximum, by selecting its *ego network*. The ego network is a network consisting of a central node (*ego*), the nodes it is directly connected to (*alters*), and the ties between them [31]. In the context of this work, an ego network consists of the induced subgraph containing a given local maximum clique and the cliques that are directly connected to it. The size of an ego network is given by its *order*. When the order is 1, the ego network includes the ego node plus its immediate neighbours. The ACPM extracts the ego network of order 2, meaning that it will contain the ego node, the immediate neighbours and the neighbours of neighbours, and all the links between these cliques. We found that using order 2 allows us to better select meaningful cliques that are not directly connected to the ego node. Finally, the ACPM converts the ego networks in clusters of topics.

3.3 Post-processing

The size of the community returned in the previous step span from 10 to 200, which is arguably a very large dimension for a community of topics that is fostering a new research topic. This is because the selection of ego networks of order 2, within very dense networks, can produce very large clusters, which may also contain some topics that do not necessarily exhibit a high pace of collaboration. In addition, since ACPM returns fuzzy communities, it may happen that two (or more) clusters share a large subset of their topics, because their mutual topics had the most active collaborations in both communities. ACPM addresses these issues by post-processing the clusters.

First, for each returned cluster, Augur ranks the links by their weights in descending order and selects the first 15, which potentially embody the most active collaborations between topics. It then prunes the clusters preserving only the topics connected by the selected links. Secondly, Augur removes redundant clusters by merging clusters that have Jaccard similarity above 0.7.

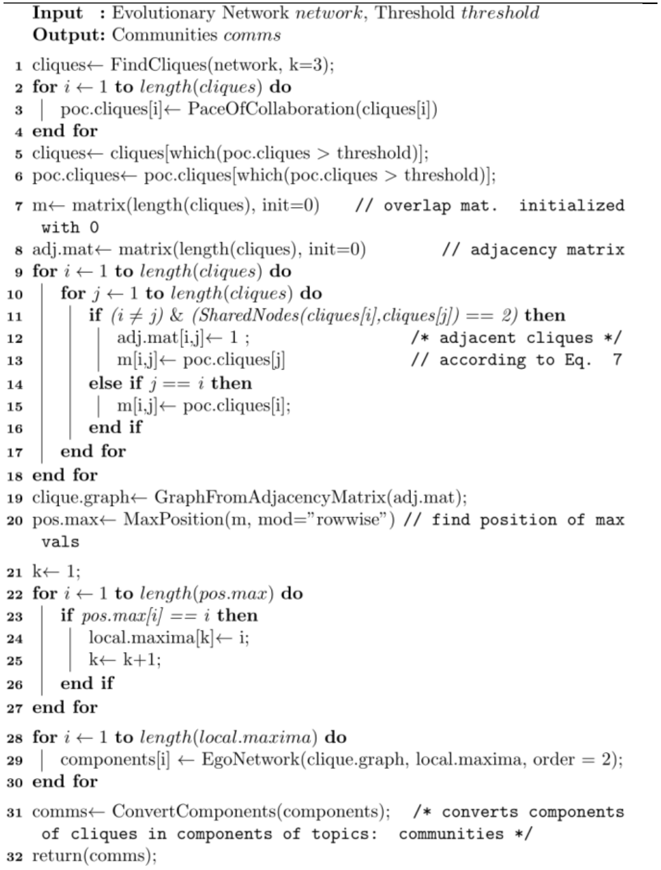
Making sense

As part of helping the user in making sense of the returned cluster of topics, Augur provides a set of influential authors and papers, relevant to cluster in question.

As influential authors, Augur identifies the authors that are actively publishing in as many topics of the identified cluster as possible. This analysis is performed in the five years prior to the detection of the cluster. Similarly, as influential papers, Augur returns the papers that have been published in the previous five years, which discuss as many of the identified topics as possible. For lack of space, we will not report here the full process for the extraction of authors and papers, which will be further addressed in future work.

In sum, Augur returns as output clusters of topics exhibiting an increase in the pace of collaboration that potentially will lead to the emergence of a new topic. In addition, for each cluster, it also returns a set of influential papers and authors that can help the user in making sense of the research dynamics in question.

Algorithm 1: *Advanced Clique Percolation Method.*



```
Input   : Evolutionary Network network, Threshold threshold
Output: Communities comms
cliques← FindCliques(network, k=3);
for $i \leftarrow 1$ to $length(cliques)$ do
    poc.cliques[i]← PaceOfCollaboration(cliques[i])
end for
cliques← cliques[which(poc.cliques > threshold)];
poc.cliques← poc.cliques[which(poc.cliques > threshold)];
m← matrix(length(cliques), init=0)      // overlap mat.  initialized
  with 0
adj.mat← matrix(length(cliques), init=0)           // adjacency matrix
for $i \leftarrow 1$ to $length(cliques)$ do
    for $j \leftarrow 1$ to $length(cliques)$ do
        if $(i \neq j)$ & $(SharedNodes(cliques[i],cliques[j]) == 2)$ then
            adj.mat[i,j]← 1 ;                   /* adjacent cliques */
            m[i,j]← poc.cliques[j]              // according to Eq.  7
        else if $j == i$ then
            m[i,j]← poc.cliques[i];
        end if
    end for
end for
clique.graph← GraphFromAdjacencyMatrix(adj.mat);
pos.max← MaxPosition(m, mod="rowwise") // find position of max
  vals
k← 1;
for $i \leftarrow 1$ to $length(pos.max)$ do
    if $pos.max[i] == i$ then
        local.maxima[k]← i;
        k← k+1;
    end if
end for
for $i \leftarrow 1$ to $length(local.maxima)$ do
    components[i] ← EgoNetwork(clique.graph, local.maxima, order = 2);
end for
comms← ConvertComponents(components);   /* converts components
  of cliques in components of topics:  communities */
return(comms);
```

4 EVALUATION AND DISCUSSION

In this section, we present an evaluation of *Augur* on the task of forecasting the emergence of new research topics. In particular, we evaluated Augur on a gold standard extracted from historical data and compared ACPM against four other algorithms: Fast Greedy (FG), Leading Eigenvector (LE), Fuzzy C-Means (FCM), and Clique Percolation Method (CPM).

To this end, we selected topic networks from the year 1995 to 2009. We then created 11 evolutionary networks taking in consideration a time interval of five years in the period 1999-2009. For instance, the evolutionary network for 1999 considers the topic networks in the years 1995-1999, while the evolutionary network of the year 2009 considers the networks in the years 2005-2009. We then processed the evolutionary networks using the five alternative algorithms and applied the post-processing described in Section 3.3.

While Fast Greedy, Leading Eigenvector, Clique Percolation Method and ACPM directly operate on networks, Fuzzy C-Means works on a feature space and needs to know a priori the number of clusters. Therefore, before evaluating FCM, we converted the evolutionary networks to adjacency matrices so that each instance (node) got as features the nodes to which it is connected. Then, we assessed the best number of clusters, using the *elbow method*. To this end, we ran several instances of FCM, in each year, iteratively increasing the number of clusters, and we observed the number of clusters in which the curve of the squared errors of prediction (SSE) starts to bend like an elbow. We found out that the optimal number of clusters for all years was 25, so we used this value for the evaluation.

The clusters resulting from an evolutionary network in a given year (e.g., 2001) were compared with a gold standard containing the ancestors of the topics that debuted in the two following years (e.g., 2002 and 2003). In the following sections, we will describe the gold standard (Section 4.1), the method for comparing the algorithm output with the gold standard (Section 4.2), the metrics adopted to assess the performance (Section 4.3), and the results of the evaluation (Section 4.4)

The data collected during the evaluation and the gold standard are available at `http://rexplore.kmi.open.ac.uk/JCDL2018/`.

4.1 Gold Standard

Very often an evaluation is carried out to compare the results of a given algorithm against a set of results determined a priori to be correct, also known as gold standard. In the context of this study, the gold standard is composed by the debutant topics that emerged from the 2000 to 2011 and a list of related topics that can be considered as their "ancestors". We consider also their related topics since all the approaches return a cluster of ancestors linked to the future emergence of a yet unlabelled topic.

In the following, we will discuss how we selected the debutant topics and the ancestors.

Generation of debutant topics.

From Rexplore dataset, with the support of CSO, we retrieved all the topics belonging to the *Computer Science* field, which emerged in the period 2000-2011.

The simplest way to find the debut of a topic is to consider the year in which the label of the topic was used for the first time as keyword in a paper. For example, according to the Rexplore corpus, the label of the topic *Cloud Computing*, made its first appearance in the year 2006. However, considering only the year in which its label firstly appeared as the year of debut can be risky. A topic label can in fact be mentioned in few papers with some meaning and then become popular years later with a completely different meaning. It is the case of "linked data", that initially was used in the context of databases to refer to pieces of data linked to each other before being adopted by the *Semantic Web* as a specific method for publishing data using the RDF format. This label misuse can create significant noise. To tackle this problem, we select as debut year of a topic the first year in which it reaches at least 5 publications. In this way, we can be more certain that a new label is already recognised by multiple authors.

Table 1 reports the number of debutant topics per year. Unfortunately, the number of debutant topics drastically decreases in the second part of the analysed period. This is probably due to missing data in our dataset. We still included the years after 2006 in the analysis for the sake of completeness, however this issue prevents us from trusting the results of the evaluation for years after 2006. As future work, we plan to analyse other scholarly datasets to provide a gold standard that will cover also more recent years.

Table 1: *Number of topics that emerged in the years between 2000 and 2011.*

Year	**2000**	**2001**	**2002**	**2003**	**2004**	**2005**
#topics	149	194	221	216	137	241
Year	**2006**	**2007**	**2008**	**2009**	**2010**	**2011**
#topics	134	60	27	12	12	5

Extraction of related topics or ancestors.

For each debutant topic in the year of analysis we select the set of its ancestors that contributed to its creation.

Our previous study [7] showed that simply selecting the most co-occurring topics as ancestors is too simplistic. Indeed, a high co-occurrence between two topics can be due to a variety of different reasons. Therefore, we consider as ancestors only the topics that most collaborate with the debutant topic during its initial stage, specifically in its first five years of activity. For each co-occurring topic, we calculate the intensity of collaboration, as showed in Eq. 8.

$$\begin{aligned} \vec{P}_{dt} &= (p_{year}, p_{year+1}, p_{year+2}, p_{year+3}, p_{year+4}) \\ \vec{C}_{dt,rt} &= (c_{year}, c_{year+1}, c_{year+2}, c_{year+3}, c_{year+4}) \\ IoC(\vec{P}_{dt}, \vec{C}_{dt,rt}) &= \sqrt{\sum_{k=0}^{4} (p_{year+k} - c_{year+k})^2} \end{aligned} \quad (8)$$

In detail, considering $\vec{P}_{dt}$ (paper vector of the debutant topic) the non-zero vector containing the amount of papers published about the debutant topic in the first five years of activity, and $\vec{C}_{dt,rt}$ (collaboration vector of the debutant topic and its related topic) the vector containing the amount of papers in which the two topics appear together in the same five years, the intensity of collaboration $IoC(\vec{P}_{dt}, \vec{C}_{dt,rt})$ between a debutant topic and a related topic can be computed as the Euclidean distance between these two vectors.

If the distance between the collaboration vector and the paper vector of the debutant topics is close to zero, it means that the debutant topic and the related topic had a very intense relationship in the first five years of life of the debutant topic and that the related topic had a major role in shaping the debutant topic.

Then, we rank the computed values in ascending order, and select the first 25 meaningful topics, considered as influential for the debutant topic. The resulting gold standard is composed by 1,408 topics in the 2000-2011 period associated with their 35,200 ancestors.

4.2 Matching clusters with debutant topics

We assess the matching between the result set of an approach and the ancestors in the gold standard by computing the Jaccard Index between the *i-th* cluster C_i and the ancestors of the *k-th* debutant topic D_k, as showed in Eq. 9. Because a topic can have more than one label (syntactic representation) referring to it, if the cluster C_i and the debutant topic D_k contain the same topic, but with different labels, the match will fail.

To tackle this problem, we employed CSO [28] to semantically enhance Eq. 9 by including all topics SA_i that have a *same-as* relationship in CSO with the topics appearing in the cluster.

In addition, we further enhanced both sets of communities and ancestors of debutant topics using the *skos:broaderGeneric* relationship between topics [28]. In CSO, this relationship is used when a topic is broader (super-area) than another one, e.g., “semantic web” is a super-area of “linked data”. We analysed four different strategies: (1) $C\ vs.\ D$ in which there is no semantic enhancement for both C and D except for the use of topics that are *same-as* in C – this semantic enhancement is included in all strategies; (2) $(C \cup Sup)\ vs.\ D$, in which clusters are enhanced with their super-areas and compared with the debutant topics; (3) $C\ vs.\ (D \cup Sup)$ in which we enhanced the ancestors of debutant topics with their super-areas, and (4) $(C \cup Sup)\ vs.\ (D\ \cup Sup)$ where at the same time we enhanced both set of clusters and ancestors with their super-areas.

The similarity measure $J(C_i, D_k, SA_i, EC_i, ED_k)$, in Eq. 9, combines the amount of topics matched between clusters and ancestors of debutant topics, and the topics matched using the semantic enhancement.

The similarity between the clusters and the debutant topics falls in the range 0 to 1. If similarity is very near 0 the two sets share only few topics, while for values of 0.1 to 0.4 there is already a good number of matching topics. When the similarity is near 1, that the two sets are almost identical.

$$\begin{aligned}
&C_i \rightarrow i\text{-}thCluster\\
&D_k \rightarrow k\text{-}thDebutant\,topic\\
&SA_i \rightarrow same\text{-}as\ of\ i\text{-}thCluster\\
&EC_i \rightarrow super\text{-}areas\ of\ i\text{-}th\ Cluster\\
&ED_k \rightarrow super\text{-}areas\ of\ k\text{-}th\ Debutant\,topic\\
&J\left(C_i, D_k, SA_i, EC_i, ED_k\right) = \frac{\left|(C_i \cup EC_i \cup SA_i) \cap (D_k \cup ED_k)\right|}{\left|C_i \cup EC_i \cup D_k \cup ED_k\right|}
\end{aligned} \tag{9}$$

We consider a positive match between a debutant topic and a cluster only when their similarity is above a threshold t. Since the similarity threshold t cannot be defined a priori, we computed precision and recall for each method in each year, for 250 similarity thresholds, from 0 to 1.

4.3 Metrics

We assessed the performance of the five algorithms by means of precision and recall. However, since a cluster can foster the emergence of more than one new topic and similarly two or more different clusters can share the same subset of ancestors that match the same debutant topic, there is not a *direct* relationship between clusters and debutant topics. Therefore, it is important to observe this relationship from different angles (matched clusters, and matched debutant topics) and focus on the following two specific questions:

1. How good is our system in identifying portions of topic networks (clusters) that will eventually lead to the emergence of new topics? (perspective of the cluster)
2. How good is our system in identifying debutant topics that have been matched with clusters? (perspective of the debutant topics)

These two questions shape the definition of the precision and recall metrics for this task. We then define **precision** as the fraction of clusters that were successfully matched with debutant topics, and **recall** as the fraction of topics that were successfully matched with the clusters, as respectively expressed by Eq. 10 and Eq. 11.

$$Precision = \frac{\left|\left\{retrieved\,clusters\right\} \ltimes \left\{debutant\,topics\right\}\right|}{\left|\left\{retrieved\,clusters\right\}\right|} \tag{10}$$

$$Recall = \frac{\left|\left\{retrieved\,clusters\right\} \rtimes \left\{debutant\,topics\right\}\right|}{\left|\left\{debutant\,topics\right\}\right|} \tag{11}$$

The symbol $\ltimes$ is called *left semijoin* and its operation in Eq. 9 returns the set of the clusters for which there is a match with the debutant topics. Similarly, the sign $\rtimes$ is called *right semijoin* and in Eq. 10, it returns the set of debutant topics that received a match from the clusters.

4.4 Results

We ran the five community detection algorithms within the Augur framework and evaluated them against the gold standard. In addition, we also ran the four different strategies described in Section 4.2.

In all cases, Augur with ACPM significantly outperforms the other approaches. In Table 2 and Table 3 we report the values of precision and recall derived by applying the four strategies on the results of the five clustering algorithms for the evolutionary networks of the years 1999 and 2000. We only show the results of the years 1999 and 2000, as the behaviour in the following years is similar. The fourth strategy, which enhances both topics in the clusters and ancestors with their super-areas, returns better values of recall and precision for all approaches, being able to identify matches that other strategies would miss. Therefore, we will adopt this strategy as default in the following analysis.

Table 2: *Values of Precision and Recall obtained using the four strategies and the five approaches in the year 1999 with a similarity threshold 0.1.*

	FG		LE		FCM		CPM		ACPM	
Strat.	Pr	Re	Pr	Re	Pr	Re	Pr	Re	Pr	Re
(1)	.12	.05	.00	.00	.00	.00	**.06**	**.01**	.68	.49
(2)	.15	.07	.00	.00	.00	.00	**.06**	**.01**	.81	.59
(3)	.19	.04	.00	.00	.00	.00	.03	.00	.69	.55
(4)	**.27**	**.11**	**.00**	**.00**	**.00**	**.00**	**.06**	**.01**	**.86**	**.76**

Table 3: *Values of Precision and Recall obtained using the four strategies and the five approaches in the year 2000 with a similarity threshold 0.1.*

	FG		LE		FCM		CPM		ACPM	
Strat.	Pr	Re	Pr	Re	Pr	Re	Pr	Re	Pr	Re
(1)	.10	.01	.00	.00	.00	.00	**.05**	**.00**	.62	.39
(2)	.17	.02	.00	.00	**.96**	**.01**	**.05**	**.00**	.76	.58
(3)	.14	.02	.00	.00	.00	.00	.00	.00	.62	.48
(4)	**.21**	**.07**	**.14**	**.02**	**.96**	**.01**	**.05**	**.00**	**.78**	**.70**

Table 4, Table 5 and Table 6 report the values of precision (Pr) and recall (Re) for the five algorithms, when the matches between clusters and debutant topics respectively have similarity equal or above 0.1, 0.15 and 0.2. As we can see, the ACPM outperforms the other four clustering algorithms both in precision and recall for all the similarity thresholds.

Table 4: *Values of Precision and Recall for the five approaches along time, at similarity value of 0.10. In bold the best results.*

	FG		LE		FCM		CPM		**ACPM**	
Years	Pr	Re	Pr	Re	Pr	Re	Pr	Re	**Pr**	**Re**
1999	.27	.11	.00	.00	.00	.00	.06	.01	**.86**	**.76**
2000	.21	.07	.14	.02	.96	.01	.05	.00	**.78**	**.70**
2001	.13	.04	.11	.01	.00	.00	.17	.00	**.77**	**.72**
2002	.14	.04	.11	.01	.00	.00	.29	.01	**.82**	**.80**
2003	.09	.02	.20	.02	.00	.00	.08	.02	**.83**	**.79**
2004	.11	.05	.06	.00	.00	.00	.00	.00	**.84**	**.68**
2005	.07	.11	.06	.01	.00	.00	.00	.00	**.71**	**.66**
2006	.01	.01	.07	.01	.00	.00	.00	.00	**.43**	**.51**
2007	.01	.08	.00	.00	.00	.00	.00	.00	**.28**	**.44**
2008	.01	.04	.00	.00	.00	.00	.00	.00	**.15**	**.33**
2009	.00	.00	.00	.00	.00	.00	.00	.00	**.09**	**.76**

Table 5: *Values of Precision and Recall for the five approaches along time, at similarity value of 0.15. In bold the best results.*

	FG		LE		FCM		CPM		**ACPM**	
Years	Pr	Re	Pr	Re	Pr	Re	Pr	Re	**Pr**	**Re**
1999	.12	.06	.00	.00	.00	.00	.03	.00	**.81**	**.65**
2000	.14	.02	.00	.00	.96	.01	.00	.00	**.75**	**.55**
2001	.04	.00	.00	.00	.00	.00	.17	.00	**.73**	**.63**
2002	.07	.01	.00	.00	.00	.00	.29	.01	**.81**	**.70**
2003	.07	.01	.20	.01	.00	.00	.08	.02	**.80**	**.73**
2004	.07	.03	.06	.00	.00	.00	.00	.00	**.74**	**.58**
2005	.05	.04	.00	.00	.00	.00	.00	.00	**.68**	**.59**
2006	.00	.00	.00	.00	.00	.00	.00	.00	**.35**	**.38**
2007	.00	.00	.00	.00	.00	.00	.00	.00	**.24**	**.36**
2008	.00	.00	.00	.00	.00	.00	.00	.00	**.14**	**.25**
2009	.00	.00	.00	.00	.00	.00	.00	.00	**.07**	**.59**

Table 6: *Values of Precision and Recall for the five approaches along time, at similarity value of 0.2. In bold the best results.*

	FG		LE		FCM		CPM		**ACPM**	
Years	Pr	Re	Pr	Re	Pr	Re	Pr	Re	**Pr**	**Re**
1999	.08	.03	.00	.00	.00	.00	.03	.00	**.64**	**.41**
2000	.03	.00	.00	.00	.00	.00	.00	.00	**.57**	**.33**
2001	.04	.00	.00	.00	.00	.00	.17	.00	**.63**	**.44**
2002	.04	.00	.00	.00	.00	.00	.07	.01	**.70**	**.50**
2003	.02	.00	.20	.01	.00	.00	.08	.02	**.70**	**.51**
2004	.04	.02	.06	.00	.00	.00	.00	.00	**.66**	**.42**
2005	.04	.02	.00	.00	.00	.00	.00	.00	**.56**	**.43**
2006	.00	.00	.00	.00	.00	.00	.00	.00	**.30**	**.28**
2007	.00	.00	.00	.00	.00	.00	.00	.00	**.15**	**.28**
2008	.00	.00	.00	.00	.00	.00	.00	.00	**.12**	**.17**
2009	.00	.00	.00	.00	.00	.00	.00	.00	**.03**	**.24**

An analysis on the clusters revealed that Leading Eigenvector, Fuzzy C-Means, and Clique Percolation Method, returned for all years a very large cluster (above 1000 topics) and several other smaller ones (on average around 6-7 topics). The Fast Greedy method yielded few large clusters per year (with at least 300 topics) and many small ones. This confirms the difficulty of standard approaches in handling evolutionary networks, which tend to be very dense since they represent every interaction that occurred during a 5 years period. Indeed, the Fast Greedy and Leading Eigenvector algorithms try to optimise a quality function called *modularity*. During this phase, they force small communities into larger ones, offering a misleading characterisation of the underlying community structure and returning very coarse-grained clusters. Fuzzy C-Means and CPM similarly fail to correctly identify coherent clusters. As an example, Table 7 shows some statistics of the evolutionary network in the year 2000, suggesting the dense structure of the network.

Table 7: *Statistics for the evolutionary network (EN) of year 2000.*

Network parameter	EN-2000	Network parameter	EN-2000
Nodes	2263	Max degree	184
Edges	13327	Diameter	20.67
Average degree	11.77	Average clustering coefficient	0.163

Figure 2 compares the dimension of the clusters generated by CMP (left) and ACMP (right) from the evolutionary network produced in the 1996-2000 interval. In this example, CPM identifies a very large cluster containing 1,124 topics and 10,939 connections and other 53 smaller clusters. Conversely, ACPM is able to better handle the density of the evolutionary networks and detects 103 clusters of comparable dimensions. The clusters produced in the other years exhibit the same trend.

Figure 3 and Figure 4 shows precision and recall obtained respectively by ACPM and Fast Greedy (which obtained the second best results) for varying values of similarity. Each coloured line represents the values of precision and recall in a particular year.

As highlighted by Table 4-6 and Figure 3-4 the values of precision and recall are much lower in most recent years. This happens for all algorithms and is due to the aforementioned fact that the number of debutant topics in the gold standard (see Table 1) significantly decrease in the last part of the analysed period, and thus many correct clusters are unable to find a match in the gold standard.

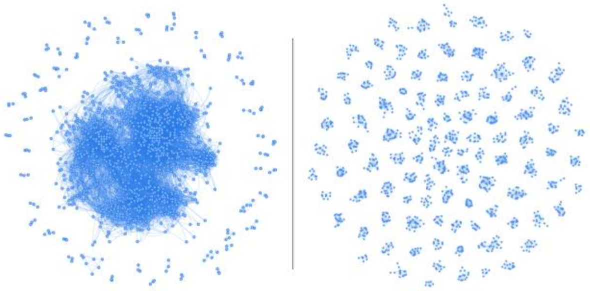
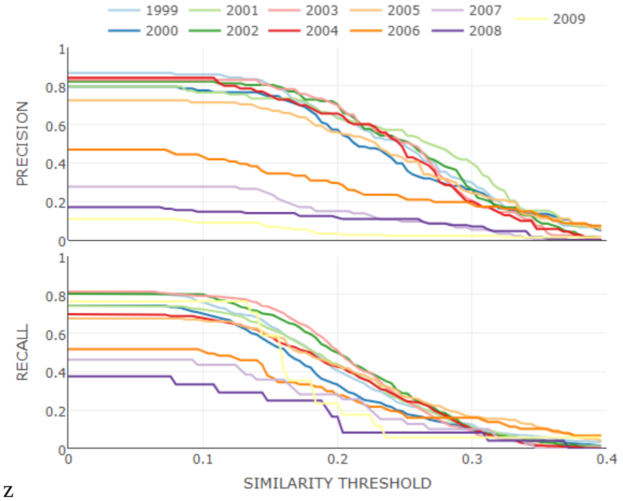


Figure 2: *On the left, we show the output of CPM in the year 2000 (54 clusters). The largest cluster counts 1,124 topics and 10,393 edges. On the right, the output of ACPM in the year 2000 (103 clusters).*



z

Figure 3: *Performance of the Advanced Clique Percolation Method.*

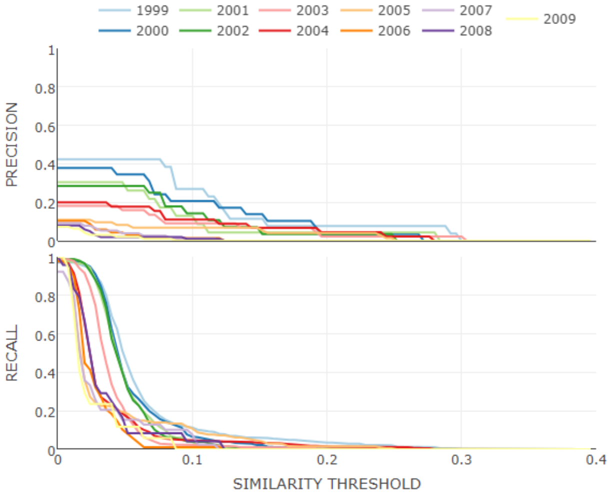


Figure 4: *Performance of Fast Greedy algorithm.*

Table 8 shows an example of a cluster produced by Augur from the topic networks in the period 1998-2002. The cluster (top-left) contains topics such as “world wide web”, “query languages”, “metadata”, “content base retrieval”, and “search engines” that exhibit a strong increment in their pace of collaboration in the period under analysis and match the ancestors of *Semantic Search*, a topic that debuted in 2003. Therefore, we considered this cluster as correctly predicting *Semantic Search* (with a similarity of 0.38). *Semantic Search* aims to improve search accuracy by understanding the contextual meaning of terms and combines research in semantic technologies and information retrieval. The topics in bold are the direct ancestors of semantic search, but, even among the other ones, we find many topics conducive to semantic search or that produced technologies adopted by this field, such as “text processing”, “electronic commerce”, “digital libraries”, and “web browser”. This is an exemplary case of the dynamics exploited by Augur, in which some topics, previously less connected, started to collaborate and moulded a novel research area that inherited their domains (e.g., “information retrieval”, “digital libraries”), formats (e.g., “xml”), software (e.g., “search engines”), and applications (e.g., “content-based retrieval”). As part of the making sense process, in Table 8 we also report the top 10 authors (top-right) and the top 5 papers (bottom) relevant to this cluster.

Table 8: *Example of output produced by Augur. Top-left, we show the cluster associated with the emergence of the semantic search topic (in bold the topics that match its ancestors). Top-right, the top 10 influential authors. At the bottom, the top 5 papers.*

Cluster	**Influential Authors**
world wide web, query languages, metadata, content-based retrieval, information retrieval, search engines, xml, information systems, information retrieval systems, multi agent systems, intelligent agents, servers, digital libraries, electronic commerce, text processing, information management, indexing, web browsers, classification	W. Bruce Croft, Dieter Fensel, Dan Suciu, William W. Cohen, Berthier Ribeiro-Neto, Clement T. Yu, James Allan, Justin Zobel, Dragomir R. Radev, Victor Vianu
Influential Papers	
- A Sheth et al. "*Managing semantic content for the Web*" (2002) - RWP Luk et al. "*A survey in indexing and searching XML documents*" (2002) - J Kahan et al. "*Annotea: An open RDF infrastructure for shared Web annotations*" (2002) - R Manmatha et al. "*Modeling score distributions for combining the outputs of search engines*" (2001) - S Dagtas et al. "*Models for motion-based video indexing and retrieval*" (2000)	

5 CONCLUSIONS

In this paper, we presented Augur, a new framework to detect research topics at the embryonic stage, i.e., when they have not yet been labelled or associated with a considerable number of publications. This approach takes advantage of the results of the study presented in Salatino, et al. [7], which showed a strong correlation between the pace of collaboration in a topic network and the emergence of new research topics, a few years later. Specifically, Augur uses the Advanced Clique Percolation Method (ACPM), a novel community detection algorithm, for analysing the dynamics between existent topics and returns clusters of topics associated with the future emergence of new research areas, which are then further characterised by providing a list of significant authors and publications.

We evaluated Augur and ACPM versus four alternative approaches on a gold standard of 1,408 debutant topics in the 2000-2011 timeframe. The results show that our approach outperforms state of the art solutions and is able to successfully identify clusters that will produce new topics in the two following years.

While these results are satisfactory, our analysis presents some limitations that we plan to address in future work. In the first instance, the gold standard does not cover well the years after 2007. We thus intend to consider more up-to-date scholarly datasets and to produce a more comprehensive version of the gold standard that could be adopted by the scholarly community to further study this task. In the second instance, the current version of Augur only focuses on the pace of collaboration between topics. This single indicator may not be enough to fully understand and detect the complex dynamics behind the creation of a topic. We thus plan to investigate other kinds of dynamics that could be associated with the emergence of new research areas, such as the patterns of collaboration between prominent authors, the dynamics of citations networks, or the change in the topic distributions of high-tier scientific venues. Finally, Augur has been tested only on the field of *Computer Science*. We believe that more work is needed to evaluate it on other disciplines.

Our aim is to produce a robust approach that can be used by researchers, policy makers, and academic editors to gain a better understanding of the dynamics of academic research and detect new research trends at the earliest possible stage.

ACKNOWLEDGMENTS

We would like to thank Springer Nature for partially funding this research and Elsevier B.V. for providing us with access to their large repositories of scholarly data.

REFERENCES

[1] P. O. Larsen and M. Von Ins, "The rate of growth in scientific publication and the decline in coverage provided by Science Citation Index," *Scientometrics,* vol. 84, pp. 575-603, 2010.

[2] L. Bolelli, Ş. Ertekin, and C. L. Giles, "Topic and trend detection in text collections using latent dirichlet allocation," in *Advances in Information Retrieval*, ed: Springer, 2009, pp. 776-780.

[3] A. Duvvuru, S. Kamarthi, and S. Sultornsanee, "Undercovering research trends: Network analysis of keywords in scholarly articles," *Computer Science and Software Engineering (JCSSE), 2012 International Joint Conference on,* pp. 265-270, 2012.

[4] Q. He, B. Chen, J. Pei, B. Qiu, P. Mitra, and L. Giles, "Detecting topic evolution in scientific literature: how can citations help?," *Proceedings of the 18th ACM conference on Information and knowledge management,* pp. 957-966, 2009 2009.

[5] Y. Wu, S. Venkatramanan, and D. M. Chiu, "Research collaboration and topic trends in Computer Science based on top active authors," *PeerJ Computer Science,* vol. 2, p. e41, 2016.

[6] T. Braun, A. P. Schubert, and R. N. Kostoff, "Growth and trends of fullerene research as reflected in its journal literature," *Chemical reviews,* vol. 100, pp. 23-38, 2000.

[7] A. A. Salatino, F. Osborne, and E. Motta, "How are topics born? Understanding the research dynamics preceding the emergence of new areas," *PeerJ Computer Science,* vol. 3, p. e119, 2017/06/19 2017.

[8] T. S. Kuhn, *The structure of scientific revolutions*: University of Chicago press, 2012.

[9] M. Cataldi, L. Di Caro, and C. Schifanella, "Emerging topic detection on twitter based on temporal and social terms evaluation," in *Proceedings of the Tenth International Workshop on Multimedia Data Mining*, 2010, p. 4.

[10] M. Oka, H. Abe, and K. Kato, "Extracting topics from weblogs through frequency segments," in *Proceedings of WWW 2006 Annual Workshop on the Weblogging Ecosystem: Aggregation, Analysis, and Dynamics*, 2006.

[11] Y.-H. Tseng, Y.-I. Lin, Y.-Y. Lee, W.-C. Hung, and C.-H. Lee, "A comparison of methods for detecting hot topics," *Scientometrics,* vol. 81, pp. 73-90, 2009.

[12] S. L. Decker, B. Aleman-Meza, D. Cameron, and I. B. Arpinar, "Detection of bursty and emerging trends towards identification of researchers at the early stage of trends," University of Georgia, 2007.

[13] C. Erten, P. J. Harding, S. G. Kobourov, K. Wampler, and G. Yee, "Exploring the computing literature using temporal graph visualization," *Electronic Imaging 2004,* pp. 45-56, 2004.

[14] P. H. Lv, G.-F. Wang, Y. Wan, J. Liu, Q. Liu, and F.-c. Ma, "Bibliometric trend analysis on global graphene research," *Scientometrics,* vol. 88, pp. 399-419, 2011.

[15] X. Sun, K. Ding, and Y. Lin, "Mapping the evolution of scientific fields based on cross-field authors," *Journal of Informetrics,* vol. 10, pp. 750-761, 2016.

[16] F. Osborne, G. Scavo, and E. Motta, "A hybrid semantic approach to building dynamic maps of research communities," in *Knowledge Engineering and Knowledge Management*, ed: Springer, 2014.

[17] J. Kleinberg, "Bursty and hierarchical structure in streams," *Data Mining and Knowledge Discovery,* vol. 7, pp. 373-397, 2003.

[18] J. C. Ho, E.-C. Saw, L. Y. Lu, and J. S. Liu, "Technological barriers and research trends in fuel cell technologies: A citation network analysis," *Technological Forecasting and Social Change,* vol. 82, pp. 66-79, 2014.

[19] H. Guo, S. Weingart, and K. Börner, "Mixed-indicators model for identifying emerging research areas," *Scientometrics,* vol. 89, pp. 421-435, 2011.

[20] Y. Jo, C. Lagoze, and C. L. Giles, "Detecting research topics via the correlation between graphs and texts," *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining,* pp. 370-379, 2007.

[21] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *J. Mach. Learn. Res.,* vol. 3, pp. 993-1022, 2003.

[22] T. Furukawa, K. Mori, K. Arino, K. Hayashi, and N. Shirakawa, "Identifying the evolutionary process of emerging technologies: A chronological network analysis of World Wide Web conference sessions," *Technological Forecasting and Social Change,* vol. 91, pp. 280-294, 2015.

[23] L. Di Caro, M. Guerzoni, M. Nuccio, and G. Siragusa, "A Bimodal Network Approach to Model Topic Dynamics," *arXiv preprint arXiv:1709.09373,* 2017.

[24] K. W. Boyack, R. Klavans, and K. Börner, "Mapping the backbone of science," *Scientometrics,* vol. 64, pp. 351-374, 2005.

[25] L. Leydesdorff, I. Rafols, and C. Chen, "Interactive overlays of journals and the measurement of interdisciplinarity on the basis of aggregated journal–journal citations," *Journal of the Association for Information Science and Technology,* vol. 64, pp. 2573-2586, 2013.

[26] F. Osborne and E. Motta, "Mining semantic relations between research areas," in *The Semantic Web–ISWC 2012*, ed: Springer, 2012.

[27] F. Osborne, A. Salatino, A. Birukou, and E. Motta, "Automatic classification of springer nature proceedings with smart topic miner," in *International Semantic Web Conference*, 2016, pp. 383-399.

[28] F. Osborne and E. Motta, "Klink-2: integrating multiple web sources to generate semantic topic networks," in *The Semantic Web–ISWC 2015*, ed: Springer, 2015, pp. 408-424.

[29] G. Palla, I. Derényi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *Nature,* vol. 435, pp. 814-818, 2005.

[30] I. Farkas, D. Ábel, G. Palla, and T. Vicsek, "Weighted network modules," *New Journal of Physics,* vol. 9, p. 180, 2007.

[31] L. C. Freeman, "Centered graphs and the structure of ego networks," *Mathematical Social Sciences,* vol. 3, pp. 291-304, 1982.