ΠΑΝΕΠΙΣΤΗΜΙΟ ΜΑΚΕΔΟΝΙΑΣ
ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΤΜΗΜΑΤΟΣ ΕΦΑΡΜΟΣΜΕΝΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ

Mining Historical Social Data for Detecting Persistent Labeled Communities

Διπλωματική Εργασία

της

Παπαδοπούλου Ευδοξίας

Θεσσαλονίκη, 07/2018

This page was left intentionally blank

ΕΞΟΡΥΞΗ ΙΣΤΟΡΙΚΩΝ ΚΟΙΝΩΝΙΚΩΝ ΔΕΔΟΜΕΝΩΝ ΓΙΑ ΤΗΝ ΑΝΙΧΝΕΥΣΗ ΕΠΙΣΗΜΑΣΜΕΝΩΝ ΚΟΙΝΟΤΗΤΩΝ ΜΕ ΔΙΑΡΚΕΙΑ ΣΤΟ ΧΡΟΝΟ

Παπαδοπούλου Ευδοξία

Πτυχίο Εφαρμοσμένης Πληροφορικής, Πανεπιστήμιο Μακεδονίας, 2013

Διπλωματική Εργασία

υποβαλλόμενη για τη μερική εκπλήρωση των απαιτήσεων του

ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΤΙΤΛΟΥ ΣΠΟΥΔΩΝ ΣΤΗΝ ΕΦΑΡΜΟΣΜΕΝΗ ΠΛΗΡΟΦΟΡΙΚΗ

Επιβλέπουσα Καθηγήτρια
Κολωνιάρη Γεωργία

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 10/07/2018

Κολωνιάρη Γεωργία	Σατρατζέμη Μαρία	Πιτουρά Ευαγγελία
.....	.....	.....

Παπαδοπούλου Ευδοξία

.....

Περίληψη

Οι γράφοι είναι μια πολύ συνηθισμένη αναπαράσταση δικτύων της πραγματικής ζωής και η εξαγωγή πληροφοριών από αυτούς έχει προκαλέσει το έντονο ενδιαφέρον της ερευνητικής κοινότητας. Η παρούσα προσέγγιση αφορά στην ανίχνευση κοινοτήτων σε δίκτυα που εξελίσσονται στο χρόνο και στον εντοπισμό αυτών των κοινοτήτων που συνεχίζουν να εμφανίζονται στην πάροδο του χρόνου και χαρακτηρίζονται από λέξεις - κλειδιά με συνεχή εμφάνιση στον εξεταζόμενο χρονικό ορίζοντα. Η προτεινόμενη προσέγγιση συνδυάζει έννοιες και τεχνικές που έχουν προταθεί για τη λύση των επιμέρους προβλημάτων που συνθέτουν το στόχο μας και αφορούν στον εντοπισμό κοινοτήτων σε μεγάλης κλίμακας δίκτυα (και κυρίως κοινωνικά δίκτυα), στη διαχείριση και μελέτη δικτύων που εξελίσσονται στο χρόνο, και τέλος στην εξαγωγή ετικετών που χαρακτηρίζουν τις κοινότητες κοινωνικών δικτύων.

Το πρώτο βήμα της προτεινόμενης μεθοδολογίας διαιρεί τον χρονικό ορίζοντα σε διακριτά χρονικά διαστήματα, και εστιάζει στην εύρεση των κοινοτήτων που υπάρχουν σε καθένα από αυτά τα χρονικά διαστήματα. Στη συνέχεια εξάγονται οι ετικέτες που χαρακτηρίζουν τις κοινότητες που βρέθηκαν. Αυτές μπορούν να είναι είτε νούμερα είτε αριθμοί, εξαρτώνται από τη φύση του δικτύου και παρέχουν σημασιολογική πληροφορία για το δίκτυο και τις επιμέρους κοινότητες. Για την ανίχνευση της ανθεκτικότητας των κοινοτήτων στο χρόνο, συγκρίνονται οι κοινότητες που έχουν εντοπιστεί σε διαδοχικά χρονικά διαστήματα έτσι ώστε να ανιχνευθεί αν και για πόσο μια κοινότητα συνεχίζει να υφίσταται στο χρόνο. Για να θεωρηθεί μια κοινότητα ότι έχει συνεχή εμφάνιση στο χρόνο πρέπει ένα ποσοστό των οντοτήτων που την αποτελούν να παραμένει σταθερό. Συγκεκριμένα, το εξεταζόμενο χρονικό διάστημα και το αρχικό στιγμιότυπο του δικτύου δίνονται σαν είσοδο σε έναν αναδρομικό αλγόριθμο και το αποτέλεσμα του είναι οι κοινότητες που συνεχίζουν να εμφανίζονται στο δοθέν χρονικό διάστημα μαζί με τις ετικέτες τους που παρουσιάζονται κάθε χρονιά.

Η προτεινόμενη μεθοδολογία εφαρμόστηκε σε ένα δίκτυο που αποτελείται από συγγραφείς ερευνητικών δημοσιεύσεων για το οποίο χρησιμοποιήθηκαν δεδομένα δημοσιεύσεων από το 1980 μέχρι το 2010 και σαν ετικέτες χρησιμοποιήθηκαν οι λέξεις των τίτλων των δημοσιεύσεων των συγγραφέων. Η προτεινόμενη μέθοδος εφαρμόστηκε για διαφορετικά χρονικά διαστήματα και με ποικίλα στιγμιότυπα του βιβλιογραφικού δικτύου σαν αρχική κατάσταση και εξάχθηκαν διάφορα συμπεράσματα για το υπό

μελέτη δίκτυο. Καταρχήν, εντοπίστηκαν λίγες ανθεκτικές κοινότητες που διατηρούνται για μεγάλα χρονικά διαστήματα, κι αυτές μειωνόταν όσο αυξανόταν τα εξεταζόμενα χρονικά διαστήματα. Τα μεγαλύτερα χρονικά διαστήματα δείχνουν ότι δεν υπάρχουν πολλές κοινότητες που συνεχίσουν να εμφανίζονται στο χρόνο, χαρακτηριζόμενες από τουλάχιστον μια σταθερή ετικέτα Επίσης μετά το 2000 οι ανθεκτικές κοινότητες είναι περισσότερες ίσως λόγο της αύξησης της δραστηριότητας της ερευνητικής κοινότητας. Επιπρόσθετα φαίνεται να υπάρχουν θέματα γενικού ενδιαφέροντος που συνεχίζουν να προσελκύουν το ενδιαφέρον της ερευνητικής κοινότητας στην πάροδο του χρόνου στον εξεταζόμενο χρονικό διάστημα αλλά εμφανίζονται σε διαφορετικές κοινότητες.

Λέξεις-Κλειδιά:
γράφος, εντοπισμός κοινοτήτων, εξέλιξη στο χρόνο, εξαγωγή ετικετών

Abstract

Graphs are a very common representation of real life networks and the information extraction from them has attracted the interest of the research community. In our approach, we have tried to identify communities that exist in time-evolution networks and the existence of clusters that persist over years and they are characterized by keywords that keep occur in the time interval which is under examination. The proposed approach involves clustering in time evolving mainly social networks and extraction of labels that characterize them, so the applied methodology combines approaches and concepts regarding clustering, time-evolve networks and labels extraction.

The first step of the proposed methodology divides the temporal horizon in discrete intervals and focuses in the identification of communities that exist in each of the intervals. Then the labels that characterize the identified communities are extracted. Those labels can be either numbers or text, they depend on network characteristics and provide semantic information about the network itself and the individual communities. To detect persistent communities in time, communities that have been identified at successive time intervals are compared to detect whether and how much a community continues to exist in time. For a community to be considered persistent through time, a percentage of its entities must remain constant. Specifically, the time interval and the initial network's snapshot are given as input in a recursive algorithm and the output is the clusters that continue their existence with their retained labels in the examined time interval. In order to consider a cluster persistent during time, a percentage of cluster's entities should remain stable.

The proposed methodology was applied in an authors' publications network and the dataset was included publications from 1980 till 2010. As labels of the communities that characterize them were considered the publications' title words of the authors that constitute them. The proposed method was applied for different time intervals and with various literature network snapshots as initial state and various conclusions were drawn for the considered network. First, few communities have been identified that have persisted for long periods of time, and they decreased as the test periods increased. Longer time intervals reveal that there are not many clusters that persist over the years and continue to be characterized by the same labels. Moreover after 2000 the identified

persistent communities are more and bigger, maybe due to the increase of the research activity community. Moreover, it occurs that there are topics of general interest that persist over time at attract the interest of the research community in the examined time window but they appear at different clusters.

Keywords:

graph, clustering, time-evolve, labels extraction

Acknowledgments

Graphs and information extraction from them was a topic that was of significant interest to me and this is the reason that led me to discuss this topic in my thesis. The co-existence of postgraduate studies and a professional career is not easy, but when there's a will there is also a way. At this point, I would like to thank my supervisor for giving me the opportunity to explore my research interest and for her guidance through all this time and of course my family who have been supportive to me throughout my life.

Contents

List of Figures

List of Tables

Acronyms

GN	Girvan and Newman algorithm
RRW	Repeated Random Walk algorithm
ESCG	Efficient Spectral Clustering on Graphs
ESCG-R	Efficient Spectral Clustering on Graphs with Regeneration
IAM	Incremental Arithmetic Mean
MDL	Minimum Description Length
DBMM	Dynamic Behavioral Mixed-membership Model
MCL	Markov Clustering Algorithm
LDA	Latent Dirichlet Allocation
SRC	Search Result Clustering Problem
SLM	Local Moving Algorithm
NLTK	Natural Language Toolkit

1 Introduction

Graphs are used for the representation of a large variety of real life networks, such as biological, social, banking or supply chain networks, in order to gain useful insight into a variety of network characteristics that can lead us, for example, in better decisions or in useful conclusions. The examination of each node separately can end up in important results but can be computationally demanding and provide us with too detailed insight, which sometimes is not the final goal. For this reason, community detection is very common in graphs, as it results in separate compartments of the network that play a similar role and their examination can give more valuable results than the examination of each separate node. The information extracted from the communities, depending on the type of the graph, can reveal groups of entities that have similar behavior and characteristics. These characteristics can be represented by labels with useful information for the communities and this case is actually of interest in the current work.

A special type of graphs is time-evolving graphs. Such graphs represent networks that change over time and the examination of those changes can reveal change time points where actually the behavior of the network has changed, common patterns, prediction of networks' future states etc. In the current work, the goal of the examination of time evolving graphs is to find communities that persist over time and keep stable some of their characteristics.

1.1 Problem Statement

Knowledge extraction from graphs can reveal important information regarding the network which is examined. Using clustering techniques to detect communities in a graph can depict the relations between the network's entities. The extracted characteristics of the identified communities can be used for various purposes. Taking into account the dimension of time, one of such purposes is to identify how the extracted characteristics evolve over time and if there are cases where they remain stable despite the communities and the overall network's evolution.

In the current work, a bibliographical network that depicts the relations between publications authors is used as a case study for examination of communities' evolution

and topics persistence. The existence of persistent communities and topics could reveal if there are clusters of authors that deal with specific topics, which are considered as the labels of the communities. As every year research papers are published, this network can be treated as a time evolving network. The evolution of clusters over the years can reveal how the relations of the authors evolve and if there are authors that keep cooperating. Moreover, if there are clusters that keep existing over time, the existence of topics that remain stable through time can reveal a community that keeps publishing regarding the same research area.

1.2 Objectives

Time-evolving graphs have recently attracted the interest of the research community as they are very common and can depict the evolution of the networks in node level and also in a more high level, such as the changes that occurs in the various communities that exist in the network. One of the main objectives of current work is to study the evolution of time-evolving networks at a community level. The other objective is to propose a methodology that can be used to detect stable communities and stable characteristics of communities that persist during time. Our motivation is that detecting communities that are continuously characterized by the same features could reveal important information regarding the examined network and assist in predicting its future states.

The network of publications authors was used as case study in order to test the proposed approach. The collaboration of the publication authors is very common and can be represented as a graph that depicts their relationships. Various valuable characteristics can be extracted from those networks, such as the journals and the conferences that specific clusters choose to publish their research work or the topics that interest the identified communities. For the purposes of this work, the topics that authors publish about are used as the characteristics of the identified communities. As a result, in the specific case study, we attempt to find if there are communities of authors that keep collaborating over time and keep publishing regarding the same topics. This could reveal dedicated research collaborations, and important research topics that persist through time, as well as groups of researchers with a more dedicated focus.

1.3 Thesis Structure

The structure of this thesis is as follows: In Section 2, related work is presented regarding clustering algorithms, techniques applied at time evolving networks and approaches for graphs that are characterized by labels. In Section 3, the necessary background for our methodology is described, where the main algorithms that are used are described. The proposed methodology follows in Section 4, and the experimental results of our case study in Section 5. Finally, in Section 6 the conclusions and future work suggestions are presented.

2 Related Work

Since graphs are a data structure used for the representation of a wide variety of systems in different areas, many different techniques have been developed in order to extract useful information from them. The aim is to understand the features and the behavior of the networks and sometimes predict the future states of the network.

An example of networks that are represented with graphs is the social networks. In this type of network there is a variety of questions that occurs and need an answer, such as the characteristics of the entities that form a community. For example if clusters' entities have similar race, age or educational achievements this could imply that entities with those characteristics similar, it is more possible to be connected. Another example is biological networks where entities can be genes, tissues etc. The identification of genes, for example, with common behavior can reveal their relatedness with an illness and their evolution over a medical treatment can reveal if this treatment is effective.

One of the most popular topics is identification of the communities, or clusters, that exist at a network. Clustering is a method used from various research areas in order to isolate entities that share the same characteristics. While the main idea remains the same, the algorithms are adapted in order to comply with the needs of each research area. Recently, the research community has shown interest in networks that change over time, such as social networks. The identification of existing patterns and how they change over time is important for understanding the network behavior and potentially for predicting its future state. The purpose of the graph is to efficiently represent a network and its characteristics and those characteristics can vary. There are types of networks that are characterized from labels, such as forum networks where a topic can characterize a whole cluster. A number of techniques is used to extract information from that category of networks and their applicability spans many types of systems. All the topics mentioned above are going to be presented extensively below.

2.1 Graph Clustering

Clustering data is a fundamental task in the area of machine learning. Given a set of data instances, the aim is to group them in a meaningful way given a particular domain. Elements assigned at the same cluster are connected in a predefined sense. There is a variety of measures that are used for clustering identification, such distance similarity

measures, adjacency-based measures, density measures and cut-based measures (Schaeffer, 2007). Generally, clustering nodes is a useful technique for deriving useful knowledge from the graph database. At first glance, it might seem that the initial problem in graph clustering is the definition of community, which, most of the time, depends on the specific system that is examined, but in most cases, an a priori definition is not needed, since communities arise according to the algorithm's output.

The purpose of a reliable algorithm is to identify good clusters. The definition of a cluster's quality is not an easy task, as some specific properties should be satisfied. Since different algorithms can produce different clustering outputs, one should be able to identify meaningful partitions by using a quantitative criterion. For this goal, cluster fitness functions are used that can estimate the quality of the output communities.

In order for the clustering algorithms to identify the different partitions, quality criteria are used. These criteria can be computed with two different ways, either first compute the predefined quality values for all vertices and based on them, assign the vertices into clusters, or compute a quality function for all the possible clusters and then rank them in order to find and select the best clustering option.

There are networks that are extremely large and their clustering requires the utilization of many computational resources. In these networks, the clustering of the whole graph (global clustering) is sometimes replaced by local clustering where smallest parts of the graph are processed independently. In the case of global clustering, all vertices of the initial graph are assigned to a cluster with the given algorithm, whereas in local clustering, only a subset of vertices ends up with cluster assignments.

It is important to mention that graph clustering has the same problems that the clustering techniques generally have. For example, it is rare to know exactly the number of clusters that exist at the graph and how big or small those clusters are. As a result, for different approaches we need to make some assumptions about this kind of information.

Since graph clustering is a topic that attracts the interest of the research community, various approaches have been proposed. Of course the main idea of those approaches is the same as the classic clustering approaches that appear in machine learning literature, but the implementation has been adjusted for the graphs and their specific characteristics. The fundamental graph clustering approaches are going to be presented at the next chapters (Schaeffer, 2007) (Fortunato, 2010).

2.1.1 Hierarchical Clustering

Some kinds of graphs, for example social networks, have a hierarchical structure as for example a social network which depicts neighborhoods, which are consist of schools, schools of classes, classes of teams and so on. This means that each cluster is composed of smaller clusters and so forth. If this is the case, it makes sense to use hierarchical clustering techniques which unravel the multilevel structure of the graph. It is important to define a measure for computing the similarity between the vertices and the final goal is to cluster together the vertices with high similarity. The output of a hierarchical algorithm is a dendrogram with the clusters and their sub-clusters as children. The root of the tree is a cluster which contains the whole of the data set. The algorithm stops when the resulted graph partitioning fulfills a predefined quality function, such as modularity. (Fortunato, 2010) (Schaeffer, 2007).

A hierarchical approach has been proposed by Basuchowdhuri et al (Basuchwdhuri, 2010) who make a mathematical formulation of Granovetter's hypothesis (Granovetter, 1973) for this purpose. Granovetter states that nodes that belong to the same cluster share stronger links than those that belong to different clusters. Except using their mathematical formulation in order to identify communities, they evaluate the resulted clusters using clustering coefficients.

A very popular hierarchical algorithm regarding graph community detection is the Girvan – Newman (GN) algorithm (Girvan, 2002). The GN algorithm tries to find the smallest number of central edges to communities and not the most central and this results in removing the less central edge from the initial graph, rather than adding the strongest one. For their approach they used the "betweenness" centrality proposed by Freeman (Freeman, 1977). "Betweeness" is based on the term of shortest paths and for every vertex of the graph is the number of shortest paths of all the pair of vertixes that run along this vertex. It is actually the tendency of a vertex to be more central than the others.

The same concept of "betweenness" and edge removal was proposed by Steve Gregory (Gregory, 2007) (Gregory, 2008) who proposed CONGA and CONGO algorithms. The difference between CONGA and GN is that Gregory duplicates the removed edge and as a result it is capable to find overlapping communities. CONGO is an improved version of CONGA and it is much faster as it uses local "betweenness".

2.1.2 Random Walk Based Methods

Random walks is another technique used in order to find graph communities. The inputs to this technique are the graph, along with the starting point of the algorithm and at each step it moves to a neighbor based on probabilities which calculation is independent of time and previous states and only depends in the current state. They can be depicted with a transition matrix. Clusters are identified as the random walker spends time in the cluster due to the big number of edges between nodes and the various paths that could be selected (Fortunato, 2010). Random walks are highly related with finite Markov chains, since if the graph is directed and weighted, the random walk is actually a Markov chain (Lovasz, 1993).

Macropol et al. (Macropol, 2009) proposed a random walk algorithm with restarts, the Repeated Random Walk algorithm (RRW), which finds clusters based on edges' proximity scores. An initial cluster of size 1 is expanded in order to be part of the cluster of the closest node. This process continues until the clusters are below a predefined early cut off percentage or when clusters with size equals k have been obtained. This algorithm is capable of identifying overlapping communities, but there is a threshold regarding the overlap. RRW was initially applied in genes network and Cai et al. (Cai, 2010) tested the algorithm in social networks in order to investigate its performance and it came up that RRW compared with other algorithms ended up with higher precision but lower modularity.

Pons et al. (Pons, 2005) propose a new measure, which is based on the idea of random walks and defines the closeness of graph vertices. Based on this measure, they proposed Walktrap which is actually a hierarchical clustering algorithm and the decision of merging communities is based on Ward's method (Ward, 1963). An important drawback of this method is that it demands a lot of memory.

2.1.3 Spectral Clustering

In this specific approach, eigenvectors of matrices are used for identification of clusters. Dimensionality reduction is conducted based on eigenvalues and then a common algorithm, like k-means, is used for clustering the resulting points. The advantage of spectral clustering is that it can discriminate between data points that k-means cannot directly distinguish. The main drawback of the spectral method is that the calculation of the adjacent matrix is computationally demanding (Fortunato, 2010).

Liu et al. (Liu, 2013) propose an algorithm for clustering large-scale graphs that are computationally demanding. Their basic idea is to create "supernodes" which are linked with the nodes of the initial graph. As a next step, k-means clustering is applied on the graph created by the "supernodes". They name their algorithm Efficient Spectral Clustering on Graphs (ESCG). In order to improve the results they have implemented an improved version of ESCG, Efficient Spectral Clustering on Graphs with Regeneration (ESCG-R). In the first approach, the links between "supernodes" and original nodes are selected randomly. In the second approach, those links are corrected by the clustering results.

Another approach is White et al's clustering (White, 2005), who optimize the Q objective function, first proposed by Newman and Girvan (Newman, 2004), which indicates the quality of the generated clustering. They kept the main idea of the Q function and changed the implementation in order to make it suitable for spectral clustering. Based on their implementation, they propose two spectral algorithms. The first algorithm attempts to find the global optimum of the Q function, whereas the second one attempts to make local improvements. In order to produce the clusters they use k-means.

2.1.4 Partitional Clustering

Another common clustering method for graphs is partitional clustering. In this approach the number of clusters is predefined. A measure to calculate the distance between points is defined and the final aim of the approach is to maximize or minimize the cost of the function regarding distances of the graph points. The maximization or the minimization of the function depends on the similarity measure of the points. One of the most common partitional algorithms is k-means clustering algorithm (Fortunato, 2010).

Jain et al. (Jain, 2009) extend Elkan's method (Elkan, 2003), which is a k-means algorithm applied at vector level, for graphs. At their approach, they replace the Euclidean distance that is used at the classical k-means algorithm, with geometric graph distance and they use incremental arithmetic mean (IAM) as the sample mean. They retain the "triangle inequality" from Elkan's algorithm in order to reduce the computational resources required.

Ferrer et al. (Ferrer, 2009) propose a k-means clustering that uses the concept of generalized median graph (Jiang, 2001) and since the calculation of median graph is a

demanding process regarding time and resources, they solve this problem by proposing a new approach. Each graph is embedded into a vector space and they use the two closest points and the weighted mean of the pair of graphs that correspond to those points.

2.2 Graphs and Time Evolution

There are cases of dynamic networks that evolve over time and their mining can give beneficial insight regarding their characteristics. The observation of how communities and features of the graphs change during time can result in important outcomes for various research areas. There are various types of networks that change over time, such as social and communication networks, and the research community recently started to pay attention to this type of graphs. A part of the literature that deals with time evolving graphs is going to be presented below and it is divided into works which use clustering in their approaches and works which do not evolve clustering in their methodologies. (Leskovec, 2005).

2.2.1 Clustering Approaches

A proposed scheme regarding time-evolving graphs is GraphScope (Sun, 2007). The algorithm does not require any input from the user as it use Minimum Description Length (MDL) in order to make the decisions needed for the algorithm. MDL is an information theoretic technique which implies that the model that compresses the data better and describes it with less information is the one that provides the best description. The general idea of the approach is to find the communities in different snapshots of the graph. As long as the detected communities have similar description, they are grouped together in order to form a time segment. If a new snapshot of the graph does not have similar description with the communities that have formed the previous segment, then this is considered as a change point and a new segment starts. The main differences between this approach and the proposed one is that GraphScope is applied on streaming data, user input it is not needed and it is mainly considers the bipartite graphs.

Aggarwal et al. (Aggarwal, 2005) introduce an online analysis framework with main purpose to give the user the possibility to make queries in order to identify communities in a dataset with a common characteristic over time or that present a change in a specific time interval. This process cannot be conducted online, and for this reason they have separated the offline part, which executes the exploratory algorithms for the

community detection, from the online part which creates summaries of the data. The clustering algorithm parameters of time interval and number of clusters are user defined. The main difference from the proposed methodology is that Aggarwal et al's approach is more generic regarding the queries that can answer and moreover those queries are conducted online by the user whereas we conduct the whole approach offline.

Semertzidis et al. (Semertzidis, 2016) in another work propose a method in order to find most densely connected nodes in the whole graph history or in a part of it. Those problems are called the Best Friends For Ever (BFF) problem and the On-Off BFF (O^2BFF) problem respectively and also variants of those are presented. They propose different algorithms based on the problems complexity. For example, the FINDBFF algorithm, which is "greedy-like" and it is proposed for BFF problem and the Iterative (ITR) FINDO^2BFF algorithm which is proposed for O^2BFF problem. A main difference between this approach and the proposed one is that they also examine the case where the densely connected node appear only in k snapshots of the network and moreover they do not require any additional criterion to persist.

2.2.2 Non-clustering Approaches

A proposed approach that deals with time-evolving graphs is WebRelievo (Toyoda, 2005), which is actually a tool that visualizes a time-evolving network and analyzes the changes over time. More specifically, it visualizes various web structure snapshots where each graph node is a web page and each edge depicts the relationship between different web pages. WebRelievo retains the position of the nodes over different snapshots in order for the user to understand the evolution of different web page clusters.

An analysis of the Flickr network is conducted by Cha et al. (Cha, 2009) in order to mine useful information regarding pace, width and way of information spreading. In order to collect the necessary data regarding the network's state, they started from a user and followed all friends' links. Then, they obtain the changes of the friends' links over time, tracking the links daily in order to obtain any changes at their friendship relationships. In order to get the necessary data to investigate the way of the information spread, researches get the photos that the users marked as favorite and the time point that this action occurred. This research yielded remarkable insights on how Flickr works.

An approach that focuses on nodes, their characteristics and how they change over time is proposed by Rossi et al. (Rossi, 2013) and it is actually a dynamic behavioral

mixed-membership model (DBMM). The model is scalable, does not require input from the user and it is not data-driven. Rossi's model assigns nodes to roles and not to communities. The difference is that nodes that belong to the same role share common structure whereas typical communities are characterized by many connections between the nodes. The main purpose of the model is to identify the nodes' patterns, predict changes that will take place in the future and find changes over time that are unusual by discovering features over different timestamps of the graph and learning the different roles of the nodes.

Semertzidis et al. (Semertzidis, 2016) in their work, try to provide a definition for the different types of historical queries (historical graph queries, historical time queries and historical top k queries) that can be performed in a time-evolving graph. Moreover, they propose single edge or multiple edges representation of network snapshots and they present two algorithms that process the queries asked in the graph. The selection of the algorithm depends on the representation of the network as for single edge a proposed by the authors' version of BF Straversal method is used, whereas for multiple-edge representation initial BF Straversal method is used.

2.3 Graphs and Labeled Communities

There are many applications in which one of the characteristics of the network are labels and those should be included at the applied methodology in order to extract the necessary information. Labels can be either numbers or text, which can vary from words to whole sentences. Labels' extraction can be very useful in order to find information regarding network, such as topics that are mentioned, interests of the various networks' entities and characteristics. An example is to cluster the users of a social network and investigate if there are topics that interest the users belonging to specific clusters. Some approaches that deal with labels and clustering are going to be presented below.

An approach that creates topic clusters from comments in online news is proposed by Aker et al. (Aker, 2016). The clustering of the comments is conducted by the Markov Clustering Algorithm (MCL), which does not require the number of the clusters as input. The labels of the generated clusters are extracted using Latent Dirichlet Allocation (LDA) and as a final step, clusters are labeled using DBPedia.

An approach that tries to find a solution for the Search Result Clustering Problem (SRC), is the one proposed by Scaiella et al. (Scaiella, 2012). SRC groups topically

together the results that search engines return and assign to those groups phrases that describe them. In their approach, they use the TAGME annotator for the search results in order to find the relevant text snippets, where each snippet is represented as a graph of topics that are computed by using the Wikipedia-graph linkage. As a next step, the topic-to-topic and topic-to-snippet similarities are calculated and they are used as an input to the algorithm, which uses the spectral properties in order to find the different clusters that exists

Lim et al. (Lim, 2014) (Lim, 2017) proposed a framework for graph clustering where the graphs are characterize by labels. In fact the connections between the nodes are labels, which can be numbers, words or more complicated whole phrases. The labels are members of a set and based on them the algorithm ends up with the various clusters that exist in the network and the clustering identification is based on the optimization of a weighted objective created from the various network's labels.

TimeFall is an algorithm (Ferlez, 2008) that can perform analysis in time evolving graphs regarding time and labels. This approach can find the existing clusters, track their evolution and identify time points that the evolution of the clusters change. The algorithm creates an adjacency matrix of the graph for the different timestamps and using Cross Associations (CA), the algorithm finds the different communities. Those communities form the input of Minimum Description Length (MDL) which was also mentioned in Sun et. al approach previously (Sun, 2007), that identifies the time evolution of those communities. The algorithm does not require input from the user. An important difference between this methodology and our proposed approach is that they cluster words and not authors using their adjacency matrices and their methodology concerns bipartite graphs.

3 Background

In this section some basic techniques and terminologies will be presented that they are important for the understanding of the proposed methodology.

3.1 Graphs and Communities

Graphs are data structures that are used in order to model relations between objects. There are different kind of graphs and some of them are going to be presented below, but in its most common form, a graph G is a pair G = (V, E), where V is a finite non-empty set of vertices and E is the set of edges, which actually consists of unordered pairs of vertices.

Simple graphs are graphs with no loops or multiple edges. The mathematical representation is the same as the above. The only difference with the generic definition that was presented is that E consists of unordered distinct pairs of vertices. Another type of graph is a directed graph, where the edges are characterized by a direction. In more formal terms, a directed graph G = (V, A), where V are as at the above case the vertices and A are directed edges that connect the vertices. Directed graphs are important for cases where the relationships between objects include the term of direction, like networks that represent road networks, where streets' direction can be depicted (Wilson, 1996). Additionally, labeled graphs could be mentioned as another type of graphs. They are divided into vertex-labeled and edge-labeled graphs and the vertexes or the labels respectively are characterized by labels that can vary and depend on the network that is under investigation (Hedge, 2012).

Graphs are composed from nodes and their connections and commonly nodes are grouped together in order to form communities. They are actually nodes that are densely connected and divide the network into smaller subareas. Nodes of those subareas interact a lot among them and this indicates that maybe they share common characteristics and depending on the network reveals valuable features about it. Of course the community/cluster identification process is not conducted arbitrarily but based on specific criteria such as modularity, that are defined by the applied methodology.

3.2 Graphs and Snapshots

Not all networks are static; there are also types of networks that evolve over time, like the graph modeling the interactions between users of various social networks. This type of networks has attracted the interest of the research community, which tries to find answers regarding the different patterns that occur, evolve, remain stable or disappear during time and sometimes create a model from all this information in order to predict their future state, find interesting time points where the behavior of the network changed etc.

To achieve that, researchers capture snapshots of different network time points and try to extract interesting information regarding each of those static states of the network. The selected intervals of the snapshots and their time duration in order for them to be representative of the evolving trend and informative, depends on the domain.

In a more formal way, the collection of the different graph snapshots of the evolving network can be represented as $G = \{G_1, G_2, \ldots, G_N\}$, where N are the different snapshots of the network that depict it's evolution over time. There are various research works that deal with time-evolving graphs, which indicates that research community is active regarding this type of graphs (Sun, 2007) (Leskovec, 2005) (Hyland-Wood, 2005) (Montgolfier, 2011), (Chen, 2008) (Asur, 2007).

3.3 Graphs and Labels

The information that can be extracted from a network varies and depends on the research objective. A very common question that has to be answered relates to the communities that exist in the network. Sometimes the question pertains not only to the existing communities but also the labels that characterize those communities. Those labels can describe features that characterize the obtained clusters, such as topics of interests or age range. The labels can vary and depend on the characteristics of the communities that are valuable for the research purposes (Aker, 2016).

To attain this information, one approach is to first find the communities and then find the labels that describe those communities. Another approach is to find the labels of interest and then assign the nodes of the network graph to the different labels. The labels can be separate words, word snippets, sentences, numbers or whatever is considered valuable for a particular domain. (Honch, 2011).

An example use case is a recommendation system for the users of a movie rental site. Except for finding the different communities of users, the system has to know the type of movies that those users prefer. This information will be used by the system to suggest movies that fit users' preferences and can be obtained by labeling the networks' communities with the movies' details that characterize them for example. A number of approaches have been conducted regarding labeles graphs in order to propose solutions in some of their problems that need answers (Leal, 2013) (Honch, 2011) (Aker, 2016).

3.4 Clustering Algorithms

There is a variety of clustering algorithms as was shown in previous sections. The choice of the clustering algorithm that will be used depends on the problem that is going to be investigated and its characteristics. For this thesis, the Louvain algorithm (Blondel, 2008) and some of its variations are selected. The reason is that is a fairly efficient, well-known algorithm which has been used successfully at many types of networks. The main algorithm and its extensions are presented in this section, but first an important and quite common clustering quality criterion for graphs, modularity, is presented, which is also used by the selected Louvain algorithm.

Modularity is a widely used criterion for defining the quality of the resulted clusters from a clustering algorithm. It means that connections inside a cluster should be dense and connections between different communities should be sparse. It was proposed by Newman et al. (Newman, 2004) and is used for the quality calculation of communities that are produced by clustering algorithms. The modularity function can be written as

$$Q = \frac{1}{2m} \sum_{i,j} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta(c_i, c_j)$$

where c_i indicates the cluster to which node i is assigned and A_{ij} indicates the presence of an edge that connects nodes i and j

$$k_i = \sum_j A_{ij}$$

is the degree of i node and,

$$m = \frac{1}{2}\sum_{i,j} A_{ij}$$

indicates how many edges exist in the graph. Finally, the function $\delta(c_i, c_j)$ denotes if nodes i and j are part of the same cluster. If all nodes belong in the same group, then the function is 1 for all i,j. The higher the modularity function, the better the resulting communities. Apart from the standard modularity function there are many variations that have been proposed. One is that of Traag et. al (Traag, 2011) that tries to reduce the resolution-limit, which describes the failure of detecting small communities in large networks and it is one of the known drawbacks of the standard modularity function.

3.4.1 Louvain Algorithm

The classical Louvain algorithm was proposed by Blondel et al. (Blondel, 2008) and it is a very popular algorithm regarding graph clustering. It is actually a greedy optimization method and its implementation is easy.

Briefly, in the initial state of the Louvain algorithm, each node of the graph constitutes a cluster. In the next iterations, the local moving heuristic changes an individual node's cluster assignment to optimize modularity function. The local moving heuristic (Barber, 2009) (Blondel, 2008) randomly selects the nodes of the graph and changes their assignment to different clusters so as to increase modularity if possible. In case no other improvement can be achieved, the algorithm stops its execution. This heuristic is used a lot in the research community, mainly because it can be implemented efficiently.

When the modularity function is optimized, a reduced network is created. Each community of the graph from the previous iteration (original graph in the first iteration) is merged into one node into the reduced graph. The weight of an edge between two nodes in a reduced network is equal to the sum of the weights between the nodes of the two corresponding initial communities. The algorithm starts again by assigning all the nodes of the reduced network to their own clusters and applies the local moving heuristic in order to create another reduced network. This process of merging communities continues until the network cannot be reduced any further.

3.4.2 Louvain with Multilevel Refinement

Rotta et al (Rotta, 2011) proposed an improved version of classic Louvain algorithm – the Louvain algorithm with refinement. In the final graph of the Louvain algorithm, the clustering cannot be improved by additionally merging the communities, but the resulting output can potentially be improved by changing the community assignment of individual nodes. This is not supported by the classical Louvain algorithm but it is supported by the Louvain algorithm with refinement. In other words, the difference between the Louvain algorithm and the Louvain algorithm with refinement is that the second one uses the local moving heuristic for creating the initial community structure, as well as improving the final one.

3.4.3 SLM Algorithm

Another improvement of the Louvain algorithm is the Smart Local Moving Algorithm (SLM) (Waltman, 2013). The difference from the Louvain algorithm is that it identifies locally optimal solutions by splitting communities and moving sets of nodes from one community to another. The solution is optimized in the same way as the Louvain algorithm with multilevel refinement by moving individual nodes across communities. According to results found in literature, SLM outperforms the Louvain algorithm and the Louvain algorithm with multilevel refinement for large networks, but achieves the same results as those two algorithms for small and medium networks.

4 Methodology

The main purpose of this thesis is to detect communities that persist over time in time evolving graphs and labels that characterize them that keep appearing during the same time period. The proposed methodology is presented in this section with its details, such as the communities' identification and label extraction processes and the algorithm that identifies the persistent communities over time with their labels that continue to characterize them. Moreover the definition regarding basic terms are presented, such as the network snapshot and the time persistence definition.

In order to test the proposed methodology, we applied it as a case study in a time evolving bibliographical network. The nodes of the graph represent the authors of the publications, and two nodes are connected if the corresponding authors have a common publication together.

4.1 Graph Clustering

To model evolution through time, time is split into disjoint time-intervals and a graph snapshot that reflects the state of the network at that time interval is constructed. A clustering algorithm is applied on every snapshot's graph in order to identify the different communities that exist. The communities' identification is the first step of the approach. Regarding our case study, we split the time period into years and construct a separate snapshot for each year. Two nodes in the graph snapshot are connected if the corresponding authors have a common publication for the particular year of the snapshot.

As clustering methods, the three different algorithms that were presented above are selected, namely Louvain, Louvain with multilevel refinement and SLM. The reason is that the Louvain algorithm is a well known algorithm with satisfying performance. Its variations are also examined, in order to investigate whether any further improvement can be achieved. Their implementations were found at Ludo Waltman's and Nees Jan van Eck's site (http://www.ludowaltman.nl/slm/), who provide a command line executable jar file with the algorithms' implementations. Various parameterizations are supported for the clustering algorithms, which are presented in detail in Table 4-1.

Regarding the modularity function, the standard one refers to the one proposed by Newman et al. (Newman, 2004) and the alternative refers to the one proposed by Traag et al. (Traag, 2011). The number of random starts defines the number of executions of the optimization algorithm and the resolution parameter defines the granularity level at

which communities are identified. The random seed number generator refers to the local moving heuristic and defines the order that graph nodes are visited. Finally, the number of iterations defined per random start is self-explanatory. The output of each iteration is used as input for the next iteration in order to improve the result. For example, SLM algorithm can be selected as the clustering algorithm and can be executed with 10 runs of 10 iterations, trying to optimize the standard function with resolution 1.0, whereas the seed of the random number generator can be equal to 0.

Table 4-1 Clustering parameters

modularity_function	Modularity function (1 = standard; 2 = alternative)
resolution_parameter	Value of the resolution parameter
optimization_algorithm	Algorithm for modularity optimization (1 = original Louvain algorithm; 2 = Louvain algorithm with multilevel refinement; 3 = SLM algorithm)
n_random_starts	Number of random starts
n_iterations	Number of iterations per random start
random_seed	Seed of the random number generator

The input of the algorithms is a list of pair-wise graph nodes that are connected and in case that the graph is weighted, the weight of the nodes is also included. Its output lists all the nodes and the ID of the communities they are assigned to.

4.2 Community Labels

After the identification of the communities that exist at the different snapshots of the network, the next step is to define their labels. The labels depend on the network which is under examination and it's characteristics that are of interest and can be numbers or text. For example communities' entities interests in the case of a social network or communities expressed genes in the case of a biological network can be considered as their labels. For our case study, the titles of the publications are used to

derive the communities' labels. In this section, we present the tokenization and stemming process we applied to extract the labels for our clusters.

4.2.1 Tokenization/Stemming

Tokenization and stemming are basic terms in the Natural Language Processing area. The main purpose is to find the various tokens that comprise a text as a first step and next end up with their root. The processing of the publication titles is a necessary step in order to find the words that comprise the titles, make them consistent and use them as labels of the different clusters.

Each title is split into its words and stopwords are excluded. Stopwords are common language words that are usually excluded from the final results, as they add no semantic value regarding the terms that are included in the initial sentence. Examples of stopwords include: and, me, I, any, etc. There is not a specific list with stopwords that should be included at this type of preprocessing, so for the purposes of this thesis, NLTK's (Natural Language Toolkit) list is used.

After the tokenization of the title, the next step is stemming. Stemming is the process that tries to distinguish the different forms of the same word. For example, we can have the words "organization", "organized" and "organize" that have the same root. The goal of the stemming process is to retrieve the root of the words. Various algorithms exist and perform stemming but the most common one is Porter's algorithm and it is the one used in this thesis. Porter's algorithm has 5 processing steps and the basic idea of the algorithm is that each suffix is composed of smaller suffixes. The algorithm contains a predefined list of suffixes and for every suffix to be removed from the initial word a criterion should be fulfilled (Porter, 1980). The java implementation of the Porter's algorithm that is used was found at github (https://gist.github.com/ldclakmal/667d8ecb620a0cce7d3dedae80a2c013)

4.3 Time Stable Clusters Identification

After the communities and their labels in the different snapshots have been identified the next step is to find whether there are clusters that persist through time and keep being characterized by the same labels. As communities of each snapshot we consider the clusters found from the clustering algorithm and as labels their characteristics that are important for further investigation and conclusions extraction.

The definition of what is considered as persistent community can vary depending on the approach. In our work, we define a community as stable between two consecutive snapshots G_i and G_{i+1}, if at least t% of the entities (nodes) that belong in the community in snapshot G_i, also belong in the community at snapshot G_{i+1}. We call this percentage t of required persistent nodes, stability threshold.

The approach that is followed regarding the entities of the persistent clusters is keeping a pool with all the entities that are added through the years in the initial cluster without excluding those that potentially disappear from the cluster as time progresses. As a result, the threshold regarding the percentage of entities remaining constant is tested against the constructed pool. This is done to better track the evolution of the original community and to better assimilate the nature of social networks that are mostly incrementally grown. For instance, setting it to 20% and starting from the first snapshot (n), let us consider that we have a community that contains 5 entities. The second snapshot (n+1) has a cluster that contains 7 entities where the 3 are the same that exist in the starting snapshot community. 3 of the 5 entities of the initial community keep existing, which is above the 20% we demand and we consider that the cluster keep existing in the second snapshot. At the 5 entities of the n year cluster, the 4 new of the n+1 snapshot cluster are added and at those 9 entities it is examined if there is a cluster in the n+2 snapshot which is continuation of the previous. Regarding labels we retain only those that keep appearing from year to year. For example if a cluster in the first snapshot is characterized by 3 labels and only 1 of them exist in the cluster which is considered as its evolution in the n+1 snapshot, then only this one label continues in the next snapshot (n+2).

Our proposed approach examines the persistence of communities and stable labels by considering specific time windows. In particular, it accepts as input a time window, defined by a starting time point and an ending time point, the stability threshold, the clusters of each snapshot in the specified time window and the labels that characterize them. The output of the algorithm is the clusters that persist from the starting till the ending time point and their labels that continue to appear in the predefined time window. Below the steps of the algorithm for every time window are presented.

Input: starting time point
ending time point
stability threshold
snapshots' communities
communities' labels

Output: the clusters that persist
clusters' labels that continue to appear

Step 1: Identification of communities with common entities between n and n+1 snapshot

Step 2: We examine if the number of common entities is above or equal the given threshold (1 condition)

Step 3: We examine if there are common labels that characterize the clusters that fulfill condition1 (condition 2)

Step 4: If condition 1 and 2 are fulfilled, the new entities of the n+1 snapshot cluster are added in the pool of the entities of the cluster that continues, and the common labels are kept.

The above steps are repeated with input the n+1,n+2, … years and the clusters that continue existing from previous years with their common labels till the ending year is reached

Figure 4-1 Algorithm's steps

5 Case Study

The proposed methodology is applied in a network of publications in order to verify it through experiments. The purpose is to find the communities of authors that keep collaborating over time and keep publishing research papers regarding the same topic. This could reveal if there are topics that concern continuously specific parts of the research community which is dedicated to a specific research area.

5.1 Dataset

To create the publication authors' network, the dataset is downloaded from the dblp site (https://dblp.uni-trier.de/). This xml file contains information pertaining to the authors of a publication, the title, the journal, the volume, the pages, the year of publication etc. For this thesis, we are interested in the information regarding the authors, the title and the year of publication. It is important to have data for a long period of time as is it is necessary for the time parameter to be involved at the experiments so as to attain information about persistence through the network's evolution. For this reason, this dataset is suitable, as it contains all the necessary information for research papers published from 1963 till 2016.

Based on this dataset, we construct a co-authorship network, where the nodes of the graph represent the authors, and two nodes are connected if the corresponding authors they represent have a joint publication. For our experiments, publications from year 1980 to 2010 are used, as prior to 1980, the number of publications is not sufficient enough and we consider that 30 years are a satisfactory time window in order to investigate how communities evolve over time. We construct a graph snapshot for each year, illustrating the publications and corresponding authors (nodes) and connections (edges) of the respective year. In order to reduce the algorithm's computational requirements, the obtained xml file is preprocessed to generate a different xml file for every year. The idea is to isolate the information needed for every different network snapshot.

5.2 Parameters Tuning

In order to find out whether there are clusters that persist through years along with the labels that characterize them, different configurations have been tested. All three clustering algorithms, the Louvain algorithm, the Louvain algorithm with multilevel

refinement and the SLM algorithm were compared. All algorithms resulted in the same clusters that persist at a given time window and are characterized by at least one common label. For this reason, the Louvain algorithm with refinement is selected for the final experiments as the selection of the algorithm did not have any effect regarding the final results.

Moreover, different resolution parameters were tested (1.0 and 2.0) to verify if the persistent clusters would change but this did not happen and as a result the 1.0 resolution is selected. Finally, the Louvain algorithm with multilevel refinement was tested with the two different modularity functions (standard and alternative). The alternative function resulted in clusters that in their majority consisted of a single author only, whereas the standard function resulted in bigger clusters. As such, for the final results the standard modularity function is selected. For the parameters of random starts, number of iterations per random start and the seed of the random number generator, 10, 10 and 0 are selected respectively.

Regarding the stability threshold 20% is selected. Increasing the threshold did not have any observable effect in the evolution of clusters through time, since all authors that appear initially, persist through time. Moreover, it is observed that some clusters' size increases rapidly at particular time points, exhibiting peaks, so we selected a small threshold in order to not lose the evolution of those cases as well.

5.3 Experimental Results

This section contains the experimental results of the case study regarding the number of clusters, numbers of authors comprising the clusters and the labels for the examined time intervals.

5.3.1 Clusters' Number

We evaluate our approach with different time windows between 1980 and 2010 with the window resolution (size) ranging from 2 to 6 years. Our goal is to detect with different persistence and investigate whether a specific time window presents any interesting results. The stride of the window is 1 year. To this end we measure the number of identified clusters that persist over each time window.

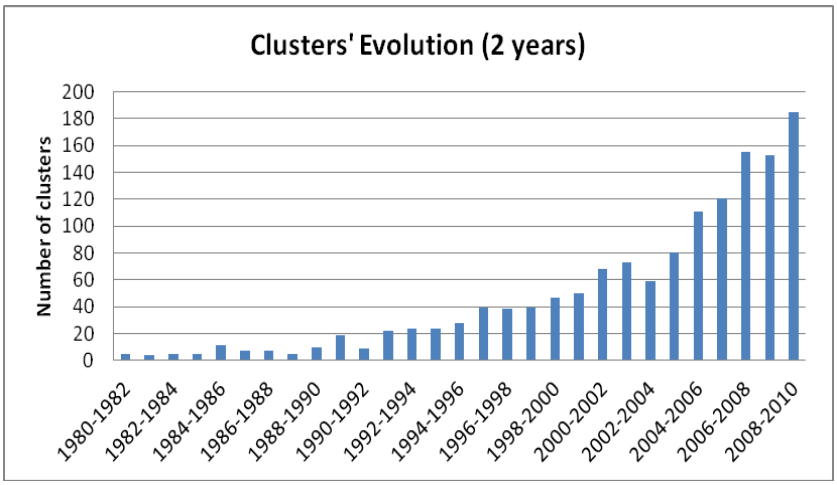

Figure 5-1 Cluster evolution (2-year window)

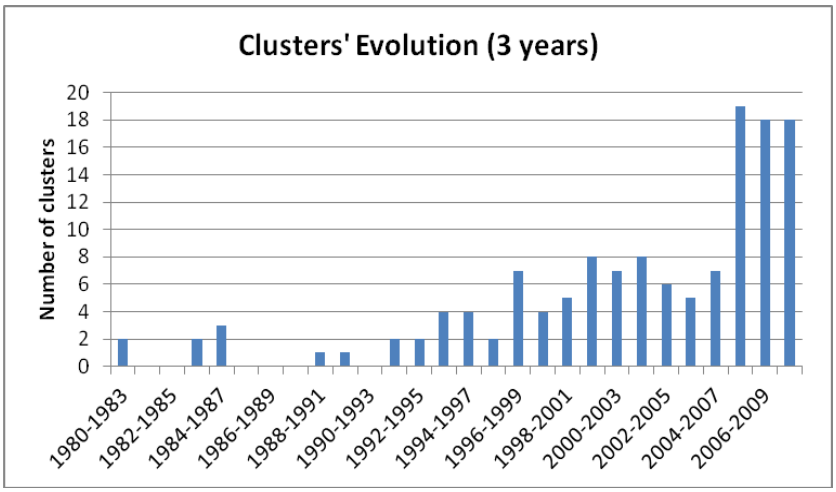

Figure 5-2 Cluster evolution (3-year window)

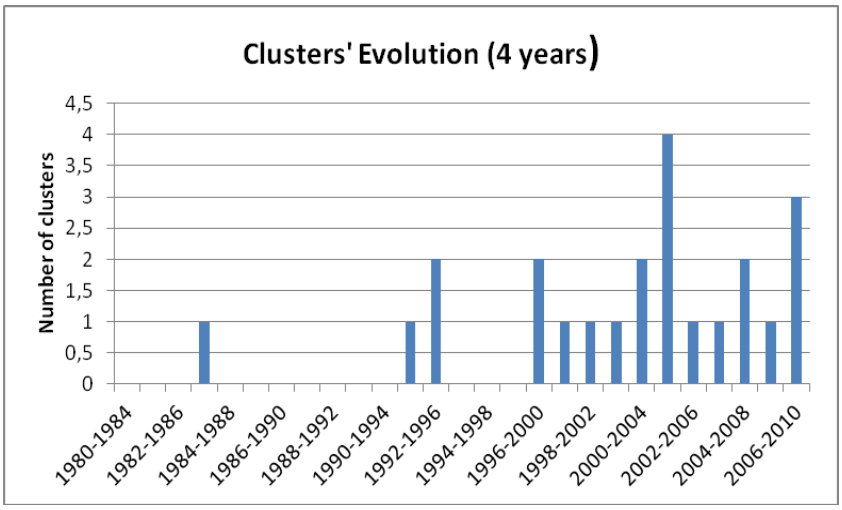

Figure 5-3 Cluster evolution (4-year window)

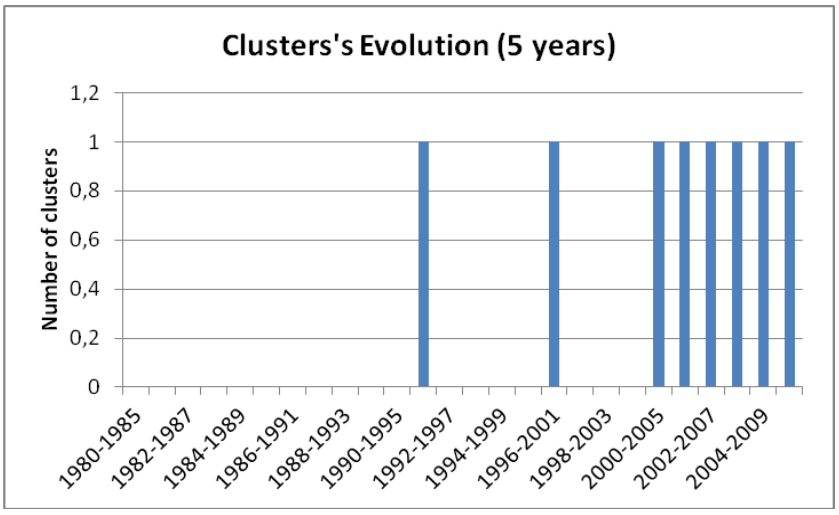

Figure 5-4 Cluster evolution (5-year window)

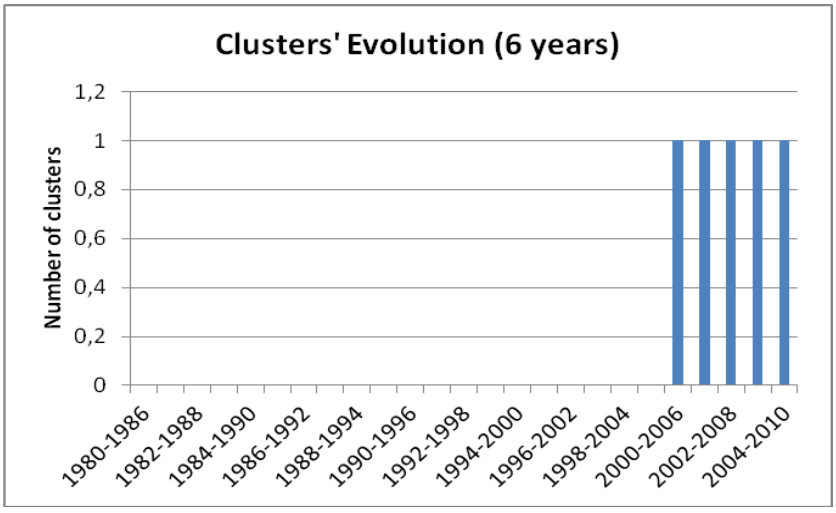

Figure 5-5 Cluster evolution (6-year window)

As is evident by the Figures 5-1 – 5-5, as the window size increases, the number of stable clusters is decreased. For the shortest window (2 years) there exist clusters for every interval, whereas for the longest window (6 years) only one cluster exists for the intervals 2000-2006, 2002-2008, 2004-2010. The 2-year window is short and it is not representative of time persistence. On the other hand, the 6-year window seems too large for the specific dataset as it fails to detect any communities for most time periods. Moreover, only one cluster has been identified at 2000-2006, 2001-2007, 2002-2008, 2003-2009, 2004-2010 time intervals, so the results are not sufficient for further investigation. Between these two extremes, the intermediate time lengths (3 and 4 years) are more appropriate for the data and seem more representative of the evolution process

of the communities. In general, it is noted that there is a trend for communities that occur post ca. 2000 to persist more than communities that occur prior to that year. Furthermore, it is apparent that different time windows lead to different granularity of results.

5.3.2 Clusters' Authors

For the chosen time windows of 3 and 4 years, not only the number of clusters is significant, but also the number of the authors that comprise those communities. It is of interest to investigate how big the identified communities are. In the figures below, the min, max and average number of authors is presented for all the stable clusters that have been identified for different time intervals in those time windows.

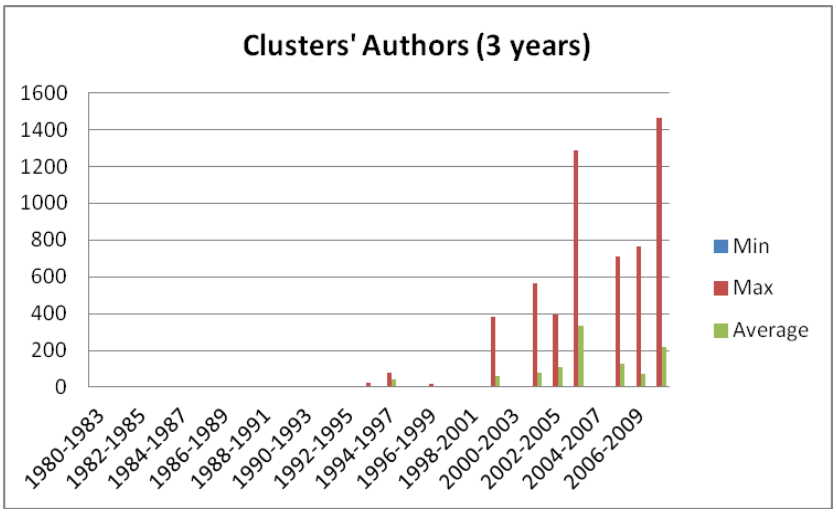

Figure 5-6 Min, max and average number of authors according to the 3-year time window for different intervals

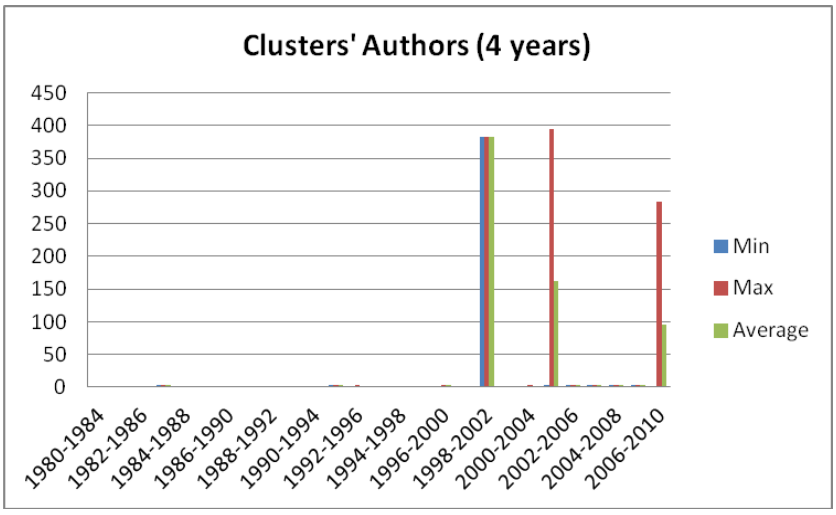

Figure 5-7 Min, max and average number of authors according to the 4-year time window for different intervals

As can be seen in Figures 5-6 and 5-7, the overall max number of authors for the 3-year time window is over 1400 and for 4 years it is about 400. The overall min is 1 author for both sliding windows. A difference of one year in the length of the time window resulted in a significant difference in the number of authors that comprise the clusters. For the 3-year time window, the clusters that consist of a significant number of authors are much more numerous than the 4-year time window where only the 1998-2002, 2001-2005 and 2006-2010 intervals consist of more than 3 authors. Moreover, it is observed that the difference between max and average number of authors for the 4-year time window is more balanced than the respective difference for the 3-year time window. The common point in the diagrams for both time windows is that before 2000 the authors that comprise the clusters are only a few. After 2000 the number of authors that comprise the cluster increases rapidly.

Moreover, it could be interesting to investigate how the size of the clusters evolves over time. That is not possible for all the clusters that have been identified, but a closer look can be taken at the two biggest clusters of those that are identified – i.e. the clusters that consist of 1466 authors and 1288 authors for the 3-year time window. The evolution of the number of authors in the graph is presented at the figures below.

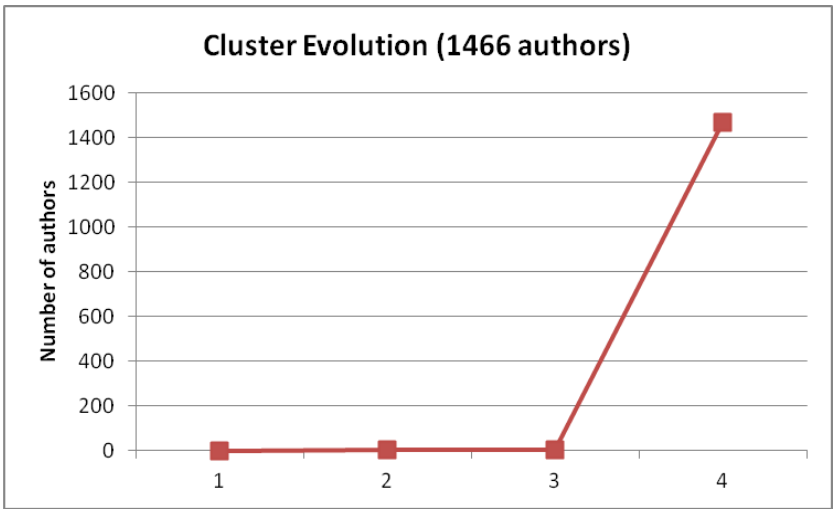

Figure 5-8 Evolution of the identified cluster consist of 1466 authors

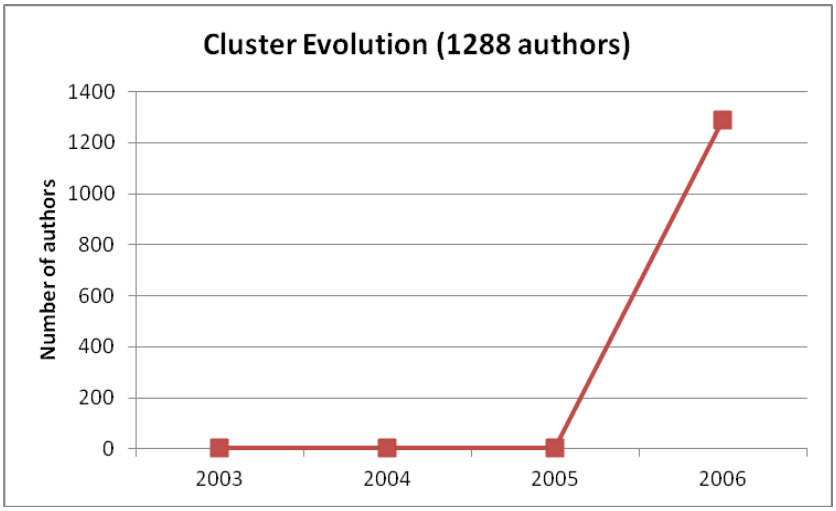

Figure 5-9 Evolution of the identified cluster consist of 1288 authors

As is observed in the Figures 5-8 and 5-9, initially the number of authors is really small and then it increases rapidly. More specifically, regarding Figure 5-8, for the first 3 years, the number of authors that comprise the cluster is really small. More specifically, in 2007 the cluster consists of 1 author and in the following 2 years, the cluster consists of 2 authors. In 2010 the number of authors that are part of the identified community increases rapidly. Regarding Figure 5-9, the number of authors from 2003 till 2005 is 3 and then increases rapidly to 1288. The conclusion from those figures is that maybe the requirement which has been set that requires a percentage of authors to be present in the following year's clusters, is possible only for clusters for which the number of authors

does not increase rapidly, or for clusters where the author number remains the same from year to year. This is a reason the authors' threshold is 20% and not greater.

5.3.3 Clusters' Labels

Except for the number of the clusters of authors, the labels that characterize the clusters that are found at those 2 intervals (3- and 4-year) are of interest in the context of this thesis. At the table below the various words that characterize the clusters persistently over time are presented. The words that belong at the same community have been placed into a parenthesis.

Table 5-1 Clusters' labels for 3 year sliding window

Time intervals	**Labels**
1980-1983	(fix), (parallel)
1983-1986	(nonlinear), (ai-rel, dissert)
1984-1987	(ai-rel, dissert), (handwritten), (**fuzzi**)
1988-1991	(time)
1989-1992	(semigroup)
1991-1994	(chemistri, topolog, organ, **graph**), (reconstruct)
1992-1995	(**graph**, topolog, organ), (**graph**)
1993-1996	(logic), (**graph**), (topolog, organ, **graph**, theori), (cluster)
1994-1997	(**fuzzi**), (use), (recurs), (reason)
1995-1998	(build, smalltalk), (**graph**)
1996-1999	(block), (rna, modif), (enzym, methylas), (translat), (histon), (genbank), (compon, independ, analysi)
1997-2000	(enzym, methylas), (block, databas, server), (method), (optim, orient, cartesian, product)
1998-2001	(methylas, enzym), (**fuzzi**), (noncoher), (distanc, code), (control, **fuzzi**)
1999-2002	(distanc, code), (**fuzzi**), (inform), (multimedia), (inform), (measur), (induc), (model)
2000-2003	(william, lowel, putnam, mathemat, competit), (control), (surfac), (match), (multimedia), (rough), (wireless)

2001-2004	(william, lowel, putnam, mathemat, competit), (stereovis, match) , (network), (compress, test), (intellig, control, approach, use), (brain), (comput, function), (rout, fault-toler)
2002-2005	(william, lowel, putnam, mathemat, competit), (network), (use, neural, control, intellig, **fuzzi**), (ultrasound), (fade), (test, data)
2003-2006	(william, lowel, putnam, mathemat, competit), (**fuzzi**), (speech), (virtual), (system, shape, understand)
2004-2007	(william, lowel, putnam, mathemat, competit), (affin, project, filtered-x), (grid, comput), (network), (data), (quadratur), (steganographi)
2005-2008	(william, lowel, putnam, mathemat, competit), (**fuzzi**), (network) , (linear, **fuzzi**), (invers), (algorithm, cdma), (**graph**), (process), (entropi, hidden, markov, chain, rate), (code), (transmiss), (algorithm), (shrinkag), (steganographi), (orthogon, polynomi), (morphism), (simul, n-qubit, quantum, system), (sequenc), (estim)
2006-2009	(integr), (transcod), (method), (design), (relai), (messag), (distribut), (channel), (tree), (flow), (william, lowel, putnam, mathemat, competit), (algorithm), (multivari, public, kei, cryptosystem, piec, hand, secur), (loop), (amplifi), (learn), (wireless, channel), (method, sixth-ord, converg)
2007-2010	(fuzzi), (william, lowel, putnam, mathemat, competit), (relai), (estim), (modul), (schedul, linear), (trf), (multiclass), (sourc), (messag), (short-tim, transform, system, identifi, domain, fourier), (anycast, ipv6), (**graph**), (optimum, alloc, power, error), (invers), (channel), (shop, flow), (code)

Table 5-2 Clusters' labels for 4 year sliding window

Time intervals	**Labels**
1983-1987	(ai-rel, dissert)
1991-1995	(topolog, organ, graph)
1992-1996	(topolog, organ, graph), (graph)
1996-2000	(methylas, enzym), (block)
1997-2001	(methylas, enzym)
1998-2002	(distanc, code)

1999-2003	(multim)
2000-2004	(**william**, **lowel**, **putnam**, **mathemat**, **competit**), (match)
2001-2005	(**william**, **lowel**, **putnam**, **mathemat**, **competit**), (intellig, control, use), (test), (network)
2002-2006	(**william**, **lowel**, **putnam**, **mathemat**, **competit**)
2003-2007	(**william**, **lowel**, **putnam**, **mathemat**, **competit**)
2004-2008	(**william**, **lowel**, **putnam**, **mathemat**, **competit**), (steganographi)
2005-2009	(**william**, **lowel**, **putnam**, **mathemat**, **competit**)
2006-2010	(**william**, **lowel**, **putnam**, **mathemat**, **competit**), (messag), (relai)

As the Tables 5-1 and 5-2 indicate, there is much more variety of labels at the various clusters that persist over time for the 3-year time window, compared to the 4-year time window. One reason is that there are more clusters for the 3-year time window. Another reason is that at Table 5-2, after 2000, the majority of the labels that characterize the clusters remain the same till 2010. Those labels have been highlighted and there are tokens *william, lowel, putnam, mathemat, competit*. This indicates that maybe there is a cluster that persists in intervals 2000-2004, 2001-2005, 2002-2006, 2003-2007, 2004-2008, 2005-2009, 2006-2010. This assumption is true, as our algorithm identified a cluster that persists from 2000 till 2010 and consists of 3 authors (Leonard F. Klosinski Gerald L. Alexanderson, Loren C. Larson) and keeps being characterized by 5 labels (*william, competition, lowell, putnam, mathematical*).

An interesting conclusion for the 3-year time window is that there are several clusters for which only one of the labels that characterize them persists over time. Another interesting observation is that some labels exist over the years in different clusters, like *graph* and *fuzzi*. The various occurrences of those words in the clusters have been highlighted in Table 5-1, and we can see that there are cases where they appear in different clusters in the same time interval and moreover that they keep appearing in various time intervals. That indicates that they attract the interest of the research community.

In order to find out the most popular labels, a Table has been created with the top 5 clusters' labels with the most occurrences in each interval for the 3- and 4-year time window. The tables below presents for each label: the number of occurrences in the given interval, the number of clusters in which it appears in the specific interval and the total

number of occurrences in the whole time period we study, from 1980 to 2010. The purpose is to find if the labels that continue to exist in each time interval appear significantly in it and moreover if these labels attract the interest of the research community in the whole time period which is examined in our tests. There are intervals that did not have 5 different labels, so in that case all the labels of the interval are included in the results. If the same label exists in two different clusters in the same interval, a comma separates the label's occurrences in those clusters.

Table 5-3 Top 5 labels for the 3-year time window

Time intervals	**Labels**	**Occurrences at clusters**	**Occurrences at time interval**	**Occurrences at 1980-2010**
1980-1983	fix	5	55	1740
	parallel	4	215	12904
1983-1986	nonlinear	6	152	10074
	dissert	4	12	98
	ai-rel	4	11	21
1984-1987	dissert	5	16	98
	ai-rel	4	13	21
	handwritten	4	15	614
	fuzzi	6	147	14733
	select	4	194	8888
1988-1991	time	5	589	13141
1989-1992	semigroup	5	38	410
1991-1994	chemistri	6	34	404
	topolog	12	229	3434
	organ	6	1939	19747
	reconstruct	4	287	3761
1992-1995	graph	4, 6	2128	19747
	topolog	12	259	3434
	organ	7	187	2557
1993-1996	logic	6	1466	10668

	graph	4, 5	2358	19747
	topolog	10	303	3434
	theori	5	1008	9873
	cluster	6	345	6945
1994-1997	fuzzi	7	1287	14733
	use	6	36	8560
	recurs	6	365	2671
	reason	4	530	3478
1995-1998	build	6	308	2852
	smalltalk	9	27	67
	graph	4	2568	19747
1996-1999	block	6	413	3743
	translat	4	196	1697
	compon	5	344	3637
	independ	5	289	2551
	analysi	5	3497	38225
1997-2000	block	5	453	3743
	databas	4	1472	8864
	method	5	3636	31506
	optim	5	2886	26197
	product	5	880	7924
1998-2001	fuzzi	7, 7	2970	14733
	noncoher	8	57	291
	distanc	4	517	3580
	code	4	6760	17328
	control	6	4685	30250
1999-2002	code	7	2329	17328
	fuzzi	22	3058	14733
	inform	13	4140	24646
	measur	4	1684	10464
	model	6	8775	59363
2000-2003	control	9	5922	30250

	surfac	5	994	5736
	match	5	770	4832
	multimedia	5	792	3845
	wireless	4	1422	11616
2001-2004	network	8	8143	53321
	control	9	5975	30250
	approach	9	5109	28145
	comput	7	5429	30789
	function	11	3173	17877
2002-2005	network	20	9622	53321
	neural	4	2591	13850
	control	9	6492	30250
	fuzzi	5	3228	14733
	data	8	5613	26471
2003-2006	fuzzi	7	3541	14733
	speech	6	952	3956
	virtual	18	1732	6631
	system	8	19552	79452
	shape	9	787	3781
2004-2007	project	5	1505	5679
	grid	15	1412	3743
	comput	13	7194	30789
	network	5	14429	53321
	data	4	7440	26471
2005-2008	network	6	17283	53321
	algorithm	4, 22	11391	41628
	process	4	5423	18728
	system	4	23084	79452
	estim	44	5557	17010
2006-2009	method	8	11556	31506
	design	4	9490	28807
	distribut	5	6021	18760

	algorithm	4	12774	41628
	wireless	4	5965	11616
2007-2010	fuzzi	10	5416	14733
	estim	21	7065	17010
	system	13	28200	79452
	graph	11	6192	19747

Table 5-4 Top 5 labels for the 4-year time window

Time intervals	**Labels**	**Occurrences at clusters**	**Occurrences at time interval**	**Occurrences at 1980-2010**
1983-1987	ai-rel	5	14	21
	dissert	6	17	98
1991-1995	topolog	14	308	3434
	organ	8	224	2557
	graph	7	2504	19747
1992-1996	topolog	14	352	3434
	organ	8	242	2557
	graph	5, 7	2816	19747
1996-2000	methylas	5	5	9
	enzym	5	23	234
	block	7	550	3743
1997-2001	methylas	5	5	9
	enzym	5	29	234
1998-2002	distanc	6	676	3580
	code	8	2828	17328
1999-2003	multimedia	6	1005	3845
2000-2004	william	5	38	145
	putnam	5	10	21
	mathemat	5	589	2786
	competit	5	565	2049
	match	6	978	4832

2001-2005	mathemat	5	636	2786
	intellig	6	1354	5814
	control	9	7882	30250
	test	18	2385	9315
	network	22	11323	53321
2002-2006	william	5	41	145
	lowel	5	8	18
	putnam	5	9	21
	mathemat	5	726	2786
	competit	5	635	2049
2003-2007	william	5	36	145
	lowel	5	9	18
	putnam	5	9	21
	mathemat	5	782	2786
	competit	5	665	2049
2004-2008	william	5	33	145
	putnam	5	8	21
	mathemat	5	936	2786
	competit	5	710	2049
	steganographi	6	80	199
2005-2009	william	5	37	145
	lowel	5	7	18
	putnam	5	8	21
	mathemat	5	1074	2786
	competit	5	772	2049
2006-2010	william	5	42	145
	mathemat	5	1196	2786
	competit	5	884	2049
	messag	9	749	1842
	relai	10	1319	1458

Table 5-3 is more interesting because the top labels have a larger variety as the clusters that persist in the 3-year time window are more numerous than in the 4-year time

window. Observing Table 5-3, it comes of that there are cases of labels that their occurrences in the time interval in which they are among the top 5 words is low and their occurrences in the time window from 1980 to 2010 is also low which indicates they actually do not interest the overall research community significantly. An example is label "*smalltalk*" in interval 1995-1998 where the occurrences in the specific interval are 27 and the occurrences in the whole time period are 67. Another example is the label "*noncoher*" in the time-interval 1998-2001. The occurrences in the specific time interval are 57 and the occurrences in the whole time are 291. On the other hand, there are cases where a label has significant occurrences in the whole time period but in specific time intervals the occurrences are comparatively rather small. For example, label *"parallel"* in the whole time window occurs 14733 times but in the time interval 1980-1983 it occurs only 215 times. Presumably, this label characterizes several communities but those either do not persist over time or the label itself does not appear continuously. There are also cases where the occurrences of the label both in the time interval and in the whole time window are big. An example is the label *"system"* in the interval 2007-2010 where the occurrences in the specific time interval are 28200 and in the whole time period are 79452. Thus, the given time interval includes about 35.5% of the label's total occurrences. A special case is label *"fuzzi"* which appears in several different time intervals and the number of its occurrences in the whole time window is significant (14733) but in the initial time windows the number of its occurrences is small and as time passes its occurrences are increased. For example, in the 1984-1987 it occurs only 147 times and in 2007-2010 5416 times which is about 36.67% of the label's occurrences in the whole time period. Therefore, one can consider this increase as an indication of the evolution of the interest of the research community for the given topic through the years. Generally, there are tokens that appear a lot between 1980 and 2010 and this indicates that they attract the interest of the research community.

In Table 5-4, the majority of the entries have small number of occurrences both in each time interval they appear and in the whole time window which is examined. For example label *"methylas"* has 5 occurrences in time intervals 1996-2000 and 1997-2001 and the sum of occurrences in the whole time is only 9. This indicates that this label does not interest the research community and appears mainly in the specific interval and moreover in the specific cluster which persists, as the occurrences in the cluster is 5, equal with the sum of occurrences in the time intervals. Another case is the label

"*putnam*", which appears in the time intervals 2000-2004, 2002-2006, 2003-2007, 2004-2008, 2005-2009 and its occurrences vary from 8 to 10 whereas its occurrences in the whole time window is only 21. Generally, in the 4-year time window, the same labels keep occurring and the majority of them are labels that do not show a significant spread of interest in the research community.

The fact that there are labels that are included at the top 5 labels in different intervals such as "*fuzzi*", and "*graph*" verify the results from Tables 5-1 and 5-2. There are words that keep appearing in different clusters during years, and those are also included in the top words of the intervals. Moreover, as their occurrences in the time interval 1980-2010 indicates, there are labels that interest the research community generally. It is interesting to investigate how their occurrences evolve over time. The following figure presents the evolution of the "*fuzzi*" and "*graph*" labels.

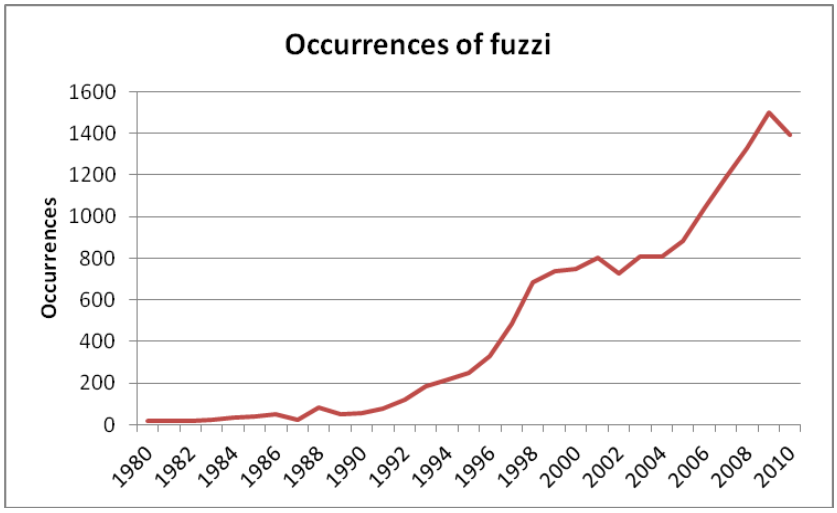

Figure 5-10 Occurrences of fuzzi label

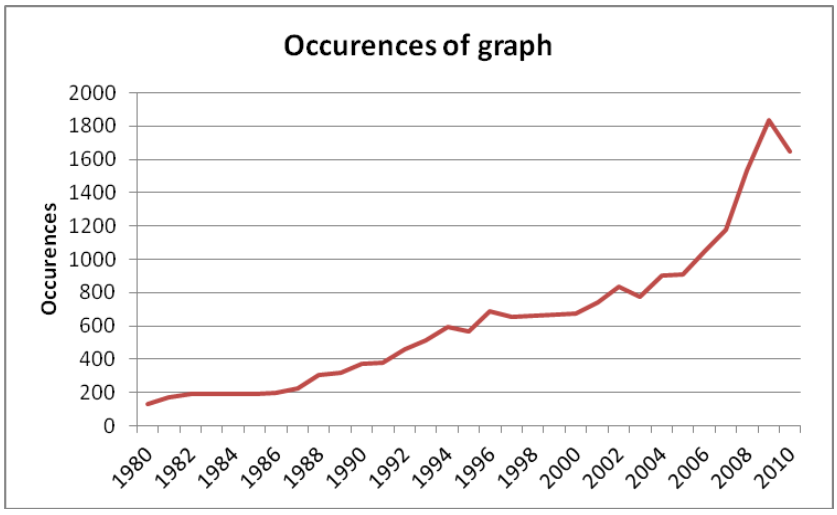

Figure 5-11 Occurrences of graph label

As Figures 5-10 and 5-11 indicate, occurrences for both labels increase through time and only for the last 2 years a small reduction is observed. The assumption is that the most important reason for this increase is that the number of publications increases rapidly as time passes in this specific time window. For example, the publications in 1990 are less than the publications in 2000. As a result, tokens may appear more often due to the greater number of publications.

The general outcome of our experiments is that actually there are clusters of authors that they keep cooperating regarding the same topic over time but as time intervals are increased, this behavior is observed in fewer clusters. Moreover, there are topics that attract the interest the research community as they characterize different clusters that persist in several of the examined time intervals and moreover the number of their occurrences in the whole time window is remarkable. The reason why on the one hand there are labels that attract the interest of the research community through years but on the other hand the cluster that persist in large intervals are a few, is that maybe that those labels characterize many different clusters which do not have continuous appearance through time or they do not use the same words at their publications. In this approach we demand the same token to appear through years and as a result maybe there are cases where the authors keep publish regarding the same topic but without using the same words continuously.

6 Conclusion

The aim of the current thesis is to find persistent communities in a time evolving network that persist in time while also keep some of their characteristics (labels) stable as various interesting outcomes can be revealed from this insight into network's communities evolution. A case study was applied in a publications network in order to test the proposed methodology. The goal was to find groups of authors that persist in collaborating over time as well as important topics characterizing the authors'' work and especially identify persistent groups of authors that insist on working on the same topics through time.

6.1 Results

Our experiments with the bibliographical network showed that as the studied time interval increases the number of communities that persist over time decreases rapidly. In the longest time interval we tested, 6 years, only a few clusters managed to persist with at least one common label throughout all these years and they consist of only 2-3 authors. Moreover, it was observed that after 2000, the research community is more active and as a result more clusters persisted and met the prerequisites that we set in our methodology.

3- and 4- year time intervals were selected for closer investigation and especially at the 3-year time window it was found that there are labels that keep existing over time but at different clusters. The occurrences of the top 5 labels that appear at those time intervals in the whole time window showed that actually there exist keywords that keep being of interest to the overall research community over time but they do not appear continuously in the same cluster.

6.2 Research Difficulties and Limitations

During the research process some difficulties and limitations occurred. One important difficulty was that the algorithm was not targeted at good performance but at the investigation of the problem. As our dataset included large-scale networks, the execution time of the experiments was significant.

Moreover, the research community is much more active in recent years and the number of publications is larger, but this time window is not sufficient to investigate

persistence over time and draw useful conclusions. As a result to target a longer time period, the initial year of the experiments was set to 1980, because firstly, the number of publications was adequate and secondly the evolution of the research community's interest could be revealed.

6.3 Future Work

Considering the problems we encountered, the limitations of our work and the results of this first study, we can see there is room for improvement of the approach. A possible improvement concerns the performance of the algorithm to improve its efficiency and scalability by using appropriate indexing structures or the design of pruning methods that could exclude communities that are not likely to persist over time from our computations.

Moreover, a more refined approach regarding community labels can be adopted. It is possible that communities keep publishing about the same topic but they did not use the same words in their publications. With the current approach those cases are not taken into consideration. A potential extension is to group together the labels that concern the same topic maybe with a clustering approach or with the use of ontologies. Another proposal is to also extract and exploit as labels the keywords, the tokens form the abstract or from the main body of the paper.

What is more, to deal with the observed peaks in the membership of the communities that lead to characterizing such communities as not stable as they lose a significant number of members in the next instance, we could use an adaptive persistence threshold based on the size of the cluster. The larger the cluster, the lower the threshold in order to be easier for large clusters to persist over time. Finally, it would be interesting to repeat the experiments after a few years with a more recent time window and maybe test different clustering algorithms or relax the clustering approach by tracking communities not by using a clustering algorithm but a measure of their similarity or relevance.

References

Aggarwal, C. C., Yu, P. S., (2005). *Online analysis of community evolution in data streams*, In Proceedings of SIAM International Conference on Data Mining, pp. 56–67.

Aker, A., Kurtic, E., Balamurali, A. R., Paramita, M., Barker, E., Hepple, M., Gaizauskas, R.., (2016). *A graph-based approach to topic clustering for online comments to news*, In Proceedings of the 38th European Conference on Information Retrieval, pp. 15-29.

Asur, S., Parthasarathy, S., Ucar, D., (2007). *An event-based framework for characterizing the evolutionary behavior of interaction graphs*, In Proceedings of the 13th ACM SIGKDD Conference, pp 913–921.

Barber, M. J., Clark, J. W., (2009). *Detecting network communities by propagating labels under constraints*. Physical Review E, 80(2), 026129.

Basuchwdhuri, P., Chen, J., (2010). *Detecting communities using social ties*, Proceedings of the IEEE International Conference on Granular Computing, San Jose, pp. 55-60.

Blondel, V. D., Guillaume, J. L., Lambiotte, R., Lefebvre, E. (2008). *Fast unfolding of communities in large networks*. Journal of Statistical Mechanics: Theory and Experiment, 10, P10008.

Cai, B., Wang, H., Zheng, H., Wang, H., (2010). *Evaluation repeated random walks in community detection of social networks*, In Proceedings of the 9th International Conference on Machine Learning and Cybernetics, Qingdao, pp. 1849-1854.

Cha, M., Mislove, A., Gummadi, K. P., (2009). *A measurement-driven analysis of information propagation in the flickr social network*, In Proceedings of the 18th International Conference on World Wide Web, pp. 721–730.

Chen, C., Yan, X., Zhu, F., Han, J., Yu, P. S., (2008). *Graph OLAP: towards online analytical processing on graphs*. In Proceedings of the 8th IEEE International Conference on Data Mining, pp. 103-112.

Elkan, C., (2003). *Using the triangle inequality to accelerate k-means*, In Proceedings of the International Conference on Machine Learning, pp. 147-153.

Ferlez, J., Faloutsos, C., Leskovec, J., Mladenic, D., Grobelnik, M., (2008). *Monitoring network evolution using MDL*, In Proceedings of the IEEE International Conference on Data Engineering, pp. 1328-1330.

Ferrer, M., Valveny, E., Serratosa, F., Bardaji, I., Bunke, H., (2009). *Graph-based k-means clustering: A Comparison of the set median versus the generalized median graph*. Lecture Notes in Computer Science, 5702:342-350.

Fortunato, S. , (2010). *Community detection in graphs*, Physics Reports, 486: 75-174.

Freeman, L. (1977), *A set of measures of centrality based on betweenness.* Sociometry, 40:35-41.

Girvan, M., Newman, M. E. J., (2002). *Community structure in social and biological networks*, Proceedings of the National Academy of Sciences, USA, pp. 8271-8276.

Granovetter, M. S., (1973). *The strength of weak ties*, American Journal of Sociology, 78: 1360-1380.

Gregory, S., (2007). *An algorithm to find overlapping community structure in networks*, Proceedings of European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, Antwerp, pp. 91-102.

Gregory, S., (2008). *A fast algorithm to find overlapping communities in networks*, Lecture Notes in Computer Science, 5211:408-423.

Hedge S. M., (2012). Labeled graphs and digraphs: theory and applications, [online] Available at: <http://cs.rkmvu.ac.in/~sghosh/public_html/nitk_igga/slides/iggaSMH.pdf> [Accessed 7 July 2018]

Honsch, M., (2011). *Detecting user communities based on latent and dynamic interest on a news portal*, Personalized Web-Science, Technologies and Engineering, (3):47-50.

Hyland-Wood, D., Carrington, D., Kaplan, Y. (2005). *Scale-free nature of java software package, class and method collaboration graphs*, In Proceedings of 5th International Symposium on Empirical Software Engineering, pp. 439-446.

Jain, B., Obermayer, K., (2009). *Elkan's k-means for graphs*, arXiv:0912.4598v1[cs.AI].

Jiang, X., Minger, A., Bunke, H., (2001). *On median graphs: Properties, algorithms and applications*. IEEE Transactions on Pattern Analysis and Machine Intelligence 23(10):1144-1151.

Leskovec, J., Kleinberg, J., Faloutsos, C., (2005). *Graphs over time: Densification laws, shrinking diameters and possible explanations*, In Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, pp. 177–187.

Lim, S. H., Chen, Y., Xu, H., (2014). *Clustering from labels and time-varying graphs*, Advances In Neural Information Processing Systemts, 27:1188-1196.

Lim, S. H., Chen, Y., Xu, H., (2017). *Clustering from general pairwise observations with applications to time-varying graphs*, Journal of Machine Learning Reseach, 18(49):1-47.

Liu, J., Wang, C., Danilevsky, M., Han, J., (2013). *Large-scale spectral clustering on graphs*, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, pp.1486–1492.

Lovasz L., (1993). *Random Walks on Graphs: A Survey*, Bolyai Society Mathematical Studies, 2:1-46.

Macropol, K., Can, T., Singh, A., (2009). *RRW: repeated random walks on genome-scale protein networks for local cluster discovery*, BMC Bionformatics. 10:283.

Montgolfier, F. D., Soto, M., Viennot, L., (2011). *Treewidth and Hyperbolicity of the Internet*. In Proceedings of the 10th IEEE International Symposium on Network Computing and Applications, pp. 25-32.

Newman, M., Girvan, M. (2004). *Finding and evaluating community structure in networks*, Physical Review E, 69:026113.

Pons, P., Latapy, M., (2005). *Computing communities in large networks using random walks*, Lecture Notes in Computer Science, 3733:284-293.

Porter, M. F., 1980, *An algorithm for suffix stripping*, Program (Automated Library and Information Systems), 14(3):130-137.

Rossi, R. A., Gallagher, B., Neville, J., Henderson, K., (2013). *Modeling Dynamic Behavior in Large Evolving Graphs*, In Proceeding of the 6th ACM International Conference on Web Search and Data Mining, pp. 667-676.

Rotta, R., Noack, A. (2011). *Multilevel local search algorithms for modularity clustering*. Journal of Experimental Algorithms, 16(2):2.3.1–2.3.27.

Scaiella, U., Ferragina, P., Marino A., Ciaramita, M., (2012). *Topical clustering of search results*, In Proceedings of WSDM-12, pp. 223-232.

Schaeffer, S. E., (2007). *Graph clustering*, Computer Science Review, 1: 27-64.

Semertzidis, K., Pitoura, E., (2016). *Time traveling in graphs using a graph database*, in Proceedings of the Workshops of the (EDBT/ICDT).

Semertzidis, K., Pitoura, E., Terzi, E., Tsaparas, P., (2016). *Best Friends Forever (BFF): finding lasting dense subgraphs*.

Sun, J., Falousos, S., Papadimitriou, S., Yu, P. S., (2007). *GraphScope: Parameter-free mining of large time-evolving graphs*, In Proceedings of the ACM SIGKDD International Conference Knowledge Discovery in Databases, San Jose, pp. 687-696.

Toyoda, M., Kitsuregawa, M. (2005). *A System for Visualizing and Analyzing the Evolution of the Web with a Time Series of Graphs*, in Proceedings of the 16th ACM Conference on Hypertext and Hypermedia, pp. 151-160.

Traag, V. A., Van Dooren, P., Nesterov, Y., (2011*). Narrow scope for resolution-limit-free community detection*, Physical Review E, 84(1):016114.

Waltman, L., Van Eck, N.J., (2013). *A smart local moving algorithm for large-scale modularity-based community detection*, European Physical Journal B, 86(11), 471.

Ward, J. H. (1963). *Hierarchical grouping to optimize an objective function, Journal of the American Statistical Association*, 58(301):236-244.

White, S., Smyth, P., (2005). *A spectral clustering approach to finding communities in graphs*, In Proceedings of the 5th SIAM International Conference on Data Mining, Philadephia, pp. 76-84.

Wilson, R., J., (1996). *Introduction to graph theory*, Fourth Edition, [online] Available at: <http://www.maths.ed.ac.uk/~v1ranick/papers/wilsongraph.pdf> [Accessed 13 June 2018].