Universitat Politècnica de Catalunya

Bachelors in Computer Science and Engineering

Minor Heading

Graph and matrix algorithms for visualizing high dimensional data

Director:
Dr. Ricard Gavalda Mestre

Co-Director:
Dr. Marta Arias Vincente

Bachelors Thesis of :
Abhinav Shankaranarayanan Venkataraman

June 27,2016


UPC

To my Mother, Father, Professors and Friends. I owe a lot to My professors Ricard Gavalda and Marta Arias and to Babaji at Gurudwara

Abstract

Motivated by the problem of understanding data from the medical domain, we consider algorithms for visually representing highly dimensional data so that "similar" entities appear close together. We will study, implement and compare several algorithms based on graph and on matrix representation of the data. The first kind are known as "community detection" algorithms, the second kind as "clustering" algorithms. The implementations should be robust, scalable, and provide a visually appealing representation of the main structures in the data.

Acknowledgement

I would like to Acknowledge the support provided by my faculty and admins at my home university – SASTRA University, Thanjavur and UPC Barcelona for supporting me throughout the project.

Contents

1 Introduction

In this section we provide an overview of the entire work. We mention the context of the project we have studied, approaches that we have used, goal of the project. We also provide the intended planning, economic estimate and sustainability of the work that has been done.

1.1 Context Of the Project

In the present day scenario, the modern science of algorithms and graph theory has brought significant advances to our understanding of complex data. Many complex systems are representable in the form of graphs. Graphs have time and again been used to represent real world networks. One of the most pertinent feature of graphs representing real system is community structures or otherwise known as clusters. Community can be defined as the organization of vertices in groups or clusters, with many edges joining the vertices of the same cluster and comparatively fewer vertices joining the vertices in another neighbouring cluster. Such communities form an independent compartment of a graph exhibiting similar role. Thus, Community detection is the key for understanding the structure of complex graphs, and ultimately educe information from them.

1.2 Approaches

1.3 For Community Identification

Virtually in every scientific field dealing with empirical data, primary approach to get a first impression on the data is by trying to identify groups having "similar" behaviour in data. There are numerous methods to achieve this objective of which

- Community Detection
- Clustering

1.3.1 Community Detection

Definition Communities are a part of the graph that has fewer ties with the rest of the system. Commmunity detection traditionally focueses on the graph structures while clusting algorithms focuses on node attributes.

1.3.2 Clustering

Traditional Clustering Methods are as follows:

- Graph Partitioning
- Hierarchical Clustering
- Partitional Clustering
- Spectral Clustering

1.3.3 For Visualization

1.3.4 Computational Complexity

The estimate of the amount of resources required for by the algorithm to perform a task is defined as computational complexity. The humongous amount of data on the real graphs or real networks that are available in the current scenario causes the efficiency of the clustering algorithm to be crucial.

community finding and clustering. separately, visualization tools

1.4 Goal of the Project

1.5 Planning

1.5.1 Task Description

The tasks for the project have been subdivided into various task phases which are enumerated below :

- **Required knowledge acquisition**
  Before any immersion into the real topic, it was necessary to acquire the knowledge necessary to understand the problem. In this phase we familiarize with the term modularity, Louvain algorithm for community detection and various other algorithms used for community detection. Acquisition of knowledge about visualization tools to be used and make conversant with python is also required.

- **Paper Analysis**
  In this phase we analyze and compare several works about community detection and clustering algorithm over high dimensional graph-like

data. Doing this we became conscious of functionalities that our proposal should have and we are thus able to guide all the subsequent phases.

- **Design and Implementation**
  In this phase the project is designed and coded implementing all the functionalities of the solution.

- **Testing I**
  In this phase we test the program in order to identify errors in the implementation. It includes the successive recoding.

- **Testing II**
  In this phase we perform tests over synthetic and real data streams. We evaluate the performance of the program and we study the effects of concept drift.

- **Report Writing**
  In this phase the report of the project is written.

1.6 Economic Budget

1.6.1 An Introduction to Economic Budget

Economic management is primarily based on an estimate of income and expenditure called as budget. Development of a sustainable budget leads to proper economic management of the project. Budget and sustainability is one of the most important phase of the project management. In this phase we analyze the budget for the project. We also aim at providing an estimate of the project budget and optimize the same. We look at the expenditure from various aspects such as software costs, hardware costs, license costs and human resource costs. Additionally we also account the software for its sustainability. One important factor to note is that the budget that we describe in this section is subject to change and it may increase depending on the unexpected obstacles that we may face. For an instance when we don't get the expected results with a particular software we may have to go in for another software that may incur extra installation and operational charges.

1.6.2 Estimation of Economic Budget

We divide the overall expenditure into three categories namely hardware, software and human resources. One very important factor that we need to consider is that we only get an estimate of the total cost. This may vary depending on the systems in use. To calculate the amortization we consider to factors namely, first the overall life of the hardware or software in use. Second that the project is completed in 5 months. Hence the amortization cost comes one eighth of the actual life of the component.

Hardware Budget Hardware budget accounts for the actual and the amortized costs of the hardware elements used by the project. The cost is fictitious as it has not been developed commercially. Table 1 intents to estimate the economic cost of each of the hardware component of the project.

Table 1 - Hardware Budget				
Sno:	Hardware Component	Useful Life	Total Cost(in €)	Amortized Cost(in €)
1	PC System	4	1000€	125 €
	Total		**1000€**	**125 €**

Software Budget The software budget shows an estimate for the various software used in the project along with the estimate of the software costs. It is a myth that the software doesn't get old with time just as a software gets but it wears out with time. Thus for every software there is a fixed time during which it gives maximum performance. In addition freeware software and open source software incur no cost. The cost is fictitious as it has not been developed commercially. Table 2 intents to estimate the economic cost of each of the software component of the project.

Table 2 - Software Budget				
Sno:	Software Component	Useful Life	Total Cost(in €)	Amortized Cost(in €)
1	Linux OS	5	0€	0 €
2	JavaScript Engine	1	0€	0 €
3	Python Components	1	0€	0 €
4	Web.py	1	0€	0 €
5	TexMaker	1	0€	0 €
	Total		**0€**	**0 €**

Human Resource Budget The human resource budget deals with the overall expenditure spent on human resources. Every phase of the project has a cost associated with it in per hour calculation. The cost is fictitious as it has not been developed commercially. Table 3 intents to estimate the economic cost of each of the phases of the project. The cost per hour is intended as an approximation of the current cost per work hour of young analysts and developers in our environment.

Table 3 - Human Resource Budget					
Sno:	Phase	Deadline	Hours	Cost(per hour in €)	Total(in €)
1	Required Knowledge Acqusition	1 Mar 2016	70	15€/h	1050
2	Paper Analysis	1 Apr 2016	150	15€/h	2250 €
3	Design and Implementation	30 Apr 2016	230	20€/h	4600 €
4	Testing I	15 May 2016	75	15€/h	0 1125€
5	Testing II	31 May 2016	75	15€/h	0 1125€
5	Report Writing	15 Jun 2016	100	15€/h	1500€
	Total		600		**10525 €**

1.7 Sustainability

2 Background Knowledge

In this section we present the background knowledge required to understand and solve the problem

2.1 Graph Notation

Graph ,G , is construct consisting of two finite sets, the set V = { $v_1, v_2, \ldots, v_n$ } of vertices and the set E = { $e_1, e_2, \ldots, e_n$ } of edges where each edge is a pair of vertices from V, for instance,

$$e_i = (v_j, v_k)$$

is an edge from v_j to v_k represented as G=(V,E).

2.2 Matrix Notation

2.3 Equivalence between Graph and Matrix Represenation

2.4 State-of-the-art in Community finding

2.5 State-of-the-art in Clustering

2.6 State-of-the-art in Graph Visualization

explain technical concepts in more detail. for exampleequivalence of graph and matrix representations. state-of-the art in community finding, and clustering state-of-the art in graph visualization