# Capitolo 1

# Summary

This thesis goes back over the work that I was done during the internship inside the university, with the supervision of the PhD student Sheikh Nasrullah for the professor Montresor Alberto.

## 1.1 Introduction

The whole internship is based on the paper named "Community-enhanced Network Representation Learning for Network Analysis" alias CNRL, this paper proposes an innovative method for detecting the communities inside a graph, so my purpose is to take the code at the base of this paper, try to replicate the results shown and after that increase if it is possible the potentiality of this algorithm.
At the start, I had to need to understand the main problem and understand the dynamics at the base of it, therefore I read out some other papers like:

- "DeepWalk: Online Learning of Social Representations"
- "node2vec: Scalable Feature Learning for Networks"
- "How exactly does word2vec work?"

Before the explaining of the work done, we need to understand why the community detection is so important.

## 1.2 What are graphs and the communities

At the base of all, there are the graphs, a particular data structure formed of the nodes that can represent an entity, eventually with some attributes, and the edges that link two nodes, the edges are all directed or not. With this special data structure, we can represent a huge amount of situations. The purpose of the community detection is to find the communities inside the graph, that are groups of nodes that are similar.
What does this mean?
Two nodes are similar if have an equivalent role inside a group of nodes, they can are a hub of connection or an isolated point or more simply they have the same value for some attribute. To find this groups can permit to who look for it, to manage those nodes in a similar way. Like the possible application of the graphs, the same is for this technology, there are lots of situation and context where those algorithms can work very well.

## 1.3 How we evaluate the community

Up to now, I have explained what are graphs and communities, and from my words seem that when you choose some groups of nodes is may be easily understood if it is a good partition or not. And if you have two partitions to decide which is the best is not difficult. This is not really true. If we have a little graph with only one community we can look at it and decide if the partition is good or not,

but this is not possible for a big graph.
So how we evaluate a partition?
First of all, we must explain what we intend with partitions, here the characteristics:

- A partition is a set of communities
- Not all the communities have the same size, or better the same number of nodes
- For the communities, there is not a minimum size neither a maximum size
- Is possible that some nodes do not have a community to which it belongs
- A node can belong to more than one community, this mean that I have an overlapping of communities

Said that, we can speak about the modularity.
The modularity is the method used for evaluating a partition, there are different implementations of it because the aim is the same but we can have different interest so we can weight different aspect, in our work we have used three different methods: mod_withMax, mod_overlap and modified modularity. The first two are unofficial names, we have chosen them for distinguishing it, the last one is the method used in the CNRL's paper. More explanation of those methods in the chapter on the modularity.

## 1.4 Our changes

The code of CNRL do $x$ random walks (a sequence of $l$ adjacent nodes) on the graph where $x = n * w$:

- $n$: is the number of nodes in the graph
- $w$: is the number of walks starting from each node
- $l$: is the maximum length of each walk

Given those walks, the algorithm calculates the partition. Note that CNRL use only the structure of the graph, the nodes and the edges, no other information is considered.
Our approach uses the code of CNRL but introduces the attributes of the nodes, this permit a better evaluation. This because two nodes that before are far, there was no one link between them, now are considered near if they share one or more attributes.

## 1.5 Results

To be continued...