Analiza omrežij

1.predavanje (21.2.2024)

Logistics of the course: 2Hws, project, openbook exam

Newman – Networks: An Introduction (2018)

Netwoek science

Graph – points(nodes) connected with edges

Web is the largest graph, second is the human brain

Internet: nodes are class C subnets and autonomus entities (like the faculty) that route by themselves, links are packet routes.

Facebook: nodes – users, links – social connections (online)

Society – firnedships – division based on race and age (offline)

Assortatitve mixing nodes connect to similar nodes (same authors collaborate with similar authors, homosexual relationships in a sex intercourse graph)

Symetric (social friendship) vs. Asymetric connections (citing another paper)

Example from ecology: food web, in topological (?) layers, cannibalism (loop connection from the node to the node itself), we will usually remove those loops (they usually have no meaning)

Hub-spokes arrangement : hub in the middle, links (spokes ? ) out of it

Networks are studied in order to understand the real systems. If they are to complex, we study them trough their structure(what is expected and what is not), evolution (how it came to such structure) and dynamics(dynamic changes traveling troutgh networks).

Small worlds networks

Scale-free networks

Connected – six degrees of separation (documentary)

Historical development of network science :

* it started with graph theory (Eulier - 7 bridges problem – first non trivial proof of graoh theory)
* Hamilton – travelling salesman problme
* Kirchhoff – laws of electrical circuits
* Kekule – Chemical structure of molecules

Operation reserach :

* Dijkstra – shortest path
* Kruskal – spanning tree
* Ford & Fulkerson – maximum flow/minimum cut
* Cartwright – signed graphs (weighted links with positive and negative weights – usually we need a nonnegative weigth) 🡪 we have stable and unstable structures based on the signs of the weights on the links
* Random graph theory –

Sociometry:

* Children sociograms (Moreno)
* Southern Women – Davis: women attending social events (they divide into two groups based on the events)
* University karate club – Zachary (most known graph) :
* Small world experiment – entire world is connected (confirmation of six degrees of separation)
* Strength of weak ties – Granovetter : strong and weak ties, weak ties are more useful (when looking for a new job social circle of me and my best friend are quite similar, while someone you only know as a collegue might know of some more job opportunities)
* Measures of centrality – Freeman : nodes importance

Bibliometrics:

* Scientific paper citations – SCI – Science Citation Index (Derek de Solla Price)
* Political scandals – Mark Lombardi
* Neural wirings – White et. Al – brain of a worm is completely mapped
* Transportation – Pelletier

Around the year 2000 3 things happened that made it possible for this field to evolve (way more citations of old papers from the field, mentioned above):

* Data became avaliable – seven orders larger than whatever exsisted before (before no one collected that many data)
* People realised thar mathematical models of graphs turn out to be useful to understand the structure of real networks (is a certain thing expected in a graph or is it suprising)
* The networks have the same structural properties – there has to be some reason for seeing these properties in science, society etc. It has to be a simple reason 🡪 modern networks from different fields are similar to one another – we can use the same tools and methods on all of them

Impact of network science: management(predicition of conflicts in a company), economic, epidemics,(predicting the spread of diseases), health(drug design), security (detecting insurance fraud), neuroscience (mapping of the human brain) and many others.

Problem – understanding the real networks

Means : study of netowrk properites, design of mathematical models, implementation of efficient algorithms

Goals: knowing the netwirk structure and evolution, knowing the network dynamic processes

**4 Layers** that we will consider when studying networks:

* Layers – whole network
* Clusters (their characteristics) – group of nodes
* Fragments – triangles and hub-spokes
* Individual nodes/links

1. GRAPHOLOGY AND NETWORKOLOGY

Terminology:

* Graph vs network: graph is a representation of a network, while network is some real-world problem.
* Graph is formal mathematical object, while network is a graph with real data
* Different terminology based on field:
  + Network scoence : edges/links
  + Graph theory: edges/relations
  + Social science: agents/brokers/units and ties

Terminology classes:

* Social networks: for example **affiliation networks** (some students are enrolled or affiliated at the university – two types of nodes, edges only appear betwen the nodes of different types), collaboration network is a projection of a bypartite graph, Facebook, offline, online, affiliation, author/actor collaboration
* Infromation networks: nodes are source of information, links show how information flows (they show information flow – usually in the other way than the links are pointed) : web, twitter, citation, communication, peer-to-peer
* Technological networks: human-made infrastructure with technological constraints(Internet, telephone, transportation, power grid, software)
* Biological networks: interaction between genes, cells, neurons in living beings.
* Ecological/lexical/financial/sports

Definition of a graph : n = number of nodes, N = set of nodes, L = set of links, m = number of links

If graph is **undirected**: links are unordered pair of nodes {i,j}

If graph is **directed**: links are ordered pairs of nodes (i,j)

**Adjecency matrix** is a nxm matrix, elements od the matrix tell you wheteher i and j nodes are connected. The edge in row i and column j tells you there is a connection from j to i (not the other way around which is more natural). We do it that way to simplify some formulas with eigenvectors later on. If there are no loops, the diagonal elements are always zero. Adjeceny matrix is symetric in an undirected graph – if we sum the whole matrix we get 2\*m connections. If the graph is directed Aij != Aji, we get the number of links if summing the whole matrix.

* Simple graph : only one link between two nodes
* Multipgraph : multiple links possible between two nodes
* Pseudographs: graph that has loops (diagonal element of the adje matrix is nonzero – 2 if there is one loop, 4 if there are two loops for this node)
* Weighted graph: links have weights, weights must be non-negative (they represent distances, larger the weight, shorter the distance)

Multipartite graph : instead of adjecency matrix we have a incidence matrix B, where one class of nodes is on the rows, and the opther class of nodes in on columns.

Projections: A = B^TB – D1; A = BB^T – D2

Multi-mode: bipartite graph(two-mode network), tripartite graph (three-mode network). Higher orders are rare.

Degree:

* Degree is the number of links from a node ni, we note it with ki.
* In directed networks the number of outgoing links is ougoing degree, and incoming links are incoming degree (ki\_out, ki\_in). The classical degree is a sum of ki\_out and ki\_in.
* Average degree: sum of all degress, divided by the number of nodes (denoted by <K>)
* If the number od edges is m, number of halfedges is m/2. If we count halfedges over the nodes we get Ki (degree of the node), so we just sum over all the nodes to get 2\*m = n\* <K>
* n \*sum(Ki)/n = sum(Ki) = 2m
* in a directed graph the average in degree is the same as the average in degree
* Average degree is rarely larger than 10 (except for Facebook network), the average degree does not scale with the average number of nodes of the network

Density(ro):

* In an undirected graph: number of links divided by number of all possible links in a graph (ro = m/(n over 2) = 2\*m/n\*(n-1) = <K>/(n-1) E [0,1].
* Density ranges from 0 to 1.
* If we have a connected graph, the sparsest graph is a tree with no loops
* Real networks are pretty sparse (cca. 0.1 – 0.2) 🡪 How is this mathematcialy defined? Graph is sparse if the density goes to zero when the n nodes go to infinity. If density converges to some nonzero value, the graph is dense.
* Connection density and network size are linearly linked (higher the density, smaller the network) – adjecency matrix is not a good way to store links because it is mostly zeroes – next week we will discuss a better way to store it.

28.2.2024 – 2.predavanje

(manjka 15min)

Degree distribution:

* Degrees are not independent one of another (if one has more, the other has less) – uncorrelated is main preposition of Gauss distribution.
* Nodes with a lot of connections are called hubs.

Connectivity of a network:

* **Path** is a sequence of links betwen i and j. Links cannot repeat, (if they repeat it is a **walk** not a path).
* **Connected component** is a subset of nodes such that exists a path between each pair of nodes. If we have an edge that if removed the whole graph falls apart, we have a **bridge**.
* **,** has only one connected component.
* What are the sizes of connected component?
* Usually, we have one giant component that contains a nontrivial fraction of nodes (90 – 95%). What remains is »change«.
* If we have a directed graph, path is a sequence of directed links.
* **Weak connectivity**: weakly connected component is connected if we ignore the direction of edges
* **Strong connectivity:** we take into account the direction of the links. If the component is connected with directed edges, we have a strongly connected componnent.
* In practice we like connectivity.
* If we have a giant componnent, we expect to have only one (it is very unlikely to have two giant componnents, because it takes only one connection from the first one to the second one to obtain one giant componnent)

Distances:

* **Length of the path** is number of links of the path.
* **Geodesic path** is the shortest path between nodes i and j (we can have multiple)
* Distance between two nodes is the length of the shortest path between the two nodes.
* Distances are non-negative, withhold the triangle rule and symmetry.
* Diameter of the graph will be the largest distance between any pair of nodes in our network.
* Maximum distance is not a very robust measure, so we instead use the **average distance** (go through all pairs of nodes, count the distances, normalize the paths accordingly)
* If we have a disconnected graph: two nodes that are not connected, their distance is infinite and the average distance is useless. To avoid that we define that distance as zero – or we can only calculate the average distance from the giant connected component (zanemarimo majhen nepovezan del)
* Average distance = <d> = l^-1 (pisani L) -> ni na -1 potenco, to je zgolj kako označimo to mero razdalje
* Distancces are short, close to the average, normally distributed,
* Average distance scales with the size if the graph was just linear nodes connected.
* <d> = 6 <<n

**Clustering coefficient:**

* **Nakopičenost**: friend of a friend is also your friend (that probability is very high)
* We formaly define it as: one node i has ki neighbours. Clustering measures what is the probability that neighbors of node i are connected. That kind of connections form triangles. Clustering coefficient is the number of triangles in the neighborhood divided by the maximum possible number of triangles. Ci = ti/(ki over 2) = 2\*ti /Ki\*(Ki – 1)
* **Average clustering coefficient:** calculate the clustering coefficient for all the nodes and then average,
* Network clustering coefficient is defined as all possible triangles in the whole network \* 3, divided by number of linked triplets/connected triads. The multiplication by 3 serves as normalization, otherwise we would count each triangle three times.
* Calculating takes a while, facebook example is calculated only one a small snapshot, not on the whole network
* For high degree we expect much higher clustering coefficient that for low degree guys

How do we represent a network?

Slika, ki vsebuje besede besedilo, diagram, posnetek zaslona

Opis je samodejno ustvarjen

1. Adjacency matrix (symmetric for undirected graph)
2. Adjacency list
3. Edge list (ordered pairs of nodes for directed graph (2,1), if they are undirected use {2,1})

What do we use in practice? All three, for different purposes.

* A matrix we use if we want to derive things analytically (because of matrix multiplication).
* A list we use when we want to represent our netweok in a computer program (most algorithms require only that we can retrieve neighbours of a given node – optimal time complexity)
* Edge list is used when we want to store the network in a file (easy to add, remove, efficient storing and manipulation).

Slika, ki vsebuje besede besedilo, posnetek zaslona, pisava, številka

Opis je samodejno ustvarjen

In practice we search for a format to store both directed and undirected graph in the same form. The best for this is an adjacency list.

Storing a graph in a file:

Slika, ki vsebuje besede besedilo, diagram

Opis je samodejno ustvarjen

Where do we get the data? Online sources, present in many standard datasets, popular network repositiores/collections

## Erdos-Reny random graph

Do determine what the coefficients tell us we need a “zero” to compare to. We get that with random graphs.

Graph model is an ensemble of random graphs. We talk about a graph model with n nodes and m edges. The random graph models is a collection of all the possible graphs with those restrictions. We do reasoning over all the possible graphs.

It is an algorithm, generating a random graph, given some parameters. This gives us a baseline to realize what statistics happen at random and which do not. We also use it for generate random graphs of different sizes and densities.

Erdoš-Reny model is the baseline for random graph.

We differentiate between G(n,m) and G(n,p) models:

**G(n,m):** n is number of nodes, m is number of edges. Randomly place m links between all (n over 2) node pairs. Computationally convenient, analytically hard (O(m) = O(n<k>)).

Slika, ki vsebuje besede besedilo, pisava, posnetek zaslona, bela

Opis je samodejno ustvarjen

**G(n,p):** n is number of nodes, p is some probability that is set in advance. Place a link between all the possible random node pairs based on probability p. Computationally hard but analytically convenient (Complexity is quadratic).

Slika, ki vsebuje besede besedilo, pisava, bela, posnetek zaslona

Opis je samodejno ustvarjen

What is the expected number of edges that we will get with set n and p?

Slika, ki vsebuje besede besedilo, pisava, rokopis, vrstica

Opis je samodejno ustvarjen

What is the probability of m? P(m) =

== we have to consider all possible places \* m coins must have said “yes – make a link” \* all the others must have said “no don’t”

== ((n choose 2) choose m) \* p ^m \* (1-p)^((n choose 2) – m).

Parameter p turns out to be the density of the network. We figure that out with throwing some formulas around.

We are working with binomial distribution – a node can in theory connect to all the other n-1 nodes. Expected number of links is (n-1)\*p. Most of the nodes will have the degree similar to the average. Because of the binomial coefficient this is hard to work it. We use Poissons distribution as an approximation. It is much easier to work with, but we have to make sure that the number of nodes is much higher than the average degree. On the picture we can see how we can adjust the top formula to get the second formula which represents the Poissons distribution. There is no n in the formula! That means that we expect the same distribution, no matter the size of the network (as long as n is big enough), it is only reliant on the degree.

Slika, ki vsebuje besede besedilo, posnetek zaslona, pisava, vrstica

Opis je samodejno ustvarjen

Degree distribution plotted on a log scale is a line. According to degree distribution, real networks are not random graphs. We will discuss the reason behind this later on. In a random graph there are no hubs (nodes with a very high degree), while we do observe them in a real network.

We are analytically trying to derive what is the size of the biggest connected component. What is the probability of a random node to be a part of this connected component. **S = ns/n**

In order for node I to be a part of the largest connected component there needs to be a link to the rest of the component. There has to be a node j that will connect to the i. This link must have formed with probability p. That node must also be part of the largest connected component – the probability is S. **S = p\*S**.

There are also other nodes that we have to consider. J’ is a random node, and be careful to not “double count the porbabilities”. It becomes a nasty long formula. Instead of computing

**S = p\*S + p\*s - …..**

that I am the part of the largest connected component, I rather compute what is the probability that I am not connected to the largest connected component. **1 – S = ((1-p)+p\*(1-S)^(n-1)**

**1 – S = (1 – p\*S)^(n-1)**

**Log(1 – S) = (n-1)\*log(1 – p\*S)** 🡨 use the Taylors approximation

**Log(1 – S) = -(n-1)\*p\*S**

**Log(1 – S) = - <K>\*S**

**1 – S = e^(-<K>\*S)**

**S = 1-e^(-<K>\*S)**

The formula seems to be recursive. Again, it does not depend on the network size. We cannot isolate S out of the formula here. The only thing to do is to resort to numerical simulations. We can consider S to be two functions. f(S)=g(S). Plot the functions and see where they intersect. That is the value we are looking for. F(S) is just a linear function. G(S) is a bit more complex. First graph show how f(s) and g(S) are plotted. If G starts above the F we have a solution (intercept), otherwise we do not. When K becomes larger than one, the large connected component emerges.

Slika, ki vsebuje besede besedilo, diagram, vrstica, grafični prikaz

Opis je samodejno ustvarjen

Slika, ki vsebuje besede besedilo, posnetek zaslona, vrstica, diagram

Opis je samodejno ustvarjen

The increase is not gradual! It is (semi)instant when K reaches the value of one.

1St homework: 7 excercises, specified what you need to submit.

1. **Mathematical** induction for the first part, for the second part use logical reasoning
2. Next week
3. How to **efficiently** (in constant time) sample
4. **Weak** and strong connectivity
5. Next two weeks
6. Figure out which network is which one (by structures)
7. Who to vaccinate? Wait until next weeks