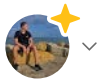
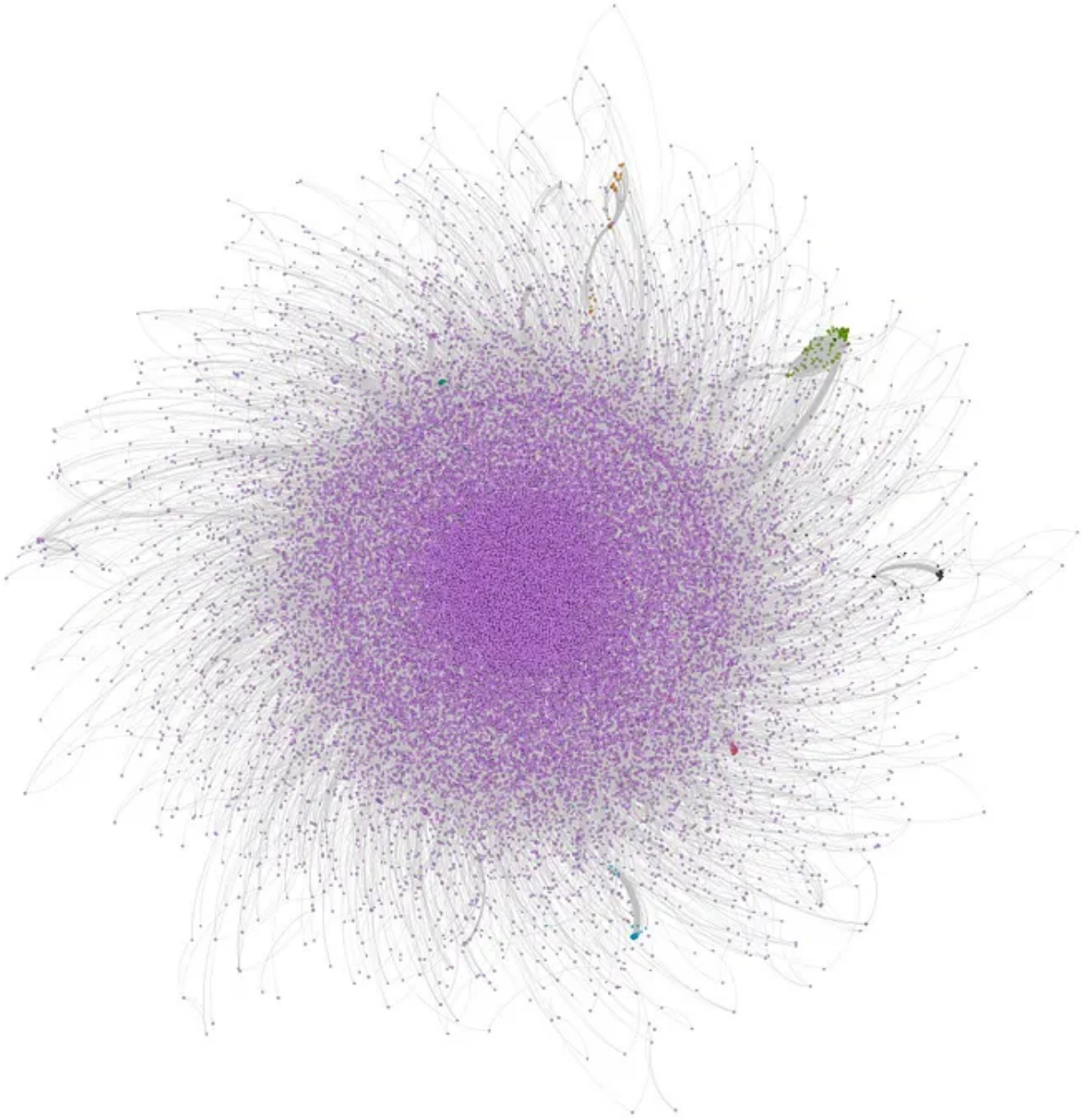




✦ Member-only story

Graph Representation Learning

Marco Brambilla · Follow
Published in Towards Data Science
5 min read · Dec 13, 2017



Graph captured on the Floating Piers study conducted in our data science lab.

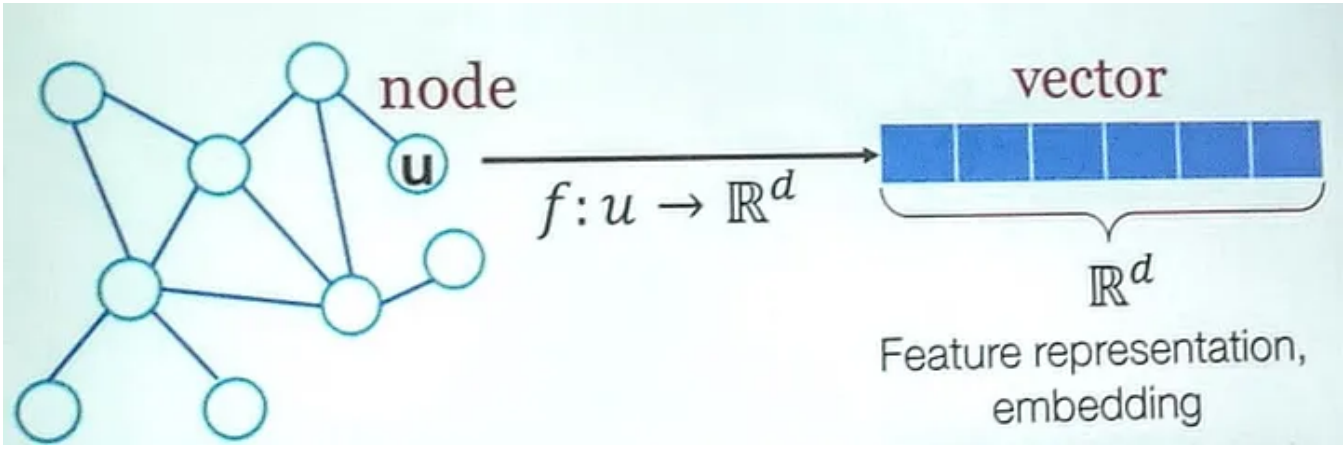
Graph models are pervasive for describing information across any scientific and industrial field where complex information is used.

The classical problems that need to be addressed in graphs are: node classification, link prediction, community detection, and many others.

When networks are huge, analysing them can become challenging. In general, machine learning techniques don't work well on networks.

Therefore, the main point becomes to **learn features of nodes in networks, considering the actual network structure.** These are called feature representation or embedding. As a result, any node is described as a vector.

The goal is to map each node into a low-dimensional space, while preserving most of the network information.



Based on this low-dimensionality representation, I can then run any analysis downstream.

The main challenge is that the structure of the network is very irregular, if compared with images, or audio, or text. Images can be seen as rigid graph grids. It's easier to run machine learning.

Graphs instead are non-Euclidean: the number of nodes are arbitrary and their connections are too. So, how to feed it into a neural network, for instance? That's why we need of graph embedding.

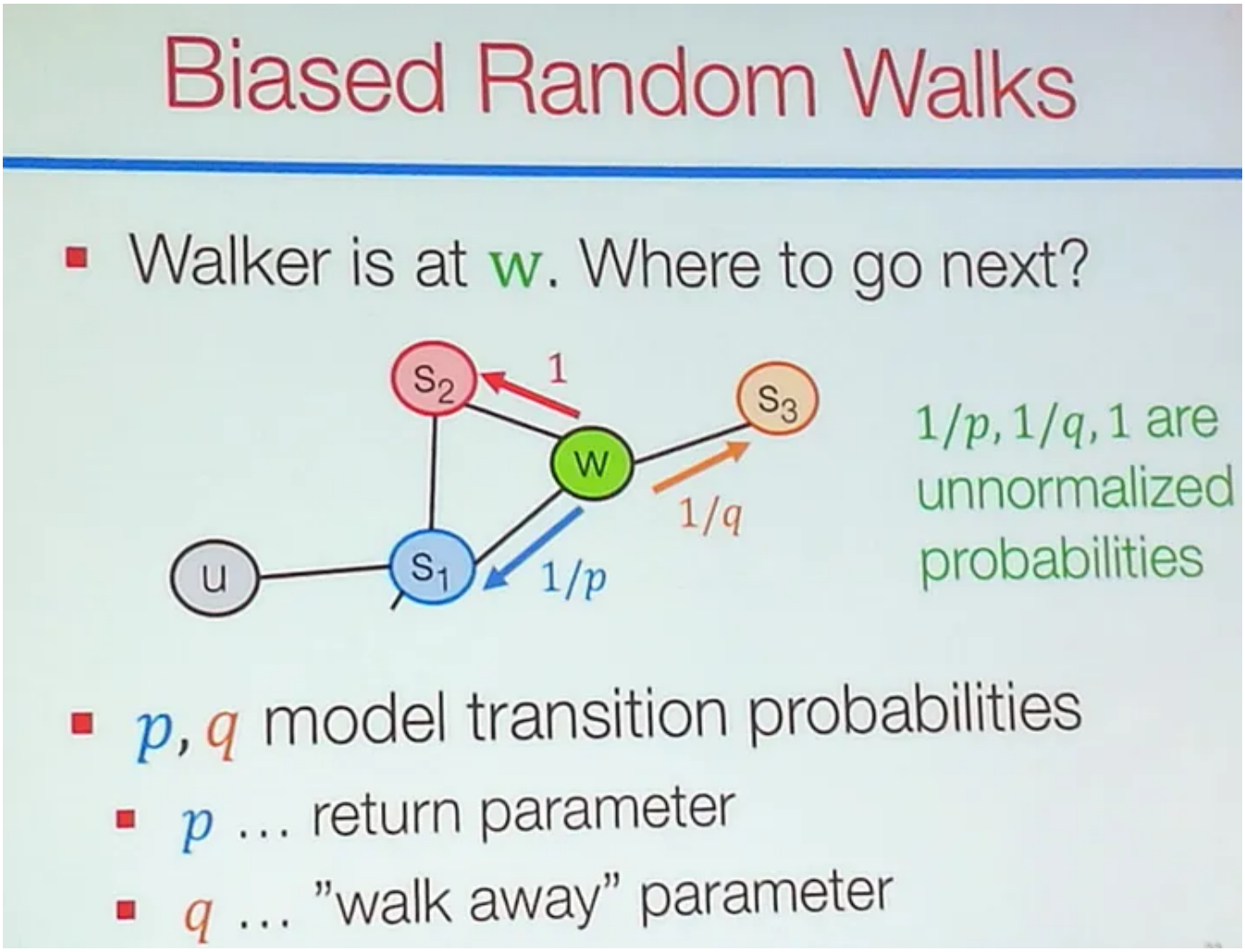
node2vec: Unsupervised feature learning

The main idea is to find embedding of nodes of dimension d that preserve similarity. The target is also that the embedding is such that nearby nodes are close together. I need to calculate the neighbourhood of a node, and then run a maximization problem that calculates the feature learning of node u in a way that

f(u) is predictive of the neighbourhood *N(u)*. This can be done through softmax functions, assuming factorization (independence) of nodes.

The definition of the neighbourhood can be based on BFS (Breadth-first) or DFS (Depth-first) strategies. BFS provides a local micro-view of the network, while DFS provides a macro-view of the network. A smart solution is an interpolation of BFS and DFS. It's a second-order random walk, because we remember the last visited node and at every step we take a decision on whether to go back to the last visited node or keep going to a new node. Essentially, at every step I have three options on where to move:

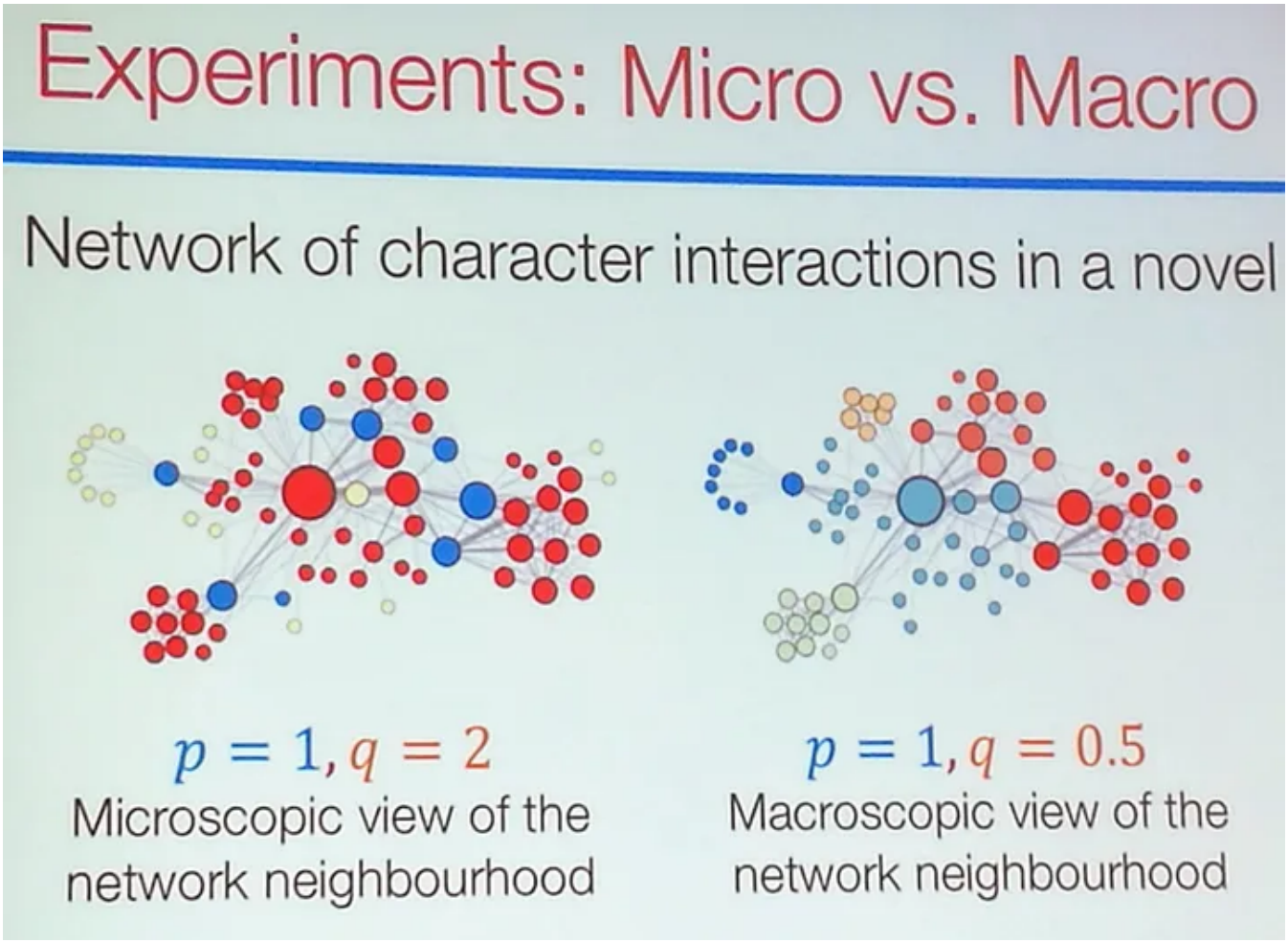
- node back closer to the original node u
- node at the same distance of the current node with respect to u
- node farther away from node u



I can set the probabilities p (return) and q (walk away) for deciding whether to go far or close to u at every step. I can then run a stochastic gradient descent (linear-time

complexity) algorithm to find the best paths.

Based on how I set the values of p and q, I can get completely different interpretation of the network: with bigger q, I tend to detect "roles" of nodes, with smaller q, I tend to detect "closeness" of nodes.

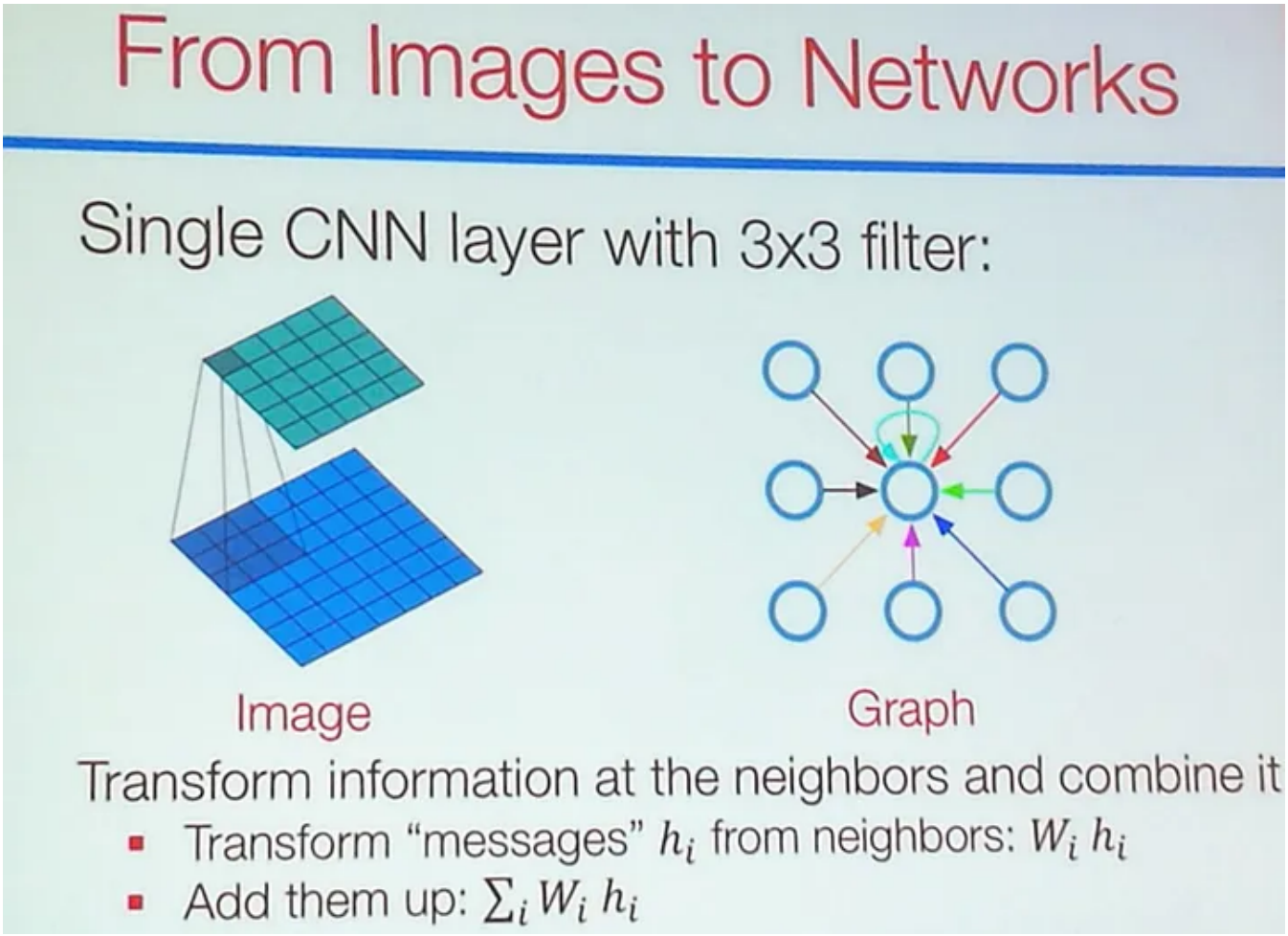


This is a task-independent feature learning approach, in the sense that it doesn't depend on the use you make of the result. The complexity is linear wrt. the size of the network.

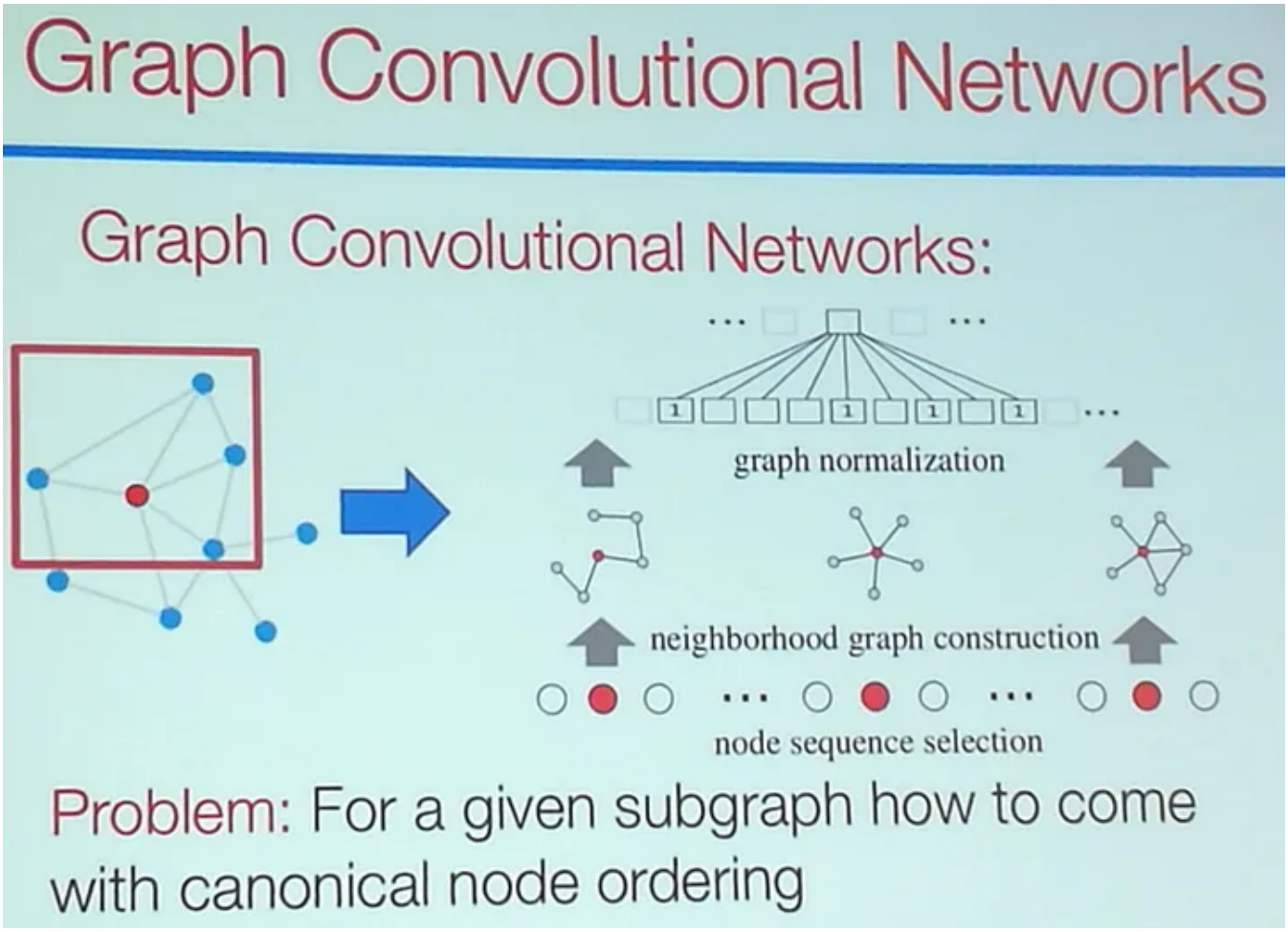
GraphSAGE: Supervised Feature Learning

This approach is instead inspired by convolutional neural networks, by generalizing them from the treatment of simple lattices (image grids) to general graphs. But how to generalize convolution?

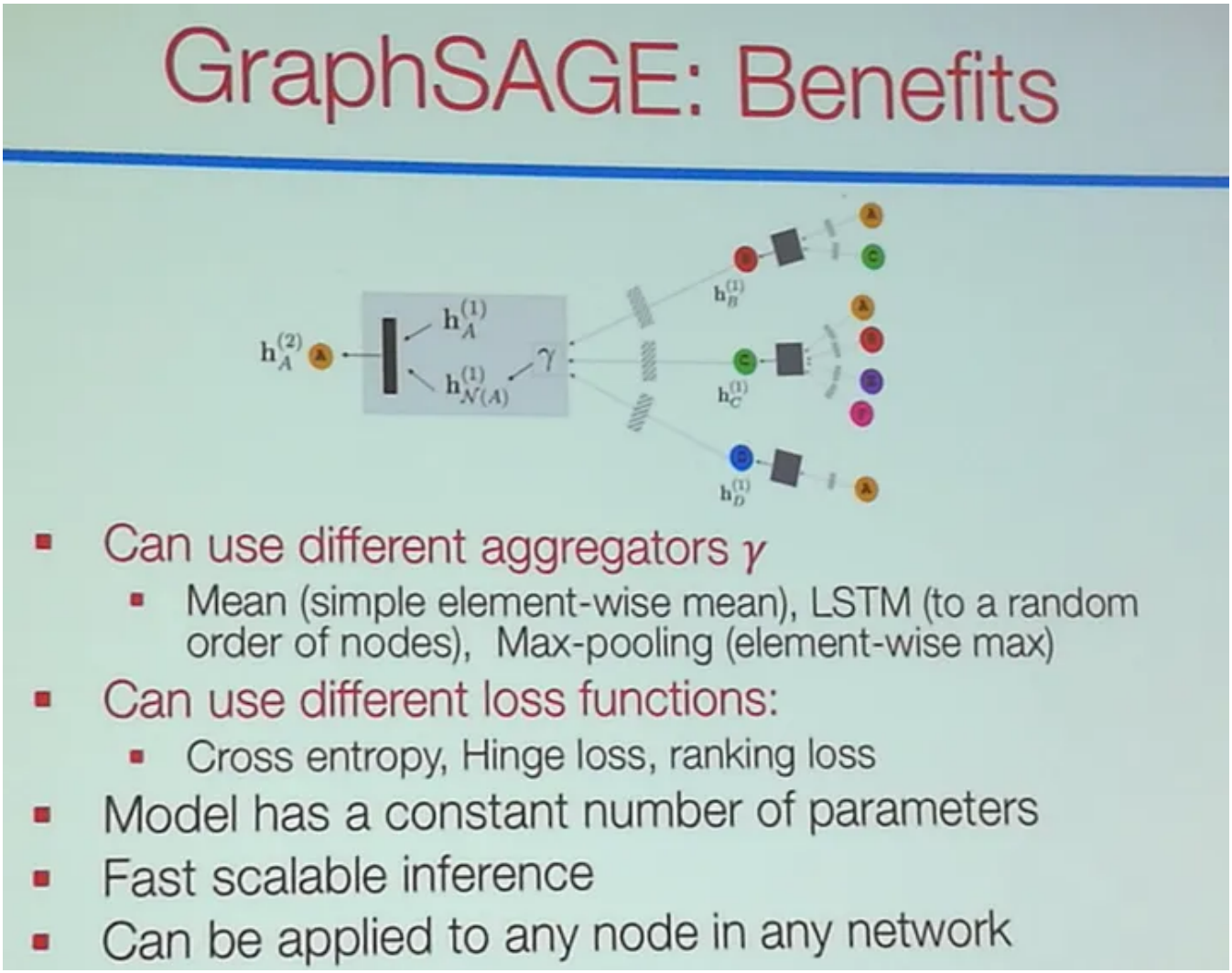
You can see the image as a graph where every pixel is connected (and influencing somehow) the close-by pixels.



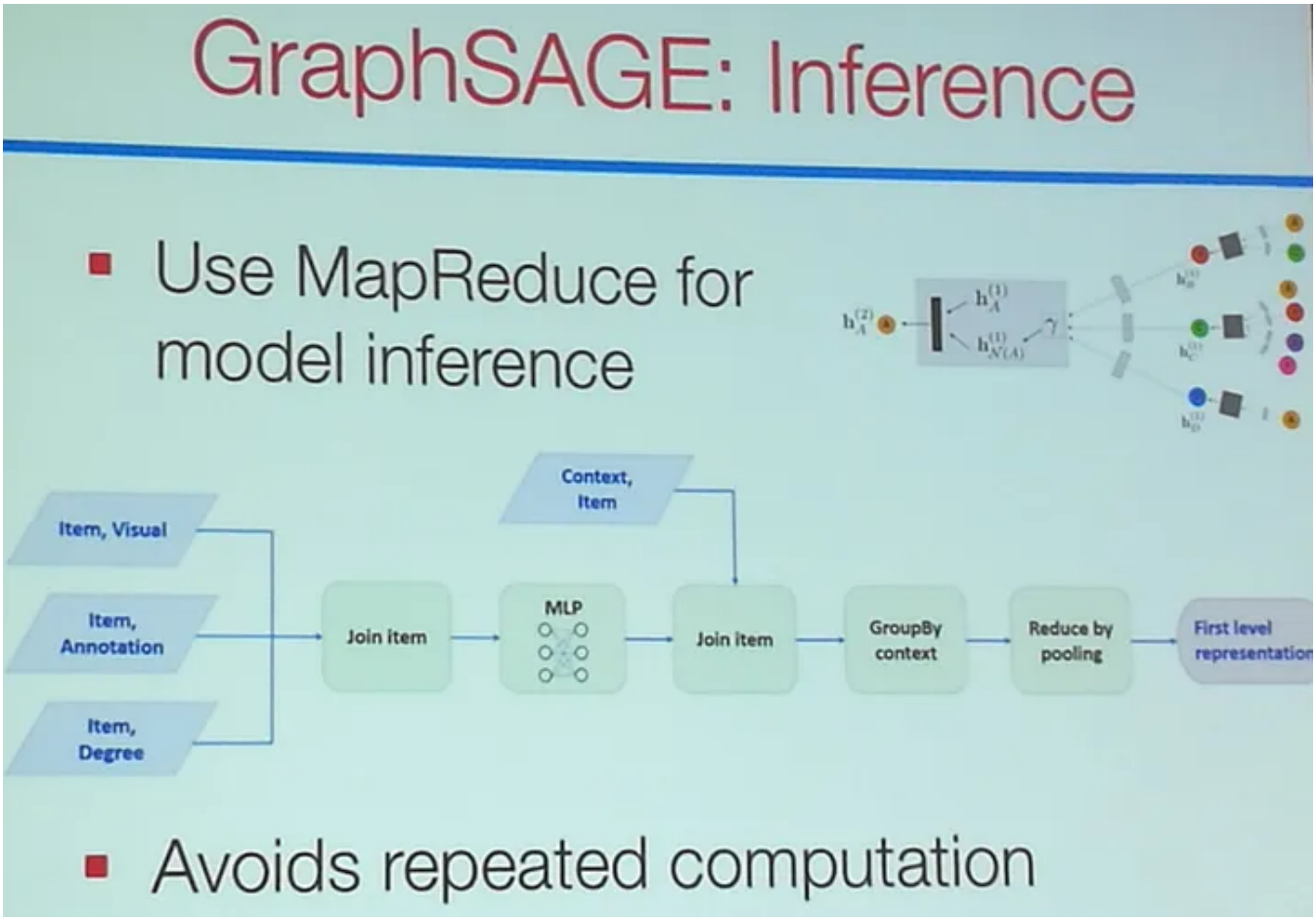
One option could be to use the adjacency matrix (plus possible features of the nodes) of the graph and feed it to a neural networks. Problem: the number of inputs is linear to the size of the network (huge).



One improvement is to use a subgraph (for instance, a patch of close-by nodes that defines a node neighborhood). This neighbourhood defines a computation graph, which I use for propagate and transform information (i.e., attributes) for all the neighbours to the node considered. Essentially, for every node the computation graph is a different neural network which I can run for learning the parameters. I have many neural networks, which use the same set of parameters, so it's like doing supervised learning.

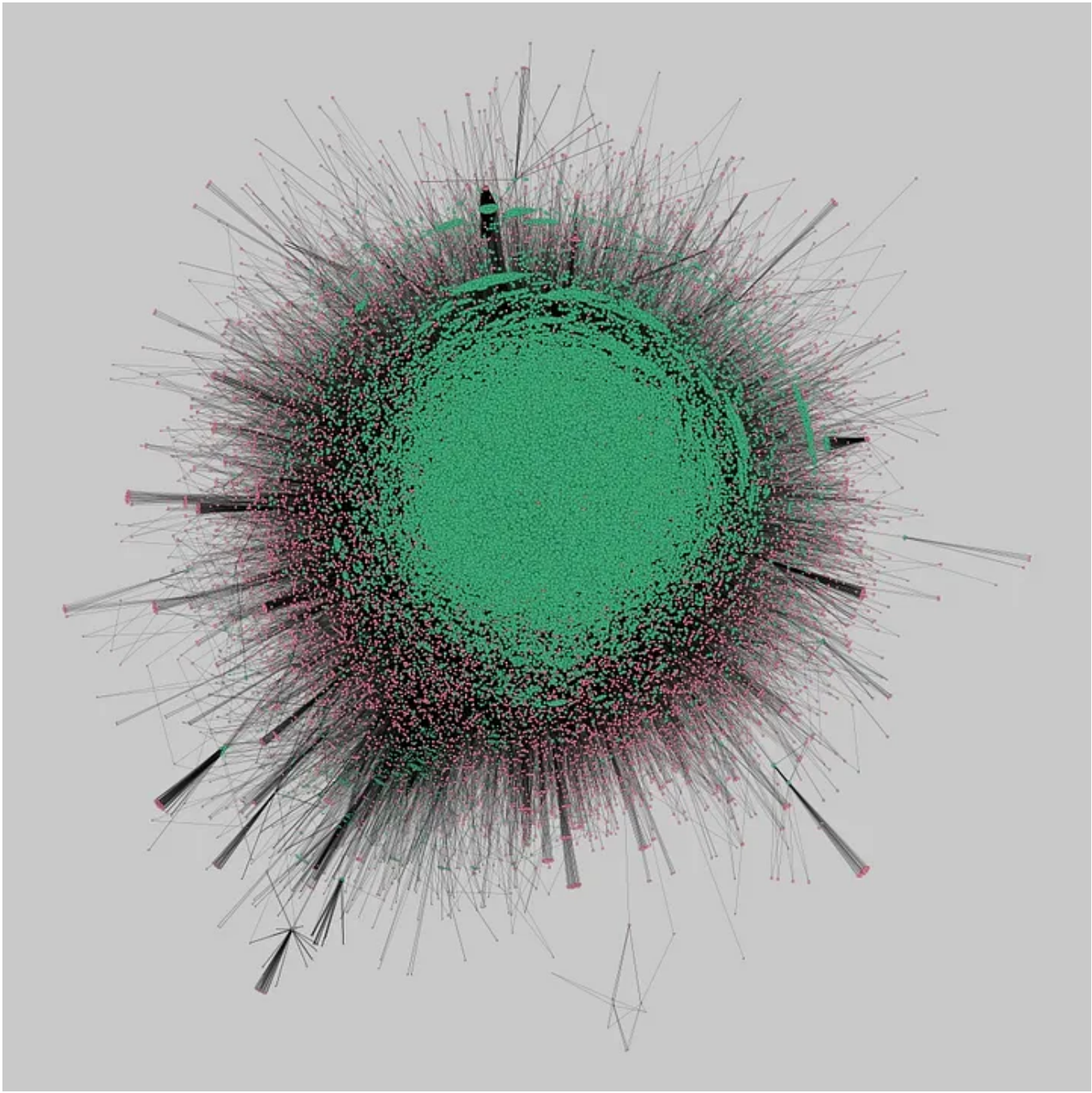


This whole framework can be conveniently executed in a MapReduce fashion, as reported below, because this automatically avoids replication of computation tasks.



Essentially, we defined a convolution network concept for graphs.

Experiments in Pinterest have run successfully over 3 BLN nodes and 17 BLN edges.



Graph captured on the Floating Piers study conducted in our data science lab.

This story is reporting on a keynote speech by **Jure Leskovec** (with Stanford University and Pinterest) given at the 4th IEEE BigData Conference, held in Boston in December 2017.

The full slidedeck of the presentation is available at Jure's home page:

http://i.stanford.edu/~jure/pub/talks2/graphsage-ieee_bigdata-dec17a.pdf

Machine Learning | Graphs | Neural Networks | Convolutional | Towards Data Science

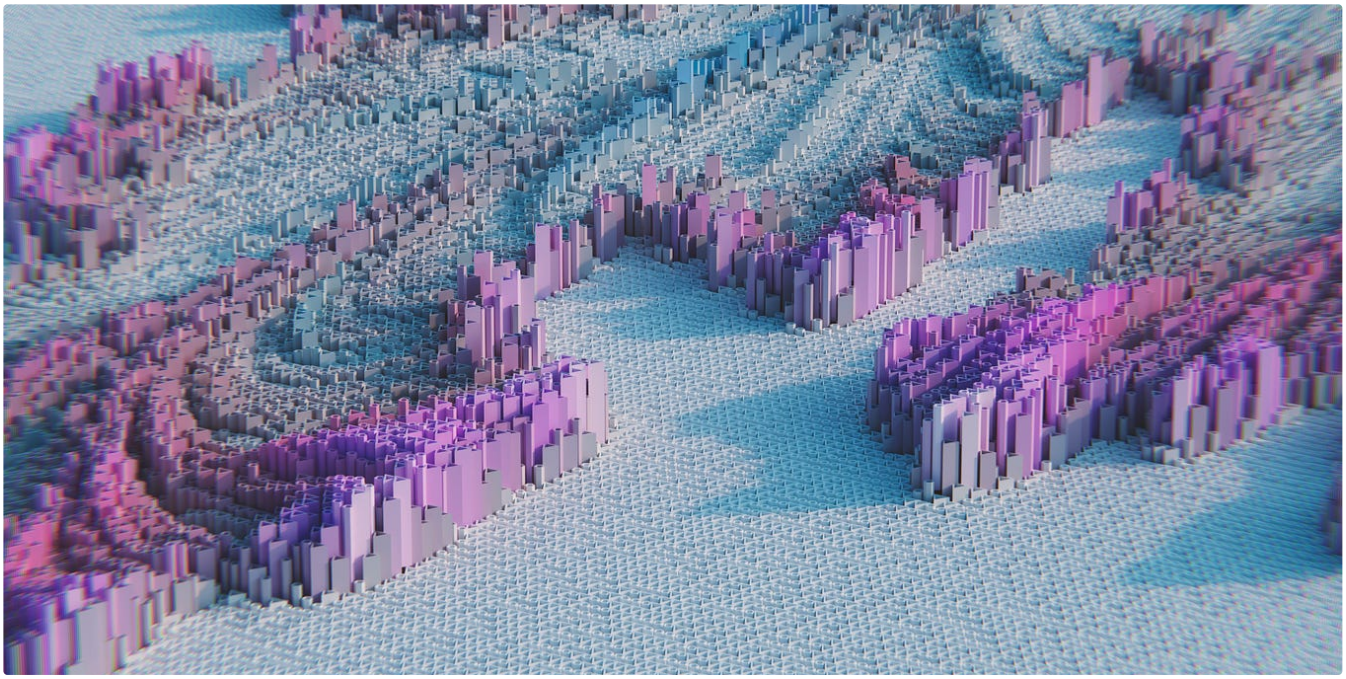
Follow

Written by Marco Brambilla

1.2K Followers · Writer for Towards Data Science

Data science, social and media analysis. Data, software and models all around.

More from Marco Brambilla and Towards Data Science



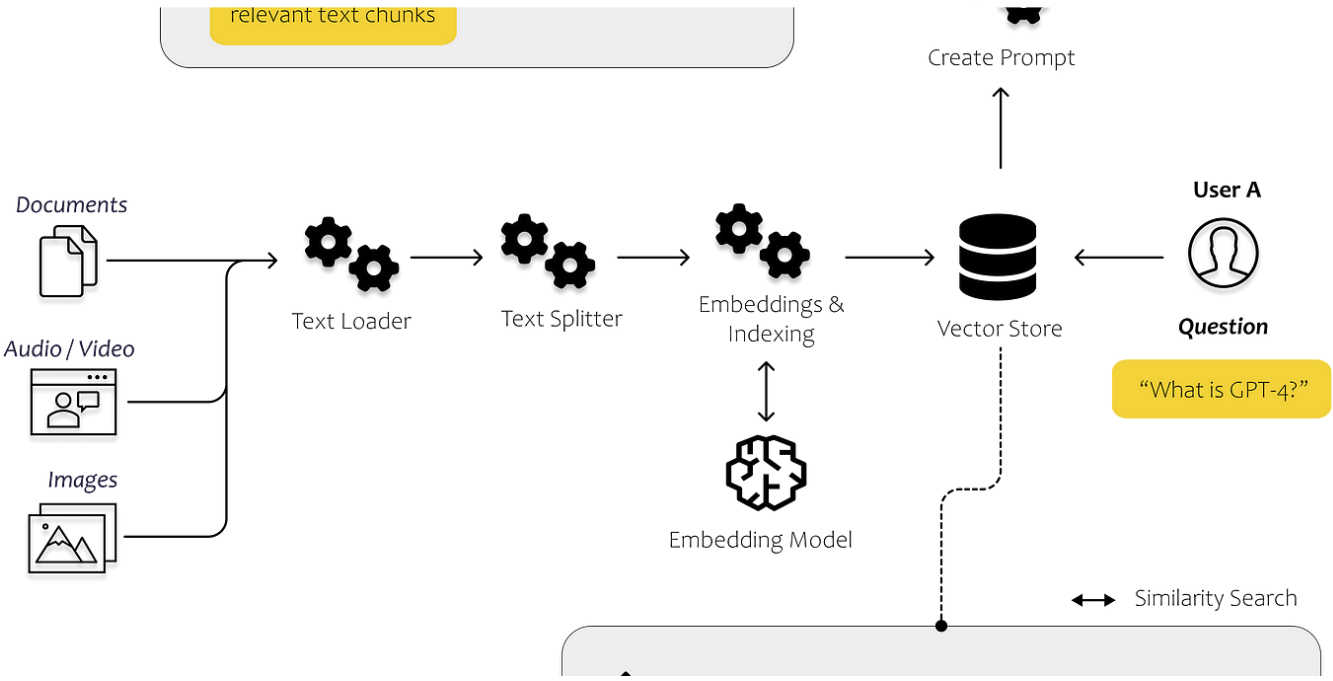
Marco Brambilla in Towards Data Science

The Path towards AI Regulation

The contribution of the AI Act its impact

· 7 min read · Mar 10

42



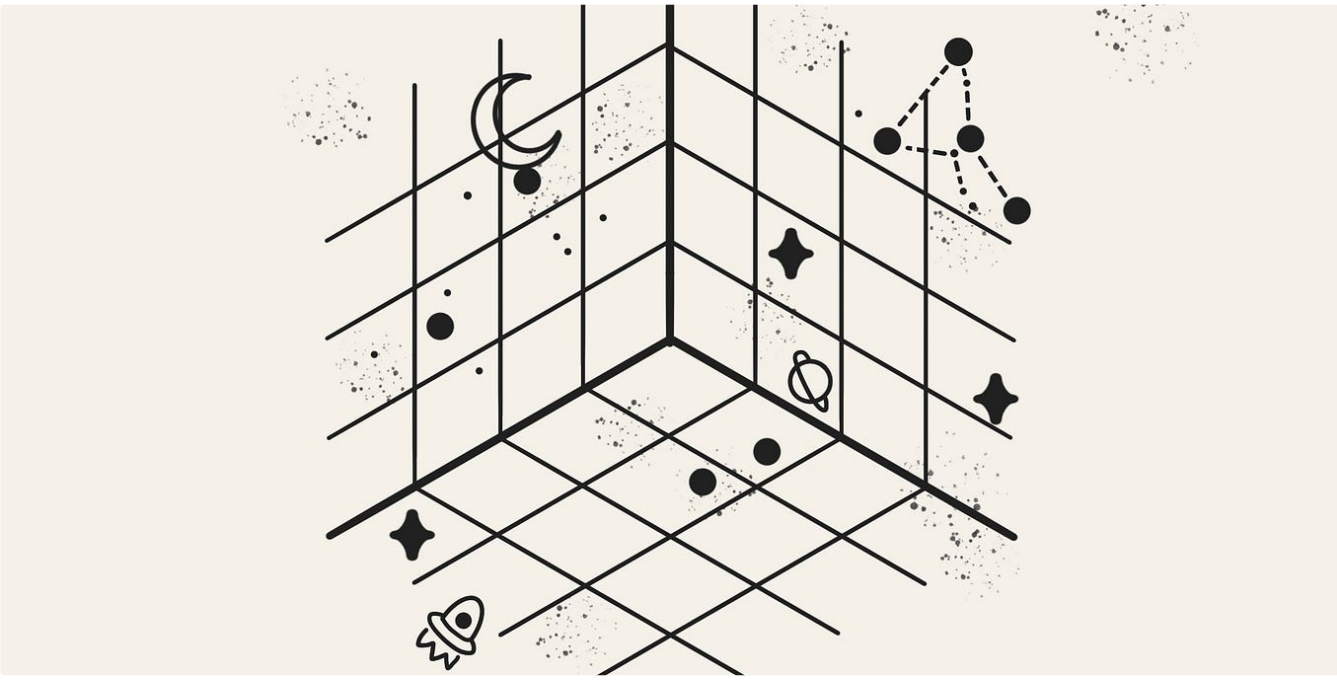
Dominik Polzer in Towards Data Science

All You Need to Know to Build Your First LLM App

A step-by-step tutorial to document loaders, embeddings, vector stores and prompt templates

· 26 min read · Jun 22

3.4K 31



Leonie Monigatti in Towards Data Science

Explaining Vector Databases in 3 Levels of Difficulty

From noob to expert: Demystifying vector databases across different backgrounds

· 8 min read · Jul 4

1.8K 18

Marco Brambilla in Towards Data Science

Large-Scale Analysis of On-line Conversation about Vaccines before COVID-19

Discussion over the role and need of vaccines has never been so strong. How was it before COVID-19? Social media can reveal that.

· 5 min read · Jan 30, 2021