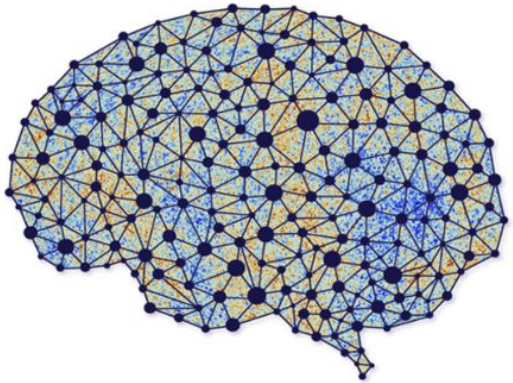
GNNs in Physics

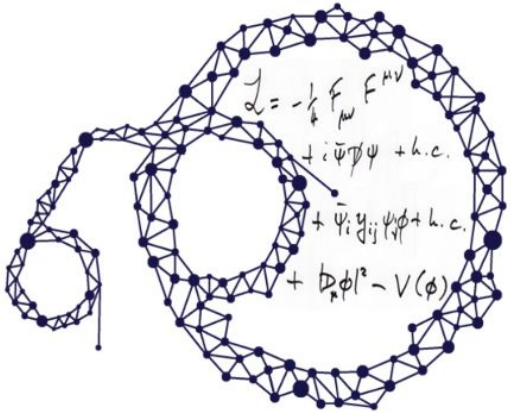
03/05/2024



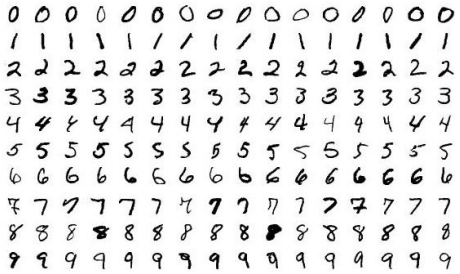
AI

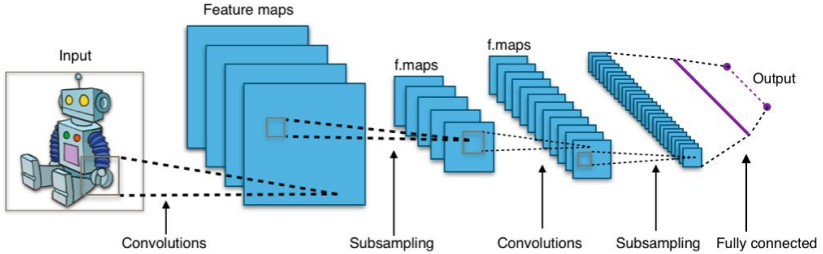
∩

Universe



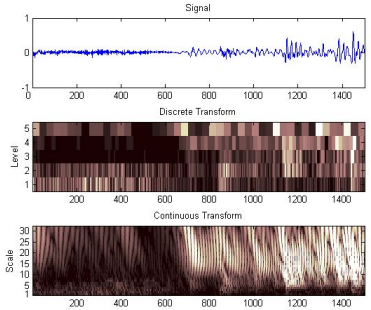
A lot of data live on grids.





Grids:

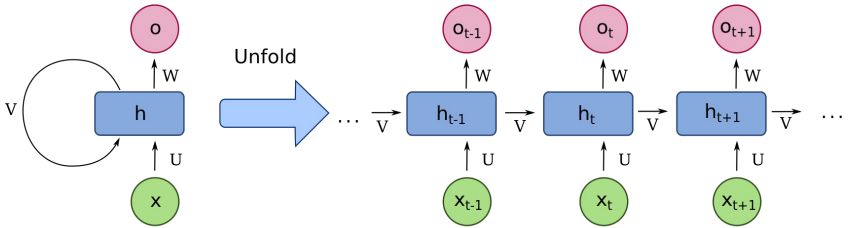
- Image: (HxWxC)
- Video: (TxHxWxC)



Time Series (also grids)

- Text: (N)-dim sequence
- Speech: (N)-dim sequence

“I love DL”



Images: Wikipedia

How about these:

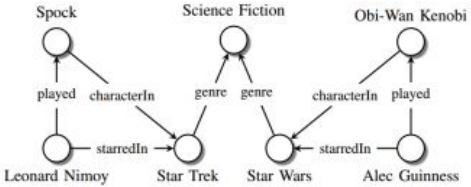
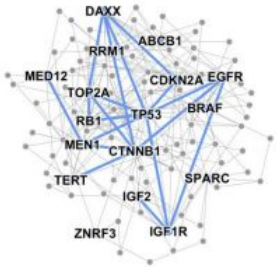


Image credit: Maximilian Nickel et al

Knowledge Graphs

Image credit: SalientNetworks

Computer Networks



Disease Pathways

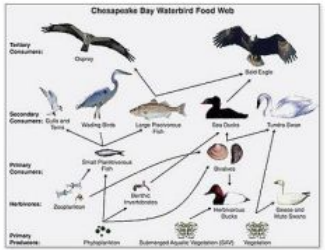


Image credit: Wikipedia

Food Webs

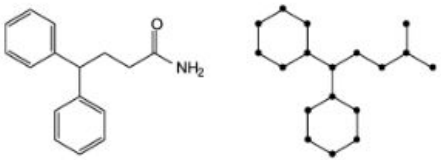


Image credit: MDPI

Molecules

Image credit: visitlondon.com

Underground Networks

https://snap.stanford.edu/class/cs224w-2023/slides/01-intro.pdf

Image credit: Medium

Social Networks

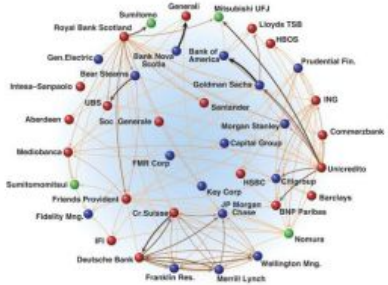
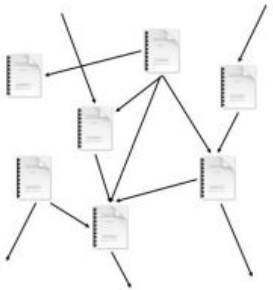


Image credit: Science

Economic Networks

Image credit: Lumen Learning

Communication Networks



Citation Networks

Image credit: Missoula Current News

Internet

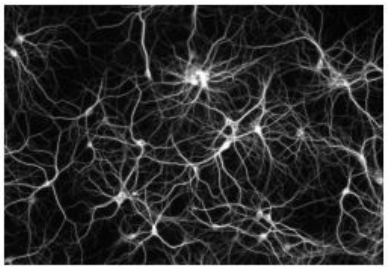


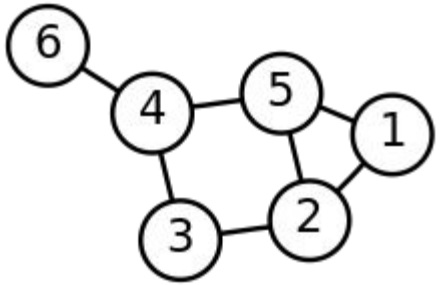
Image credit: The Conversation

Networks of Neurons

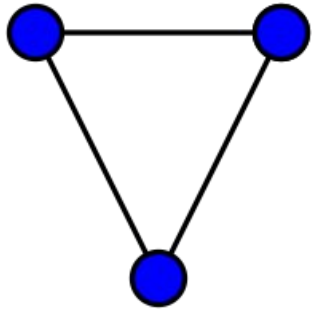
Data doesn’t always have a fixed structure. However, all those examples can be represented as graphs!

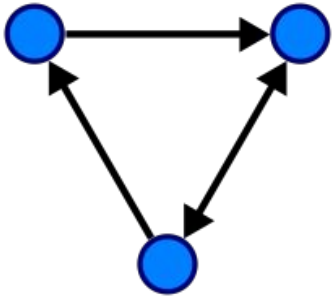
What is a Graph?

Graph - is a pair (V,E), where V - is a set whose elements are called *vertices* and E is a set of (un)ordered pairs of vertices {v1, v2}, whose elements are called *edges*. (Occasionally, this definition is being modified to include a *general feature* of the graph. In that case, graph is defined as a three-tuple (V,E,u).)



Im. Source: Wikipedia





Directed and undirected fully-connected graphs

ML with Graphs

The key objective is to provide a framework to learn (and predict) from graph-represented data.

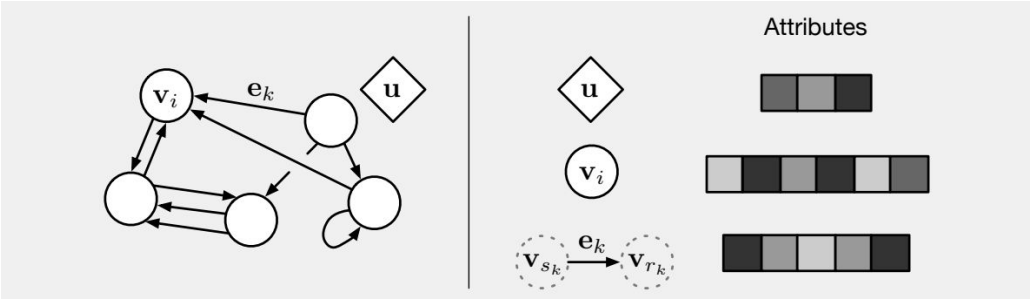
Examples include:

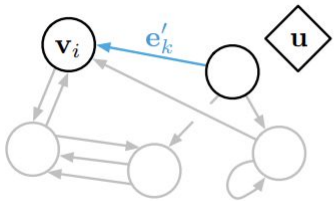
- node-level prediction (eg: predict a property of a given node (vertex))
- edge-level prediction (predict the connection bw. two given nodes)
- graph classification (predict a general property of a graph)
- graph generation

Message Passing Framework

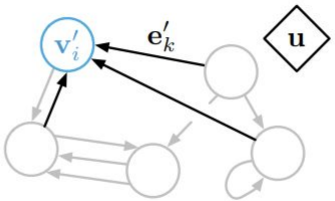
(Relational inductive biases, deep learning, and graph networks, https://arxiv.org/abs/1806.01261)

Message Passing Graph Neural Nets is a general class of architectures for learning from graph-represented data.

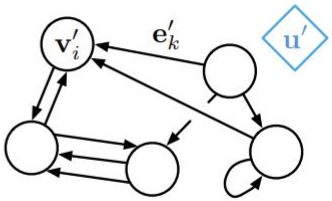




(a) Edge update



(b) Node update



(c) Global update

Algorithm 1 Steps of computation in a full GN block.

function GRAPHNETWORK(E, V, $\mathbf{u}$)

  for $k \in \{1 \dots N^e\}$ **do**

    $\mathbf{e}'_k \leftarrow \phi^e\left(\mathbf{e}_k, \mathbf{v}_{r_k}, \mathbf{v}_{s_k}, \mathbf{u}\right)$ ▷ 1. Compute updated edge attributes

  end for

  for $i \in \{1 \dots N^n\}$ **do**

    let $E'_i = \{(\mathbf{e}'_k, r_k, s_k)\}_{r_k=i,\ k=1:N^e}$

    $\bar{\mathbf{e}}'_i \leftarrow \rho^{e\rightarrow v}\left(E'_i\right)$ ▷ 2. Aggregate edge attributes per node

    $\mathbf{v}'_i \leftarrow \phi^v\left(\bar{\mathbf{e}}'_i, \mathbf{v}_i, \mathbf{u}\right)$ ▷ 3. Compute updated node attributes

  end for

  let $V' = \{\mathbf{v}'\}_{i=1:N^v}$

  let $E' = \{(\mathbf{e}'_k, r_k, s_k)\}_{k=1:N^e}$

  $\bar{\mathbf{e}}' \leftarrow \rho^{e\rightarrow u}\left(E'\right)$ ▷ 4. Aggregate edge attributes globally

  $\bar{\mathbf{v}}' \leftarrow \rho^{v\rightarrow u}\left(V'\right)$ ▷ 5. Aggregate node attributes globally

  $\mathbf{u}' \leftarrow \phi^u\left(\bar{\mathbf{e}}', \bar{\mathbf{v}}', \mathbf{u}\right)$ ▷ 6. Compute updated global attribute

  return $(E', V', \mathbf{u}')$

end function

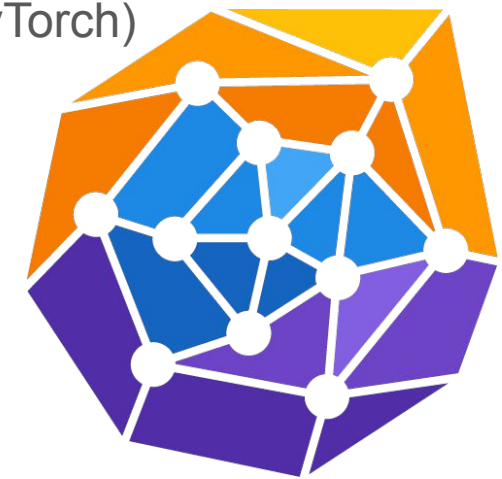
- Functions $\{\phi_e\ \phi_n\ \phi_u\}$ parametrize messages.
- Permutation-invariant aggregation functions ρ aggregate computed messages to get updated feature embeddings

Practical implementation

tfgnn (TensorFlow)

jraph (Jax)

PyG (PyTorch)



tensorflow/**gnn**

TensorFlow GNN is a library to build Graph Neural Networks on the TensorFlow platform.

31 Contributors · 17 Used by · 1k Stars · 163 Forks

Drive link