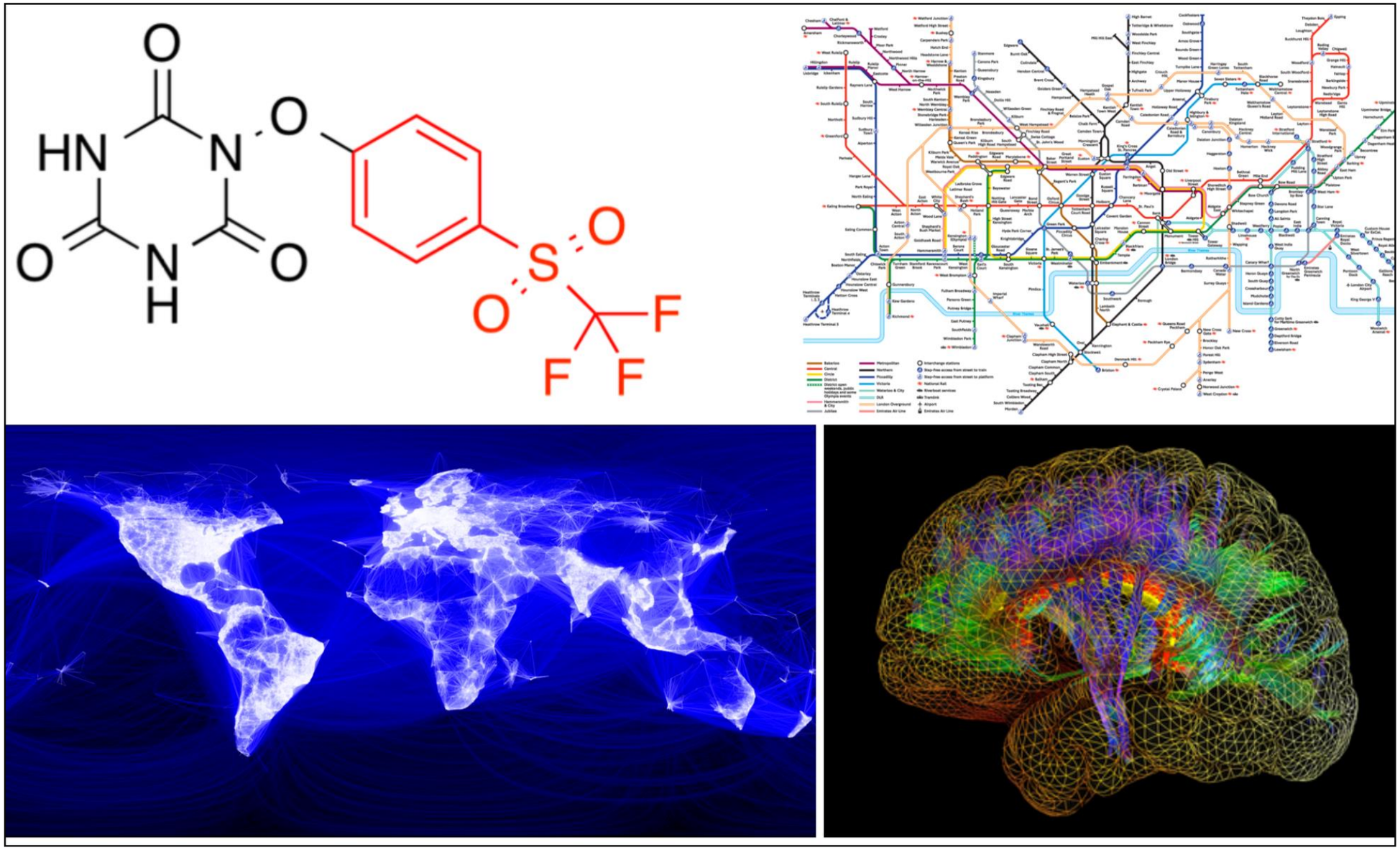
GRAPH ATTENTION NETWORKS

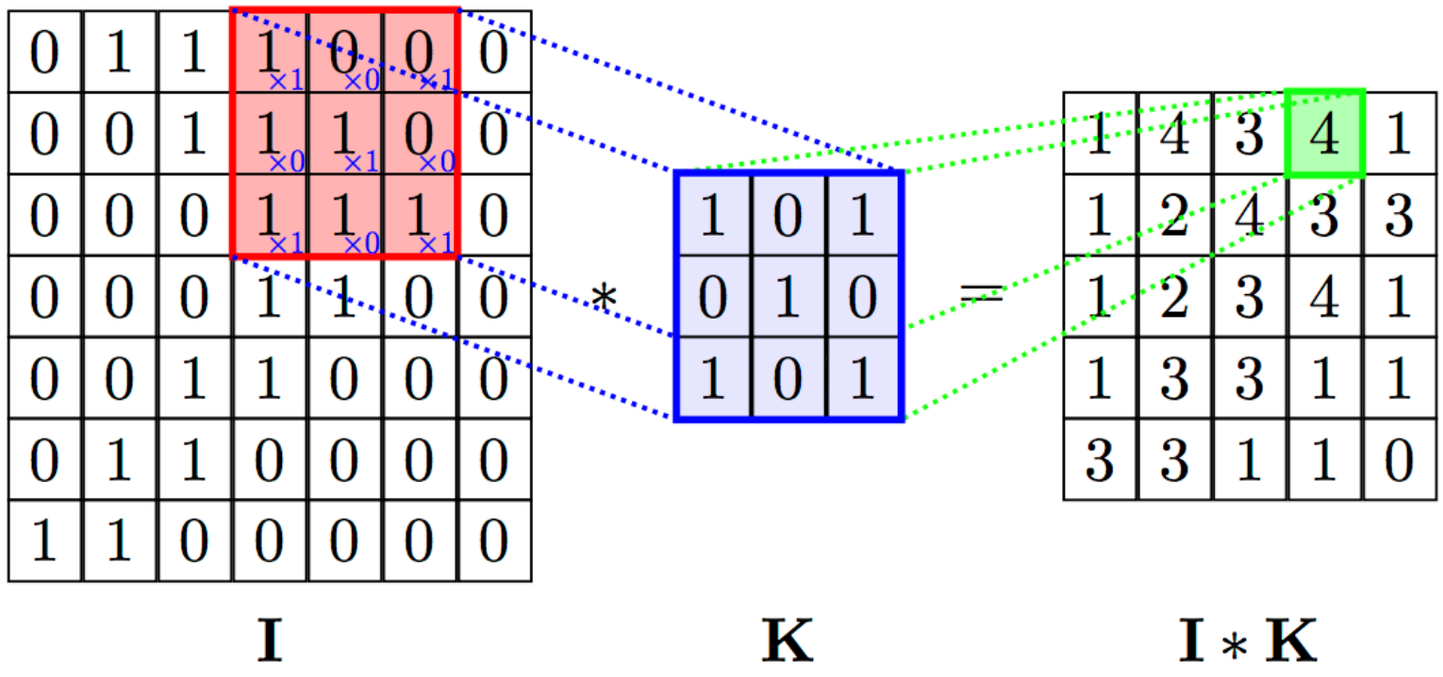
Станислав Рыбин, 151

Graph-structured inputs

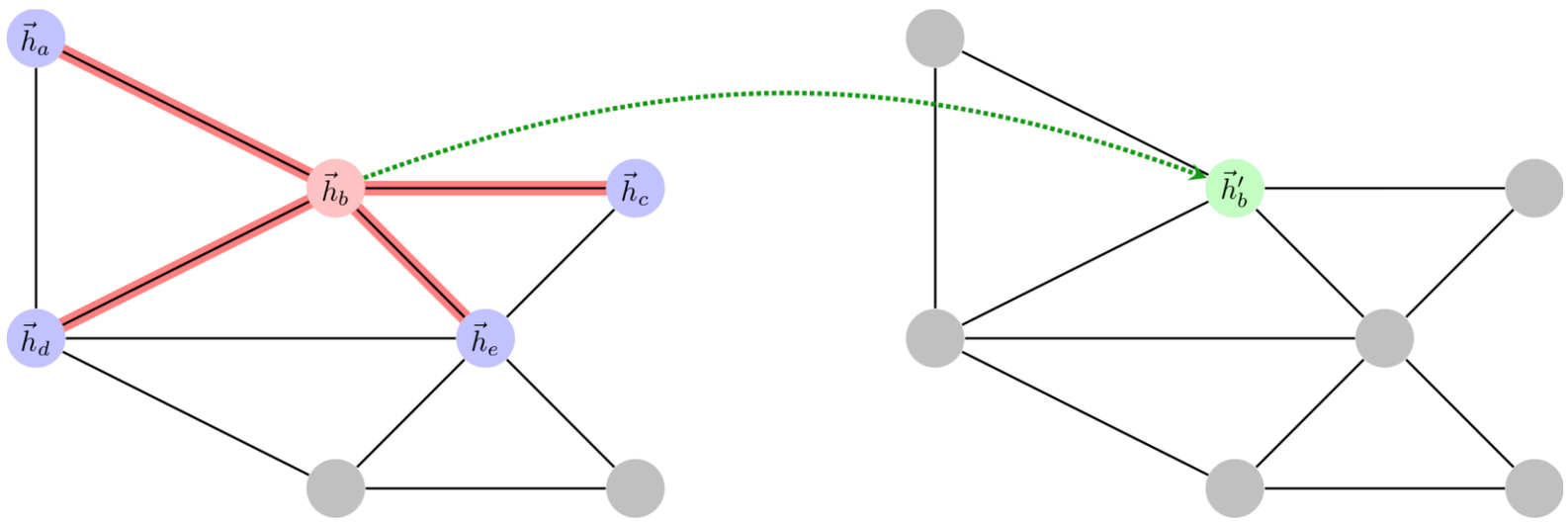


Motivating examples of graph-structured inputs: molecular networks, transportation networks, social networks and brain connectome networks.

Motivation for graph convolutions



2D convolutional operator as applied to a grid-structured input (e.g. image).



A desirable form of a graph convolutional operator.

Desirable properties of graph convolutional layer

- **Computational and storage efficiency** (requiring no more than O(V + E) time and memory);
- **Fixed** number of parameters (independent of input graph size);
- **Localisation** (acting on a local neighbourhood of a node);
- Ability to specify **arbitrary importances** to different neighbours;
- Applicability to **inductive problems** (arbitrary, unseen graph structures).

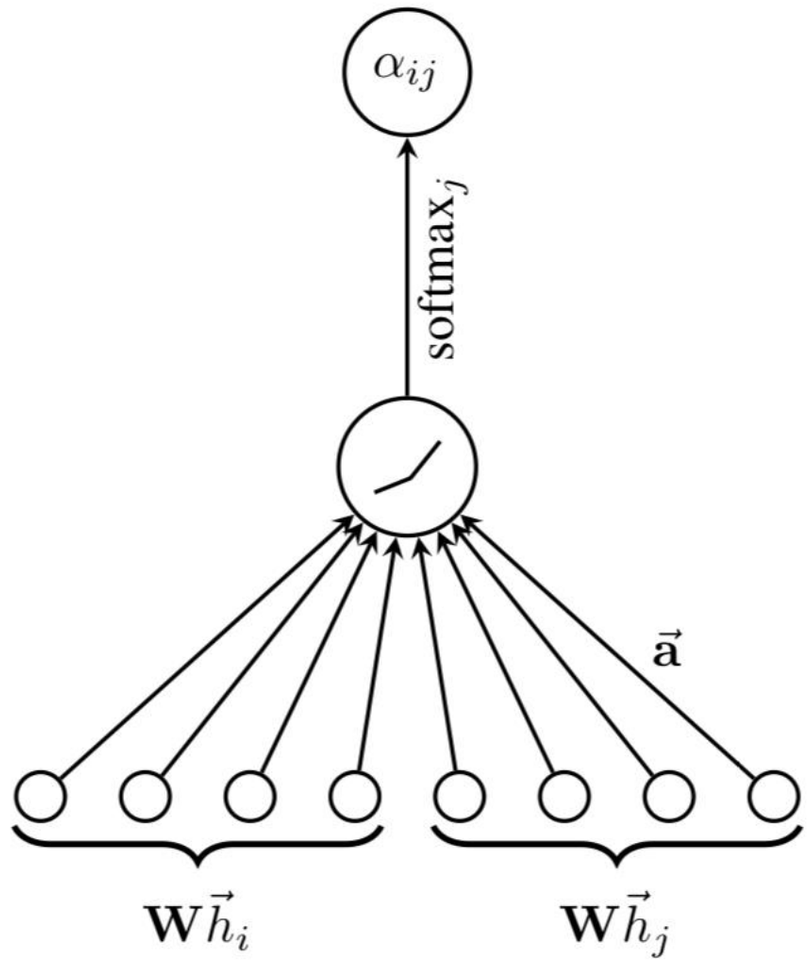
Viable graph convolution

- Graph of n nodes
- Layer input $\mathbf{h} = \{\vec{h}_1, \vec{h}_2, \ldots, \vec{h}_N\}, \vec{h}_i \in \mathbb{R}^F$
- Layer output $\mathbf{h}' = \{\vec{h}'_1, \vec{h}'_2, \ldots, \vec{h}'_N\}, \vec{h}'_i \in \mathbb{R}^{F'}$
- Shared node-wise feature transformation specified by a weight matrix **W**

$$\vec{g}_i = \mathbf{W}\vec{h}_i$$

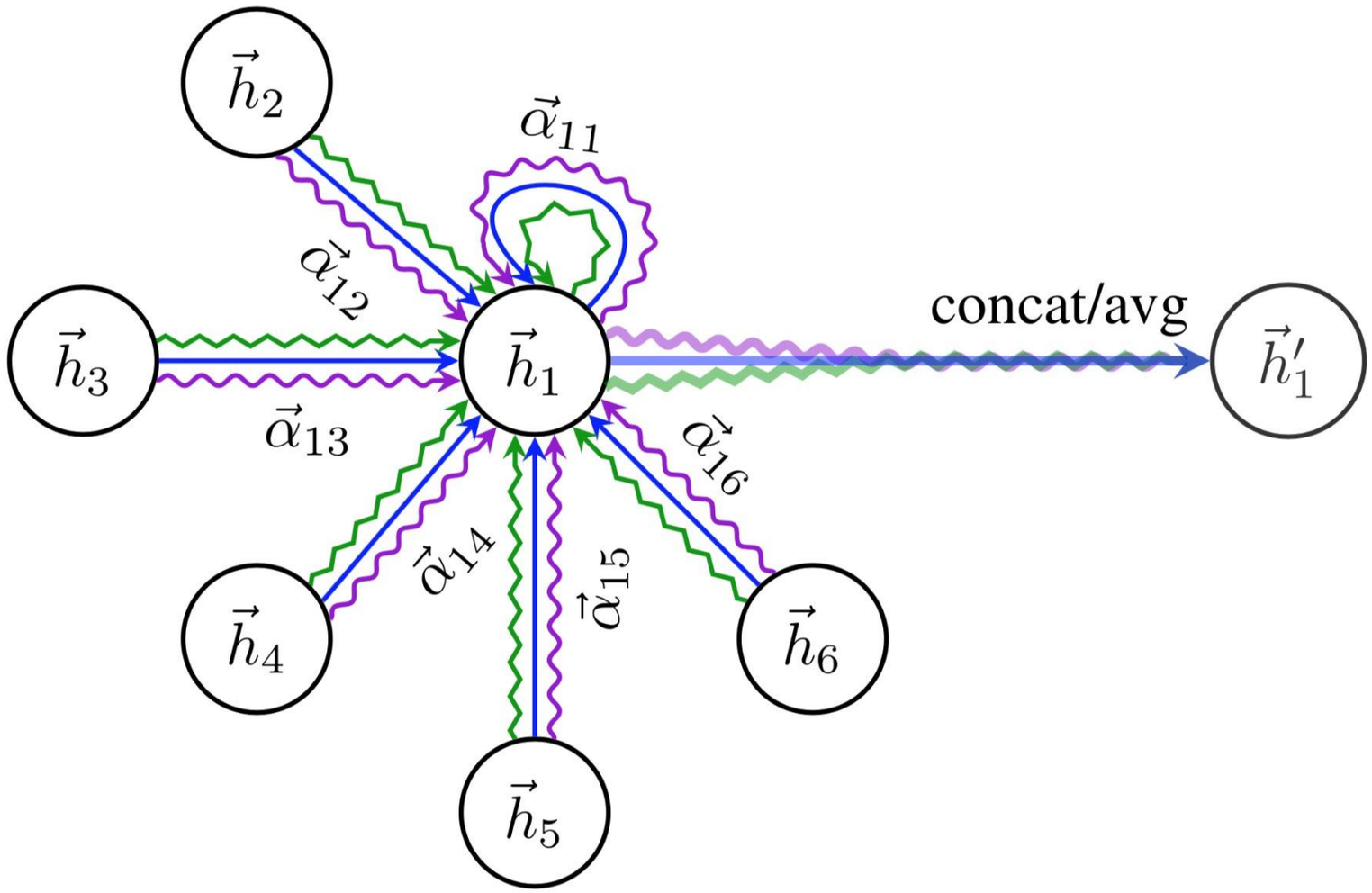
$$\vec{h}'_i = \sigma\left(\sum_{j \in \mathcal{N}_i} \alpha_{ij} \vec{g}_j\right)$$

The attention mechanism



$$e_{ij} = a(\mathbf{W}\vec{h}_i, \mathbf{W}\vec{h}_j) \qquad \alpha_{ij} = \mathrm{softmax}_j(e_{ij}) = \frac{\exp(e_{ij})}{\sum_{k \in \mathcal{N}_i} \exp(e_{ik})}.$$

The multi-head attention



$$\vec{h}'_i = \Big\Vert_{k=1}^{K} \sigma\left(\sum_{j\in\mathcal{N}_i} \alpha_{ij}^k \mathbf{W}^k \vec{h}_j\right)$$

Key properties

- **Computationally efficient**: the computation of attentional coefficients can be parallelised across all edges of the graph, and the aggregation may be parallelised across all nodes;

- **Storage efficient**: It is possible to implement a GAT layer using sparse matrix operations only, requiring no more than O(V + E) entries to be stored anywherle;

- **Fixed** number of parameters, irrespective of the graph’s node or edge count;

- Trivially **localised**, as we only attend over neighbourhoods;

- Allows for (implicitly) specifying **different importances** to **different neighbours**;

- Readily applicable to **inductive problems**, as it is a shared edge-wise mechanism and therefore does not depend on the global graph structure!

Summary of results

Method	Cora	Citeseer	Pubmed
MLP	55.1%	46.5%	71.4%
ManiReg (Belkin et al., 2006)	59.5%	60.1%	70.7%
SemiEmb (Weston et al., 2012)	59.0%	59.6%	71.7%
LP (Zhu et al., 2003)	68.0%	45.3%	63.0%
DeepWalk (Perozzi et al., 2014)	67.2%	43.2%	65.3%
ICA (Lu & Getoor, 2003)	75.1%	69.1%	73.9%
Planetoid (Yang et al., 2016)	75.7%	64.7%	77.2%
Chebyshev (Defferrard et al., 2016)	81.2%	69.8%	74.4%
GCN (Kipf & Welling, 2017)	81.5%	70.3%	**79.0%**
MoNet (Monti et al., 2016)	$81.7 \pm 0.5\%$	—	$78.8 \pm 0.3\%$
GCN-64*	$81.4 \pm 0.5\%$	$70.9 \pm 0.5\%$	**79.0** $\pm 0.3\%$
GAT (ours)	**83.0** $\pm 0.7\%$	**72.5** $\pm 0.7\%$	**79.0** $\pm 0.3\%$

References

1.Graph Attention Networks //
Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero,
Pietro Liò, Yoshua Bengio // https://arxiv.org/pdf/1710.10903.pdf