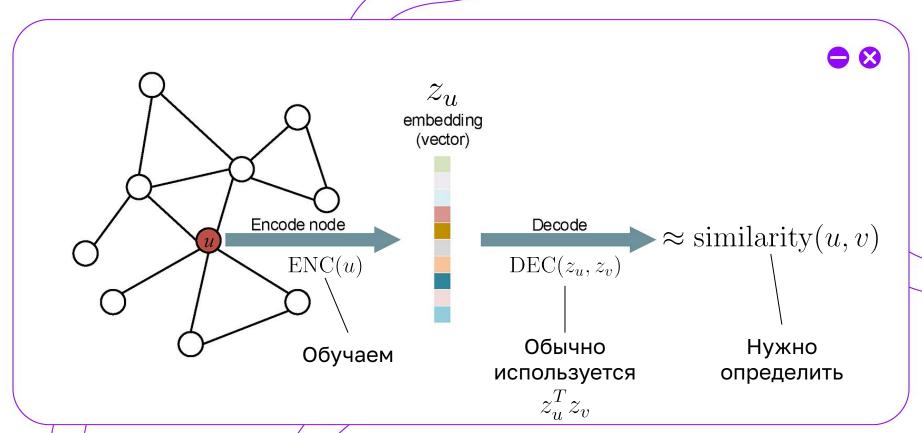
VITMO

Анализ графовых данных и глубокое обучение

Азимов Рустам Высшая школа цифровой культуры

В предыдущих сериях





Ограничения

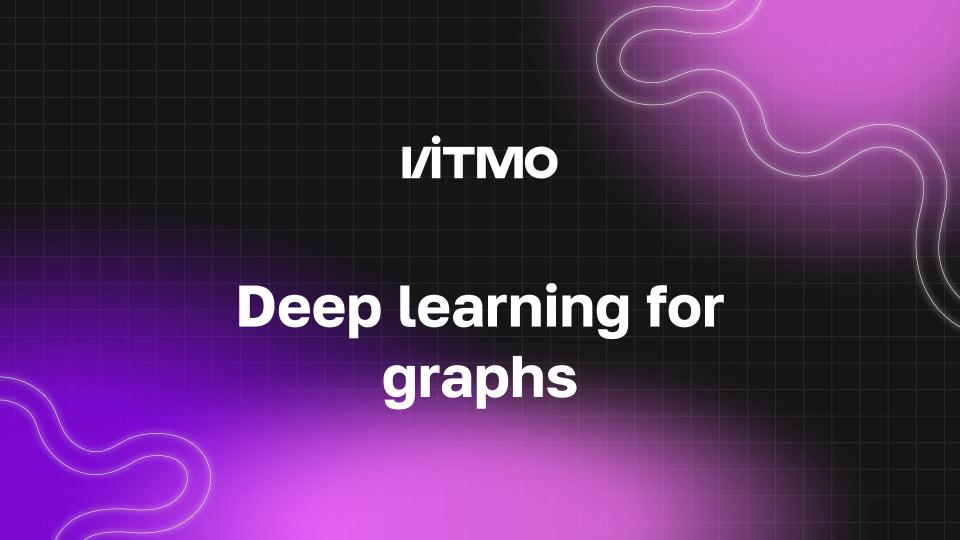


 Рассмотренные методы для получения эмбеддингов не позволяют работать с новыми вершинами





- Отсутствие общих параметров у эмбеддингов вершин, всего O(|V|d) параметров
- Не используют признаки вершин, рёбер, графов



Deep Graph Encoders



Мы рассматривали

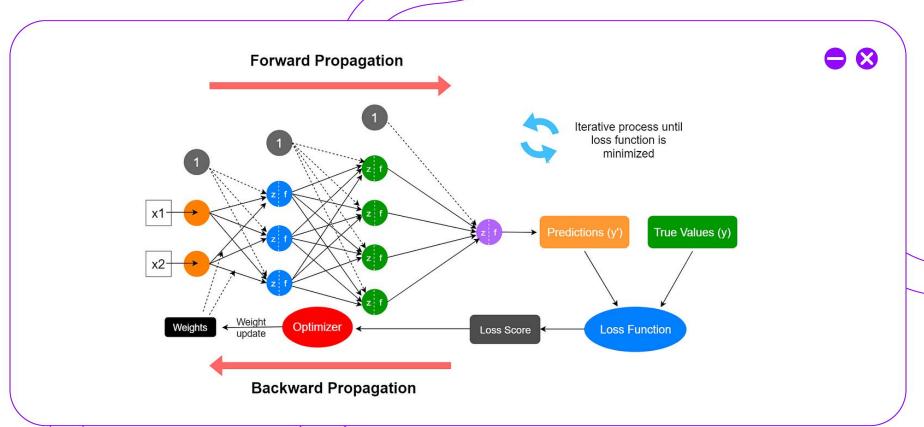
$$ENC(u) = Z_u$$

Основная идея теперь

 $\mathrm{ENC}(u)$ = MLP (основанный на структуре графа)

Deep Learning





Графовые данные



• Пусть имеется граф G с множеством вершин Vи матрицей смежности A



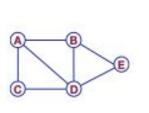
- N(v)– соседи вершины v
- ullet Будем рассматривать числовые признаки вершин $X \in \mathbb{R}^{|V| imes m}$
 - Информация о профиле в социальных сетях
 - Свойства молекул в биологических сетях
 - Если нету признаков, можно добавить one-hot кодирование вершин или вектор из единиц

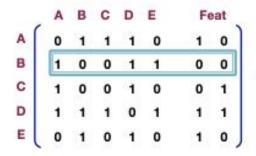
Наивный подход

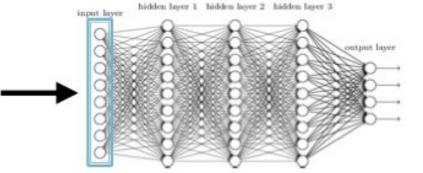


 Добавить к признакам матрицу смежности и скормить нейронной сети







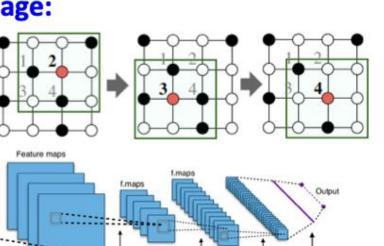


- Много параметров O(|V|)
- Зависит от размера графа и порядка вершин

Сверточные сети



 Идея — обобщить классические сверточные сети от изображений/текстов (простые графы) до произвольных графов CNN on an image:







Сложность





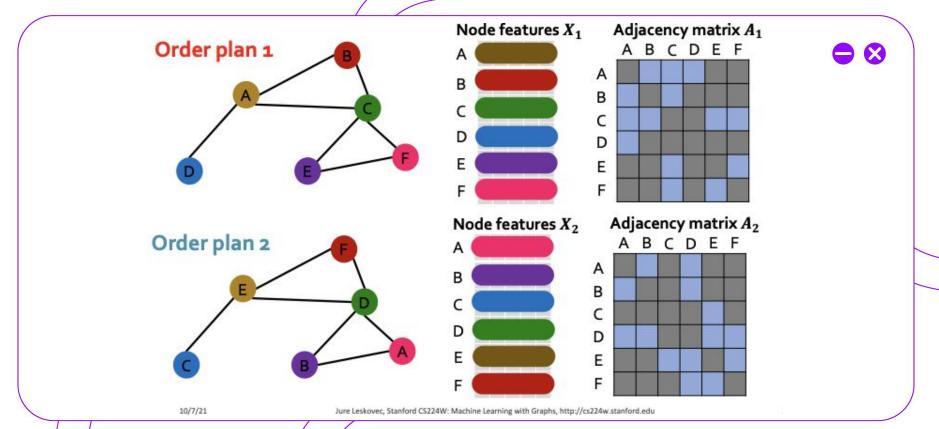




- Нету фиксированного определения локальности или скользящего окна по вершинам графа
- Граф инвариантен к перестановкам вершин
- Граф и представления вершин должны быть одинаковыми, не важно какой порядок обхода мы выбрали

Инвариантность к перестановкам

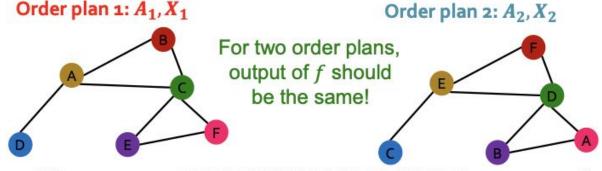




Инвариантность к перестановкам

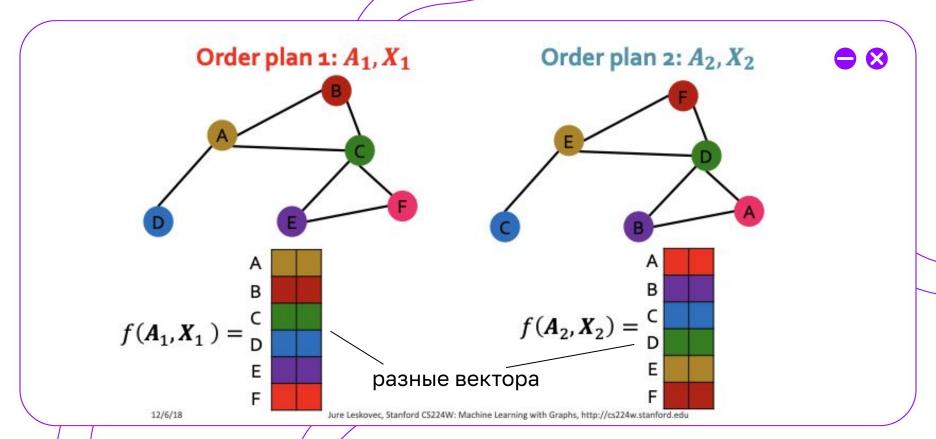


- ullet Пусть мы обучаем функцию f, которая отображает граф G=(A,X) ullet в вектор \mathbb{R}^d
- Тогда f инвариантна к перестановкам, если $f(A,X) = f(PAP^T,PX)$ для любой перестановки P
- ullet Например, $f(A_1, X_1) = f(A_2, X_2)$



Эмбеддинги для каждой вершины





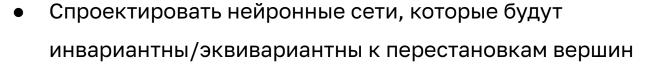
Эквивариантность к перестановкам



- ullet Пусть мы обучаем функцию f, которая отображает граф G=(A,X) ullet в матрицу $\mathbb{R}^{|V| imes d}$
- Тогда f эквивариантна к перестановкам, если $Pf(A,X) = f(PAP^T,PX)$ для любой перестановки P
- Например, GNN состоят из инвариантных и эквивариантных к перестановкам вершин функций

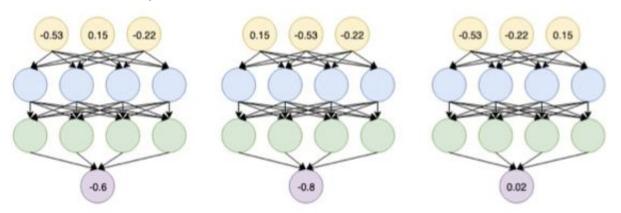
Цель







• Классические нейронные сети не являются таковыми, поэтому наивный подход не работает



VİTMO

Graph Convolutional Networks

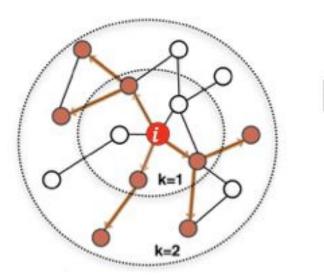
GCN

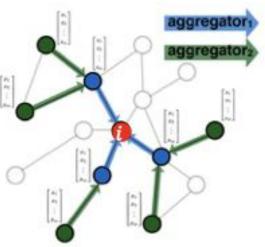


• Граф вычислений определяется соседями вершин



• Через них информация распространяется и агрегируется





Графы вычислений





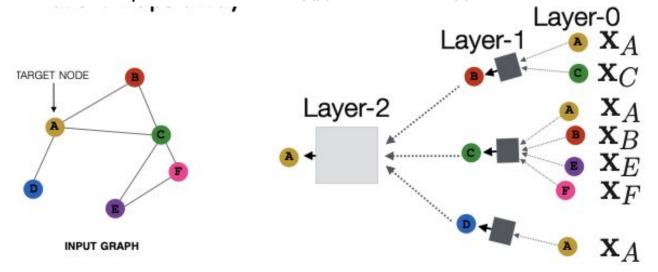
Глубина графа вычислений



 Глубина может быть любой, от нее зависит насколько далекое соседство рассматривается для распространения информации

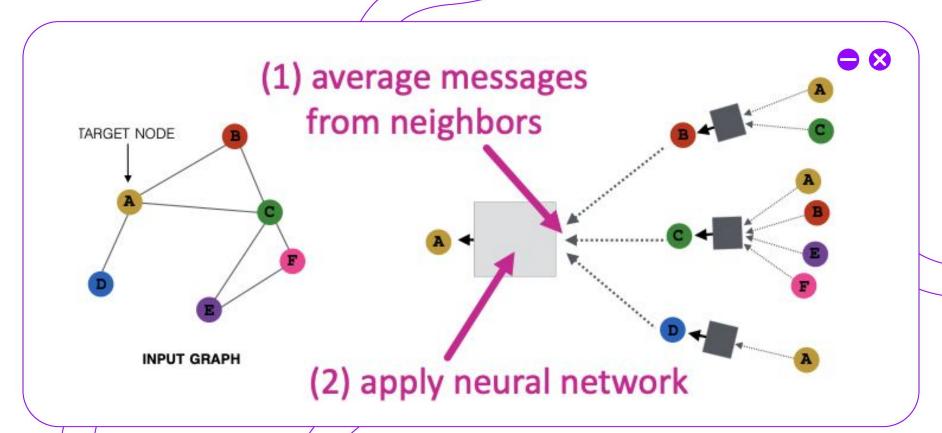


• Вершины имеют различные эмбеддинги на каждом слое



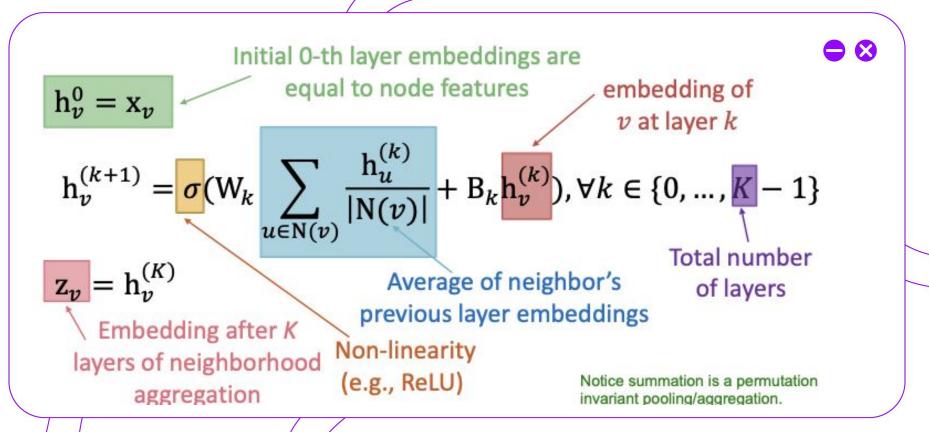
Базовый подход





Базовый подход



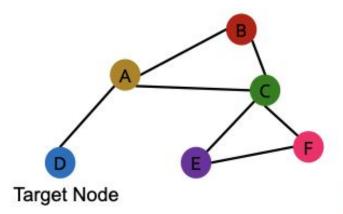


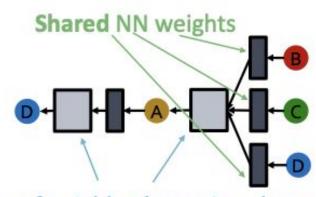
Свойства GCN для одной вершины **ИТМО**



GCN, вычисляющая эмбеддинг одной вершины, инвариантна относительно перестановок вершин



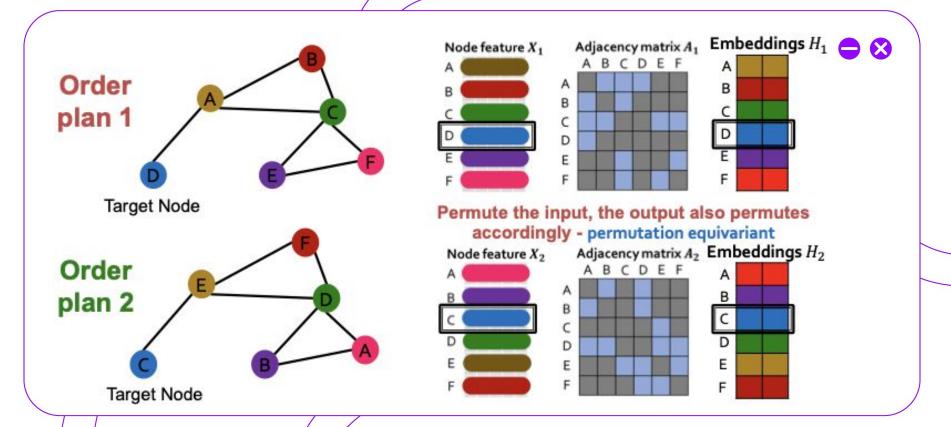




Average of neighbor's previous layer embeddings - Permutation invariant

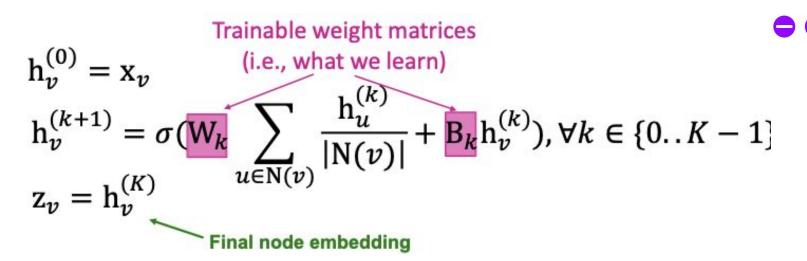
Эквивариантность GCN





Обучение



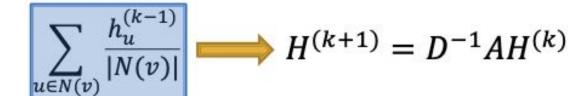


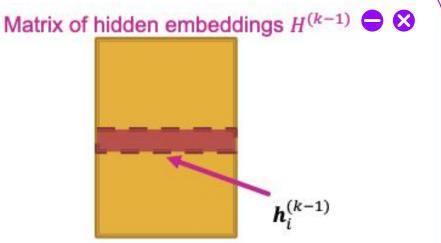
 Оптимизируем любую loss-функцию от полученных эмбеддингов, например с помощью SGD

Агрегация в матричной форме



- Let $H^{(k)} = [h_{1k}^{(k)} ... h_{|V|}^{(k)}]^T$ Then: $\sum_{u \in N_v} h_u^{(k)} = A_{v,:} H^{(k)}$
- Let D be diagonal matrix where $D_{v,v} = \text{Deg}(v) = |N(v)|$
 - The inverse of $D: D^{-1}$ is also diagonal: $D_{v,v}^{-1} = 1/|N(v)|$
- Therefore,





GCN в матричной форме



Re-writing update function in matrix form:



$$H^{(k+1)} = \sigma(\tilde{A}H^{(k)}W_k^{\mathrm{T}} + H^{(k)}B_k^{\mathrm{T}})$$

where $\tilde{A} = D^{-1}A$

$$H^{(k)} = [h_1^{(k)} \dots h_{|V|}^{(k)}]^T$$

- Red: neighborhood aggregation
- Blue: self transformation
- Можно использовать реализации умножения разреженных матриц ($ilde{A}$ разреженная матрица)
- Другие GNN с более сложными агрегациями не всегда могут быть записаны в такой простой матричной форме

Процесс обучения



• Полученные эмбеддинги используются для предсказания



• Обучение с учителем: у – метки в тренировочном наборе

$$\min_{\Theta} \mathcal{L}(\mathbf{y}, f_{\Theta}(\mathbf{z}_v))$$

• Обучение без учителя: нету меток, используем структуру графа

Пример обучения без учителя



• Схожие вершины должны иметь близкие эмбеддинги



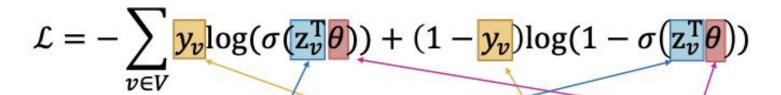
$$\min_{\Theta} \mathcal{L} = \sum_{z_u, z_v} CE(y_{u,v}, DEC(z_u, z_v))$$

- where $y_{u,v} = 1$ when node u and v are similar
- $z_u = f_{\Theta}(u)$ and DEC (\cdot, \cdot) is the dot product
- Схожесть вершин определяем самостоятельно
 - Случайные блуждания (DeepWalk, node2vec, ...)

Пример обучения с учителем



• Классификация вершин



Encoder output:

node embedding

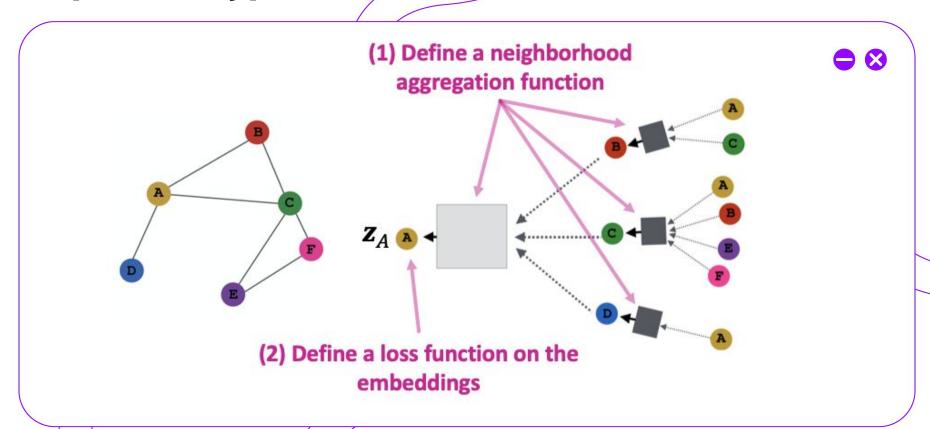
Classification weights

Node class label

Safe or toxic drug?

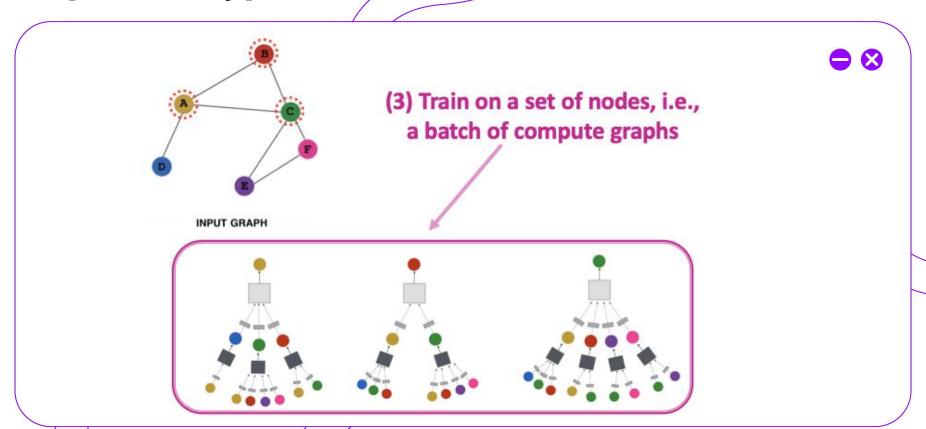
Архитектура





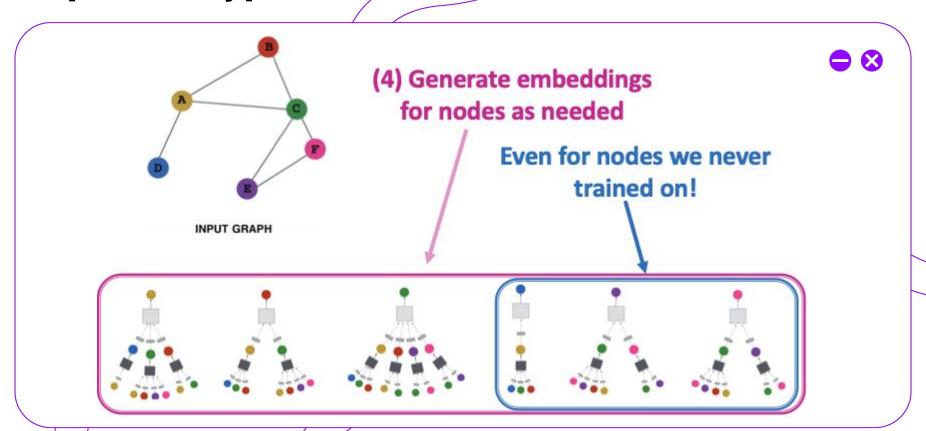
Архитектура

VITMO



Архитектура

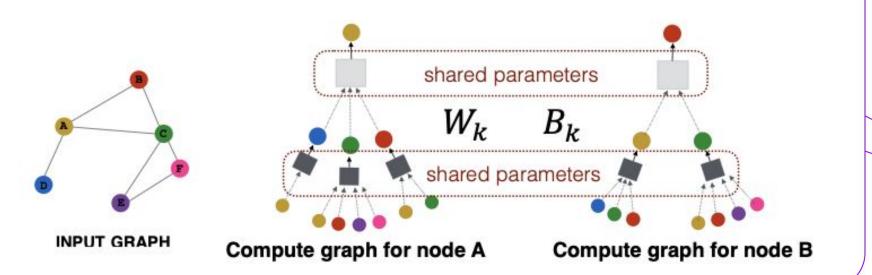




Общие параметры

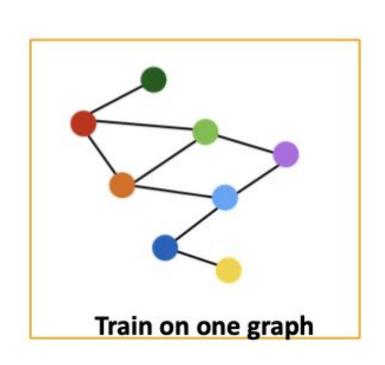


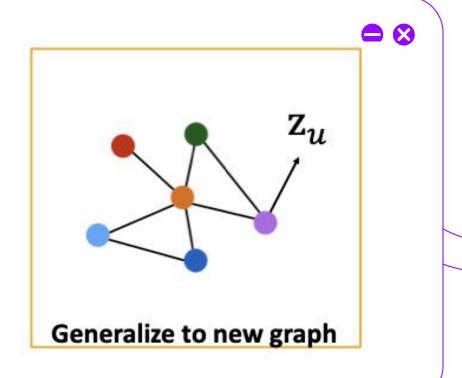
- Количество параметров сублинейно от |V|
- Модель может получать эмбеддинги и для новых вершин



Использование на новых графах

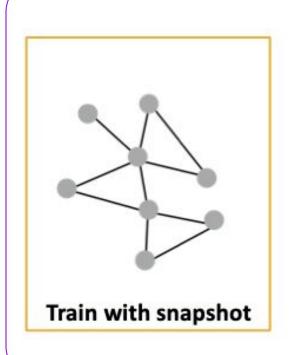


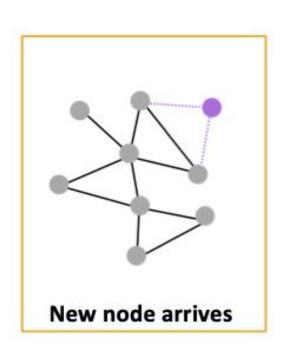


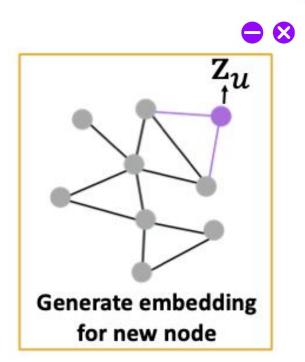


Добавление новых вершин

VITMO





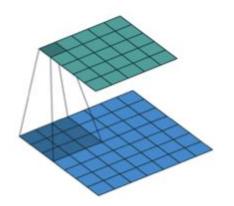


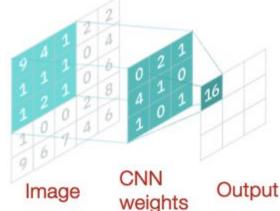
CNN



Convolutional neural network (CNN) layer with 3x3 filter:







CNN formulation:
$$\mathbf{h}_v^{(l+1)} = \sigma(\sum_{u \in \mathbf{N}(v) \cup \{v\}} \mathbf{W}_l^u \mathbf{h}_u^{(l)}), \ \ \forall l \in \{0, \dots, L-1\}$$

N(v) represents the 8 neighbor pixels of v.

GNN vs. CNN

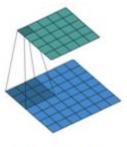


Convolutional neural network (CNN) layer with

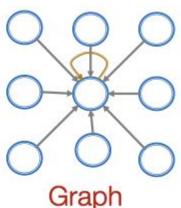












- GNN formulation: $h_v^{(l+1)} = \sigma(\mathbf{W}_l \sum_{u \in N(v)} \frac{h_u^{(l)}}{|N(v)|} + B_l h_v^{(l)}), \forall l \in \{0, ..., L-1\}$
- CNN formulation: (previous slide) $\mathbf{h}_v^{(l+1)} = \sigma(\sum_{u \in \mathbf{N}(v) \cup \{v\}} \mathbf{W}_l^u \mathbf{h}_u^{(l)}), \forall l \in \{0, \dots, L-1\}$ $\mathbf{h}_{v}^{(l+1)} = \sigma(\sum_{u \in \mathbf{N}(v)} \mathbf{W}_{l}^{u} \mathbf{h}_{u}^{(l)} + \mathbf{B}_{l} \mathbf{h}_{v}^{(l)}), \forall l \in \{0, ..., L-1\}$ if we rewrite:

Процесс обучения







- CNN не инвариантна/эквивариантна относительно перестановок вершин
- GNN можно рассмотреть как обобщение CNN от изображений до любых графов

Заключение



