

- [Обзор задач](#)
- [Введение](#)
- [Базовые идеи графовых нейронных сетей](#)
- [Задачи, решаемые с помощью графовых нейронных сетей](#)
  - [Классификация вершин графа](#)
  - [Классификация непосредственно графа](#)
- [Итог обзора архитектуры графовых нейронных сетей](#)
- [Задача предсказания трендов отдельных акций, деривативов](#)
- [Определения](#)
  - [Концепции акций](#)
  - [Ценовой тренд акций](#)
  - [Непосредственно задача](#)
- [Архитектура модели](#)
- [Целевая функция](#)
- [Обзор результатов стратегии](#)
  - [Описание эксперимента](#)
  - [Метрика](#)
    - [Information Coefficient \(IC\)](#)
    - [Rank IC](#)
  - [Подходы для сравнения:](#)
  - [Результаты](#)
    - [Метрики](#)
    - [Кол-во параметров модели](#)
- [Выводы](#)
- [Код](#)
- [Источники](#)

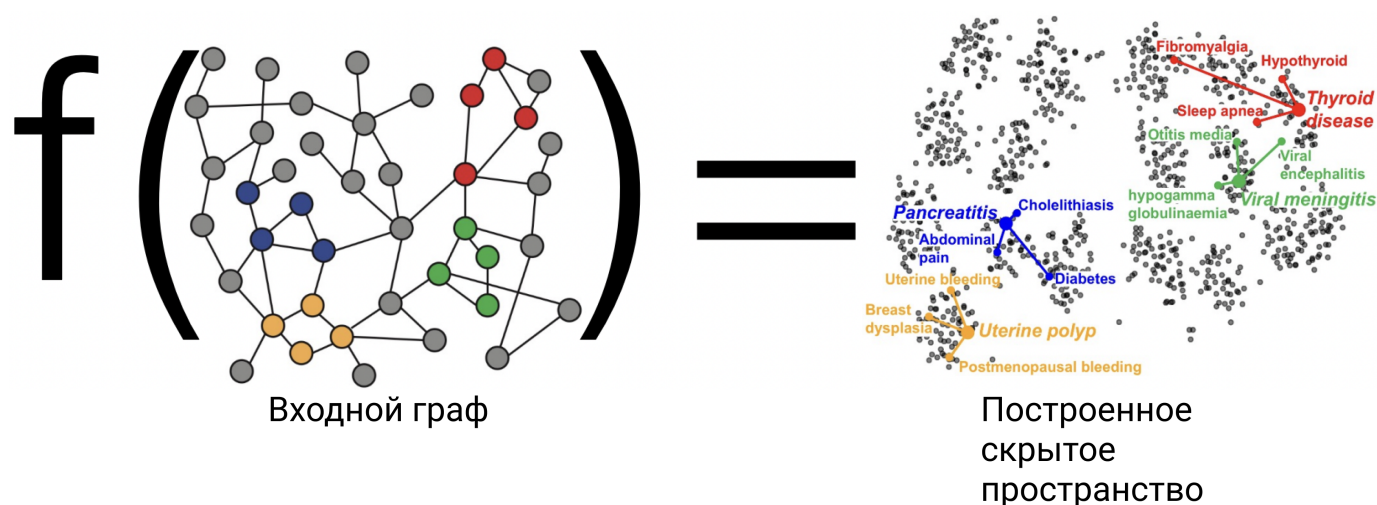
## Обзор задач

В современном мире исследователи всё чаще сталкиваются с данными, по структуре представляющими собой некоторый граф, это могут быть например данные о пользователях в социальных сетях, где рёбрами будут связи между пользователями, а вершинами будут пользователи или же информация об устройстве молекул, тогда в вершинах будут некоторые структурные элементы

молекул (атомы, ионы и так далее).

## Введение

При этом классические методы машинного обучения хоть и могут решать задачи с подобными данными, скрытое их представление в результате работы классических алгоритмов кодирования с помощью нейронных сетей не гарантирует сохранение взаимных отношений скрытых представлений соответствующих вершин графа. Для лучшего решения задач с данными в виде графов предлагается использовать так называемые графовые нейронные сети, основная задача которых – правильно построение скрытого пространства.



## Базовые идеи графовых нейронных сетей

В рассмотрении задачи обучения графовой нейронной сети рассматривают следующие параметры:

- $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  непосредственно входной граф
- $\mathcal{X} \in \mathbb{R}^{d \times |\mathcal{V}|}$  – множество признаков
- $\mathbf{z}_u \forall u \in \mathcal{V}$  – скрытое представление исходных вершин графа

Обновление информации о скрытом представлении вершин графа можно предствавить следующим образом:

$$\mathbf{h}_u^{(k+1)} = \text{UPDATE}^{(k)}(\mathbf{h}_u^{(k)}, \text{AGGREGATE}^{(k)}(\{\mathbf{h}_v^{(k)} \mid \forall v \in \mathcal{N}(u)\})) = \text{UPDATE}^{(k)}(\mathbf{h}_u^{(k)}, m^{(k)}(\mathcal{N}(u)))$$

Таким образом для операции обновления для  $\mathbf{h}_u^{(k)}$  происходят

исходя из агрегированной информации о соседях  $\mathbin{u}$  в графе, обозначим множество соседей  $\mathbin{u}$  через  $\mathcal{N}(u)$

На каждом шаге  $\text{AGGREGATE}$  функция принимает в качестве входа  $\mathcal{H}(t) = \{\mathbin{h}_u, \forall u \in \mathcal{N}(t)\}$  и генерирует информационное "сообщение"  $\mathbin{m}_{\mathcal{N}(t)^{(k)}}$ , далее функция  $\text{UPDATE}$  комбинирует  $\mathbin{m}_{\mathcal{N}(t)^{(k)}}$  и  $\mathbin{h}_{t^{(k-1)}}$  для создания  $\mathbin{h}_{t^{(k)}}$ . На нулевом шаге  $\mathbin{h}_{t^{(0)}} = \mathbin{x}$ . В результате процесса обработки графа имеем множество скрытых представлений

$$\mathbin{z}_u = \mathbin{h}_{u^{(K)}}, u \in \mathcal{V}$$

Итоговый проход графовой нейронной сети можно описать следующим образом

$$h_u^{(k)} = \sigma(W_{self}^{(k)} h_u^{(k-1)} + W_{neigh}^{(k)} \sum_{v \in \mathcal{N}(u)} h_v^{(k-1)} + b^{(k)})$$

Где  $\mathbin{W}_{self}^{(k)}, \mathbin{W}_{neigh}^{(k)} \in \mathbb{R}^{d^{(k)} \times d^{(k-1)}}$  – обучаемые параметры таким образом операцию обновления можно описать как  $\text{UPDATE}(\mathbin{h}_u, \mathbin{m}_{\mathcal{N}(u)}) = \sigma(\mathbin{W}_{self} \mathbin{h}_u + \mathbin{W}_{neigh} \mathbin{m}_{\mathcal{N}(u)})$ , где, напомним,  $\mathbin{m}_{\mathcal{N}(u)} = \text{AGGREGATE}^{(k)}(\{\mathbin{h}_v^{(k)}, \forall v \in \mathcal{N}(u)\})$ .

## Задачи, решаемые с помощью графовых нейронных сетей

### Классификация вершин графа

Часто в различных задачах построения, например, рекомендательных систем (рекомендация друзей в социальных сетях) необходимо каким либо образом классифицировать вершины графа, относя их к тому или иному классу. Данную задачу можно неплохо решать с помощью классических подходов задач классификации, но они никак не будут учитывать взаимные отношения между участниками графа. Данная проблема удачно решается с помощью применения графовых нейронных сетей.

Тогда функционал ошибки можно описать следующим образом:

$$\mathcal{L} = \sum_{u \in \mathcal{V}} \{-\log \text{softmax}(\mathbf{z}_u, \mathbf{y}_u)\}$$

## Классификация непосредственно графа

Также порой необходимо классифицировать не отдельные вершины сети, а всю сеть целиком. Например, это может применяться в задачи классификации молекул, или в задаче классификации отдельных сетей пользователей.

Тогда функционал ошибки описывается следующим образом:

$$\mathcal{L} = \sum_{\mathcal{G}_i \in \mathcal{T}} \|\text{MLP}(\mathbf{z}_{\mathcal{G}_i}) - y_{\mathcal{G}_i}\|_2^2$$

Где  $\text{MLP}$  – полносвязная нейросеть, построенная над скрытым представление вершин графа  $\mathbf{z}_{\mathcal{G}}$

## Итог обзора архитектуры графовых нейронных сетей

Особенности и отличия Графовых Нейронных Сетей от классических методов глубокого обучения:

- Графовые Нейронные сети строятся как система распространения информации по графу; число слоев = глубина обхода
- Графовые Нейронные сети это совместное обучение структуры с признаками вершин
- Графовые Нейронные сети это обучение с учителем, под конкретную задачу
- Размер модели Графовой Нейронной сети почти не зависит от размера графа

---

## Задача предсказания трендов отдельных акций, деривативов

Фондовый рынок – один из самых прибыльных инвестиционных каналов в реальном мире. Чтобы добиться высокой доходности в последние годы привлекает все большее внимание как фундаментальный компонент многих сложных инвестиционных стратегий. Многие из существующих подходов предполагают, что цены на различные акции независимы друг от друга, и модель прогнозирования строятся просто на основе информации, относящейся к каждой акции по отдельности, такой как временные ряды исторических цен и объема акций.

Однако на практике ценовые тренды различных акций, как правило, сильно коррелируют друг с другом, особенно акции из одного сектора или разделяющие общие высокоровневые концепции.

Многие недавние исследования обращают особое внимание на использование такой информации в прогнозировании фондовых тенденций. Например, некоторые простые методы напрямую использовали predetermined концепции акций в качестве входных параметров модели линейного прогнозирования.

## Определения

### Концепции акций

Предопределенные концепции акций – это некоторые определяемые человеком концепции акций, такие как сектор акций, отрасль и основные предприятия. Скрытые концепции акций – это некоторые скрытые концепции, которые эксперты-люди не определяют заранее и отражают некоторую аналогичную тенденцию цен на акции среди акций в рамках одной и той же скрытой концепции.

Например:

Возьмём 4 акции: Apple, Microsoft, Amazon, eBay. В итоге имеем 4 явных концепта: Технологический сектор, Электронная коммерция, Облачные вычисления, Интернет ретейлеры. В итоге для Microsoft имеем имеет два явных концепта Технологический сектор и Облачные вычисления.

### Ценовой тренд акций

$$d_i^t = \frac{\text{Price}_i^{t+1} - \text{Price}_i^t}{\text{Price}_i^t}$$

### Непосредственно задача

Имея некоторую информацию об акции (историческая стоимость, объём, текстовая информация из новостей и соцсетей) для акции  $A_i$  на момент времени  $t$  уметь предсказывать ценовой тренд акции  $d_i^t$ .

## Архитектура модели

Модель состоит из 3 принципиальных блоков.

1. Блок для извлечения сырой информации, это различные временные признаки, обработанные признаки из открытых источников

2. Блок для извлечения скрытых концептов, графовая нейросеть, которая извлекает некоторые скрытые концепты, взаимосвязи между акциями.
3. Регрессионный модуль для предсказания тренда

## Целевая функция

$$\mathcal{L} = \sum_{t \in \mathcal{T}} \text{MSE} \left( p^t, d^t \right) = \sum_{t \in \mathcal{T}} \sum_{i \in \mathcal{S}^t} \frac{\left( p_i^t - d_i^t \right)^2}{\left| \mathcal{S}^t \right|}$$

- $\mathcal{T}$  – множество рассматриваемых дат
- $\mathcal{S}^t$  – множество акций на день  $t$
- $p_i^t, d_i^t$  – предсказанные и целевые тренды акции  $i$  в день  $t$

## Обзор результатов стратегии

### Описание эксперимента

Предлагается сравнить различные подходы для предсказания тренда акций на примере индекса CSI 100. Данный индекс включает в себя 100 китайских компаний с наибольшей рыночной капитализацией, торгующихся на Китайских (Шанхайская и Шеньженьская) биржах с 01/01/2017 по 12/31/2020.

### Метрика

Для оценки качества стратегии используется несколько различных метрик, как статистических, так эмпирических

1. Information Coefficient (IC)
2. Rank IC
3. Precision@N  $N \in \{3, 5, 10, 30\}$

### Information Coefficient (IC)

Данная метрика выражается как мера корреляции Пирсона между целевой переменной и предсказанием

$$r_{xy} = \frac{\sum_{i=1}^m \left( x_i - \bar{x} \right) \left( y_i - \bar{y} \right)}{\sqrt{\sum_{i=1}^m \left( x_i - \bar{x} \right)^2} \sqrt{\sum_{i=1}^m \left( y_i - \bar{y} \right)^2}} = \frac{\text{cov}(x, y)}{\sqrt{s_x^2} \sqrt{s_y^2}}$$

## Rank IC

Вычисляется по формуле  $\text{Rank IC}(y^t, \hat{y}^t) = \text{corr}(\text{rank}\{y^t\}, \text{rank}\{\hat{y}^t\})$ , где  $\text{corr}(\cdot)$  – коэффициент корреляции пирсона. Далее в каждый день  $t$  предсказания и целевая переменная сортируются по возрастанию тренда акции.

## Подходы для сравнения:

Для сравнения качества графовой нейронной сети применяются два других распространённых подхода.

- 1. MLP модель
- 2. LSTM модель

## Результаты

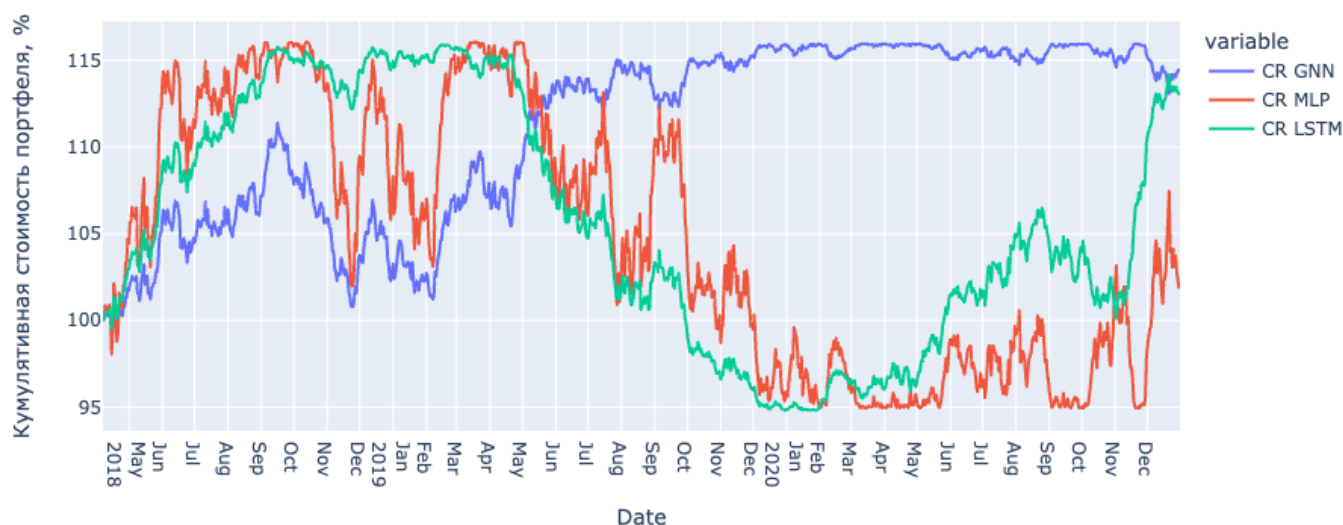
### Метрики

	GNN	MLP	LSTM
IC	0.120	0.0071	0.0097
Rank IC	0.115	0.0061	0.0091
Precision@10	0.6107	0.5549	0.5904

### Кол-во парамтеров модели

- GNN – 1 000 000
- MLP – 300 000 000
- LSTM – 50 000 000

## CR стратегии



## Выводы

Графовые сети благодаря особому подходу к структурированным данным могут быть использованы во множестве областей, начиная с классификации молекул и заканчивая построением алгоритмической стратегии. Особенное прикладное пространство позволяет работать не с отдельными элементами выборки, а с выборкой в целом, учитывая взаимные отношения отдельных элементов выборки. Особенно хорошо себя показывают графовые нейронные сети в задачах среднесрочной алгоритмической торговли, за счёт того, что позволяют агрегировать разрозненную информацию об отдельных акциях, выстраивая некоторые скрытые признаки, объединяющие отдельные деривативы.

## Код

(GitHub)[[link](#)]

## Источники

1. Graph Representation Learning
2. Stanford, cs224w