Graph WaveNet для глубокого пространственно-временного моделирования графов

Цзунхань Ву1 , Шируй Пан2∗ , Годун Лун1 , Цзин Цзян1 , Чэнци Чжан1 1 Центр искусственного интеллекта, FEIT, Технологический университет Сиднея, Австралия 2 Факультет информационных технологий, Университет Монаша, Австралия zonghan.wu-3@student.uts.edu.au, shirui.pan@monash.edu,

{guodong.long, jing.jiang, chengqi.zhang}@uts.edu.au

Абстрактный

Пространственно-временное моделирование графов является важной задачей для анализа пространственных отношений и временных трендов компонентов в системе. Существующие подходы в основном учитывают пространственную зависимость от фиксированной структуры графа, предполагая, что базовая связь между объектами предопределена. Однако явная структура графа (отношение) не обязательно отражает истинную зависимость, и подлинное отношение может отсутствовать из-за неполных связей в данных. Кроме того, существующие методы неэффективны для фиксации временных трендов, поскольку РНС или СНС, используемые в этих методах, не могут захватывать долгосрочные временные последовательности. Чтобы преодолеть эти ограничения, в данной статье мы предлагаем новую архитектуру графовой нейронной сети, Graph WaveNet, для пространственно-временного моделирования графов. Разработав новую адаптивную матрицу зависимостей и изучив ее с помощью встраивания узлов, наша модель может точно уловить скрытую пространственную зависимость в данных. Благодаря многоуровневой расширенной 1D сверточной составляющей, рецептивное поле которой растет экспоненциально по мере увеличения количества слоев, Graph WaveNet способен обрабатывать очень длинные последовательности. Эти два компонента органично интегрированы в единую инфраструктуру, и вся среда изучается сквозным образом. Экспериментальные результаты на двух наборах данных сети общественного трафика, METR-LA и PEMS-BAY, демонстрируют превосходную производительность нашего алгоритма.

# Знакомство

arXiv:1906.00121v1 [cs. LG] 31 мая 2019

Пространственно-временное моделирование графов привлекает все большее внимание с развитием графовых нейронных сетей. Он направлен на моделирование динамических входных данных на уровне узлов, предполагая взаимозависимость между подключенными узлами, как показано на рисунке 1. Пространственно-временное графовое моделирование имеет широкое применение при решении сложных системных задач, таких как прогнозирование скорости движения [Li *et al.*, 2018b], прогнозирование спроса на такси [Yao *et al.*, 2018], распознавание действий человека [Ян

|  |
| --- |
| ∗  Автор, отвечающий за переписку. |



Рисунок 1: Пространственно-временное моделирование графов. В пространственно-временном графе каждый узел имеет динамические входные функции. Цель состоит в том, чтобы смоделировать динамические особенности каждого узла с учетом структуры графа.

*et al.*, 2018], и предвидение маневра водителя [Jain *et al.*, 2016]. Например, при прогнозировании скорости движения датчики скорости на дорогах города формируют график, где веса ребер оцениваются по евклидову расстоянию двух узлов. Поскольку заторы на одной дороге могут привести к снижению скорости движения на въездных дорогах, естественно рассматривать базовую графовую структуру транспортной системы как априорные знания о взаимозависимостях между узлами при моделировании данных временных рядов скорости движения на каждой дороге.

Основное предположение, лежащее в основе пространственно-временного моделирования графов, заключается в том, что будущая информация узла обусловлена его исторической информацией, а также исторической информацией его соседей. Таким образом, вопрос о том, как одновременно охватить пространственные и временные зависимости, становится основной проблемой. Последние исследования по моделированию пространственно-временных графов в основном следуют двум направлениям. Они либо интегрируют сети свертки графов (GCN) в рекуррентные нейронные сети (RNN) [Seo *et al.*, 2018; Li *et al.*, 2018b] или в сверточные нейронные сети (CNN) [Yu *et al.*, 2018; Yan *et al.*, 2018]. Несмотря на то, что эти подходы показали эффективность внедрения графовой структуры данных в модель, они сталкиваются с двумя основными недостатками.

Во-первых, эти исследования предполагают, что графическая структура данных отражает подлинные отношения зависимости между узлами. Однако бывают обстоятельства, когда связь не влечет за собой отношения взаимозависимости между двумя узлами и когда отношения взаимозависимости между двумя узлами существуют, но связь отсутствует. Чтобы привести каждое обстоятельство в качестве примера, рассмотрим рекомендательную систему. В первом случае два пользователя связаны, но у них могут быть разные предпочтения в отношении товаров. Во втором случае два пользователя могут иметь схожие предпочтения, но они не связаны друг с другом. Zhang *et al.* [2018] использовали механизмы внимания для устранения первого обстоятельства путем корректировки веса зависимости между двумя связанными узлами, но не учли второе обстоятельство.

Во-вторых, современные исследования по моделированию пространственно-временных графов неэффективны для изучения временных зависимостей. Подходы, основанные на RNN, страдают от трудоемкого итеративного распространения и градиентного взрыва/исчезновения для захвата дальних последовательностей [Seo *et al.*, 2018; Li *et al.*, 2018b; Zhang *et al.*, 2018]. Напротив, подходы, основанные на СНС, пользуются преимуществами параллельных вычислений, стабильных градиентов и низких требований к памяти [Yu *et al.*, 2018; Yan *et al.*, 2018]. Тем не менее, в этих работах необходимо использовать много слоев, чтобы захватить очень длинные последовательности, потому что они используют стандартную 1D-свертку, размер рецептивного поля которой линейно увеличивается с увеличением количества скрытых слоев.

В этой работе мы представляем основанный на CNN метод под названием Graph WaveNet, который устраняет два вышеупомянутых недостатка. Мы предлагаем слой свертки графа, в котором самоадаптивная матрица смежности может быть изучена на основе данных с помощью сквозного контролируемого обучения. Таким образом, самоадаптивная матрица смежности сохраняет скрытые пространственные зависимости. Мотивировано WaveNet [Oord *et al.*, 2016], мы применяем многоуровневые расширенные случайные свертки для захвата временных зависимостей. Размер рецептивного поля сложенных расширенных сетей случайной свертки растет экспоненциально с увеличением числа скрытых слоев. Благодаря поддержке сложенных расширенных случайных свертков, Graph WaveNet может эффективно обрабатывать пространственно-временные данные графа с долгосрочными временными последовательностями. Основными результатами этой работы являются следующие:

* Мы строим самоадаптивную матрицу смежности, которая сохраняет скрытые пространственные зависимости. Предложенная нами самоадаптирующаяся матрица смежности способна автоматически выявлять невидимые структуры графа на основе данных без какого-либо предварительного руководства. Эксперименты подтверждают, что наш метод улучшает результаты, когда пространственные зависимости известны, но не предоставляются.
* Мы представляем эффективную и действенную структуру для одновременного захвата пространственно-временных зависимостей. Основная идея состоит в том, чтобы собрать предложенную нами свертку графа с расширенной случайной сверткой таким образом, чтобы каждый слой свертки графа решал пространственные зависимости информации об узлах, извлеченной слоями расширенной случайной свертки на разных уровнях детализации.
* Мы оцениваем предложенную модель на наборах данных о дорожном движении и получаем самые современные результаты с низкими затратами на вычисления. Исходные коды Graph WaveNet находятся в открытом доступе по адресу [https://github.com/ nnzhan/Graph-WaveNet.](https://github.com/nnzhan/Graph-WaveNet)

# Связанные работы

## Сети свертки графов

Сети свертки графов являются строительными блоками для обучения графово-структурированных данных [Wu *et al.*, 2019]. Они широко применяются в таких областях, как встраивание узлов [Pan *et al.*, 2018], классификация узлов [Kipf and Welling, 2017], классификация графов [Ying *et al.*, 2018], прогнозирование связей [Zhang and Chen, 2018] и кластеризация узлов [Wang *et al.*, 2017]. Существует два основных направления сетей свертки графов: спектральные подходы и пространственные подходы. Спектральные подходы сглаживают входные сигналы узла с помощью графовых спектральных фильтров [Bruna *et al.*, 2014; Defferrard *et al.*, 2016; Kipf and Welling, 2017]. Пространственные подходы извлекают высокоуровневое представление узла путем агрегирования информации об объектах из окрестностей [Atwood and Towsley, 2016; Gilmer *et al.*, 2017; Hamilton *et al.*, 2017]. В этих подходах матрица смежности рассматривается как априорное знание и фиксируется на протяжении всего обучения. Monti *et al.* [2017] узнал вес соседа узла с помощью гауссовых ядер. Velickovic *et al.* [2017] обновлен вес соседа узла с помощью механизмов внимания. Liu *et al.* [2019] предложил адаптивный слой пути для исследования ширины и глубины окрестности узла. Хотя эти методы предполагают, что вклад каждого соседа в центральный узел различен и должен быть изучен, они все же полагаются на предопределенную структуру графа. Li *et al.* [2018a] внедрил метрики расстояния для адаптивного изучения матрицы смежности графа для задач классификации графов. Эта сгенерированная матрица смежности обусловлена входными данными узлов. Поскольку входные данные пространственно-временного графа являются динамическими, их метод нестабилен для пространственно-временного моделирования графов.

## Пространственно-временные графовые сети

Большинство пространственно-временных графовых сетей следуют двум направлениям, а именно: на основе RNN и на основе CNN. Один из ранних методов, основанных на RNN, захватывал пространственно-временные зависимости путем фильтрации входных данных и скрытых состояний, передаваемых в рекуррентную единицу с использованием графовой свертки [Seo *et al.*, 2018]. В более поздних работах использовались различные стратегии, такие как диффузионная свертка [Li *et al.*, 2018b] и механизмы внимания [Zhang *et al.*, 2018] для повышения производительности модели. В другой параллельной работе использовались РНС узлового уровня и РНС-уровня ребра для обработки различных аспектов временной информации [Jain *et al.*, 2016]. Основные недостатки подходов, основанных на RNN, заключаются в том, что они становятся неэффективными для длинных последовательностей, а их градиенты с большей вероятностью взрываются при сочетании с сетями свертки графов. Подходы, основанные на СНС, сочетают свертку графа со стандартной одномерной сверткой [Yu *et al.*, 2018; Yan *et al.*, 2018]. Будучи эффективными с вычислительной точки зрения, эти два подхода должны объединять множество слоев или использовать глобальный пул для расширения рецептивного поля модели нейронной сети.

# Методология

В этом разделе мы сначала дадим математическое определение проблемы, которую мы рассматриваем в этой статье. Далее мы опишем два стандартных блока нашей инфраструктуры: слой свертки графа (GCN) и слой темпоральной свертки (TCN). Они работают вместе, чтобы зафиксировать пространственно-временные зависимости. Наконец, мы обрисовываем архитектуру нашего фреймворка.

## Постановка задачи

Граф представлен как *G* = (*V,E*), где *V* — множество узлов, а *E* — множество ребер. Матрица смежности, полученная из графа, обозначается **как A** ∈ **RN×N**. Если *vi,vj* ∈ *V* и (*vi,vj*) ∈ *E*, то **Aij** равно единице, в противном случае оно равно нулю. На каждом временном шаге *t* граф *G* имеет динамическую матрицу признаков **X**(*t*) ∈ **RN×D**. В данной работе матрица признаков используется как взаимозаменяемость с графическими сигналами. Имея график *G* и его исторические *сигналы на графике S-шага, наша задача состоит в том, чтобы изучить функцию* f*, которая способна предсказывать свои следующие*  сигналы на графике T шага. Отношение отображения представлено следующим образом

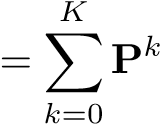
[**X**(*т−С*):*t,Г*] −→f **X**(*t+1):(t+T*)*,* (1) где **X**(*т−С*):*t* ∈ **РН×Д×С** и **X**(*t+1):(t+T*) ∈ **РН×Д×Т**.

## Слой свертки графа

Свертка графа — это важная операция для извлечения особенностей узла с учетом его структурной информации. Kipf *et al.* [2017] предложил первое приближение спектрального фильтра Чебышева [Defferrard *et al.*, 2016]. С пространственной точки зрения, он сглаживает сигнал узла путем агрегирования и преобразования информации о его окрестности. Преимущества их метода в том, что это композиционный слой, его фильтр локализован в пространстве, он поддерживает многомерные входы. **Пусть A ̃** ∈ **RN×N** обозначают нормализованной матрицу смежности с автопетлями, **X** ∈ **RN×D** обозначают входные сигналы, **Z** ∈ **RN×M** обозначают выход, а **W** ∈ **RD×M** обозначают матрицу параметров модели, в [Kipf and Welling, 2017] слой свертки графа определяется как

**Z** = **АХВ ̃***.* (2)

Li *et al.* [2018b] предложил диффузионный слой свертки, который оказался эффективным в пространственно-временном моделировании. Они смоделировали процесс диффузии сигналов графа с *K* конечными шагами. Мы обобщаем его слой диффузионной свертки в виде уравнения 2, что приводит к

**Z** **XWk***,* (3)

где **Pk** представляет степенной ряд матрицы переходов. В случае неориентированного графа **P** = **A***/rowsum*(**A**). В случае ориентированного графа процесс диффузии имеет два направления, прямое и обратное, где матрица прямого перехода **Pf** = **A***/rowsum*(**A**) и матрица обратного перехода **Pb** = **AT***/rowsum(***AT)**. С помощью матрицы прямого и обратного переходов слой свертки диффузионного графа записывается как

*K*

**Z** = XPkfXWk1 + **ПКБХВк2***.* (4)

*к=0*

Самоадаптивная матрица смежности: В нашей работе мы предлагаем самоадаптивную матрицу смежности **A ̃adp**. Эта самоадаптирующаяся матрица смежности не требует каких-либо предварительных знаний и изучается сквозным путем стохастического градиентного спуска. При этом мы позволяем модели самостоятельно обнаруживать скрытые пространственные зависимости. Мы достигаем этого путем случайной инициализации двух словарей встраивания узлов с изучаемыми параметрами **E**1*,* **E**2 ∈ **RN×c**. Предложена самоадаптивная матрица смежности в виде

**А ̃***.* (5)

Мы называем **E1** в качестве внедрения исходного узла и **E2** в качестве целевого узла. Умножая **E1** и **E2**, мы получаем веса пространственной зависимости между исходными и целевыми узлами. Мы используем функцию активации ReLU для устранения слабых соединений. Функция SoftMax применяется для нормализации самоадаптирующейся матрицы смежности. Таким образом, нормализованную самоадаптивную матрицу смежности можно рассматривать как матрицу перехода скрытого диффузионного процесса. Комбинируя предопределенные пространственные зависимости и самообучающиеся зависимости скрытых графов, мы предлагаем следующий слой свертки графа

*K*

**Z** = XPkfXW**XW****XWk3***.* (6)

*к=0*

Когда структура графа недоступна, мы предлагаем использовать только самоадаптивную матрицу смежности для захвата скрытых пространственных зависимостей, т.е.

*K*

**Z** **XWk***.* (7)

*к=0*

Стоит отметить, что наша свертка графов относится к пространственным подходам. Несмотря на то, что мы используем сигналы графа взаимозаменяемо с матрицей признаков узлов для обеспечения согласованности, свертка графа в уравнении 7 действительно интерпретируется как агрегирование преобразованной информации о признаках из окрестностей разного порядка.

## Слой темпоральной свертки

Мы используем расширенную причинную свертку [Yu and Koltun, 2016] в качестве слоя временной свертки (TCN) для фиксации временных трендов узла. Расширенные сети причинной свертки позволяют создавать экспоненциально большое рецептивное поле за счет увеличения глубины слоя. В отличие от подходов, основанных на RNN, расширенные сети случайной свертки способны правильно обрабатывать дальнодействующие последовательности нерекурсивным образом, что облегчает параллельные вычисления и решает проблему градиентного взрыва. Расширенная причинно-следственная свертка сохраняет временной причинно-следственный порядок, добавляя нули к входным данным, так что прогнозы, сделанные на текущем временном шаге, включают только историческую информацию. В качестве частного случая стандартной свертки 1D, операция расширенной причинно-следственной свертки скользит по входам, пропуская значения с определенным шагом, как показано на рисунке 2. Математически, при заданных входе 1D-последовательности **x** ∈ **RT** и фильтре **f** ∈ **RK**, расширенная операция причинной свертки **x** с **f** на шаге *t* представляется как

*К−1*

**x** *?* **f**(*t*) = X **f**(*s*)**x**(*t* − *d* × *s*)*,* (8)

*s=0*

где *d* — коэффициент расширения, который контролирует расстояние пропуска. Путем наложения расширенных слоев причинной свертки с факторами расширения в порядке возрастания, рецептивное поле

**ТСН**

Расширение = 1

**ТСН**

Расширение = 2

**ТСН**

Дилатация = 4

Рисунок 2: Расширенная случайная свертка с размером ядра 2. С помощью

коэффициент расширения *k*, он выбирает входные данные на каждом *k* шаге и применяет стандартную одномерную свертку к выбранным входам.

Модель растет в геометрической прогрессии. Это позволяет расширенным сетям причинно-следственной свертки захватывать более длинные последовательности с меньшим количеством слоев, что экономит вычислительные ресурсы.

Стробируемый TCN: Механизмы стробирования имеют решающее значение в рекуррентных нейронных сетях. Было показано, что они обладают мощными возможностями для управления потоком информации через слои для сетей временной свертки [Dauphin *et al.*, 2017]. Простой стробированный TCN содержит только выходной вентиль. С учетом входных данных

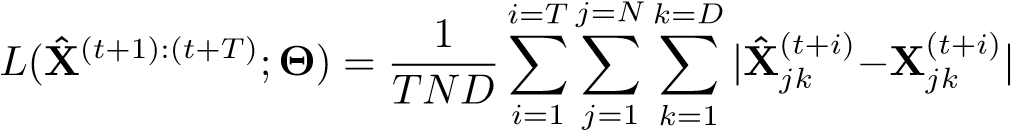
X ∈ *RN×D×S*, он принимает вид

**h** *,* (9), где **Θ1**, **Θ2**, **b** и **c** — параметры модели, — произведение по элементам, *g*(·) является функцией активации выходов, а *σ*(·) — сигмоидальная функция, определяющая соотношение информации, передаваемой на следующий слой. Мы применяем Gated TCN в нашей модели для изучения сложных временных зависимостей. Хотя мы эмпирически устанавливаем касательную гиперболическую функцию как функцию активации *g*(·), другие формы закрытого TCN могут быть легко вписаны в нашу структуру, такие как LSTM-подобный закрытый TCN [Kalchbrenner *et al.*, 2016].

## Фреймворк Graph WaveNet

Мы представляем структуру графа WaveNet на рисунке 3. Он состоит из сложенных пространственно-временных слоев и выходного слоя. Пространственно-временной слой состоит из слоя графовой свертки (GCN) и стробированного слоя временной свертки (Gated TCN), который состоит из двух параллельных временных сверточных слоев (TCN-a и TCN-b). Объединяя несколько пространственно-временных слоев, Graph WaveNet способен обрабатывать пространственные зависимости на разных временных уровнях. Например, на нижнем уровне GCN получает краткосрочную временную информацию, в то время как на верхнем уровне GCN обрабатывает долгосрочную временную информацию. Входными данными **h** для слоя свертки графа на практике являются трехмерные тензоры с размером [N,C,L], где *N* — количество узлов, *C* — скрытое измерение, *L* — длина последовательности. Мы применяем слой свертки графа к каждому из **h**[:*,*:*,i*] ∈ **RN×C**.

В качестве цели обучения Graph WaveNet мы решили использовать среднюю абсолютную ошибку (MAE), которая определяется как



(10) В отличие от предыдущих работ, таких как [Li *et al.*, 2018b; Yu *et al.*, 2018], наш граф WaveNet выводит **Xˆ**(*t+*1):(*t+T*) целиком, а не генерирует **Xˆ**(*t*) рекурсивно через *T* шагов. В нем решается проблема несоответствия между обучением и

ГКН

Тань

s

Остаточный

+

×

Линейный

Ввод

ТСН

-

a

ТСН

-

b

K слоев

Линейный

Линея

r

Выпуск

РеЛУ

+

Скип

блат

РеЛУ

Закрытый TCN

Рисунок 3: Фреймворк Graph WaveNet. Он состоит из *K* пространственно-временных слоев слева и выходного слоя справа. Входные данные сначала преобразуются линейным слоем, а затем передаются в модуль стробированной темпоральной свертки (Gated TCN), за которым следует слой свертки графа (GCN). Каждый пространственно-временной слой имеет остаточные связи и пропускается с выходным слоем.

|  |  |  |  |
| --- | --- | --- | --- |
| Данные | #Nodes | #Edges | #Time Шаги |
| МЕТР-ЛА | 207 | 1515 | 34272 |
| ПЕМС-БЭЙ | 325 | 2369 | 52116 |

Таблица 1: Сводная статистика по METR-LA и PEMS-BAY.

Тестирование связано с тем, что модель учится делать прогнозы для одного шага во время обучения и ожидается, что она будет создавать прогнозы для нескольких шагов во время вывода. Для этого мы искусственно проектируем размер рецептивного поля Graph WaveNet, равный длине последовательности входов таким образом, чтобы в последнем пространственно-временном слое временная размерность выходов была точно равна единице. После этого мы устанавливаем количество выходных каналов последнего слоя в множитель длины шага *T* , чтобы получить желаемый выходной размер.

# Эксперименты

Мы проверили Graph WaveNet на двух наборах данных общедоступной сети трафика, METR-LA и PEMS-BAY, выпущенных Li *et al.* [2018b]. METR-LA записывает статистику за четыре месяца о скорости движения на 207 датчиках на автомагистралях округа Лос-Анджелес. PEMS-BAY содержит информацию о скорости движения за шесть месяцев на 325 датчиках в районе залива. Мы применяем те же процедуры предварительной обработки данных, что и в [Li *et al.*, 2018b]. Показания датчиков агрегируются в 5-минутные окна. Матрица смежности узлов строится по расстоянию от дорожной сети с пороговым гауссовским ядром [Shuman *et al.*, 2012]. К входным данным применяется нормализация Z-баллов. Наборы данных разделены в хронологическом порядке: 70 % для обучения, 10 % для проверки и 20 % для тестирования. Подробная статистика по набору данных представлена в таблице 1.

## Базовых линий

Мы сравним Graph WaveNet со следующими моделями.

* АРИМА. Авторегрессионная модель интегрированного скользящего среднего с фильтром Калмана [Li *et al.*, 2018b].

|  |
| --- |
| 15 мин 30 мин 60 мин  Данные Модели  МАЕ RMSE МАПЕ МАЕ RMSE МАПЕ МАЕ RMSE МАПЕ  ARIMA [Ли *et al.*, 2018b] 3.99 8.21 9.60% 5.15 10.45 12.70% 6.90 13.23 17.40%  FC-LSTM [Li *et al.*, 2018b] 3.44 6.30 9.60% 3.77 7.23 10.90% 4.37 8.69 13.20%  WaveNet [Курорт *et al.*, 2016] 2.99 5.89 8.04% 3.59 7.28 10.25% 4.45 8.93 13.62%  DCRNN [Li *et al.*, 2018b] 2.77 5.38 7.30% 3.15 6.45 8.80% 3.60 7.60 10.50%  GGRU [Чжан *et al.*, 2018] 2.71 5.24 6.99% 3.12 6.36 8.56% 3.64 7.65 10.62%  STGCN [Yu *et al.*, 2018] 2,88 5,74 7,62% 3,47 7,24 9,57% 4,59 9,40 12,70% График WaveNet 2,69 5,15 6,90% 3,07 6,22 8,37% 3,53 7,37 10,01%  МЕТР-ЛА  ПЕМС-БЭЙ  ARIMA [Ли *et al.*, 2018b] 1.62 3.30 3.50% 2.33 4.76 5.40% 3.38 6.50 8.30%  FC-LSTM [Li *et al.*, 2018b] 2.05 4.19 4.80% 2.20 4.55 5.20% 2.37 4.96 5.70%  WaveNet [Курорт *et al.*, 2016] 1.39 3.01 2.91% 1.83 4.21 4.16% 2.35 5.43 5.87%  DCRNN [Li *et al.*, 2018b] 1.38 2.95 2.90% 1.74 3.97 3.90% 2.07 4.74 4.90%  GGRU [Чжан *et al.*, 2018] - - - - - - - - -  STGCN [Yu *et al.*, 2018] 1.36 2.96 2.90% 1.81 4.27 4.17% 2.49 5.69 5.79%  График WaveNet 1.30 2.74 2.73% 1.63 3.70 3.67% 1.95 4.52 4.63%  Таблица 2: Сравнение производительности Graph WaveNet и других базовых моделей. Graph WaveNet достигает наилучших результатов на обоих наборах данных. |

* FC-LSTM Рекуррентная нейронная сеть с полностью связными скрытыми единицами LSTM [Li *et al.*, 2018b].
* WaveNet. Архитектура сверточной сети для данных последовательностей [Oord *et al.*, 2016].
* DCRNN. Рекуррентная нейронная сеть диффузионной свертки [Li *et al.*, 2018b], который сочетает в себе сети свертки графов с рекуррентными нейронными сетями по принципу энкодер-декодер.
* ГГРЮ. Графовая сеть рекуррентных единиц [Zhang *et al.*, 2018]. Рекуррентные подходы. GGRU использует механизмы внимания для свертки графов.
* STGCN. Пространственно-временная сеть свертки графа [Yu *et al.*, 2018], который сочетает в себе свертку графа с 1D-сверткой.

## Экспериментальные установки

Наши эксперименты проводятся в компьютерной среде с одним процессором Intel(R) Core(TM) i9-7900X CPU @ 3.30GHz и одной графической картой NVIDIA Titan Xp. Чтобы покрыть длину входной последовательности, мы используем восемь слоев Graph WaveNet с последовательностью коэффициентов расширения  *1,2,1,2,1,2,1,2*. Мы используем уравнение 4 в качестве слоя свертки графика с шагом диффузии *K* = 2. Мы случайным образом инициализируем вложения узлов равномерным распределением с размером 10. Мы обучаем нашу модель с помощью оптимизатора Адама с начальным коэффициентом обучения 0,001. Дропаут с p=0.3 применяется к выходам слоя свертки графа. Мы выбираем следующие метрики оценки: средняя абсолютная ошибка (MAE), среднеквадратичная ошибка (RMSE) и средняя абсолютная процентная ошибка (MAPE). Отсутствующие значения исключаются как из обучения, так и из тестирования.

## Экспериментальные результаты

В таблице 2 сравнивается производительность Graph WaveNet и базовых моделей для прогнозирования на 15 минут, 30 минут и 60 минут вперед на наборах данных METR-LA и PEMS-BAY. Graph WaveNet получает превосходные результаты на обоих наборах данных.

45

50

55

60

65

70

0

100

200

300

400

Время

Ценность

Волновая сеть

График WaveNet

Реальная стоимость

Рисунок 4: Сравнение кривых прогнозирования между WaveNet и Graph WaveNet на 60 минут вперед на снимке тестовых данных METR-LA.

Он значительно превосходит временные модели, включая ARIMA, FCLSTM и WaveNet. По сравнению с другими пространственно-временными моделями, Graph WaveNet значительно превосходит предыдущий сверточный подход STGCN и в то же время превосходит рекуррентные подходы DCRNN и GGRU. Что касается второй наилучшей модели GGRU, как предложено в таблице 2, Graph WaveNet достигает небольшого улучшения по сравнению с GGRU на 15-минутных горизонтах; Тем не менее, реализует большее улучшение на 60-минутном горизонте. Мы думаем, что это связано с тем, что наша архитектура более способна обнаруживать пространственные зависимости на каждом временном этапе. GGRU использует рекуррентные архитектуры, в которых параметры слоя GCN являются общими для всех рекуррентных юнитов. В отличие от этого, Graph WaveNet использует многоуровневые пространственно-временные слои, которые содержат отдельные слои GCN с различными параметрами. Таким образом, каждый слой GCN в Graph WaveNet может фокусироваться на своем собственном диапазоне временных входных данных.

Мы строим график прогнозируемых значений на 60 минут вперед по сравнению с реальными значениями графика WaveNet и WaveNet на снимке тестовых данных на рисунке 4. Это показывает, что Graph WaveNet генерирует более стабильные прогнозы, чем WaveNet. В частности, есть красный резкий всплеск производства WaveNet, который отклоняется далеко от реальных значений. Напротив, кривая графика WaveNet все время идет в середине реальных значений.



|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Название модели  Конфигурация матрицы смежности  Набор данных  Среднее значение MAE  Среднее значение RMSE  Средний MAPE  МЕТР-  ЛР  Тождество  [  **Я**  ]  3.58  7.18  %  10.21  [  Только пересылка  **P**  ]  %  6.26  8.65  3.13  Только адаптивный  [  ˜  **A**  *АДП*  ]  6.21  8.68  %  3.10  Вперед-назад  [  **P**  *f*  ,  **P**  *b*  ]  6.13  8.25  3.08  %  [  Прямой-обратный-адаптивный  **P**  *f*  ,  **P**  *b*  ,  ˜  **A**  *АДП*  ]  6.09  3.04  8.23  %  ПЭМС-  БУХТА  Тождество  [  **Я**  ]  4.05  4.18  %  1.80  Только пересылка  [  **P**  *f*  ]  3.61  %  1.62  3.72  Только адаптивный  [  ˜  **A**  *АДП*  ]  1.61  %  3.59  3.63  Вперед-назад  [  **P**  *f*  ,  **P**  *b*  ]  3.55  1.59  3.57  %  Прямой-обратный-адаптивный  [  **P**  *f*  ,  **P**  *b*  ,  ˜  **A**  *АДП*  ]  3.55  3.52  1.58  %  Таблица 3: Экспериментальные результаты различных конфигураций матрицы смежности. Модель с прямой и обратной адаптацией достигает наилучших результатов в обоих наборах данных. Адаптивная модель обеспечивает почти такую же производительность, как и модель с прямым управлением.   |  |  | | --- | --- | | Модель | Обучение времени вычислений (с/эпоха) Выводы | | DCRNN | 249.31 18.73 | | СТГКН | 19.10 11.37 | | График WaveNet | 53.68 2.27 | |

для первых 50 узлов. Карты Google.

Рисунок 5: Выученная самоадаптивная матрица смежности.

Эффект самоадаптивной матрицы смежности

Чтобы проверить эффективность предложенной нами адаптивной матрицы смежности, мы провели эксперименты с Graph WaveNet с использованием пяти различных конфигураций матрицы смежности. В таблице 3 показан средний балл MAE, RMSE и MAPE по 12 горизонтам прогнозирования. Мы обнаружили, что адаптивная модель работает даже лучше, чем прямая модель со средним MAE. Когда структура графа недоступна, Graph WaveNet все равно сможет реализовать хорошую производительность. Модель с прямой и обратной адаптацией дает самые низкие баллы по всем трем метрикам оценки. Это указывает на то, что если задана структурная информация графа, добавление самоадаптирующейся матрицы смежности может ввести новую и полезную информацию в модель. На рисунке 5 мы дополнительно исследуем изученную самоадаптивную матрицу смежности в конфигурации модели прямо-обратно-адаптивной модели, обученной на наборе данных METR-LA. Согласно рисунку 5а, некоторые столбцы имеют больше точек с высокой ценностью, чем другие, такие как столбец 9 в левом поле по сравнению со столбцом 47 в правом поле. Это говорит о том, что некоторые узлы оказывают влияние на большинство узлов графа, в то время как другие узлы оказывают более слабое влияние. Рисунок 5b подтверждает наше наблюдение. Видно, что узел 9 расположен рядом с пересечением нескольких основных дорог, в то время как узел 47 лежит на одной дороге.

Время вычислений

Мы сравниваем стоимость вычислений Graph WaveNet с DCRNN и STGCN на наборе данных METR-LA в таблице 4.

Таблица 4: Стоимость вычислений на наборе данных METR-LA.

Graph WaveNet работает в пять раз быстрее, чем DCRNN, но в два раза медленнее, чем STGCN при обучении. Для логического вывода мы измеряем общие временные затраты каждой модели на основе данных валидации. График WaveNet является наиболее эффективным из всех на этапе логического вывода. Это связано с тем, что Graph WaveNet генерирует 12 прогнозов за один прогон, в то время как DCRNN и STGCN должны выдавать результаты, обусловленные предыдущими прогнозами.

# Заключение

В данной статье мы представляем новую модель пространственно-временного моделирования графов. Наша модель эффективно и результативно фиксирует пространственно-временные зависимости, сочетая свертку графа с расширенной случайной сверткой. Мы предлагаем эффективный метод автоматического изучения скрытых пространственных зависимостей на основе данных. Это открывает новое направление в пространственно-временном моделировании графов, где структура зависимостей системы неизвестна, но должна быть обнаружена. На двух наборах данных сети общественного трафика Graph WaveNet достигает самых современных результатов. В будущей работе мы изучим масштабируемые методы применения Graph WaveNet на крупномасштабных наборах данных и исследуем подходы к изучению динамических пространственных зависимостей.

# Подтверждения

Это исследование финансировалось правительством Австралии через Австралийский исследовательский совет (ARC) в рамках грантов: 1) LP160100630 партнерстве с Департаментом здравоохранения правительства Австралии и 2) LP150100671 партнерстве с Австралийским исследовательским альянсом для детей и молодежи (ARACY) и Глобальным бизнес-колледжем Австралии (GBCA).

# Ссылки

[Этвуд и Таусли, 2016] Джеймс Этвуд и Дон Таусли. Диффузионно-сверточные нейронные сети. В *NIPS,* страницы 1993–2001, 2016.

[Bruna *et al.*, 2014] Джоан Бруна, Войцех Заремба, Артур Слам и Ян Лекун. Спектральные сети и локально связанные сети на графах. В *ICLR*, 2014.

[Dauphin *et al.*, 2017] Ян Н. Дофин, Анжела Фан, Майкл Аули и Дэвид Гренжье. Языковое моделирование с помощью шлюзованных сверточных сетей. В *ICML,* стр. 933–941, 2017.

[Defferrard *et al.*, 2016] Михаэль Дефферрар, сын Ксавье Бре и Пьер Вандергейнст. Сверточные нейронные сети на графах с быстрой локализованной спектральной фильтрацией. В *NIPS,* страницы 3844–3852, 2016.

[Gilmer *et al.*, 2017] Джастин Гилмер, Сэмюэл С. Шонхольц, Патрик Ф. Райли, Ориол Виньялс и Джордж Э. Даль. Передача нейронных сообщений для квантовой химии. В *ICML*, страницы 1263–1272, 2017.

[Гамильтон *и др.*, 2017] Уилл Хэмилтон, Житао Ин и Юре Лесковец. Обучение индуктивным представлениям на больших графах. В *NIPS,* страницы 1024–1034, 2017.

[Jain *et al.*, 2016] Ашеш Джайн, Амир Р. Замир, Сильвио Саварезе и Ашутош Саксена. Structural-rnn: Глубокое обучение на пространственно-временных графах. В *CVPR*, страницы 5308–5317, 2016.

[Kalchbrenner *et al.*, 2016] Нал Кальхбреннер, Лассе Эспехольт, Карен Симонян, Аарон ван ден Орд, Алекс Грейвс и Корай Кавукчуоглу. Нейронный машинный перевод в линейном времени. *Препринт arXiv arXiv:1610.10099,* 2016.

[Кипф и Веллинг, 2017] Томас Н. Кипф и Макс Веллинг. Полуконтролируемая классификация с графовыми сверточными сетями. В *ICLR*, 2017.

[Li *et al.*, 2018а] Руоюй Ли, Шэн Ван, Фэйюнь Чжу и Цзюньчжоу Хуан. Сверточные нейронные сети с адаптивным графом. В *AAAI*, страницы 3546–3553, 2018.

[Li *et al.*, 2018b] Ягуан Ли, Роуз Юй, Сайрус Шахаби и Янь Лю. Диффузионная сверточная рекуррентная нейронная сеть: прогнозирование трафика на основе данных. В *ICLR*, 2018.

[Liu *et al.*, 2019] Цзыци Лю, Чаочао Чэнь, Лунфэй Ли, Цзюнь Чжоу, Сяолун Ли, Лэ Сун и Юань Ци. Geniepath: Графовые нейронные сети с адаптивными рецептивными путями. В *AAAI,* 2019.

[Монти *et al.*, 2017] Федерико Монти Давиде Боскаини,

Джонатан Маски, Эмануэле Родола, Ян Свобода и Майкл М. Бронштейн. Геометрическое глубокое обучение на графах и многообразиях с использованием моделей смесей cnn. В *CVPR*, страницы 5115–5124, 2017.

[Oord *et al.*, 2016] Аарон ван ден Орд, Сандер Дилеман, Хейга Зен, Карен Симонян, Ориол Виньялс, Алекс Грейвс, Нал Калчбреннер, Эндрю Сениор и Корай Кавукчуоглу. Wavenet: генеративная модель для необработанного звука. *Препринт arXiv arXiv:1609.03499*, 2016.

[Pan *et al.*, 2018] Шируй Пань, Жуйци Ху, Сай-фу Фун, Годун Лун, Цзин Цзян и Чэнци Чжан. Встраивание графов обучения с помощью состязательных методов обучения. В *IJCAI,* 2018.

[Seo *et al.*, 2018] Ёнджу Сео, Михаэль Дефферрар, Пьер Вандергейнст и Ксавье Брессон. Структурированное моделирование последовательностей с помощью графовых сверточных рекуррентных сетей. В *NIPS,* страницы 362–373, 2018.

[Шуман *и др.*, 2012] Давид I Шуман, Сунил К. Наранг, Паскаль Фроссар, Антонио Ортега и Пьер Вандергейнст. Новая область обработки сигналов на графах: расширение многомерного анализа данных на сети и другие нерегулярные домены. *Препринт arXiv arXiv:1211.0053*, 2012.

[Velickovic *et al.*, 2017] Петар Величкович, Гиллем Кукурулл, Аранча Казанова, Адриана Ромеро, Пьетро Лио и Йошуа Бенджио. Построение графиков сетей внимания. В *ICLR*, 2017.

[Wang *et al.*, 2017] Чунь Ван, Шируй Пань, Годун Лун, Синцюань Чжу и Цзин Цзян. Mgae: Маргинальный автоэнкодер графов для кластеризации графов. В *CIKM,* стр. 889–898. ACM, 2017.

[Wu *et al.*, 2019] Цзунхань Ву, Шируй Пань, Фэнвэнь Чэнь, Годун Лун, Чэнци Чжан и Филип С. Юй. Всесторонний обзор графовых нейронных сетей. *Препринт arXiv arXiv:1901.00596*, 2019.

[Ян *и др.*, 2018] Сицзе Янь, Юаньцзюнь Сюн и Дахуа Линь. Сверточные сети пространственного временного графа для распознавания действий на основе скелета. В *AAAI*, страницы 3482–3489, 2018.

[Яо *и др.*, 2018] Хуасю Яо, Фэй Ву, Цзиньтао Кэ, Сяньфэн

Тан, Итянь Цзя, Сию Лу, Пинхуа Гун, Цзепин Е и Чжэньхуэй Ли. Глубокая многовидовая пространственно-временная сеть для прогнозирования спроса на такси. В *AAAI*, страницы 2588–2595, 2018.

[Ying *et al.*, 2018] Чжитао Ин, Цзясюань Ю, Кристофер Моррис, Сян Жэнь, Уилл Гамильтон и Юре Лесковец. Обучение иерархическому представлению графов с дифференцируемым пулингом. В *NIPS,* страницы 4800–4810, 2018.

[Ю и Колтун, 2016] Фишер Ю и Владлен Колтун. Многомасштабная агрегация контекста с помощью расширенных сверток. В *ICLR*, 2016.

[Yu *et al.*, 2018] Бин Юй, Хаотэн Инь и Чжаньсин Чжу. Сверточные сети пространственно-временного графа: среда глубокого обучения для прогнозирования трафика. В *IJCAI,* страницы 3634–3640, 2018.

[Чжан и Чэнь, 2018] Мухань Чжан и Исинь Чэнь. Предсказание связей на основе графовых нейронных сетей. В *NIPS,* страницы 5165–5175, 2018.

[Zhang *et al.*, 2018] Цзяни Чжан, Синцзянь Ши, Цзюньюань Се, Хао Ма, Ирвин Кинг и Дит-Янь Юнг. Гаан: Закрытые сети внимания для обучения на больших и пространственно-временных графах. *Препринт arXiv arXiv:1803.07294*, 2018.