Повышение эффективности состязательных атак на модель прогнозирования трафика TGC-LSTM

Д.Д. Тарасов, О.Р. Лапонина

Аннотация **– В работе исследуется устойчивость модели прогнозирования городского трафика TGC-LSTM к состязательным атакам уклонения в режиме черного ящика и предлагается ускоренный вариант такой атаки на основе обучаемой сверточной сети. В качестве целевой модели рассматривается архитектура TGC-LSTM. В работе описана реализация одноточечной атаки 1VITA, в которой возмущение ищется без доступа к градиентам целевой модели при помощи дифференциальной эволюции (DE), что соответствует идее разреженной атаки на прогнозирующие модели временных рядов. Далее на основе найденных 1VITA-примеров собирается обучающий набор, и по нему обучается вспомогательная сверточная сеть, предсказывающая карту чувствительности входного окна и карту величин возмущения. В данной работе предложена полностью двумерная архитектура из трех слоев Conv2d + ReLU и двух независимых выходных голов 1×1, что соответствует матричному входу вида «время × датчики». На 50 одинаковых тестовых примерах чистая модель показала MAE = 2.8699, RMSE = 4.2018, MAPE = 7.0519. Одноточечная атака 1VITA ухудшила прогноз лишь незначительно до RMSE = 4.2284 и MAPE = 7.1013, но потребовала в среднем 314 запросов и 2.6985 секунды на пример. Предложенный ускоренный режим LBA_S дал существенно более сильную деградацию качества: MAE = 15.6818, RMSE = 17.0757, MAPE = 31.1117, при средней стоимости 15 запросов и 0.1009 секунды на пример. Для модели TGC-LSTM уязвимость слабо проявляется на уровне одиночного изменения одного значения во входном окне, но становится значительной, если атака использует более богатую структуру возмущения и опирается на выученную карту чувствительных координат. Практически это означает, что анализ устойчивости графовых моделей прогноза трафика нельзя ограничивать только самыми простыми одноточечными атаками. Теоретически это показывает, что даже интерпретируемая и физически мотивированная пространственно-временная архитектура сохраняет существенную поверхность атаки, если противник способен предварительно накопить и переиспользовать информацию о типовых состязательных примерах.**

Ключевые слова **– прогнозирование трафика дорожного движения, графовые нейронные сети, TGC-LSTM, состязательные атаки, режим черного ящика, 1VITA, обучаемая атака, ускоритель атак, Learning-Based Attack, LBA, сверточная сеть.**

I. Введение

Современные города стремительно трансформируются в «умные» мегаполисы, в которых информационно-телекоммуникационные технологии интегрируются в самые разные сферы городской жизни. Одним из ключевых компонентов такой инфраструктуры являются системы управления транспортной инфраструктурой.

Рост урбанизации и увеличение количества транспортных средств приводят к обострению проблем, связанных с заторами на дорогах. Это негативно сказывается на экологии, экономике и качестве жизни населения. Традиционные системы управления дорожным движением, основанные на статических алгоритмах и датчиках ограниченного охвата, такие как индукционные петли и камеры видеонаблюдения, часто не справляются с динамикой современных транспортных потоков.

На смену им приходят Интеллектуальные транспортные системы (ITS), которые используют большие данные и искусственный интеллект для адаптивного управления трафиком.

Прогнозирование транспортных потоков является одной из центральных задач интеллектуальных транспортных систем. Его практическая значимость определяется тем, что точные прогнозы скорости и состояния дорожной сети используются в маршрутизации, управлении перегрузкой сети и поддержке решений в реальном времени. В этой задаче данные одновременно обладают временной и графовой структурой: с одной стороны, сигнал меняется по времени, а с другой - определяется топологией дорожной сети, где вершины соответствуют датчикам, а ребра отражают физические связи между ними. Именно поэтому графовые нейронные сети и гибридные пространственно-временные архитектуры заняли ключевое место в современных задачах анализа и прогноза временных рядов на транспортных сетях [2].

Одной из характерных моделей данного класса является TGC-LSTM, предложенная в работе [1]. В ней пространственная зависимость моделируется специальной графовой сверткой, которая учитывает не только соседство узлов, но и физическую достижимость одного сегмента сети из другого за фиксированное время свободного движения. Временная зависимость далее описывается с помощью LSTM-блока, модифицированного дополнительным шлюзом состояния ячейки, учитывающим влияние соседних узлов. В исходной

статье показано, что TGC-LSTM превосходит базовые модели ARIMA, SVR, обычную LSTM и несколько графовых вариантов на датасетах LOOP и INRIX, а также обладает лучшей физической интерпретируемостью благодаря регуляризации весов графовой свертки [1].

За последние годы вместе с ростом качества пространственно-временных моделей усилился интерес к их безопасности и устойчивости. Для временных рядов показано, что даже малые специально подобранные возмущения входа способны заметно исказить прогноз [5], [3]. Графовые и транспортные модели также подвержены пространственным и временным целенаправленным воздействиям: предложены практические атаки на модели прогнозирования дорожного трафика, атаки на отдельную вершину графа и атаки с диффузией возмущения по графу [6]-[8]. Эти результаты делают задачу анализа устойчивости TGC-LSTM не только академически интересной, но и практически значимой.

Целью работы является:

a) Проанализировать существующие технологии состязательных атак на графовые модели глубокого обучения, используемые для прогнозирования городского трафика.

b) Проверить устойчивость модели прогнозирования трафика TGC-LSTM к состязательной атаке в режиме черного ящика и реализовать ускоренный вариант атаки.

Специфика задачи прогнозирования по временному окну и по графу создает дополнительные естественные ограничения на атаку: требуется ограничивать амплитуду возмущения, учитывать его правдоподобие с точки зрения физики процесса и, по возможности, сохранять временную гладкость сигнала. Тем не менее существующая литература показывает, что даже при таких ограничениях атаки на предсказывающие модели остаются возможными и заслуживают отдельного исследования [3], [6], [8].

II. Обзор литературы

A. *Графовые модели для прогнозирования временных рядов*

В обзоре [2] подчеркивается, что графовые модели для временных рядов особенно эффективны там, где необходимо одновременно учитывать сложные зависимости переменных и их развитие во времени. В транспортных приложениях это означает необходимость совместно моделировать локальную динамику каждого датчика и влияние соседних участков сети. Работа [1] относится именно к этому направлению: авторы формализуют дорожную сеть как граф и строят для нее специализированную трафиковую графовую свертку, учитывающую как топологию сети, так и свойства потока.

С практической точки зрения TGC-LSTM представляет интерес по двум причинам. Во-первых, модель использует физически осмысленную матрицу достижимости по свободному потоку. То есть извлекаемые пространственные признаки лучше согласуются с транспортной физической интерпретацией. Во-вторых, архитектура специально проектировалась так, чтобы сохранить интерпретируемость графовой части через регуляризацию весов и признаки свертки [1]. Это делает TGC-LSTM хорошим объектом для анализа устойчивости: если атака успешна даже в такой постановке, то речь идет не о слабости простой последовательной сети, а о более глубокой уязвимости пространственно-временной модели.

B. *Состязательные атаки на временные ряды и транспортные графы*

Для временных рядов в литературе можно выделить по меньшей мере три направления исследований. Первое направление - градиентные атаки, в которых возмущение строится по производной целевой функции по входу. Сюда относятся, например, адаптации классических подходов к задаче прогноза временных рядов [5]. Второе - разреженные атаки в режиме черного ящика, в которых вместо изменения всего окна атакующий ищет несколько наиболее чувствительных точек, а оптимизация ведется без знания градиентов. В статье [3] для этой цели предложена атака nVITA, где каждая потенциальная атака кодируется набором троек «время - признак - величина возмущения», а поиск ведется методом дифференциальной эволюции. Третье направление - обучаемые ускорители атак. В работе [4] сформулирована идея Learning-Based Attack (LBA): если сначала построить состязательные примеры более дорогим методом, то затем можно обучить компактную модель, которая будет быстро воспроизводить их структуру без постоянных обращений к целевой системе.

Для пространственно-временных графовых моделей трафика картина похожая, но к ней добавляются ограничения, связанные с причинностью, графовой диффузией и физической правдоподобностью сигнала. В [6] предложена практическая двухшаговая атака, где сначала выбираются наиболее значимые узлы, а затем строится возмущение с ограниченной величиной. По отчету авторов это приводит к деградации качества вплоть до 67.8%. В [7] разработана пространственно целенаправленная атака на одну вершину графа и показано, что эффект может распространяться на значительную часть сети. В [8] предложена атака с диффузией возмущения по графу, в которой для режима черного ящика градиент аппроксимируется методом SPSA, а атакуемые узлы выбираются жадным алгоритмом. Для настоящей работы особенно важны два обстоятельства: пространственно-временные модели прогноза трафика действительно уязвимы, и разреженная локальная атака по-прежнему остается содержательной и практически обоснованной постановкой.

III. Постановка задачи

Рассматривается задача краткосрочного прогноза скорости трафика на графе дорожной сети. Пусть G = (V, E) - граф транспортной сети, где |V| = N равно числу датчиков, а E - это множество ребер-дорог, соединяющих эти датчики. В настоящей работе используется набор LOOP, содержащий данные 323 индуктивных датчиков, расположенных на четырех связанных автомагистралях района Сиэтла. Измерения даны с шагом 5 минут и покрывают весь 2015 год [1]. В каждом экспериментальном примере вход представляет собой окно из T = 10 временных шагов, а выходом служит прогноз на следующий шаг, что соответствует исходной постановке модели TGC-LSTM [1].

$$\mathbf{X_T} = [x_1, \dots, x_T] \in \mathbb{R}^{\mathbb{T}\times\mathbb{N}}, x_t \in \mathbb{R}^{\mathbb{N}}, y = x_{T+1} \in \mathbb{R}^{\mathbb{N}}. \quad (1)$$

Целевая модель f_{tar} получает на вход окно X_T и выдает прогноз следующего вектора скоростей:

$$\hat{y} = f_{\text{tar}}(\boldsymbol{X_T}). \quad (2)$$

Состязательная атака уклонения в режиме черного ящика строит возмущенный вход

$$X_T^{adv} = X_T + \Delta, \quad (3)$$

так, чтобы как можно сильнее исказить прогноз целевой модели при ограниченном бюджете на возмущение. В наиболее общей форме задачу можно записать так:

$$\max_{(\Delta)} \mathcal{J}(\mathrm{X_T} + \Delta) \quad \text{при условиях} \quad |\Delta|_0 \le n, \quad |\Delta|_\infty \le \varepsilon, \quad X_T + \Delta \in [0,1]^{T\times N}. \quad (4)$$

Здесь n - число изменяемых точек, ε - допустимая амплитуда возмущения, а ограничение на интервал [0,1] соответствует нормированной форме входа, используемой в реализации атак.

В работе рассматриваются два варианта целевой функции. Если в момент атаки известна истинная цель $y = x_{T+1}$, то используется усиленная исследовательская постановка:

$$\mathcal{J}_{true}(X_T + \Delta) = \mathrm{MSE}(f_{tar}(X_T + \Delta), y). \quad (5)$$

Если истинная цель недоступна, то атака может быть полностью построена как режим черного ящика через отклонение от исходного прогноза:

$$\mathcal{J}_{clean}(X_T + \Delta) = \mathrm{MSE}\big(f_{tar}(X_T + \Delta), f_{tar}(X_T)\big). \quad (6)$$

Основной массив обучающих примеров для ускоренной атаки в настоящей работе собирался по варианту (5), что нужно учитывать при интерпретации итоговых результатов.

IV. Целевая модель TGC-LSTM

Ключевым отличием TGC-LSTM от обычных последовательных моделей является то, что вход сначала обрабатывается специальной трафиковой графовой сверткой, а затем передается в LSTM-блок [1]. Для этого вводится матрица достижимости по свободному потоку $FFR \in \mathbb{R}^{\mathbb{N}\times\mathbb{N}}$:

$$FFR_{i,j} = \begin{cases} 1, & S_{i,j}^{FF}\, m\Delta t - Dist_{i,j} \\ 0, & \text{иначе} \end{cases} \quad (7)$$

Здесь $S_{i,j}^{\mathrm{FF}}$ - скорость свободного движения между узлами i и j, $\mathrm{Dist_{i,j}}$ - расстояние между ними, Δt - длительность временного шага, а m определяет число временных интервалов, за которое допускается достижимость [1]. Таким образом, матрица FFR отражает не просто графовое соседство, а физическую возможность переноса транспортного потока за заданное время.

Для k-го порядка вводится трафиковая графовая свертка

$$GC_t^k = \big(W_{gc,k} \odot \widetilde{A^k} \odot FFR\big)x_t \quad (8)$$

, где $\widetilde{A^k}$ - k-шаговая матрица соседства, $\odot$ - поэлементное умножение, а $W_{gc,k} \in \mathbb{R}^{\mathbb{N}\times\mathbb{N}}$ - обучаемая матрица весов [1]. Далее признаки разных порядков конкатенируются:

$$GC_t^{\{K\}} = [GC_t^1, GC_t^2, \dots, GC_t^K]. \quad (9)$$

Затем полученные графовые признаки подаются в LSTM-блок. На последнем временном шаге h_T является прогнозом ŷ следующего состояния сети. Базовая функция потерь определяется как среднеквадратичная ошибка прогноза:

$$\mathcal{L}_{pred} = L(x_{T+1}, h_T) \quad (10)$$

К этой функции в исходной модели добавляются два регуляризатора: L_1-штраф на веса графовой свертки и L_2-штраф на различие соседних порядков графовых признаков. Эти регуляризаторы не только ограничивают переобучение, но и повышают интерпретируемость модели.

В исходной статье модель TGC-LSTM показала на наборе LOOP результат MAE = 2.57 ± 0.10, MAPE = 6.01% и RMSE = 4.63, превзойдя все рассмотренные базовые подходы [1]. Это подтверждает, что целевая модель является сильным и нетривиальным объектом атаки.

V. Реализация атаки 1VITA

Основной эталонной атакой в работе является 1VITA - одноточечный вариант разреженной атаки семейства VITA, ищущий изменение только в одной

координате входного окна. Идея такого класса атак восходит к работе [3]. Вместо построения сплошного возмущения по всему окну выбирается малое число наиболее чувствительных точек, а поиск их положения и амплитуды выполняется без доступа к внутренним градиентам целевой модели.

В реализованной функции de_vita_attack каждая построенная атака кодируется тройкой

$$\eta = (t, f, p) \tag{11}$$

, где t ∈ {0, ..., T-1} - индекс времени, f ∈ {0, ..., N-1} - индекс датчика, а p ∈ [-ε, ε] - величина возмущения. Поскольку в настоящей реализации используется n = 1, каждый атакующий пример в популяции дифференциальной эволюции задает ровно одну изменяемую координату. Бюджет по амплитуде вычисляется индивидуально для каждого окна:

$$\varepsilon = \beta(\max X_T - \min X_T) \tag{12}$$

Такой выбор согласуется с постановкой LBA и nVITA, где величина допустимого возмущения задается через коэффициент β относительно динамического диапазона окна [3], [4]. Далее после декодирования триады (t, f, p) значение p обрезается до интервала [-ε, ε], а сам атакованный вход формируется как

$$X_T^{adv} = \mathrm{clip}\big(X_T + p\, e_{t,f}, 0{,}1\big) \tag{13}$$

, где $e_{(t,f)}$- матрица той же размерности, что и вход, имеющая единицу в координате (t, f) и нули в остальных позициях.

Поиск лучшей триады выполняется методом дифференциальной эволюции. В представленной реализации используются параметры, совпадающие с настройками, рекомендованными в статье LBA: размер популяции pop_size = 15, число итераций iters = 60, а при построении обучающего набора берется именно одноточечная атака (n = 1) [4]. Функция приспособленности задается либо формулой (5), либо формулой (6). На каждом шаге алгоритм многократно вызывает оракул целевой модели, сравнивает потомков и родителей и оставляет вариант с большим значением целевой функции. Достоинство этой схемы состоит в полном отсутствии потребности в градиентах. Но при этом есть недостаток в виде высокой вычислительной нагрузки по числу запросов.

VI. Построение набора состязательных примеров

Следующий этап состоит в использовании дорогой поисковой атаки для обучения вспомогательной модели. Для этого по результатам 1VITA формируется набор

$$D_{adv} = \{(X_i, loc_i, p_i)\}_{i=1}^{M} \tag{14}$$

, где X_i- исходное окно трафика, $\mathrm{loc_i}$ - индекс наиболее чувствительной точки, найденной 1VITA, а $\mathrm{p_i}$ - величина возмущения в этой точке. Индекс определяется как

$$loc_i = t_i\, N + f_i. \tag{15}$$

Это используется для того, чтобы вместо обучения сети сразу генерировать весь возмущенный тензор, задача разбивается на две подзадачи. Первая - поиск чувствительной координаты. Вторая - оценка нужной амплитуды воздействия. Эта идея непосредственно соответствует подходу LBA, где обучаемая модель получает на вход исходные данные, а на выходе строит последовательность чувствительности и последовательность возмущений [4].

Функция build_dadv_dataset реализует этот процесс следующим образом. Для каждого примера она запускает 1VITA, извлекает найденную триаду (t, f, p), переводит пару (t, f) в плоский индекс (15) и сохраняет кортеж (X, loc, p). Тем самым дорогой алгоритм 1VITA используется как генератор разметки для более быстрой модели второго уровня. В этом и состоит ключевая идея ускоренной атаки: офлайн мы платим за поиск чувствительных точек, а онлайн переиспользуем уже выученные закономерности.

VII. Ускоренная атака на основе сверточной сети

A. *Архитектура вспомогательной модели FlearnCNN2D*

В статье [4] в качестве обучаемой модели LBA использовалась небольшая сверточная сеть с байесовскими сверточными слоями, полносвязным слоем, нормализацией и двумя независимыми выходными слоями: один предсказывает чувствительные точки, второй - значения возмущения. В настоящей работе эта идея адаптируется к пространственно-временной матрице входа и реализуется в виде полностью двумерной архитектуры:

$$F_{learn}\colon \mathbb{R}^{\mathbb{B}\times 1\times \mathbb{T}\times \mathbb{N}} \to (\mathbb{R}^{\mathbb{B}\times \mathbb{T}\times \mathbb{N}},\ \mathbb{R}^{\mathbb{B}\times \mathbb{T}\times \mathbb{N}}) \tag{16}$$

Здесь первый выход отвечает за карту чувствительности, а второй - за карту предсказанных возмущений.

Архитектура сети FlearnCNN2D представляет собой последовательность трех двумерных сверточных слоев:

$$\begin{aligned} H_1 &= \mathrm{ReLU}(\mathrm{Conv}_{3\times 3}^{1\to 32}(X)),\\ H_2 &= \mathrm{ReL\,U}(\mathrm{Conv}_{3\times 3}^{32\to 32}(H_1)),\\ H_3 &= \mathrm{ReLU}(\mathrm{Conv}_{3\times 3}^{32\to 32}(H_2)). \end{aligned} \tag{17}$$

После общего блока идут две независимые

выходные головы размерности 1×1:

$$Z = \mathrm{Conv}_{1\times1}^{32\to1}(H_3) \in \mathbb{R}^{\mathbb{T}\times\mathbb{N}},$$
$$P = \mathrm{Conv}_{1\times1}^{32\to1}(H_3) \in \mathbb{R}^{\mathbb{T}\times\mathbb{N}}. \quad (18)$$

Здесь Z интерпретируется как карта чувствительности, а P - как карта величин возмущения.

Выбор именно такой архитектуры обусловлен свойствами задачи:

a) Двумерная форма входа. Окно трафика уже имеет естественную матричную структуру «время × датчики», поэтому двумерная свертка лучше соответствует данным, чем одномерная. Она позволяет одновременно учитывать локальный временной и пространственный контекст.

b) Ядро 3×3 и сохранение разрешения. Ядро 3×3 захватывает ближайшее окружение каждой координаты без резкого роста вычислений. Дополнение padding = 1 сохраняет размер карты, а значит предсказанная координата напрямую соответствует координате входа.

c) Отсутствие пулинга. В сети нет слоев уменьшения размерности, поскольку задача требует не только извлечь признаки, но и точно указать, какую клетку тензора нужно изменить.

d) Две независимые головы. Карта чувствительности и карта амплитуд решают разные задачи. Разделение на две головы после общего признакового блока позволяет совместно использовать извлеченный контекст, но не смешивать классификацию позиции и регрессию величины.

e) Компактность. В отличие от целевой модели TGC-LSTM, ускоряющая сеть не обязана моделировать сложную пространственно-временную динамику всего трафика. Ее задача - по виду входного окна быстро угадать, где атака вероятнее всего будет наиболее эффективной и насколько нужно сдвинуть значение.

B. Функция потерь и метрики обучения

Пусть для батча размера B сеть выдает карту чувствительности Z и карту возмущений P. После разворачивания в векторы длины L = TN получаем

$$z_i = \mathrm{vec}(Z_i) \in \mathbb{R}^{\mathbb{L}},\ q_i = \mathrm{vec}(P_i) \in \mathbb{R}^{\mathbb{L}}. \quad (19)$$

Тогда первая часть функции потерь отвечает за локализацию чувствительной точки и задается перекрестной энтропией:

$$\mathcal{L}_{loc} = -\frac{1}{B}\sum_{i=1}^{B}\log\left(\frac{\exp(z_i[loc_i])}{\sum_{j=1}^{L}\exp(z_i[j])}\right). \quad (20)$$

Вторая часть функции потерь - это регрессия амплитуды на истинной чувствительной позиции:

$$\mathcal{L}_{p} = \frac{1}{B}\sum_{i=1}^{B}(q_i[loc_i] - p_i)^2. \quad (21)$$

Полная функция потерь имеет вид

$$\mathcal{L}_{flearn} = \mathcal{L}_{loc} + \lambda_p\,\mathcal{L}_{p}. \quad (22)$$

Тем самым сеть обучается одновременно отвечать на два вопроса: где атаковать и на сколько изменять выбранную точку.

В предложенной реализации для обучения используется оптимизатор Adam с параметрами lr = 5·10^(-3), размером батча 8 и числом эпох 50. Такие же базовые гиперпараметры обучения LBA указываются и в статье [4]. Помимо функции потерь, по эпохам считаются три метрики качества: точность точного попадания AR@1, точность попадания в множество первых k позиций AR@k и ошибка предсказания амплитуды в истинной точке p_RMSE@trueLoc. Это разделение особенно важно, потому что ускоренная атака может работать плохо либо из-за ошибочной локализации, либо из-за неточной оценки амплитуды даже при верной локализации.

VIII. Режимы генерации ускоренной атаки

A. Прямой разреженный режим

Функция lba_attack_batch является наиболее близким аналогом исходной идеи LBA и наиболее честным ускоренным суррогатом 1VITA. Для каждого примера вычисляется карта чувствительности Z и карта возмущений P, после чего выбирается наиболее чувствительная позиция:

$$\widehat{loc} = \arg\max_{j\in\{1,\dots,TN\}} z[j]. \quad (23)$$

Амплитуда берется из второй головы:

$$\hat{p} = \mathrm{clip}\big(\delta\, q[\widehat{loc}], -\varepsilon, \varepsilon\big) \quad (24)$$

Затем формируется состязательный пример:

$$X_T^{adv} = \mathrm{clip}(X_T + \hat{p}\, e_{\widehat{loc}}, 0, 1) \quad (25)$$

Если брать только одну лучшую точку, получается по-настоящему одноточечная обучаемая атака. Метод практически не требует обращений к целевой модели после завершения обучения, что и делает его привлекательным как ускоритель атаки.

B. Усиленный гибридный режим LBA_S

Вторая функция - lba_sign_search_batch_poison - реализует более сильный гибридный режим, который в данной работе обозначается как LBA_S. Здесь вспомогательная сеть сначала строит карту чувствительности, после чего атака ограничивается последними двумя временными шагами окна. Это делается через маску, оставляющую только наиболее поздние наблюдения, поскольку именно они обычно сильнее всего влияют на краткосрочный прогноз.

Далее выбирается множество наиболее

чувствительных координат

$$I_k(X_T) = \mathrm{TopK}(Z(X_T), k),$$
$$k = \max\big(1, round(\rho\, TN)\big) \quad (26)$$

, где ρ = poison_frac - доля атакуемых координат. После выбора множества I_k строятся два кандидата:

$$X_T^+ = \mathrm{clip}\left(X_T + \varepsilon \sum_{j \in I_k} e_j, 0{,}1\right)$$
$$X_T^- = \mathrm{clip}\left(X_T - \varepsilon \sum_{j \in I_k} e_j, 0{,}1\right) \quad (27)$$

после чего выбирается знак, дающий большее смещение прогноза относительно чистого входа:

$$X_T^{adv} = \arg \max_{X' \in \{X_T^+, X_T^-\}} |f_{tar}(X') - f_{tar}(X_T)|_2^2. \quad (28)$$

Этот режим уже нельзя считать полностью беззапросным, поскольку на этапе выбора знака он требует ограниченного числа обращений к целевой модели через оракул. Тем не менее он на порядки дешевле исходной поисковой атаки, потому что основная трудность - локализация чувствительных координат, которая уже решается быстрым прямым проходом обученной сверточной сети. Именно этим объясняется сочетание двух свойств LBA_S: малой стоимости и высокой силы воздействия.

IX. Экспериментальная реализация

Эксперименты проводятся на наборе LOOP. Входной пример представляет собой окно длины T = 10, а прогноз строится на один следующий шаг. В реализации атак входы рассматриваются в нормированном диапазоне [0, 1], что соответствует коду атакующих функций и общей логике работ по LBA и nVITA [3], [4].

Целевая модель TGC-LSTM была обучена и запущена на этих данных. Затем была реализована 1VITA как эталонная поисковая атака в режиме черного ящика. После этого на основе 1VITA был собран набор D_adv и обучена сверточная сеть f_learn, которая далее использовалась в двух режимах: как прямой ускоренный аналог 1VITA и как усиленный гибридный режим LBA_S. Итоговое сравнение выполнено на 50 одинаковых тестовых примерах (count = 50).

Для качества прогноза используются стандартные для прогноза транспортного трафика метрики MAE, MAPE и RMSE, принятые и в исходной статье TGC-LSTM [1]. Дополнительно в отчете эксперимента учитываются показатель RSE, нормированная сумма квадратов ошибок REL_SSE, среднее число запросов к целевой модели и среднее время генерации одного состязательного примера.

$$\mathrm{MAE} = \frac{1}{m}\sum_{i=1}^{m} |\hat{y}_i - y_i|,$$
$$\mathrm{MAPE} = \frac{100}{m}\sum_{i=1}^{m} \left|\frac{\hat{y}_i - y_i}{y_i}\right|,$$
$$\mathrm{RMSE} = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(\hat{y}_i - y_i)^2}, \quad (29)$$
$$\mathrm{REL\_SSE} = \frac{\sum_{i=1}^{m}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{m}(y_i - \bar{y})^2},$$
$$\mathrm{RSE} = \sqrt{\mathrm{RELSSE}}.$$

Программная реализация модели, атак и сценариев эксперимента доступна в репозитории [9].

X. Результаты

Основные результаты приведены в таблице I. В первой строке показано качество чистой модели без атаки. Во второй строке - эффект строго одноточечной поисковой атаки 1VITA. В третьей строке - результат ускоренного гибридного режима LBA_S.

Таблица 1. Сравнение качества прогноза и стоимости атаки на 50 одинаковых тестовых примерах

Метод	MAE	RMSE	MAPE	RSE	Запросы	Время, с
Без атаки	2.8699	4.2018	7.0519	0.3349	0.0	0.0000
1VITA	2.8823	4.2284	7.1013	0.3370	314.0	2.6985
LBA_S	15.6818	17.0757	31.1117	1.3609	15.0	0.1009

Полученные числа позволяют сделать несколько содержательных выводов.

Во-первых, TGC-LSTM в исследуемой конфигурации оказывается достаточно устойчивой к строго одноточечной атаке 1VITA. При переходе от чистого режима к 1VITA значение MAE возрастает лишь с 2.8699 до 2.8823 (примерно на 0.43%), RMSE - с 4.2018 до 4.2284 (примерно на 0.63%), а MAPE - с 7.0519 до 7.1013 (примерно на 0.70%). Иными словами, оптимально подобранное изменение одной клетки входного окна при выбранном бюджете по амплитуде почти не разрушает прогноз. Это важный результат сам по себе, поскольку сильная графовая модель, учитывающая соседство и достижимость по свободному потоку, оказывается не слишком чувствительной к одноточечному вмешательству.

Во-вторых, 1VITA оказывается очень дорогой атакой. На один пример уходит в среднем 314 запросов к целевой модели и 2.6985 секунды. Это согласуется с общей природой метода: отсутствие

доступа к градиентам компенсируется большим числом внешних запросов и дорогостоящим поиском по пространству триад (t, f, p) [3].

В-третьих, ускоренный режим LBA_S дает качественно иной эффект. Ошибка прогноза растет очень резко: MAE увеличивается до 15.6818, RMSE - до 17.0757, MAPE - до 31.1117. По сравнению с чистой моделью это означает рост MAE примерно на 446.42%, RMSE - на 306.39%, а MAPE - на 341.18%. При этом среднее число запросов снижается с 314 до 15 (снижение примерно на 95.22%), а среднее время - с 2.6985 до 0.1009 секунды (снижение примерно на 96.26%). Таким образом, обучаемая локализация чувствительных точек действительно решает основную вычислительную проблему поисковой атаки.

В-четвертых, сопоставление 1VITA и LBA_S необходимо интерпретировать осторожно. Согласно журналу эксперимента, для 1VITA количество измененных точек равно 1, тогда как для LBA_S оно равно 646. Это означает, что ускоренный режим вносит возмущения в более широкий набор точек. Следовательно, LBA_S не является строгим одноточечным эквивалентом 1VITA. Это более сильная гибридная атака, в которой разреженный поиск заменен обучаемой локализацией, а затем дополнительно используется знак, выбираемый по ограниченному числу запросов к целевой модели. Поэтому корректнее говорить, что одноточечная атака почти не нарушает прогноз TGC-LSTM, но выученная карта чувствительности позволяет построить гораздо более сильную и одновременно существенно более дешевую атаку в более широком классе возмущений.

XI. Анализ результатов

Полученные результаты допускают несколько интерпретаций.

Первая интерпретация связана с самой архитектурой TGC-LSTM. В отличие от обычной предсказывающей модели, она не полагается только на локальную временную память одного датчика, а агрегирует признаки по соседним узлам графа, причем область агрегации ограничивается физически осмысленной матрицей достижимости FFR [1]. Это означает, что единичное возмущение одной координаты частично «растворяется» в пространственно-временном контексте, что и может объяснять слабый эффект 1VITA.

Вторая интерпретация состоит в том, что для задач прогноза по окну особенно важны последние наблюдения. В режиме LBA_S атака намеренно ограничивается последними двумя временными шагами. С точки зрения причинности краткосрочного прогноза это выглядит содержательно оправданным. Ближайшее будущее обычно сильнее зависит от последних наблюдений, чем от ранних фрагментов истории. Обучаемая сеть при этом выступает не как замена целевой модели, а как быстрый индикатор того, где именно в хвосте окна расположены наиболее чувствительные координаты.

Наконец, результаты согласуются с литературой по устойчивости пространственно-временных моделей. Работы [6]-[8] показывают, что сильные атаки на транспортные графы обычно опираются не на одно изолированное изменение, а на более структурированное воздействие: выбор важных узлов, использование нескольких временных точек, диффузию по графу или оптимизацию по более богатому классу возмущений. В этом смысле полученная разница между 1VITA и LBA_S не выглядит случайной, а наоборот, отражает общий закон: чем лучше атака использует структуру пространственно-временной модели, тем разрушительнее оказывается ее эффект.

XII. Ограничения исследования

a) Одна целевая архитектура и один набор данных. Эксперименты выполнены для TGC-LSTM на наборе LOOP. Для более общего вывода о защищенности графовых моделей прогноза трафика требуются дополнительные эксперименты на других архитектурах и сетях.

b) При построении обучающего набора D_{adv} использовалась цель (5), то есть с доступом к истинному следующему шагу. Для полностью строгого режима черного ящика необходимо дополнительно исследовать вариант (6) на всех этапах.

c) Разный бюджет возмущения у сравниваемых режимов. 1VITA является одноточечной атакой, тогда как LBA_S использует многоточечное воздействие и дополнительный выбор знака с использованием оракула. Поэтому их сравнение корректно прежде всего с точки зрения практической опасности и вычислительной цены, но не как строгое сравнение при одинаковой L0-мощности возмущения.

d) Отсутствие явной проверки физической правдоподобности. Ограничение по амплитуде и локализации уменьшает нереалистичность атаки, однако в настоящей работе отдельно не исследуются дополнительные условия вроде монотонности, гладкости или соответствия статистике реальных шумов датчиков.

XIII. Заключение

В работе была исследована устойчивость модели TGC-LSTM к состязательным атакам в режиме черного ящика и предложен ускоренный вариант атаки на основе обучаемой двумерной сверточной сети. В качестве эталонной поисковой атаки была реализована 1VITA, в которой единственная изменяемая точка входного окна подбирается методом дифференциальной эволюции. На ее основе был построен набор D_adv, содержащий исходные окна трафика, локализации чувствительных точек и найденные амплитуды возмущений. Далее по этому набору была обучена компактная сеть f_learn,

имеющая общий сверточный блок и две независимые головы для поиска точки и регрессии амплитуды.

Эксперименты показали, что TGC-LSTM в исследуемой постановке остается сравнительно устойчивой к одноточечной атаке: 1VITA почти не изменяет ошибки прогноза, но требует большого числа запросов к целевой модели. Одновременно показано, что обучаемая карта чувствительности позволяет построить гораздо более дешевую атаку, а в усиленном гибридном режиме LBA_S - и существенно более разрушительную. Тем самым основная вычислительная трудность поисковой атаки переносится в офлайн-этап обучения, а на этапе исполнения заменяется очень быстрым прямым проходом компактной сверточной сети.

Главный вывод работы состоит в том, что для модели TGC-LSTM уязвимость слабо проявляется на уровне одиночного изменения одного значения во входном окне, но становится значительной, если атака использует более богатую структуру возмущения и опирается на выученную карту чувствительных координат. Практически это означает, что анализ устойчивости графовых моделей прогноза трафика нельзя ограничивать только самыми простыми одноточечными атаками. Теоретически это показывает, что даже интерпретируемая и физически мотивированная пространственно-временная архитектура сохраняет существенную поверхность атаки, если противник способен предварительно накопить и переиспользовать информацию о типовых состязательных примерах.

В качестве естественного продолжения работы представляются перспективными следующие направления: a) строгий паритетный эксперимент между 1VITA и LBA_S, в котором используется только одна найденная точка; b) полный перенос всего конвейера в режим без знания истинной цели, то есть с использованием только функции (6); c) исследование физической правдоподобности, детектируемости и переносимости атак на другие модели и другие дорожные сети.

Благодарности

Авторы благодарят сотрудников кафедры Информационной безопасности за критику и ценные обсуждения. Вопросы использования Искусственного интеллекта в кибербезопасности являются одним из основных научных направлений кафедры ИБ факультета ВМК МГУ имени М.В. Ломоносова и рассматривались во множестве магистерских диссертаций и научных работ [10, 11, 12]. Также, традиционно, отмечаем работы В.П. Куприяновского и его соавторов, положивших начало цифровой тематике в журнале INJOIT [13, 14].

Библиография

[1] Cui Z., Henrickson K., Ke R., Pu Z., Wang Y. Traffic Graph Convolutional Recurrent Neural Network: A Deep Learning Framework for Network-Scale Traffic Learning and Forecasting // IEEE Transactions on Intelligent Transportation Systems. 2020. Vol. 21. No. 11. P. 4883-4894.

[2] Jin M., Koh H. Y., Wen Q., Zambon D., Alippi C., Webb G. I., King I., Pan S. A survey on graph neural networks for time series: Forecasting, classification, imputation, and anomaly detection // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2024. Vol. 46. No. 12. P. 10466-10485.

[3] Chen Z., Dost K., Zhu X., Chang X., Dobbie G., Wicker J. Targeted Attacks on Time Series Forecasting // Pacific-Asia Conference on Knowledge Discovery and Data Mining. Cham: Springer Nature Switzerland, 2023. P. 314-327.

[4] Xiao Y., Yang Z., Zou Q., Zhang P. Learn from Adversarial Examples: Learning-Based Attack on Time Series Forecasting // Information Technology and Control. 2025. Vol. 54. No. 2. P. 613-628.

[5] Xu A., Wang X., Zhang Y., Wu T., Xian X. Adversarial Attacks on Deep Neural Networks for Time Series Prediction // 2021 10th International Conference on Internet Computing for Science and Engineering. 2021. P. 8-14.

[6] Liu F., Liu H., Jiang W. Practical Adversarial Attacks on Spatiotemporal Traffic Forecasting Models // Advances in Neural Information Processing Systems. 2022. Vol. 35. P. 19035-19047.

[7] Liu F., Miranda-Moreno L., Sun L. Spatially Focused Attack against Spatiotemporal Graph Neural Networks // arXiv preprint. 2021.

[8] Zhu L., Feng K., Pu Z., Ma W. Adversarial Diffusion Attacks on Graph-based Traffic Prediction Models // IEEE Internet of Things Journal. 2023. Vol. 11. No. 1. P. 1481-1495.

[9] LSTM https://github.com/Danilka776/TGC-LSTM-attack

[10] Лапонина, О. Р., and Р. Н. Костин. "Разработка программного обеспечения моделирования угроз для систем на базе LLM-агентов." International Journal of Open Information Technologies 13.6 (2025): 132-146.

[11] Бербер, Д. В., and О. Р. Лапонина. "Разработка подходов к увеличению устойчивости моделей машинного обучения для обнаружения распределенных атак отказа обслуживания." International Journal of Open Information Technologies 13.6 (2025): 16-24.

[12] Намиот, Д. Е. Схемы атак на модели машинного обучения / Д. Е. Намиот // International Journal of Open Information Technologies. – 2023. – Т. 11, № 5. – С. 68-86. – EDN YVRDOB.

[13] Куприяновский, В. П. Демистификация цифровой экономики / В. П. Куприяновский, Д. Е. Намиот, С. А. Синягов // International Journal of Open Information Technologies. – 2016. – Т. 4, № 11. – С. 59-63. – EDN WXQLIJ..

[14] Цифровая железная дорога - инновационные стандарты и их роль на примере Великобритании / Д. Е. Николаев, В. П. Куприяновский, Г. В. Суконников [и др.] // International Journal of Open Information Technologies. – 2016. – Т. 4, № 10. – С. 55-61. – EDN WXBAZN.

Статья получена 12 апреля 2026.

Д.Д. Тарасов – МГУ имени М.В. Ломоносова (email: s02240560@gse.cs.msu.ru).
О.Р. Лапонина – МГУ имени М.В. Ломоносова (email: laponina@oit.cmc.msu.ru)

Improving the Efficiency of Adversarial Attacks on the TGC-LSTM Traffic Prediction Model

D.D. Tarasov, O.R. Laponina

***Abstract* – This paper examines the robustness of the TGC-LSTM urban traffic forecasting model to black-box adversarial evasion attacks and proposes an accelerated version of this attack based on a trainable convolutional network. The TGC-LSTM architecture is considered as the target model. This paper describes an implementation of the single-point 1VITA attack, in which a perturbation is found without access to the target model's gradients using differential evolution (DE), which is consistent with the idea of a sparse attack on time series predictive models. Next, a training set is constructed from the identified 1VITA examples, and an auxiliary convolutional network is trained on it, predicting the input window sensitivity map and the perturbation magnitude map. This paper proposes a fully two-dimensional architecture consisting of three Conv2d + ReLU layers and two independent 1×1 output heads, corresponding to a time × sensors matrix input. On 50 identical test examples, the pure model achieved MAE = 2.8699, RMSE = 4.2018, and MAPE = 7.0519. The single-point 1VITA attack degraded the prediction only slightly to RMSE = 4.2284 and MAPE = 7.1013, but required an average of 314 queries and 2.6985 seconds per example. The proposed accelerated LBA_S mode yielded significantly more severe performance degradation: MAE = 15.6818, RMSE = 17.0757, and MAPE = 31.1117, with an average cost of 15 queries and 0.1009 seconds per example. For the TGC-LSTM model, the vulnerability is weak at the level of a single change in a single value in the input window, but becomes significant if the attack uses a richer perturbation structure and relies on a learned map of sensitive coordinates. In practice, this means that the robustness analysis of graph traffic forecasting models cannot be limited to the simplest single-point attacks. Theoretically, this shows that even an interpretable and physically motivated spatiotemporal architecture retains a significant attack surface if the adversary is able to accumulate and reuse information about typical adversarial examples.**

***Keywords – Traffic forecasting, graph neural networks, TGC-LSTM, adversarial attacks, black-box mode, 1VITA, trainable attack, attack accelerator, Learning-Based Attack, LBA, convolutional network*.**

REFERENCES

[1] Cui Z., Henrickson K., Ke R., Pu Z., Wang Y. Traffic Graph Convolutional Recurrent Neural Network: A Deep Learning Framework for Network-Scale Traffic Learning and Forecasting // IEEE Transactions on Intelligent Transportation Systems. 2020. Vol. 21. No. 11. P. 4883-4894.

[2] Jin M., Koh H. Y., Wen Q., Zambon D., Alippi C., Webb G. I., King I., Pan S. A survey on graph neural networks for time series: Forecasting, classification, imputation, and anomaly detection // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2024. Vol. 46. No. 12. P. 10466-10485.

[3] Chen Z., Dost K., Zhu X., Chang X., Dobbie G., Wicker J. Targeted Attacks on Time Series Forecasting // Pacific-Asia Conference on Knowledge Discovery and Data Mining. Cham: Springer Nature Switzerland, 2023. P. 314-327.

[4] Xiao Y., Yang Z., Zou Q., Zhang P. Learn from Adversarial Examples: Learning-Based Attack on Time Series Forecasting // Information Technology and Control. 2025. Vol. 54. No. 2. P. 613-628.

[5] Xu A., Wang X., Zhang Y., Wu T., Xian X. Adversarial Attacks on Deep Neural Networks for Time Series Prediction // 2021 10th International Conference on Internet Computing for Science and Engineering. 2021. P. 8-14.

[6] Liu F., Liu H., Jiang W. Practical Adversarial Attacks on Spatiotemporal Traffic Forecasting Models // Advances in Neural Information Processing Systems. 2022. Vol. 35. P. 19035-19047.

[7] Liu F., Miranda-Moreno L., Sun L. Spatially Focused Attack against Spatiotemporal Graph Neural Networks // arXiv preprint. 2021.

[8] Zhu L., Feng K., Pu Z., Ma W. Adversarial Diffusion Attacks on Graph-based Traffic Prediction Models // IEEE Internet of Things Journal. 2023. Vol. 11. No. 1. P. 1481-1495.

[9] LSTM https://github.com/Danilka776/TGC-LSTM-attack

[10] Laponina, O. R., and R. N. Kostin. "Razrabotka programmnogo obespecheniya modelirovaniya ugroz dlya sistem na baze LLM-agentov." International Journal of Open Information Technologies 13.6 (2025): 132-146.

[11] Berber, D. V., and O. R. Laponina. "Razrabotka podhodov k uvelicheniyu ustojchivosti modelej mashinnogo obucheniya dlya obnaruzheniya raspredelennyh atak otkaza obsluzhivaniya." International Journal of Open Information Technologies 13.6 (2025): 16-24.

[12] Namiot, D. E. Skhemy atak na modeli mashinnogo obucheniya / D. E. Namiot // International Journal of Open Information Technologies. – 2023. – T. 11, № 5. – S. 68-86. – EDN YVRDOB.

[13] Kupriyanovskij, V. P. Demistifikaciya cifrovoj ekonomiki / V. P. Kupriyanovskij, D. E. Namiot, S. A. Sinyagov // International Journal of Open Information Technologies. – 2016. – T. 4, № 11. – S. 59-63. – EDN WXQLIJ..

[14] Cifrovaya zheleznaya doroga - innovacionnye standarty i ih rol' na primere Velikobritanii / D. E. Nikolaev, V. P. Kupriyanovskij, G. V. Sukonnikov [i dr.] // International Journal of Open Information Technologies. – 2016. – T. 4, № 10. – S. 55-61. – EDN WXBAZN