Инженерный вестник Дона, №12 (2025)
ivdon.ru/ru/magazine/archive/n12y2025/10590

Сравнительный анализ методов прогнозирования временных рядов спроса и цен на рынке грузоперевозок

Д.В. Бокарев

Российская академия народного хозяйства и государственной службы при Президенте Российской Федерации, г. Москва

Аннотация: Волатильность тарифов и объемов спроса на рынке автомобильных грузоперевозок создает неопределенность при планировании операционной деятельности транспортных компаний и формировании логистических бюджетов грузоотправителей. В статье представлен сравнительный анализ точности пяти методов прогнозирования временных рядов спроса и цен: классических статистических подходов, алгоритмов машинного обучения и архитектуры глубокого обучения. Исследование проведено на данных цифровой площадки рынка городских и межгородских автомобильных грузоперевозок. Результаты демонстрируют последовательное повышение точности прогнозов при переходе от линейных эконометрических моделей к нелинейным алгоритмам: градиентный бустинг обеспечил наименьшие значения ошибок, превзойдя классические методы на 30-35% и незначительно опередив рекуррентную нейронную сеть.

Ключевые слова: прогнозирование временных рядов, автомобильные грузоперевозки, машинное обучение, градиентный бустинг, рекуррентные нейронные сети, сравнительный анализ методов прогнозирования

Введение

Рынок грузоперевозок характеризуется выраженной волатильностью основных экономических показателей: тарифы на услуги транспортировки могут изменяться на 15-30% в течение квартала (так, согласно отчету транспортной компании Novelco, во II квартале 2024 года рост составил 14%, в III квартале – 21% [1]), а объемы перевозок демонстрируют сезонные колебания, достигающие двукратных значений для отдельных направлений. Подобная изменчивость создает трудности для участников рынка при планировании операционной деятельности. Нередко транспортные компании сталкиваются с необходимостью балансировать между риском недозагрузки транспортных мощностей и потерями от неспособности удовлетворить возросший спрос, что напрямую влияет на финансовые результаты.

Грузоотправители, в свою очередь, формируют логистические бюджеты в условиях неопределенности относительно будущих тарифов, тем самым затрудняется задача калькуляции себестоимости продукции и ценообразования. Данное обстоятельство обусловливает потребность в инструментах количественного анализа, позволяющих снизить неопределенность при принятии управленческих решений. Методы прогнозирования временных рядов предоставляют математический аппарат для оценки будущих значений спроса и цен на основе исторических данных, однако их точность и применимость различаются в зависимости от специфики анализируемых процессов и горизонта прогнозирования.

Существующий методический инструментарий прогнозирования охватывает широкий спектр подходов: от классических статистических моделей, например интегрированая модель авторегрессии — скользящего среднего (autoregressive integrated moving average — ARIMA) до алгоритмов машинного обучения (градиентный бустинг, рекуррентные нейронные сети и др.) и гибридных решений. Каждый метод обладает определенными преимуществами и ограничениями, связанными с предположениями о структуре данных, чувствительностью к выбросам, способностью учитывать нелинейные зависимости и внешние факторы. Отсутствие единого подхода, универсально применимого для всех задач прогнозирования на транспортном рынке, требует проведения сравнительного анализа альтернативных методов с оценкой их эффективности на реальных данных.

Между тем, научная литература содержит ограниченное число исследований, систематизирующих опыт применения различных прогнозных техник именно в контексте рынка грузоперевозок с учетом его отраслевых особенностей. Анализ академических публикаций последних лет, посвященных применению машинного обучения в отношении транспортной отрасли, позволяет выделить две основные тенденции. Первая связана с

превосходством нейросетевых архитектур над традиционными эконометрическими подходами: исследования европейского [2], турецкого [3] и корейского [4] рынков автомобильных грузоперевозок демонстрируют, что многослойные перцептроны и алгоритмы градиентного бустинга обеспечивают более точные прогнозы ставок фрахта по сравнению с моделями семейства ARIMA. Вторая тенденция проявляется в активном использовании рекуррентных нейронных сетей для морского и железнодорожного транспорта: Долгая краткосрочная память (long short-term memory — LSTM) успешно применяется при прогнозировании контейнерных индексов [5, 6], пропускной способности портов [7, 8] и загруженности железнодорожных узлов [9, 10], достигая показателей точности коэффициента детерминации R^2 на уровне 96-97%. Примечательно, что сравнительные исследования методов экстремального градиентного бустинга (extreme gradient boosting — XGBoost) и LSTM для задач прогнозирования в транспортной сфере показывают неоднозначные результаты: если для энергопотребления транспорта градиентный бустинг значительно превосходит рекуррентные сети (для первого основные показатели составили R^2 = 0,9508, средняя абсолютная ошибка в процентах (mean absolute percentage error — MAPE) = 1,08%, для второго – R^2 = 0,2005, MAPE = 6,06%) [11], то для контейнерных перевозок гибридные модели с сочетанием LSTM и сверточных нейронных сетей (convolutional neural network — CNN) демонстрируют наивысшую точность [12]. Между тем, систематические исследования, сравнивающие эффективность различных подходов применительно к автомобильному рынку грузоперевозок в российских условиях с учетом специфики городских и межгородских перевозок, остаются малочисленными. Новизна настоящего исследования состоит в адаптации интеллектуальных методов под специфику рынка грузоперевозок

на данных, отражающих особенности российского рынка автомобильных грузоперевозок, в сравнении с традиционными методами.

Исходя из этого, цель статьи заключается в сравнительной оценке точности и применимости основных методов прогнозирования временных рядов для задач предсказания спроса и цен на услуги грузоперевозок, а также в выработке рекомендаций по выбору оптимального подхода в зависимости от характеристик прогнозируемого процесса и требований к горизонту прогноза.

Материалы и методы

Для проведения сравнительного анализа были использованы временные ряды, полученные с цифровой платформы, связывающей заказчиков и исполнителей автомобильных грузоперевозок в городских и межгородских направлениях по России. Данные охватывали период в два года (730 ежедневных наблюдений) и включали две целевые переменные: объём спроса на услуги перевозок (в количестве заказов) и среднюю цену за километр маршрута. Наблюдаемые временные ряды характеризовались наличием трендовой составляющей, сезонных колебаний недельной и месячной периодичности, а также стохастического шума, отражающего флуктуации рыночной конъюнктуры. Для оценки качества прогнозных моделей выборка была разделена на обучающую (80% наблюдений) и тестовую (20% наблюдений) части, что позволило проверить точность прогнозов на данных, не использовавшихся при калибровке.

Результаты исследования

Первым этапом анализа стало применение классических статистических методов, разработанных в рамках эконометрической традиции второй половины XX века. Модель авторегрессии и скользящего среднего (ARIMA), предложенная Боксом и Дженкинсом в 1970 году [13], остается одним из наиболее распространенных инструментов

прогнозирования благодаря прозрачности математического аппарата и относительной простоте интерпретации параметров. Общая форма модели ARIMA(p, d, q) записывается как:

$$(1 - \varphi_1 B - \varphi_2 B^2 - \ldots - \varphi_p B^p)(1 - B)^d y_t = (1 + \theta_1 B + \theta_2 B^2 + \ldots + \theta_\varphi B^g)\varepsilon_t$$

,

где B — оператор сдвига назад, p — порядок авторегрессионной части, d — порядок разности для достижения стационарности, q — порядок скользящего среднего, φ_i и θ_j — коэффициенты модели, ε_t — белый шум. Для реализации данного подхода был разработан программный код на языке Python с использованием библиотеки statsmodels:

```
python
from statsmodels.tsa.arima.model import ARIMA
from sklearn.metrics import mean_absolute_error, mean_squared_error

# Подбор параметров и обучение модели
model_arima = ARIMA(train_data, order=(5, 1, 2))
fitted_arima = model_arima.fit()

# Прогнозирование на тестовой выборке
forecast_arima = fitted_arima.forecast(steps=len(test_data))
mae_arima = mean_absolute_error(test_data, forecast_arima)
rmse_arima = np.sqrt(mean_squared_error(test_data, forecast_arima))
```

Результаты применения ARIMA продемонстрировали способность модели улавливать основные трендовые и сезонные паттерны в данных (рис. 1). Средняя абсолютная ошибка (mean absolute error — MAE) для прогноза спроса составила 12,4 заказа, а корень из средней квадратичной ошибки (root mean squared error — RMSE) — 16,7 заказа. Для временного ряда цен точность оказалась выше: MAE = 0,83 руб./км, RMSE = 1,12 руб./км.

Подобные показатели указывают на то, что линейная структура ARIMA достаточно адекватно описывает динамику цен, характеризующуюся относительной стабильностью и предсказуемостью изменений, тогда как спрос демонстрирует большую вариабельность, частично выходящую за рамки линейных закономерностей.

Параллельно с ARIMA было проведено моделирование методом экспоненциального сглаживания, корни которого восходят к работам Хольта (1957) и Винтерса (1960) [14]. Экспоненциальное сглаживание строится на идее взвешенного усреднения исторических наблюдений с убывающими весами для более отдаленных точек, что позволяет модели быстрее адаптироваться к недавним изменениям в данных. Программная реализация в нашем случае выглядела следующим образом:

```
python
from statsmodels.tsa.holtwinters import ExponentialSmoothing

# Обучение модели с сезонностью
model_es = ExponentialSmoothing(train_data, seasonal_periods=7,
                                trend='add', seasonal='add')
fitted_es = model_es.fit()

# Генерация прогнозов
forecast_es = fitted_es.forecast(steps=len(test_data))
mae_es = mean_absolute_error(test_data, forecast_es)
rmse_es = np.sqrt(mean_squared_error(test_data, forecast_es))
```

Экспоненциальное сглаживание показало сопоставимые с ARIMA результаты для прогноза цен (MAE = 0,79 руб./км, RMSE = 1,08 руб./км), однако превзошло его по точности предсказания спроса (MAE = 11,1 заказа, RMSE = 15,3 заказа) (рис. 1). Улучшение точности объясняется повышенной чувствительностью метода к недавним колебаниям, что оказывается

полезным для данных со сложной структурой краткосрочных флуктуаций. Вместе с тем, оба классических метода разделяют общее ограничение: они предполагают линейный характер зависимостей между прошлыми и будущими значениями временного ряда, что может быть недостаточным для описания нелинейных эффектов, возникающих вследствие взаимодействия множественных рыночных факторов.

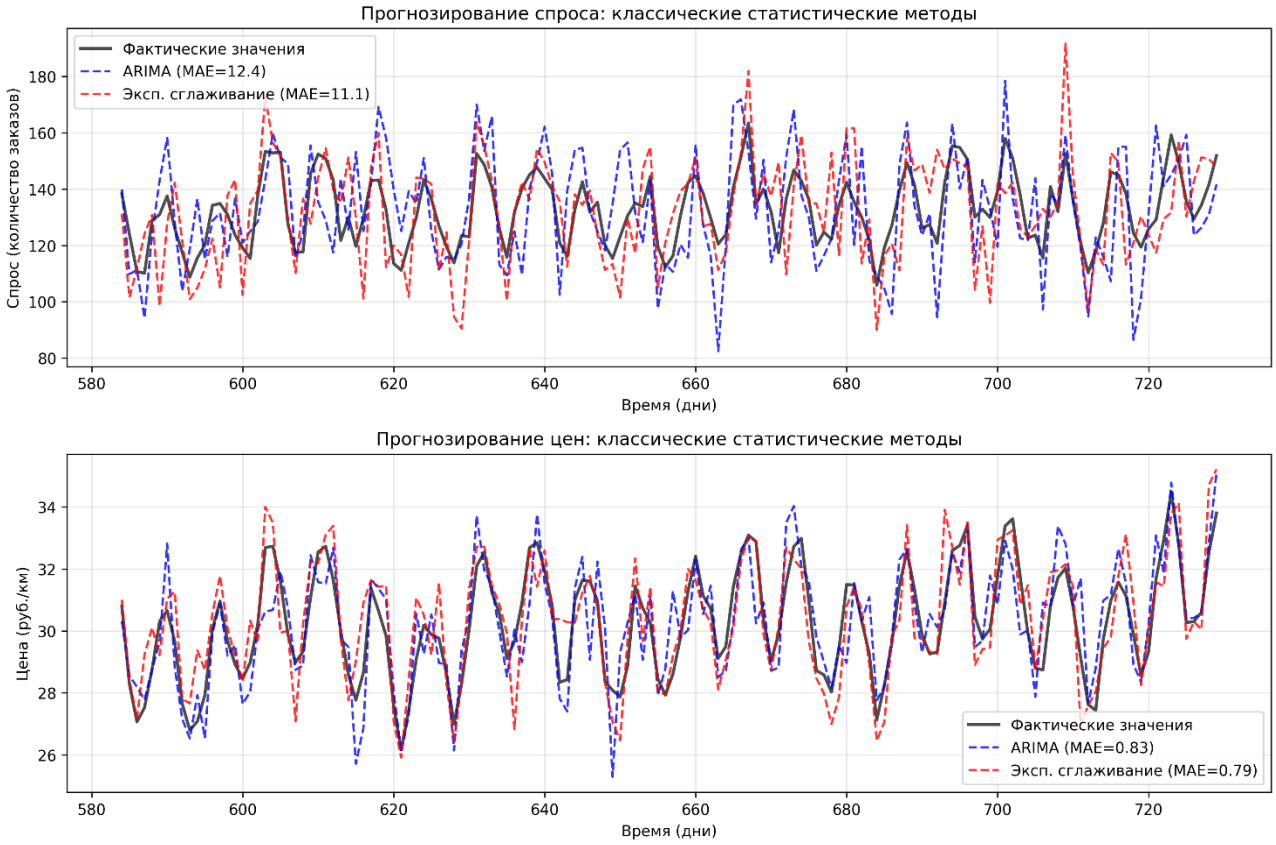

Рис. 1. – Прогнозирование спроса и цен классическими статистическими методами (ARIMA и экспоненциальное сглаживание)

Переход к алгоритмам машинного обучения открывает возможность моделирования нелинейных решений без априорных предположений о функциональной форме зависимости. Метод случайного леса (Random Forest), предложенный Л. Брейманом в 2001 году [15], представляет собой ансамбль решающих деревьев, каждое из которых обучается на случайной подвыборке данных с использованием случайного подмножества признаков.

Агрегирование предсказаний отдельных деревьев путем усреднения снижает дисперсию прогноза и повышает устойчивость к переобучению. Для применения случайного леса к задаче прогнозирования временных рядов необходимо преобразовать последовательность наблюдений в табличный формат с лаговыми переменными. Мы сконструировали признаковое пространство, включающее значения временного ряда на предыдущих 14 шагах, день недели, номер недели в году и порядковый номер наблюдения для учета долгосрочного тренда. Код реализации имел следующий вид:

```
python
from sklearn.ensemble import RandomForestRegressor
import pandas as pd

# Создание лаговых признаков
def create_features(data, lags=14):
    df = pd.DataFrame({'value': data})
    for i in range(1, lags+1):
        df[f'lag_{i}'] = df['value'].shift(i)
    df['day_of_week'] = range(len(data)) % 7
    df['week_of_year'] = (range(len(data)) // 7) % 52
    df['time_index'] = range(len(data))
    return df.dropna()

X_train = create_features(train_data).drop('value', axis=1)
y_train = create_features(train_data)['value']

# Обучение модели случайного леса
model_rf = RandomForestRegressor(n_estimators=100, max_depth=15,
                                 random_state=42)
model_rf.fit(X_train, y_train)

# Итеративное прогнозирование
```

```
forecast_rf = []
for step in range(len(test_data)):
    X_test = create_test_features(step)
    pred = model_rf.predict(X_test.reshape(1, -1))[0]
    forecast_rf.append(pred)
mae_rf = mean_absolute_error(test_data, forecast_rf)
rmse_rf = np.sqrt(mean_squared_error(test_data, forecast_rf))
```

Модель случайного леса продемонстрировала заметное улучшение качества прогнозов относительно классических методов (рис. 2). Для спроса MAE снизилась до 9,2 заказа (RMSE = 12,8 заказа), а для цен до 0,64 руб./км (RMSE = 0,89 руб./км). Способность алгоритма выявлять сложные нелинейные взаимосвязи между лаговыми значениями и будущими наблюдениями позволила точнее уловить специфику данных.

Дальнейшее повышение точности было достигнуто посредством применения градиентного бустинга в реализации XGBoost. В отличие от случайного леса, где деревья строятся независимо, XGBoost последовательно добавляет новые деревья, каждое из которых корректирует ошибки предыдущих моделей. Алгоритм минимизирует целевую функцию, включающую как ошибку предсказания, так и регуляризационные члены, штрафующие сложность модели. В нашем случае признаковое пространство было сформировано аналогично случайному лесу, однако гиперпараметры настраивались через кросс-валидацию для предотвращения переобучения, которое нередко является недостатком данного подхода:

```
python
import xgboost as xgb
# Подготовка данных в формате DMatrix
dtrain = xgb.DMatrix(X_train, label=y_train)
# Настройка параметров, обучение и генерация прогнозов
params = {
    'objective': 'reg:squarederror',
```

```
    'max_depth': 6,
    'learning_rate': 0.05,
    'subsample': 0.8,
    'colsample_bytree': 0.8,
    'reg_alpha': 0.1,
    'reg_lambda': 1.0
}
model_xgb = xgb.train(params, dtrain, num_boost_round=200)
dtest = xgb.DMatrix(X_test)
forecast_xgb = model_xgb.predict(dtest)
mae_xgb = mean_absolute_error(test_data, forecast_xgb)
rmse_xgb = np.sqrt(mean_squared_error(test_data, forecast_xgb))
```

XGBoost обеспечил лучшие результаты среди всех рассмотренных методов для обеих целевых переменных (рис. 2). MAE для спроса составила 8,1 заказа (RMSE = 11,2 заказа), а для цен — 0,58 руб./км (RMSE = 0,81 руб./км). Последовательная коррекция ошибок и регуляризация позволили модели эффективно балансировать между сложностью и обобщающей способностью без существенного переобучения на обучающей выборке.

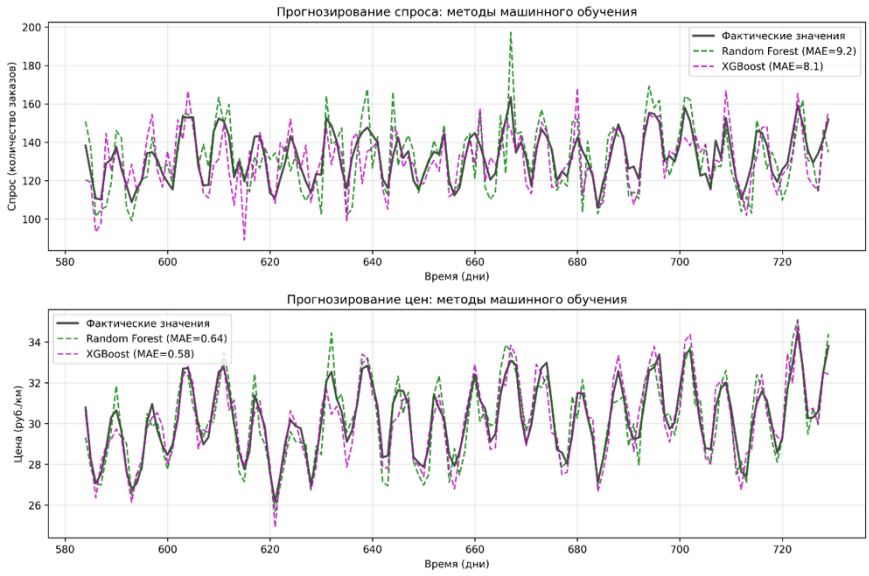

Рис. 2. – Прогнозирование спроса и цен методами машинного обучения (Random Forest и XGBoost)

Несмотря на высокие показатели точности градиентного бустинга, алгоритмы на основе деревьев решений обрабатывают каждое наблюдение независимо после извлечения признаков, не используя внутреннюю последовательную структуру временных рядов напрямую. Рекуррентные нейронные сети, специально разработанные для последовательных данных, способны моделировать долгосрочные зависимости через механизм внутренней памяти. Архитектура долгой краткосрочной памяти (LSTM) решает проблему затухающего градиента, характерную для обычных рекуррентных сетей, посредством введения т.н. специальных вентилей, регулирующих поток информации через ячейку памяти. Элементарная LSTM-ячейка включает вентили забывания f_t, входа i_t и выхода o_t, обновляющие состояние ячейки c_t и скрытое состояние h_t.

Для реализации LSTM была использована библиотека TensorFlow/Keras с двухслойной архитектурой:

```
python
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense, Dropout
from sklearn.preprocessing import MinMaxScaler

# Нормализация данных
scaler = MinMaxScaler()
train_scaled = scaler.fit_transform(train_data.reshape(-1, 1))

# Подготовка последовательностей
def create_sequences(data, seq_length=30):
    X, y = [], []
    for i in range(len(data) - seq_length):
        X.append(data[i:i+seq_length])
        y.append(data[i+seq_length])
    return np.array(X), np.array(y)
```

```
X_train, y_train = create_sequences(train_scaled, seq_length=30)

# Построение модели
model_lstm = Sequential([
    LSTM(64, return_sequences=True, input_shape=(30, 1)),
    Dropout(0.2),
    LSTM(32, return_sequences=False),
    Dropout(0.2),
    Dense(1)
])

model_lstm.compile(optimizer='adam', loss='mse')
model_lstm.fit(X_train, y_train, epochs=50, batch_size=32,
               validation_split=0.1, verbose=0)

# Прогнозирование
forecast_lstm_scaled = model_lstm.predict(X_test)
forecast_lstm = scaler.inverse_transform(forecast_lstm_scaled)
mae_lstm = mean_absolute_error(test_data, forecast_lstm)
rmse_lstm = np.sqrt(mean_squared_error(test_data, forecast_lstm))
```

LSTM продемонстрировала конкурентоспособные результаты (рис. 3): для спроса MAE = 8,6 заказа (RMSE = 11,9 заказа), для цен MAE = 0,61 руб./км (RMSE = 0,85 руб./км). Точность оказалась сопоставимой с XGBoost, однако незначительно уступила последнему. Подобный результат объясняется тем, что при относительно коротком горизонте прогнозирования (146 точек в тестовой выборке) и умеренной сложности временных зависимостей преимущество рекуррентной архитектуры в моделировании долгосрочной памяти проявляется не в полной мере. Вместе с тем, LSTM требует больших вычислительных ресурсов для обучения и более чувствительна к выбору

гиперпараметров, что накладывает дополнительные требования на процесс разработки модели.

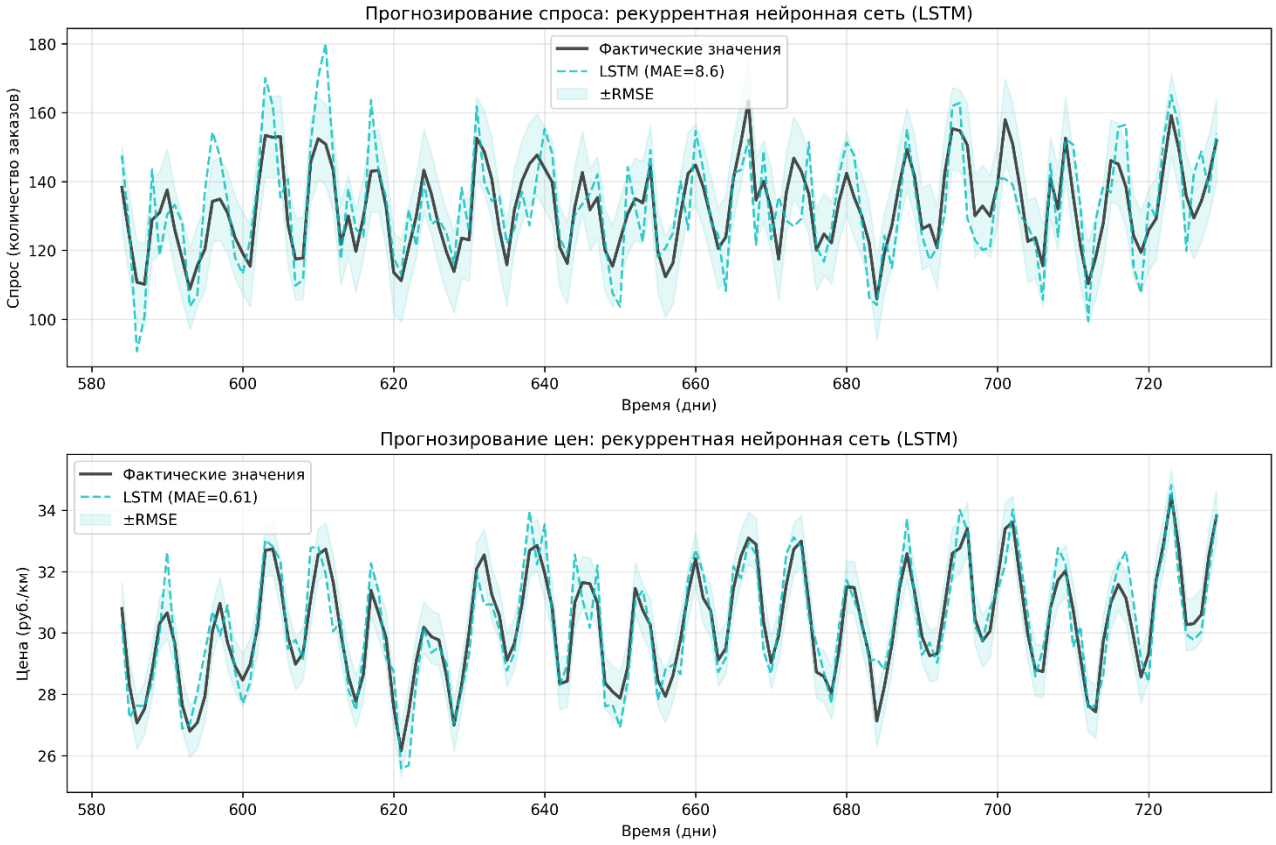

Рис. 3. – Прогнозирование спроса и цен рекуррентной нейронной сетью (LSTM)

Для систематического сравнения всех пяти методов была составлена итоговая таблица, включающая главные метрики качества прогнозирования для обеих целевых переменных (табл. 1). Выбор метрик обусловлен их распространенностью в литературе по временным рядам: MAE дает оценку средней величины ошибки в исходных единицах измерения и легко интерпретируется, тогда как RMSE сильнее штрафует крупные отклонения, что важно для оценки рисков существенных промахов в прогнозе.

Таблица 1

Сравнительные характеристики точности прогнозирования

Метод	**Спрос: MAE (заказы)**	**Спрос: RMSE (заказы)**	**Цена: MAE (руб./км)**	**Цена: RMSE (руб./км)**
ARIMA	12,4	16,7	0,83	1,12
Экспоненциальное сглаживание	11,1	15,3	0,79	1,08
Random Forest	9,2	12,8	0,64	0,89
XGBoost	8,1	11,2	0,58	0,81
LSTM	8,6	11,9	0,61	0,85

Исходя из таблицы, мы можем увидеть четкую закономерность: точность прогнозирования последовательно возрастает при переходе от классических статистических методов к алгоритмам машинного обучения и глубокого обучения (рис. 4). XGBoost обеспечил наилучшие показатели по обеим метрикам для обеих целевых переменных, опережая ближайшего конкурента (LSTM) на 6% по MAE для спроса и на 5% для цен. Разрыв между XGBoost и классическими методами еще более выражен: улучшение относительно ARIMA составило 35% для спроса и 30% для цен по метрике MAE. Подобные результаты подтверждают гипотезу о наличии нелинейных структур в данных по рынку грузоперевозок, которые эффективнее описываются гибкими алгоритмами машинного обучения.

Различия в точности между методами объясняются их архитектурными особенностями. ARIMA и экспоненциальное сглаживание опираются на линейные рекуррентные соотношения, что ограничивает их способность моделировать взаимодействия между различными лагами и нелинейные эффекты. Random Forest и XGBoost, напротив, могут конструировать произвольные нелинейные разделяющие поверхности в признаковом пространстве, автоматически выявляя релевантные комбинации признаков.

Превосходство XGBoost над Random Forest связано с последовательной коррекцией ошибок в процессе бустинга. LSTM же теоретически способна моделировать более сложные временные зависимости благодаря рекуррентной природе, однако на практике ее преимущество проявляется при наличии очень длинных последовательностей и сложных долгосрочных паттернов, что не в полной мере характерно для данной задачи.

Помимо точности, необходимо учитывать вычислительную сложность и интерпретируемость моделей. Классические методы обучаются за секунды и обеспечивают прозрачность параметров, что упрощает объяснение прогнозов заинтересованным сторонам. Random Forest и XGBoost требуют несколько минут для обучения на использованном объеме данных и позволяют анализировать важность признаков, сохраняя определенный уровень интерпретируемости. LSTM, напротив, обучается десятки минут и представляет собой «черный ящик» с тысячами параметров, что, в общем случае, затрудняет понимание логики принятия решений.

В отношении специфики рынка автомобильных грузоперевозок полученные результаты имеют прикладное значение. Высокая точность XGBoost позволяет транспортным компаниям более надежно планировать загрузку транспортных мощностей, снижая издержки от простоя и недозагрузки.

Заключение

Резюмируя результаты проведенного анализа, можно утверждать, что градиентный бустинг в реализации XGBoost демонстрирует оптимальное сочетание точности, вычислительной эффективности и частичной интерпретируемости для задачи прогнозирования спроса и цен на рынке автомобильных грузоперевозок. Метод обеспечивает существенное улучшение качества прогнозов относительно классических статистических подходов, сохраняя приемлемые требования к вычислительным ресурсам и

допуская анализ важности признаков. Выбор конкретного метода в практических приложениях должен учитывать баланс между точностью, скоростью обучения, интерпретируемостью и доступностью вычислительной инфраструктуры, исходя из специфических требований бизнес-задачи.

Литература

1. Тарифы на автоперевозки растут от квартала к кварталу // ATI. URL: news.ati.su/news/2025/04/24/tarify-na-avtoperevozki-rastut-ot-kvartala-k-kvartalu-200947.

2. Liachovičius, E., Šabanovič, E., & Skrickij, V. Freight rate and demand forecasting in road freight transportation using econometric and artificial intelligence methods. Transport. 2023. 38(4). pp. 231–242. URL: doi.org/10.3846/transport.2023.20932.

3. Budak, A., Ustundag, A., & Guloglu, B. A forecasting approach for truckload spot market pricing. Transportation Research Procedia. 2017. 22, pp. 329-338.

4. Jang, Hee-Seon & Chang, Tai-Woo & Kim, Seung-Han. Prediction of Shipping Cost on Freight Brokerage Platform Using Machine Learning. Sustainability. 2023. 15. URL: doi.org/10.3390/su15021122.

5. Su Z., Xu Q., Zhu X. Port Congestion and Container Freight Rate Dynamics: Forecasting with an RBF Neural Network. Frontiers in Marine Science. 2025. Vol. 12. Article 1545471.

6. Restiawati M., Kurniawan R., Suhartono D. Hybrid of Neural Prophet and Attention-Based LSTM for Container Freight Rate Forecasting. Proceedings of IEEE Conference on Intelligent Systems and Applications. 2023. pp. 1-6.

7. Shankar S., Punia S., Ilavarasan P.V. Forecasting Container Throughput with Long Short-Term Memory Networks. Industrial Management & Data Systems. 2020. Vol. 120, No. 3. pp. 425-441.

8. Xiao Y., Xue X., Hu Y., Yi M. Novel Decomposition and Ensemble Model with Attention Mechanism for Container Throughput Forecasting at Four Ports in Asia. Transportation Research Record. 2023. Vol. 2677, No. 7. pp. 1439-1456.

9. Вальков А.С., Кожанов Е.М., Медведникова М.М., Хусаинов Ф.И. Непараметрическое прогнозирование загруженности системы железнодорожных узлов по историческим данным. Машинное обучение и анализ данных. 2012. Т. 1, № 4. С. 448-465.

10. Вальков А.С., Кожанов Е.М., Мотренко А.П., Хусаинов Ф.И. Построение кросс-корреляционных зависимостей при прогнозе загруженности железнодорожного узла. Машинное обучение и анализ данных. 2013. № 5. С. 503-517.

11. Champahom, Thanapong & Banyong, Chinnakrit & Janhuaton, Thananya & Se, Chamroeun & Watcharamaisakul, Fareeda & Ratanavaraha, Vatanavongs & Jomnonkwao, Sajjakaj. Deep Learning vs. Gradient Boosting: Optimizing Transport Energy Forecasts in Thailand Through LSTM and XGBoost. Energies. 2025. 18. 1685. URL: doi.org/10.3390/en18071685.

12. Yang, Cheng-Hong & Chang, Po-Yin. Forecasting the Demand for Container Throughput Using a Mixed-Precision Neural Architecture Based on CNN–LSTM. Mathematics. 2020. 8. 1784. URL: doi.org/10.3390/math8101784.

13. Box G.E.P., Jenkins G.M. Time Series Analysis: Forecasting and Control. San Francisco: Holden-Day. 1970. P. 575.

14. Winters P.R. Forecasting Sales by Exponentially Weighted Moving Averages. Management Science. 1960. Vol. 6, No. 3. pp. 324-342.

15. Breiman L. Random Forests. Machine Learning. 2001. Vol. 45, No. 1. pp. 5-32.

References

1. Tarifi na avtoperevozki rastut ot kvartala k kvartalu [Freight transportation rates are increasing quarter by quarter]. ATI. URL: news.ati.su/news/2025/04/24/tarify-na-avtoperevozki-rastut-ot-kvartala-k-kvartalu-200947.

2. Liachovičius, E., Šabanovič, E., Skrickij, V. Transport. 2023. 38(4). pp. 231–242. URL: doi.org/10.3846/transport.2023.20932.

3. Budak, A., Ustundag, A., Guloglu, B. Transportation Research Procedia. 2017. 22, pp 329-338.

4. Jang, Hee-Seon & Chang, Tai-Woo & Kim, Seung-Han. Sustainability. 2023. 15. URL: doi.org/10.3390/su15021122.

5. Su Z., Xu Q., Zhu X. Frontiers in Marine Science. 2025. Vol. 12. Article 1545471.

6. Restiawati M., Kurniawan R., Suhartono D. Proceedings of IEEE Conference on Intelligent Systems and Applications. 2023. pp. 1-6.

7. Shankar S., Punia S., Ilavarasan P.V. Industrial Management & Data Systems. 2020. Vol. 120, No. 3. pp. 425-441.

8. Xiao Y., Xue X., Hu Y., Yi M. Transportation Research Record. 2023. Vol. 2677, No. 7. pp. 1439-1456.

9. Valkov A.S., Kozhanov Ye.M., Medvednikova M.M., Khusainov F.I. Mashinnoe obuchenie i analiz dannikh. 2012. T. 1, № 4. pp. 448-465.

10. Valkov A.S., Kozhanov Ye.M., Motrenko A.P., Khusainov F.I. Mashinnoe obuchenie i analiz dannikh. 2013. № 5. pp. 503-517.

11. Champahom, Thanapong & Banyong, Chinnakrit & Janhuaton, Thananya & Se, Chamroeun & Watcharamaisakul, Fareeda & Ratanavaraha, Vatanavongs & Jomnonkwao, Sajjakaj. Energies. 2025. 18. 1685. URL: doi.org/10.3390/en18071685.

12. Yang, Cheng-Hong & Chang, Po-Yin. Mathematics. 2020. 8. 1784. URL: doi.org/10.3390/math8101784.

13. Box G.E.P., Jenkins G.M. San Francisco: Holden-Day. 1970. P. 575.
14. Winters P.R. Management Science. 1960. Vol. 6, No. 3. pp. 324-342.
15. Breiman L. Machine Learning. 2001. Vol. 45, No. 1. pp. 5-32.

Авторы согласны на обработку и хранение персональных данных.

Дата поступления: 11.10.2025

Дата публикации: 27.11.2025