

## ОПТИМИЗАЦИЯ РЕКЛАМНЫХ СТРАТЕГИЙ С ИСПОЛЬЗОВАНИЕМ РЕКУРРЕНТНОЙ НЕЙРОННОЙ СЕТИ С ДОЛГОЙ КРАТКОСРОЧНОЙ ПАМЯТЬЮ

**Е.С. Пискун, Д. В. Нуансенгси**

Белорусский государственный университет информатики  
и радиоэлектроники, Минск, Беларусь, [diana.nuansengsi@mail.ru](mailto:diana.nuansengsi@mail.ru)

*Modern e-commerce advertising requires the use of advanced data analysis technologies to improve the effectiveness of user interaction. One of the effective approaches to managing advertising companies is the use of neural networks such as Long Short-Term Memory (LSTM). In this paper, we consider the process of numerical modeling of advertising strategies using LSTM, which allows us to take into account time dependencies in the data and increase the accuracy of predictions of key campaign performance indicators such as CTR and ROI.*

### **Введение**

Рынок рекламы в электронной коммерции стремительно меняется, что требует от маркетологов оперативного принятия решений на основе анализа имеющихся данных. Традиционные методы анализа не всегда учитывают временные зависимости между различными параметрами рекламных кампаний, такими как бюджет, аудитория, время показа и поведение пользователей. В связи с этим, перспективным подходом для предсказания результатов рекламных кампаний является использование рекуррентных нейронных сетей, в частности, нейронной сети с долгой краткосрочной памятью (англ. *Long Short-Term Memory, LSTM*) [1]. Она обладает возможностью хранить информацию о предыдущих шагах, что делает её особенно полезной для работы с длинными временными рядами данных, характерными для рекламных кампаний.

### **Описание процесса и математическая модель**

Процесс оптимизации рекламных стратегий начинается с анализа данных о предыдущих действиях, выполненных пользователем. Эти данные включают такие параметры, как бюджет, время показа, показатели эффективности, включая показатель кликабельности (англ. *Click-Through Rate, CTR*) и метрику стоимости одного перехода по ссылке (англ. *Cost per Click, CPC*), а также демографические характеристики аудитории. На основе этих данных обучается модель *LSTM*, которая затем прогнозирует результаты будущих рекламных кампаний. Также её важной особенностью является способность сохранять и обновлять информацию о предыдущих шагах в процессе анализа, что позволяет учитывать изменения поведения пользователей со временем [2].

Модель *LSTM* основана на трёх ключевых элементах: забывающий слой, входной слой и выходной слой [3 – 6]. Эти элементы управляют потоком

информации в сети и обеспечивают ее точное предсказание на каждом временном шаге. Рассмотрим данные слои более подробно.

1. Забывающий слой (англ. *forget layer*) необходим для определения информации на каждом шаге, которую целесообразно не учитывать (забыть). Выходной сигнал в модели забывания вычисляется по формуле 1 [7].

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f), \quad (1)$$

где  $f_t$  – коэффициент забывания (значение от 0 до 1);  $h_{t-1}$  – скрытое состояние предыдущего шага,  $x_t$  – текущие входные данные,  $W_f$  – весовые коэффициенты слоя забывания,  $b_f$  – смещение,  $\sigma$  – сигмоида, которая преобразует результат в диапазон [0, 1].

Схема забывающего слоя в модели *LSTM* представлена на рисунке 1.

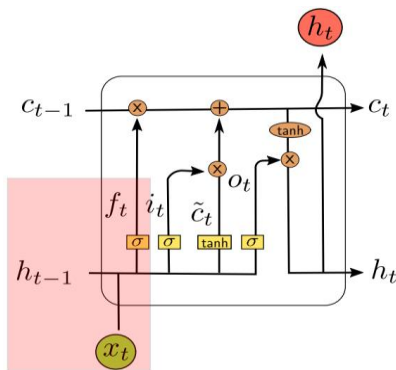


Рис 1. Схема забывающего слоя в модели *LSTM* [7]

2. Входной слой (англ. *input layer*) необходим для определения объема данных к добавлению в память.

Выходной сигнал входного элемента вычисляется по формуле 2 [7].

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i), \quad (2)$$

где  $i_t$  – коэффициент входа, определяющий количество новой информации, которое будет добавлено,  $\sigma$  – сигмоида, которая преобразует результат в диапазон [0, 1],  $W_i$  – весовой коэффициент входного слоя,  $h_{t-1}$  – скрытое состояние предыдущего шага,  $x_t$  – текущие входные данные,  $b_i$  – смещение.

Новое содержание памяти вычисляется по формуле 3 [7].

$$C_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c), \quad (3)$$

где  $C_t$  – новое содержимое памяти,  $h_{t-1}$  – скрытое состояние предыдущего шага,  $x_t$  – текущие входные данные,  $W_c$  – весовой коэффициент,  $b_c$  – смещение,  $\tanh$  – гиперболический тангенс, который масштабирует значение до диапазона  $[-1, 1]$ .

Схема входного слоя в модели *LSTM* имеет вид, приведенный на рисунке 2.

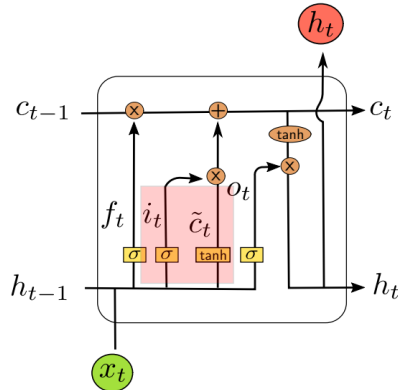


Рис 2. Схема входного слоя в модели *LSTM* [7]

3. Выходной слой (англ. *output layer*) необходим для контроля данных, передаваемых на следующий шаг.

Работа выходного слоя вычисляется по формуле 4 [7].

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o), \quad (4)$$

где  $o_t$  – коэффициент выхода (определяет, какую часть состояния нужно передать на следующий шаг),  $\sigma$  – сигмоида, которая преобразует результат в диапазон  $[0, 1]$ ,  $W_o$  – матрица весов выходного слоя,  $h_{t-1}$  – скрытое состояние предыдущего шага,  $x_t$  – входные данные,  $b_o$  – смещение.

Обновление скрытого состояния  $h_t$  на текущем временном шаге вычисляется по формуле 5 [7].

$$h_t = o_t \times \tanh(C_t), \quad (5)$$

где  $o_t$  – коэффициент выхода,  $\tanh$  – гиперболический тангенс от текущего состояния памяти  $C_t$ , который масштабирует значение до диапазона  $[-1, 1]$ .

Схема выходного слоя в модели *LSTM* представлена рисунке 3.

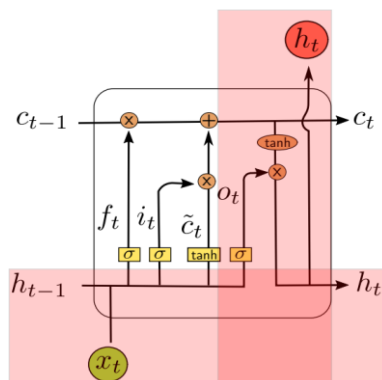


Рис 3. Схема выходного слоя в модели LSTM [7]

Модель обучается минимизировать ошибку прогнозов с помощью функции потерь, например, среднеквадратичной ошибки (англ. *mean square error*, *MSE*) для задач регрессии.

Среднеквадратичная ошибка для задач регрессии вычисляется по формуле 6 [7].

$$MSE(y_i, \hat{y}_i) = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2, \quad (6)$$

где  $y_i$  – реальное значение метрики,  $\hat{y}_i$  – предсказанное значение,  $N$  – количество примеров в обучающей выборке [3].

Вышеприведённые формулы 1 – 6 описывают способ поступления информации в рассматриваемую модель нейронной сети, ее обработку и дальнейшую передачу. Обучение модели направлено на минимизацию ошибки между реальными и предсказанными значениями ключевых метрик. Для этого используется метод обратного распространения ошибки с оптимизацией весов модели [4].

Стоит также отметить, что одним из важнейших факторов успеха модели LSTM является её способность учитывать долгосрочные зависимости, что особенно важно для рекламных кампаний с длительными временными интервалами. Эффективность модели подтверждена использованием таких стандартных метрик, как среднеквадратичная ошибка для регрессионных задач и кросс-энтропия для классификационных задач, что позволило достичь точности предсказаний, значительно превышающей традиционные методы [5].

В настоящее время, LSTM активно совершенствуется для повышения точности и её адаптации к различным сценариям рекламных кампаний, в частности:

1 Использование *механизма внимания* позволяет ей фокусироваться на ключевых временных шагах, что может повысить точность предсказаний в случае, когда определенные временные моменты имеют особую важность для конечного результата [3];

2 *Многослойность архитектуры LSTM* позволяет сети лучше распознавать сложные зависимости в данных, что позволит обеспечить более глубокую обработку информации и приведет к более точным предсказаниям [4];

3 *Включение дополнительных данных*, таких как время года, праздничные дни и макроэкономические индикаторы, может значительно улучшить прогнозирование модели, в частности, поведение пользователей в преддверии распродаж или праздничных дней, что важно учитывать при планировании рекламных кампаний [5 – 6].

### **Заключение**

В работе показано, что внедрение модели *LSTM* в процесс оптимизации рекламных стратегий позволяет существенно улучшить точность прогнозирования ключевых метрик. Модель успешно учитывает временные зависимости, что делает её эффективным инструментом для автоматизации управления рекламными кампаниями.

Внедрение таких усовершенствований, как механизм внимания, многослойные архитектуры и учет внешних факторов, может дать возможность для создания более точных и адаптивных систем управления рекламой в электронной коммерции.

### **Список использованных источников**

1. Hochreiter S., Schmidhuber J. Long Short-Term Memory // *Neural Computation*. 1997. Т. 9, № 8. С. 1735–1780.
2. Olah C. Understanding LSTM Networks [Электронный ресурс] // Colah's Blog. 2015. URL: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/> (дата обращения: 16.10.2024).
3. Bengio Y., Courville A., Vincent P. Representation Learning: A Review and New Perspectives // *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2013. Т. 35, № 8. С. 1798–1828.
4. Vaswani A., Shazeer N., Parmar N., et al. Attention is All You Need // *Advances in Neural Information Processing Systems*. 2017. Т. 30. С. 5998–6008.
5. Sutskever I., Vinyals O., Le Q. V. Sequence to Sequence Learning with Neural Networks // *Advances in Neural Information Processing Systems*. 2014. Т. 27. С. 3104–3112.
6. Brownlee J. Deep Learning for Time Series Forecasting [Текст] // *Machine Learning Mastery*. 2018.
7. Gao S., Li H. An Application of Innovative Algorithm of Integrated Social Network Analysis with Statistical LSTM Chain Network Analysis (SLSTM-CNA) for Entrepreneurial Team Member Selection // *Journal of Electrical Systems*. – 2024. – Vol. 20, no. 3s. – pp. 1592–1602. DOI:10.52783/jes.1699.