

Для цитирования: Бурдо Г.Б., Болотов А.Н. Оценка решений при подготовке производства // Вестник Тверского государственного технического университета. Серия «Технические науки». 2023. № 2 (18). С. 94–101.

EVALUATION OF SOLUTIONS DURING PRE-PRODUCTION

G.B. BURDO, Dr. Sc., A.N. BOLOTOV, Dr. Sc.

Tver State Technical University
22, Af. Nikitin emb., Tver, 170026, Russian Federation; e-mail: gbtms@yandex.ru

It is shown that the specifics of single and small-scale instrument-making and machine-building production presupposes the implementation of a significant number of contracts at the same time, which implies simplified, and not always adequate, technical preparation of production. The basic principles of organizational and technological design are revealed, the set-theoretic and time models of the production system are analyzed, which allows identifying criteria during technological preparation. The subjects and objects of the production system and time points for determining the specified criteria are established. It is shown that the set-theoretic model of the procedure for establishing criteria for technological preparation of production reflects the information structure of the production system, and the time structure ensures synchronization of the procedures performed in time.

Keywords: computer-aided design of technological processes, technological preparation of production, artificial intelligence, objective function, unified information space, system approach.

Поступила в редакцию/received: 30.10.2022; после рецензирования/revised: 12.11.2022;
принята/accepted: 24.11.2022

УДК 004.032.26

АРХИТЕКТУРА ГЛУБОКОЙ НЕЙРОННОЙ СЕТИ В ЗАДАЧЕ ПРОГНОЗИРОВАНИЯ КООРДИНАТ ДВИЖУЩЕЙСЯ ЦЕЛИ*

В.К. КЕМАЙКИН, канд. техн. наук, А.С. ПОЛЫГАЕВ, студ.

Тверской государственный технический университет
170026, Тверь, наб. Аф. Никитина, 22; e-mail: vk-kem@mail.ru

© Кемайкин В.К., Польшаев А.С., 2023

Рассмотрено использование глубокой рекуррентной нейронной сети в задаче прогнозирования координат движущейся цели в условиях неточности измерений. Обучение нейронной сети проводится на предварительной выборке координат ограниченной длины, моделируются ошибки измерения координат. Характер движения

* Материалы были представлены на научном семинаре «Золотовские чтения», посвященном 100-летию со дня рождения выдающегося российского математика, академика АН СССР Золотова Евгения Васильевича (6–7 октября 2022 г., Тверь, Тверской государственный технический университет).

цели при тестировании сети отличается от модели, используемой при обучении нейронной сети и положенной в основу фильтра Калмана.

Ключевые слова: искусственная нейронная сеть, нейронная сеть с долгой кратковременной памятью, фильтр Калмана, прогнозирование временных рядов, фильтрация измерений, экстраполяция.

DOI: 10.46573/2658-5030-2023-2-101-112

ВВЕДЕНИЕ

Задача прогнозирования координат движущейся цели в условиях неточности измерений координат и недостатка априорной информации о характере движения цели крайне актуальна в системах траекторной обработки радиолокационной информации.

Применительно к задаче траекторной обработки прогнозирование координат включает в себя последовательное решение двух задач: фильтрации измерений и прогнозирования. Задача фильтрации измерений сводится к прогнозированию временных рядов, которое традиционно можно осуществить применением фильтров калмановского типа [1, 2]. Как известно, фильтр Калмана успешно используется в случаях, когда модель системы линейна, а шум измерений некоррелирован и имеет плотность вероятности, соответствующую плотности вероятности нормального (гауссовского) распределения [3]. Чтобы обойти данные ограничения, применяют такие модификации, как расширенный фильтр Калмана (Extended Kalman Filter) [4] и ансцентный фильтр Калмана (Unscent Kalman Filter) [5]. Однако, как и классический фильтр Калмана, данные модификации требуют априорных знаний в виде модели процесса, отклонение от которой ведет к некорректным результатам [3].

Задача прогнозирования может быть решена двумя способами: экстраполяцией отфильтрованных координат цели и получением априорной оценки фильтра на следующий момент времени.

В качестве альтернативы традиционному подходу к фильтрации и экстраполяции координат движущейся цели в данной статье предлагается использовать многослойную искусственную нейронную сеть с долговременной краткосрочной памятью (Long Short-Term Memory), обученную на референсной выборке неточных измерений и истинных положений цели. Такой подход не требует построения точной модели наблюдаемого процесса и позволяет обеспечить адаптивность подсистемы фильтрации и экстраполяции к изменениям характера движения цели в пределах заданной модели движения.

Целью данной статьи является иллюстрация использования глубокой рекуррентной нейронной сети в задаче прогнозирования координат движущейся цели в условиях неточности измерений.

ФИЛЬТР КАЛМАНА

Алгоритм работы фильтра Калмана заключается в последовательном выполнении двух основных шагов: прогнозирования измерений на основе заложенной модели процесса и обновления состояния фильтра по результатам измерений [6].

Основным достоинством фильтра Калмана применительно к задаче траекторной обработки является то, что он не требует накопления измерений за несколько шагов и дает оценку согласно модели процесса на каждом шаге измерения.

Для решаемой задачи также важно, что фильтр позволяет дать априорную оценку состояния процесса (положения цели) на следующий за текущим измерением момент времени.

Состояние процесса в момент времени n обозначается $x(n)$. Изменение состояния системы описывается уравнением

$$x(n) = F(n) \cdot x(n-1) + B(n) \cdot u(n) + W(n), \quad (1)$$

где $F(n)$ – матрица перехода состояний; B – матрица управления; $u(n)$ – вектор управляющих воздействий; $W(n)$ – шум процесса.

Измерения положения цели осуществляются регулярно и связаны с истинными значениями следующим образом:

$$z(n) = H(n) \cdot x(n) + V(n), \quad (2)$$

где $H(n)$ – модель измерений; $V(n)$ – белый шум измерений с нулевым средним значением.

Оценка состояния системы в момент времени $(n+1)$

$$\hat{x}(n+1|n) = F(n) \cdot \hat{x}(n|n), \quad (3)$$

где $\hat{x}(n|n)$ – предыдущая оценка состояния.

Оценка измерения на момент времени $(n+1)$ получается из оценки состояния системы с учетом модели измерений:

$$\tilde{z}(n+1|n) = H(n+1) \cdot \hat{x}(n+1|n). \quad (4)$$

После получения очередного измерения оно сравнивается с ранее полученной оценкой измерения:

$$\tilde{z}(n+1) = z(n+1) - \tilde{z}(n+1|n). \quad (5)$$

Далее оценка состояния системы уточняется на основе полученного измерения и коэффициента усиления фильтра:

$$\hat{x}(n+1|n+1) = \hat{x}(n+1|n) + K(n+1) \cdot \tilde{z}(n+1), \quad (6)$$

где $K(n+1)$ – коэффициент усиления фильтра Калмана.

Усиление фильтра Калмана требует вычислений ковариационной матрицы состояния $P(n+1|n)$ и ковариационной матрицы ошибок $S(n+1)$. Ковариационная матрица состояния вычисляется следующим образом:

$$P(n+1|n) = F(n) \cdot P(n|n) \cdot F(n)' + Q(n), \quad (7)$$

где $Q(n)$ – матрица ковариации белого шума процесса.

Ковариационная матрица ошибок

$$S(n+1) = H(n+1) \cdot P(n+1|n) \cdot H(n+1)' + R(n+1), \quad (8)$$

где $R(k+1)$ – ковариация белого шума измерений.

Коэффициент усиления фильтра Калмана

$$K(n+1) = P(n+1|k) \cdot H(n+1)' \cdot S(n+1)^{-1}. \quad (9)$$

Затем ковариационная матрица состояния обновляется с использованием рассчитанного коэффициента усиления фильтра Калмана:

$$P(n+1|n+1) = P(n+1|n) - K(n+1) \cdot S(n+1) \cdot W(n+1)'. \quad (10)$$

Вычисление выражений (3)–(10) выполняется в цикле на каждом шаге измерений состояния наблюдаемой системы (рис. 1).

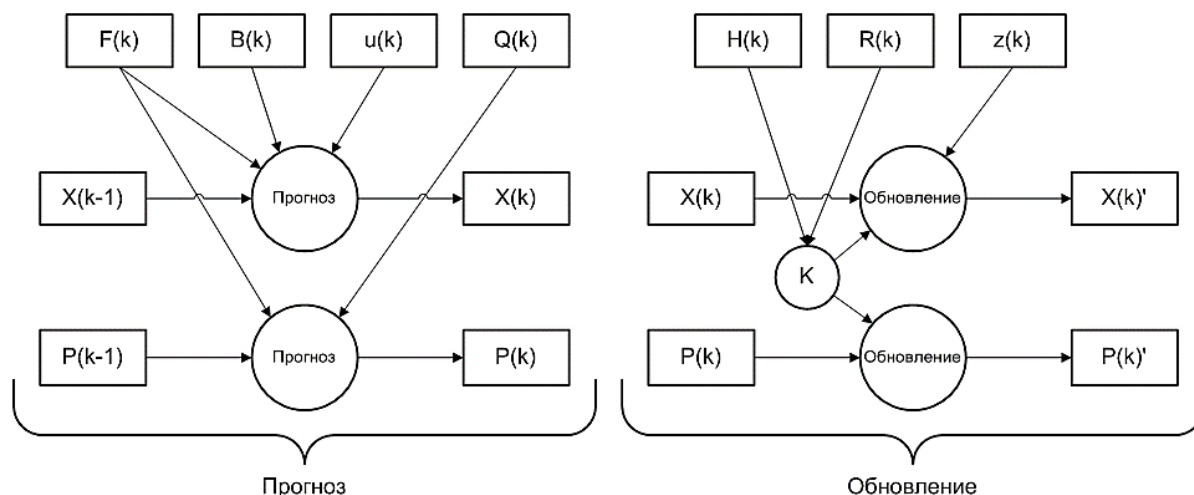


Рис. 1. Принцип работы фильтра Калмана

НЕЙРОННАЯ СЕТЬ LSTM

Сети с долгой краткосрочной памятью (LSTM-сети, от англ. Long Short-Term Memory) – это модификация классической рекуррентной нейронной сети, способная находить долгосрочные и краткосрочные зависимости. Такие сети хорошо приспособлены к обучению на задачах прогнозирования временных рядов в тех случаях, когда измерения достаточно сильно отделены друг от друга по времени [7].

LSTM-сети, как и рекуррентные нейронные сети, имеют цепную структуру, но устройство клетки сети более сложное: она состоит из четырех нейронов, соединенных специальным образом (рис. 2). В источниках [8, 9] можно увидеть отличия нейронной LSTM-сети от классической рекуррентной нейронной сети.

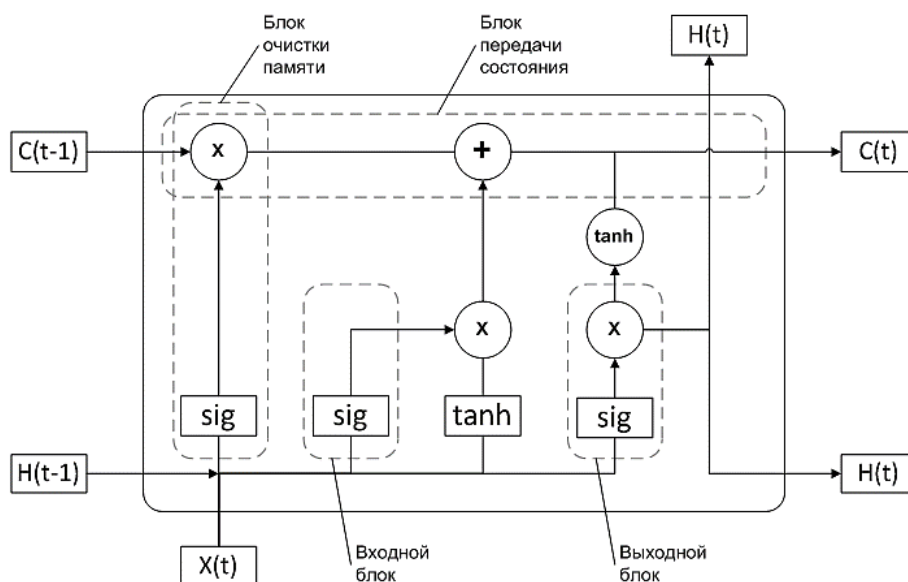


Рис. 2. Структура клетки LSTM-сети
(блоками обозначены нейроны клетки, кругами – операции)

LSTM-сети способны как отслеживать долгосрочные зависимости в данных, так и удалять (забывать) информацию. Этот процесс регулируется структурами, называемыми блоками или вентилями (от англ. gates).

Как указано на рис. 2, у каждой клетки имеются две рекуррентные компоненты: выходной вектор H_t и вектор состояния C_t . На вход сети в момент времени t подается вектор измерений X_t , а также выходной вектор H_{t-1} и вектор состояния C_{t-1} предыдущей клетки.

Состояние входного блока

$$I_t = \text{sig}(x_t W_i + H_{t-1} W_i + b_i), \quad (11)$$

где x_t – вектор измерений; W_i – вес нейрона входного блока; H_{t-1} – выходной вектор предыдущей клетки; b_i – смещение (от англ. bias) функции активации.

Далее работает блок очистки памяти:

$$F_t = \text{sig}(x_t W_f + H_{t-1} W_f + b_f), \quad (12)$$

где W_f – вес нейрона блока очистки памяти; b_f – смещение функции активации.

Изменение памяти для блока передачи состояния

$$\tilde{C}_t = \text{tanh}(x_t W_c + t - 1 W_c + b_c), \quad (13)$$

где W_c – вес нейрона блока передачи состояния; b_c – смещение функции активации.

Затем рассчитывается выходной вектор состояния клетки:

$$C_t = F_t C_{t-1} + I_t \tilde{C}_t, \quad (14)$$

где F_t – состояние блока очистки памяти; C_{t-1} – вектор состояния предыдущей клетки; I_t – состояние входного блока; \tilde{C}_t – оценки изменения памяти.

Состояние выходного блока рассчитывается по формуле, аналогичной формуле для входного блока:

$$O_t = \text{sig}(x_t W_o + H_{t-1} W_o + b_o), \quad (15)$$

где W_o – вес нейрона выходного блока; b_o – смещение функции активации.

Затем применяется тангенциальный слой:

$$H_t = O_t \cdot \text{tanh}(C_t), \quad (16)$$

где O_t – состояние выходного блока клетки; C_t – выходной вектор состояния клетки.

Для обучения нейронных сетей LSTM и GRU в данной работе применяется алгоритм ADAM (от англ. Adaptive Moment Estimation) – «метод адаптивной оценки моментов» [10, 11]. В данном методе используются скользящие средние значения, рассчитанные как для градиентов весов, так и для вторых моментов градиентов. Суть алгоритма заключается в делении скорости обучения для каждого веса в отдельности на скользящие средние значения градиентов веса.

Значения скользящих среднего первого и второго моментов градиента при обучении рассчитываются следующим образом (m_t (от англ. mean) – значение скользящего среднего для первого момента градиента; v_t (от англ. variance) – для второго момента градиента):

$$\begin{aligned} m_t &= \beta_1 m_{t-1} + (1 - \beta_1) g_t; \\ v_t &= \beta_2 v_{t-1} + (1 - \beta_2) g_t^2, \end{aligned} \quad (17)$$

где t – номер итерации; g_t – значение первого момента градиента; g_t^2 – значение второго момента градиента; β_1, β_2 – коэффициенты забывания для первого и второго моментов градиента соответственно.

Первый и второй моменты градиента являются частью гиперпараметра алгоритма обучения нейронной сети. По умолчанию в начале обучения они имеют значения 0,9 и 0,999 соответственно.

Оценки скользящего среднего для первого и второго моментов градиента на очередном шаге обучения рассчитываются по формулам:

$$\begin{aligned}\hat{m}_t &= \frac{m_t}{1-\beta_2^t}, \\ \hat{v}_t &= \frac{v_t}{1-\beta_2^t},\end{aligned}\tag{18}$$

где \hat{m}_t – оценка скользящего среднего для первого момента градиента; \hat{v}_t – оценка скользящего среднего для второго момента градиента.

Затем полученные оценки скользящего среднего для первого и второго моментов градиента используются для обновления значений веса нейрона, рассчитанного на основе предыдущего значения:

$$W_{t+1} = W_t - \frac{\eta}{\sqrt{\hat{v}_{t+\epsilon}}} \hat{m}_t,\tag{19}$$

где η – размер шага итерации; ϵ – относительно малое значение, используемое, чтобы избежать деления на ноль.

По умолчанию для значения ϵ используется значение 10^{-8} .

АРХИТЕКТУРА ГЛУБОКОЙ НЕЙРОННОЙ СЕТИ

Предлагаемая архитектура нейронной сети включает в себя два сложносоставных слоя, каждый из которых состоит из рабочего слоя LSTM с заданным количеством нейронов, слоя активации и слоя исключения (от англ. dropout) (рис. 3, табл. 1).

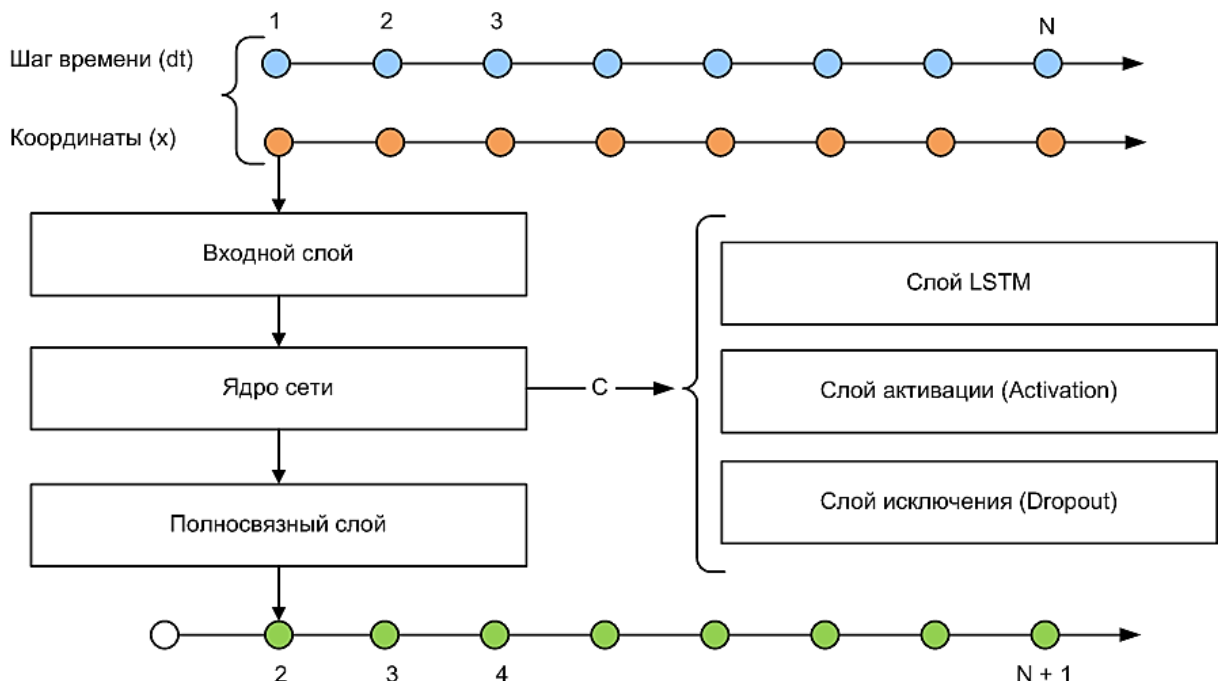


Рис. 3. Функциональная схема экстраполирующего фильтра на основе глубокой нейронной сети LSTM

Таблица 1. Архитектура нейронной сети

Номер слоя	Наименование слоя	Параметры слоя
1	Входной слой	Количество входов: 2
2	LSTM-слой	Функция активации состояния: Tanh Функция активации вентилей: Sigmoid
3	Слой активации	Тип функции активации: ReLu
4	Слой исключения	Вероятность исключения: 0,2
5	Полносвязный слой	Количество выходов: 2
6	LSTM-слой	Функция активации состояния: Tanh Функция активации вентилей: Sigmoid
7	Слой активации	Тип функции активации: ReLu
8	Слой исключения	Вероятность исключения: 0,2
9	Полносвязный слой	Количество выходов: 1

Ожидается, что правильно подобранная функция активации и значение вероятности исключения данных для слоя исключения помогут избежать переобучения сети и повысить адаптивность сети к отклонениям от обучающей выборки и пропускам данных.

Используя свойство нейронной сети LSTM запоминать более ранние измерения и состояния нейронов и учитывая, что размер предварительной выборки существенно не влияет на точность предсказания такой нейронной сетью, зададим размер предварительной выборки равным 1. При этом размер выборки, используемой для обучения (англ. Mini-Batch Sequence Length), будет установлен по результатам оптимизации гиперпараметра.

ОПИСАНИЕ ЭКСПЕРИМЕНТА

В качестве модели движения цели использовался процесс изменения во времени некоторой величины, заданный уравнением

$$x = A \cdot \sin(w \cdot t + phi), \quad (20)$$

где $A = 5$; $w = 3 \cdot \pi$; $phi = 0$; время моделирования фиксировано в диапазоне $t \in [0, 10]$; шаг времени равен 0,01.

Данная модель использовалась для обучения нейронной сети, а также в качестве модели, лежащей в основе используемого для сравнения результатов работы нейронной сети фильтра Калмана.

Для экспериментальной отработки сформировано несколько наборов измерений (обучающих наборов) на основе выражения (20) с наложением на полученные координаты цели белого гауссового шума с соотношением «сигнал – шум» (Signal – Noise Rate (SNR)), равным 10.

Чтобы имитировать несоответствие модели процесса и модели, заложенной в фильтр Калмана, дополнительно был сформирован тестовый набор измерений с

соотношением «сигнал – шум», равным 10, на основе модифицированной модели измерений, для которой $A = f(t)$, $w = 3 \cdot \pi$, $\phi = 0$.

Обучающий и тестовый наборы данных представлены на рис. 4.

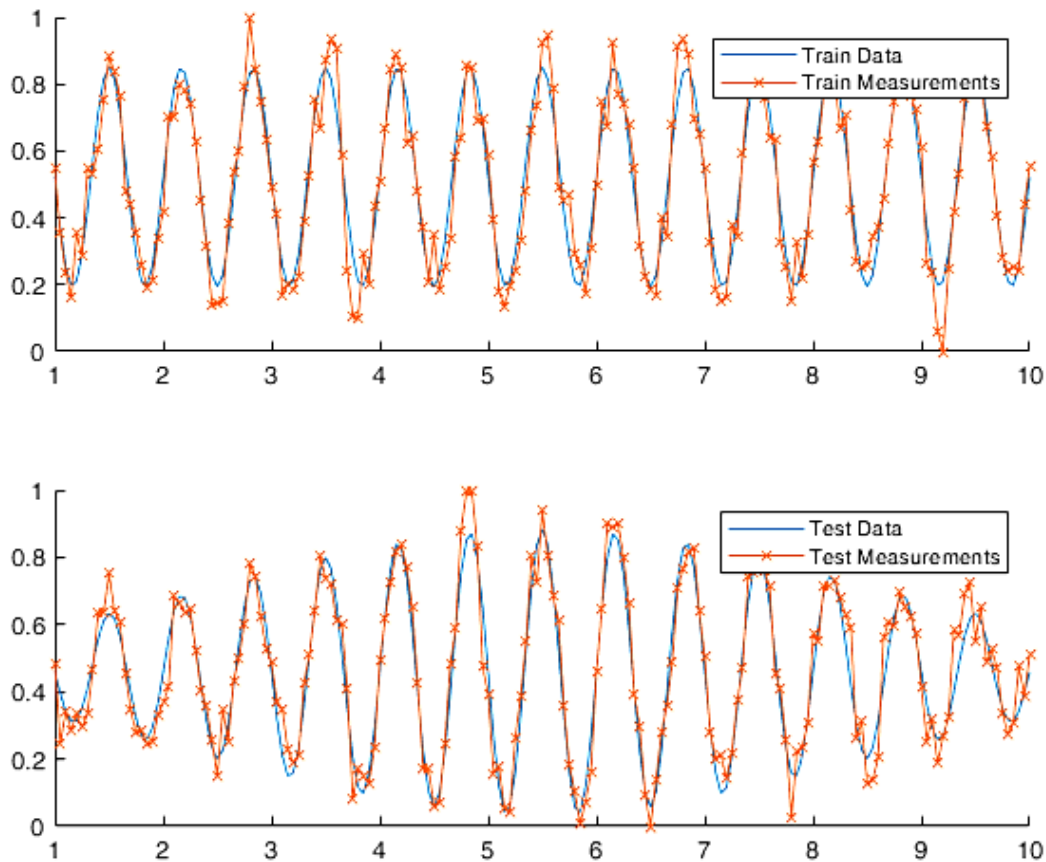


Рис. 4. Обучающий и тестовый наборы данных, SNR = 10

Размер обучающей выборки для рассматриваемой сети составляет 2/3 от общей длины набора данных. Данные измерений, используемые для обучения и валидации нейронной сети, нормализованы в диапазоне от 0 до 1.

На вход нейронной сети для оценки следующего положения цели в пространстве подаются два параметра: координата цели на текущий момент времени x и разница по времени с ранее полученным значением координаты dt (для обнаружения пропусков в данных), как показано на рис. 3.

Для анализа результатов для каждого выходного набора данных рассчитывались: средняя абсолютная ошибка (Mean Absolute Error, MAE):

$$MAE = \frac{1}{m} \sum_{i=1}^m |a_i - y_i|; \quad (21)$$

среднеквадратическое отклонение (Mean Squared Error, MSE):

$$MSE = \frac{1}{m} \sum_{i=1}^m |a_i - y_i|^2; \quad (22)$$

корень среднеквадратического отклонения (Root MSE, RMSE):

$$RMSE = \sqrt{\frac{1}{m} \sum_{i=1}^m |a_i - y_i|^2}. \quad (23)$$

РЕЗУЛЬТАТЫ ИССЛЕДОВАНИЯ

Оптимизация выполнялась в течение 15 часов на процессоре Intel Xeon-1230 v2 (четыре ядра, восемь потоков, линейная частота 3,6 ГГц), за это время было выполнено около трехсот итераций (рис. 5).

Из графика хода процесса оптимизации видно, что наилучший результат был получен в районе двухсотой итерации и дальнейшие гипотезы результатов не дали.

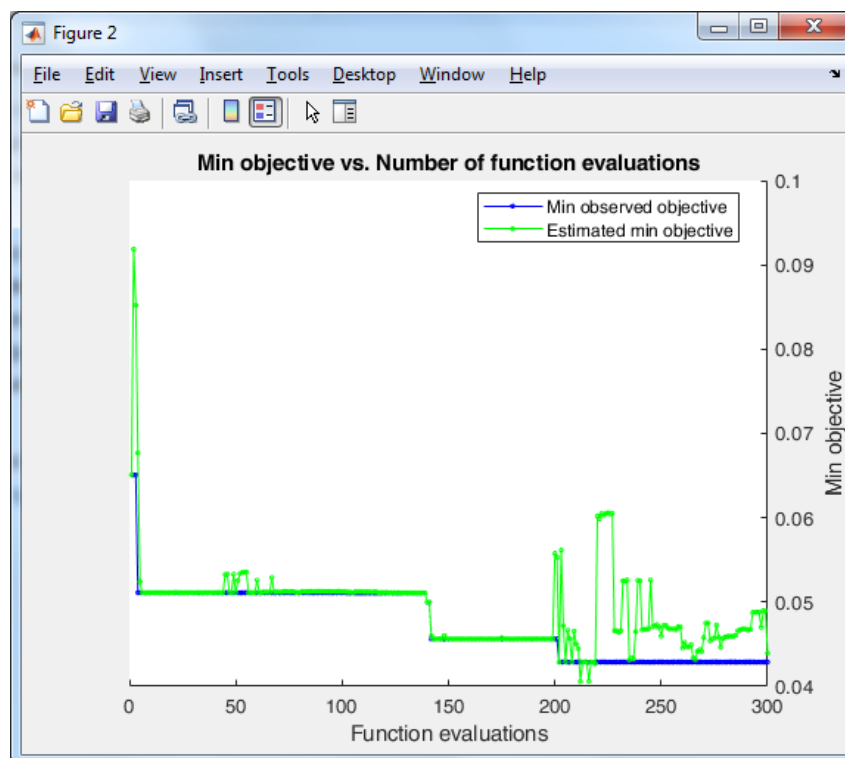


Рис. 5. Процесс обучения нейронной сети

Расчитанные по результатам моделирования средняя абсолютная ошибка, максимальная ошибка и среднеквадратическое отклонение представлены в табл. 2, фильтры обозначены следующим образом: KF – фильтр Калмана; LSTM – нейронная сеть LSTM; GRU – нейронная сеть GRU; Deep LSTM – разработанная глубокая нейронная сеть.

Таблица 2. Результаты моделирования

Фильтр	Средняя абсолютная ошибка	Максимальная ошибка	Среднеквадратическое отклонение
Шаг времени модели $ts = 0,1$			
KF	0,0900	0,2596	0,0111
LSTM	0,0565	0,2111	0,0052
GRU	0,0570	0,2275	0,0048
Deep LSTM	0,0557	0,1947	0,0049

Фильтр	Средняя абсолютная ошибка	Максимальная ошибка	Среднеквадратическое отклонение
Шаг времени модели $t_s = 0,05$			
KF	0,1169	0,2979	0,0173
LSTM	0,0695	0,2102	0,0073
GRU	0,0748	0,1979	0,0081
Deep LSTM	0,0709	0,1790	0,0077

Результаты экстраполяции по проверочному набору данных разработанной глубокой нейронной сети в сравнении с фильтром Калмана, нейронной сетью LSTM и нейронной сетью GRU представлены на рис. 6.

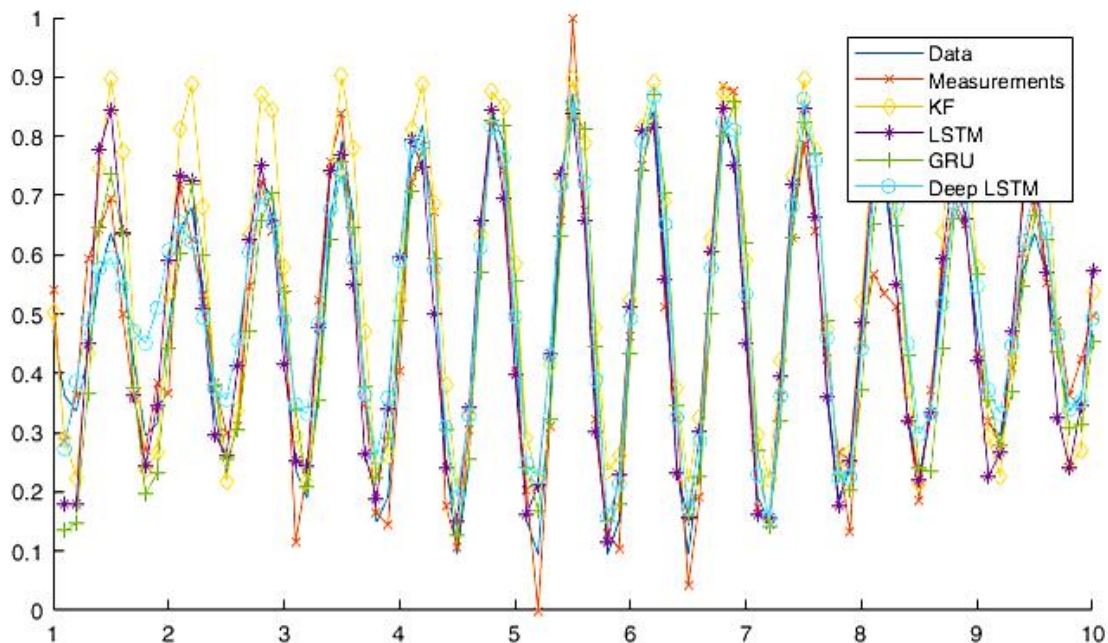


Рис. 6. Данные измерения: выход нейронной сети и выход фильтра Калмана в сравнении

По результатам моделирования видно, что фильтр Калмана вне конкуренции в ситуации, когда шум измерений относительно высок, а модель процесса в точности соответствует модели, положенной в основу фильтра.

Рекуррентные нейронные сети, в свою очередь, позволяют выявлять долговременные зависимости во входных данных и адаптироваться к изменяющейся модели движения цели.

Разработанная глубокая нейронная сеть в сравнении с однослойными решениями показывает более высокий уровень адаптации к измененной относительно применяемой для обучения модели движения и, как следствие, имеет более низкие среднюю и максимальную ошибки.

ЗАКЛЮЧЕНИЕ

Глубокая рекуррентная нейронная сеть LSTM, описанная в статье, может быть успешно применена в задаче фильтрации измерений (прогнозирования временных рядов) в случае, когда модель протекания процесса не соответствует заранее заданной, а шум измерений по своему характеру не является гауссовым.

Из недостатков данного подхода стоит отметить необходимость обучения нейронной сети на тестовой выборке, что представляет собой определенную вычислительную сложность. Подобная выборка может быть получена на основе модели измерений, лежащей в основе применяемого фильтра Калмана.

Способности нейронной сети к прогнозированию напрямую следуют из ее способности к обобщению и выделению скрытых зависимостей между входными и выходными данными. После обучения сеть способна предсказать будущее значение некой последовательности на основе нескольких предыдущих значений и (или) существующих в настоящий момент факторов. Прогнозирование возможно только тогда, когда предыдущие изменения действительно в какой-то степени определяют будущее.

Выбор архитектуры нейронной сети зависит от характера входных данных, их пригодности к классификации и структурированию (для обучения без учителя), особенностей применения нейронной сети: рекуррентные нейронные сети имеют более сложную структуру и дольше обучаются, но за счет обратной связи лучше адаптируются к шуму измерений и дают меньшую ошибку.

Применение в архитектуре разработанного экстраполирующего фильтра более двух рекуррентных блоков для описанных в данной работе входных данных не повышает точность фильтрации и экстраполяции координат, увеличивает время обучения и вычислительную сложность алгоритма.

Разработанная исполняемая модель позволяет проводить оптимизацию глубоких нейронных сетей для задачи фильтрации и экстраполяции координат движущейся цели методом байесовской оптимизации, а также сравнивать результаты работы полученной нейронной сети с классическим фильтром Калмана, нейронной сетью прямого распространения, однослойными рекуррентными нейронными сетями LSTM и GRU.

Разработанный экстраполирующий фильтр координат цели показал себя лучше классического фильтра Калмана в условиях, когда наблюдаемая модель движения не вполне соответствует модели, используемой при обучении нейронной сети (положенной в основу фильтра Калмана). Авторы полагают, что использование экстраполирующих фильтров, построенных на основе рекуррентных нейронных сетей, может быть полезно при разработке систем траекторной обработки, использующих фильтры на основе множественных взаимодействующих моделей, за счет способности рекуррентной нейронной сети к обобщению данных во времени и, как следствие, быстрой адаптации к изменяющейся модели движения цели.

ЛИТЕРАТУРА

1. Adaptive AR modeling of nonstationary time series by means of Kalman filtering / M. Arnold [et al.] // *IEEE Transactions on Biomedical Engineering*. 1998. V. 45. № 5. P. 553–562.
2. Sorensen J.V.T., Madsen H. Water level prediction skill of an operational marine forecast using a hybrid Kalman filter and time series modeling approach // *Oceans 2003. Celebrating the Past... Teaming Toward the Future (IEEE Cat. No. 03CH37492)*. IEEE. 2003. V. 2. P. 790.
3. Brookner E. Tracking and Kalman filtering made easy. New Jersey: John Wiley & Sons, Inc. 1998. 480 p.

4. Kalman R.E. A new approach to linear filtering and prediction problems // *Journal of Basic Engineering*. 1960. V. 82. Is. 1. P. 35–45.
5. Julier S.J., Uhlmann J.K. New extension of the Kalman filter to nonlinear systems // *Signal Processing, Sensor Fusion, and Target Recognition VI*. 1997. V. 3068. P. 182–193.
6. Zarchan P., Musoff H. *Fundamentals of Kalman filtering: a practical approach*. United States: American Institute of Aeronautics and Astronautics. 2000. 664 p.
7. Zell A. *Simulation neuronaler netze*. Boston: Addison-Wesley. 1994. 624 p.
8. Schmidhuber J. Deep learning in neural networks: An overview // *Neural networks*. 2015. V. 61. P. 85–117.
9. Time-series well performance prediction based on Long Short-Term Memory (LSTM) neural network model / X. Song [et al.] // *Journal of Petroleum Science and Engineering*. 2020. V. 186. P. 106682.
10. Goodfellow I., Bengio Y., Courville A. 6.5 Back-propagation and other differentiation algorithms. *Deep Learning*. 2016. 800 p.
11. Kingma D.P., Ba J. Adam: A method for stochastic optimization // *arXiv preprint arXiv: 1412.6980*. 2014. URL: <https://arxiv.org/abs/1412.6980> (дата обращения: 12.11.2022).

Для цитирования: Кемайкин В.К., Польшаев А.С. Архитектура глубокой нейронной сети в задаче прогнозирования координат движущейся цели // Вестник Тверского государственного технического университета. Серия «Технические науки». 2023. № 2 (18). С. 101–112.

ARCHITECTURE OF A DEEP NEURAL NETWORK IN THE PROBLEM OF PREDICTING THE COORDINATES OF A MOVING TARGET

V.K. KEMAYKIN, Cand. Sc., A.S. POLYGAEV, stud.

Tver State Technical University
22, Af. Nikitin emb., Tver, 170026, Russian Federation;
e-mail: vk-kem@mail.ru

The use of a deep recurrent neural network in the problem of predicting the coordinates of a moving target under conditions of measurement inaccuracy is considered. The training of the neural network is carried out on a preliminary sample of coordinates of a limited length, errors in measuring the coordinates are modeled, the nature of the movement of the target during network testing differs from the model used in training the neural network and underlying the Kalman filter.

Keywords: artificial neural network, long short-term memory neural network, Kalman filter, time series forecasting, measurement filtering, extrapolation.

Поступила в редакцию/received: 30.10.2022; после рецензирования/revised: 12.11.2022;
принята/accepted: 24.11.2022