

УДК 004.02

DOI: [10.26102/2310-6018/2019.27.4.011](https://doi.org/10.26102/2310-6018/2019.27.4.011)

## ГЛУБОКИЕ НЕЙРОННЫЕ СЕТИ ДЛЯ ПРЕДСКАЗАТЕЛЬНОГО ТЕХНИЧЕСКОГО ОБСЛУЖИВАНИЯ

**Сай Ван Квонг**

*Волгоградский государственный технический университет,*

*Волгоград, Россия*

*e-mail: [svcuonghvktqs@gmail.com](mailto:svcuonghvktqs@gmail.com)*

**Резюме:** Глубокие нейронные сети в настоящее время становятся одним из самых популярных подходов при решении различных практических задач из самых разнообразных областей, таких как распознавание изображений и речи, обработка естественного языка, компьютерное зрение, медицинская информатика и др. В статье рассматривается возможность применения глубоких нейронных сетей при реализации проактивной стратегии технического обслуживания и ремонта (ТОиР) – предсказательного технического обслуживания (Predictive maintenance, PdM). Рассмотрены различные методы построения предсказательных моделей для PdM. В настоящее время для построения предсказательных моделей для PdM наиболее перспективным представляются подходы, основанные на обработке данных с использованием глубоких нейронных сетей. Одна из причин успешного применения глубоких нейронных сетей заключается в том, что сеть автоматически выделяет из данных важные признаки, необходимые для решения задачи. Рассмотрены наиболее часто используемые нейронные сети для PdM: сеть долго-краткосрочной памяти (Long Short-Term Memory, LSTM), свёрточная нейронная сеть (convolutional neural network, CNN) и автоэнкодер (autoencoder). Дан обзор мощных фреймворков для проектирования и обучения нейронных сетей, которые сделали возможным широкое практическое применение данной технологии.

**Ключевые слова:** предсказательное техническое обслуживание, методы на основе данных, глубокие нейронные сети, LSTM, CNN, автоэнкодер.

**Для цитирования:** Сай Ван Квонг. Глубокие нейронные сети для предсказательного технического обслуживания. *Моделирование, оптимизация и информационные технологии*. 2019;7(4). Доступно по: [https://moit.vivt.ru/wp-content/uploads/2019/11/SaiVanCuong\\_4\\_19\\_1.pdf](https://moit.vivt.ru/wp-content/uploads/2019/11/SaiVanCuong_4_19_1.pdf)  
DOI: 10.26102/2310-6018/2019.27.4.011.

## DEEP NEURAL NETWORKS FOR PREDICTIVE MAINTENANCE

**Sai Van Cuong**

*Volgograd State Technical University, Volgograd, Russia*

**Abstract:** At present, deep neural networks are becoming one of the most popular approaches in solving various practical problems from a wide variety of fields, such as image and speech recognition, natural language processing, computer vision, medical informatics, etc. The article considers the possibility of using deep neural networks in the implementation of proactive maintenance strategy – predictive maintenance (PdM). Various methods of constructing predictive models for PdM are considered. Currently, the data-driven approaches using deep neural networks for constructing predictive models for PdM are most promising methods. One of the reasons for the successful application of deep neural networks is that the networks automatically selects important features from the data needed to solve the problem. The most commonly used neural networks for PdM are considered: Long short-term memory (LSTM), convolutional neural networks (CNN) and autoencoders. An overview of powerful frameworks for the design and training of neural networks is given.

**Keywords:** predictive maintenance, data-driven methods, deep neural networks, LSTM, CNN, autoencoders.

**For citation:** Sai Van Cuong. Deep neural networks for predictive maintenance. *Modeling, Optimization and Information Technology*. 2019;7(4). Available from: [https://moit.vivt.ru/wp-content/uploads/2019/11/SaiVanCuong\\_4\\_19\\_1.pdf](https://moit.vivt.ru/wp-content/uploads/2019/11/SaiVanCuong_4_19_1.pdf) DOI: 10.26102/2310-6018/2019.27.4.011 (In Russ).

## Введение

Обеспечение надежного функционирования оборудования на этапе эксплуатации с минимальными затратами является приоритетной и актуальной задачей для различных производств [1]. Простой промышленного оборудования вызывает излишний расход производственных ресурсов, что отражается на повышении себестоимости производства, и является неприемлемым в нынешней экономической ситуации. В современной производственной промышленности оборудование и системы становятся более сложными, интегрированными и автоматизированными. Эта сложность приводит к тому, что нарушение нормального функционирования небольшого объекта инфраструктуры может привести к созданию аварийных ситуаций, сопровождающихся значительным ущербом и нарушению работы всей системы в целом [2]. Классические подходы к организации ТОиР (корректирующее и планово-профилактическое обслуживание) при таких условиях в значительной мере теряют свою эффективность и не удовлетворяют требованиям современной промышленности.

Исходя из анализа работ [1, 3–12], следует сделать вывод, что предсказательное техническое обслуживание (Predictive maintenance, PdM) является наиболее эффективной и целесообразной стратегией технического обслуживания в современных экономических условиях. Философия PdM заключается в выполнении ТОиР только в случае необходимости. В идеале, график технического обслуживания может быть оптимизирован для минимизации затрат на ТОиР и достижения нулевого отказа производства с помощью этой политики. При реализации стратегии PdM применяются различные методы построения предсказательных моделей. Эти методы в основном можно разделить на 2 типа: методы, основанные на обработке данных (data-driven methods), методы, основанные на моделях (model-based methods) [1].

Методы, основанные на моделях (МОМ), используют математические модели для описания физического поведения и процессов деградации оборудования, при этом значения параметров изменяются на основе собранных данных [3, 4]. Одним из наиболее часто используемых методов, основанных на моделях являются модель Марковского процесса [5], модель Винеровского процесса [4], модель Гауссовой смеси [6], итд. Основной недостаток МОМ заключается в необходимости регулярной структурной и параметрической оптимизации моделей из-за динамически меняющегося окружения [7]. Кроме того, сложные системы требуют значительных затрат на создание и настройку МОМ, в том числе и с привлечением экспертов [8]. Для преодоления данных недостатков были предложены и в настоящее время активно развиваются методы, основанные на использовании данных.

Методы, основанные на обработке данных (МОД) описывают процесс деградации оборудования на основе измеряемых данных с использованием статистических методов или методов искусственного интеллекта [7]. МОД обладают свойствами универсальности, поскольку они абстрагированы от физической природы объектов, не требуют знаний его внутренней структуры и функциональных связей между элементами [9]. МОД могут быть построены на основании классических алгоритмов машинного обучения, таких как метод опорных векторов, случайный лес, деревья решений,

экстремальный градиентный бустинг, метод k-ближайших соседей, итд. Следует отметить критику применения этих подходов в решении задачи прогнозирования [1], в частности связанную с тем, что методы дают слабые результаты при обработке больших сложных многомерных сенсорных данных. Кроме того, эти классические модели машинного обучения обычно основаны на процессе инженерии признаков (*feature engineering*), в результате которого создаются и отбираются нужные признаки используя опыт в данной области. Это как правильно, затрудняет повторное использование этих моделей, поскольку инженерия признаков специфична для конкретного сценария проблемы. Наиболее перспективным представляется использование искусственных нейронных сетей для PdM [10]. Одна из причин успешного применения глубоких нейронных сетей заключается в том, что сети могут автоматически выделять нужные признаки из данных, устраняя необходимость в ручном *feature engineering* с помощью экспертов (Рисунок 1). Кроме этого, при обработке больших объемов данных нейронная сеть справляется с выделением признаков гораздо лучше, чем человек.

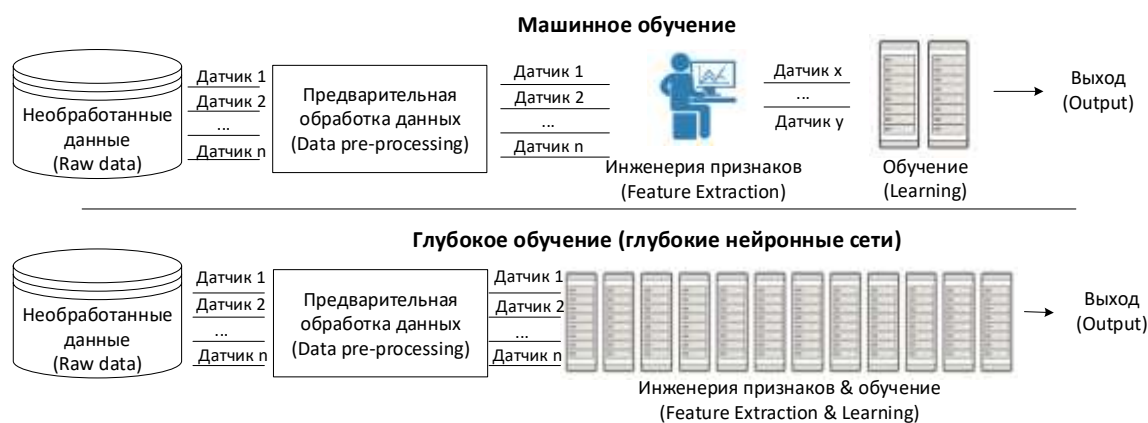


Рисунок 1- Разница между машинным обучением и глубоким обучением в PdM

В дальнейшем описывается постановка задачи построения предсказательных моделей на основе обработки собранных датчиками данных о состоянии оборудования и об условиях его работы. Рассмотрены наиболее эффективные и часто используемые нейронные сети в данной области и ведущие платформы для их проектирования и обучения.

### Постановка задачи

МОД включают в себя несколько этапов: сбор данных, обработка данных, анализ данных для создания предсказательных моделей, принятие решений по ТОиР для предотвращения отказов оборудования (Рисунок 2).

Развитие повсеместных систем и удешевление систем сбора данных позволяет собирать большие объемы данных о многообъектных целевых системах. Будем считать, что имеются значения характеристик системы, записанные в течение эксплуатационного срока службы одного или нескольких экземпляров однотипного оборудования. Мы обозначаем этот набор экземпляров как *ID*.

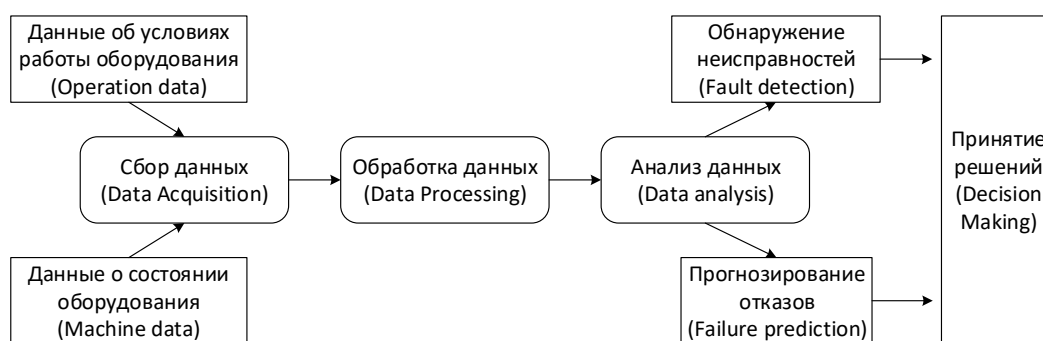


Рисунок 2 - Модель предсказательного технического обслуживания, основанная на обработке данных

Датчики являются источниками информации о состоянии узлов и агрегатов оборудования и об условиях эксплуатации оборудования. Для каждого экземпляра  $i \in ID$ , многомерные сенсорные данные могут быть представлены в виде временных рядов  $X^{(i)} = \{X_1^{(i)}, X_2^{(i)}, \dots, X_{T^{(i)}}^{(i)}\}$ , где  $T^{(i)}$  – длина временных рядов для  $i$ -ого экземпляра,  $X_t^{(i)} = \{x_{t,1}^{(i)}, x_{t,2}^{(i)}, \dots, x_{t,n}^{(i)}\}$  это  $n$ -мерный вектор, соответствующий показаниями  $n$  датчиков в момент времени  $t$ . Для экземпляра оборудования  $i$  для которого был зафиксирован отказ, длина  $T^{(i)}$  соответствует полному эксплуатационному сроку службы (от начальной приработки до отказа), а для текущего работающего экземпляра, длина  $T^{(i)}$  соответствует истекшему эксплуатационному сроку службы (от начальной приработки до текущего наблюдаемого момента времени).

Для системы для которой были зафиксирован отказ мы можем вычислить остаточный ресурс (Remaining Useful Life, RUL) на каждом шаге времени:  $RUL(t_0) = t_1 - t_0$ , если  $t_1 \geq t_0$ , где:  $t_1$  – время, на котором произошёл отказ (максимальное время работы системы),  $t_0$  – текущее время наблюдения за работой системы (Рисунок 3).

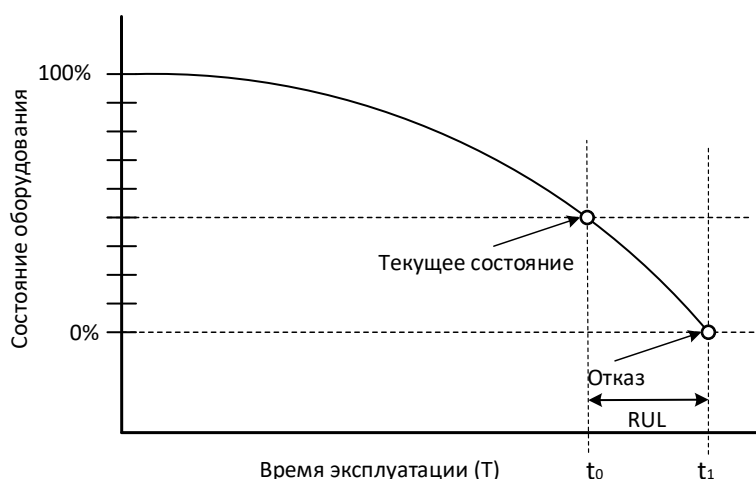


Рисунок 3 - Иллюстрация к изменению состояния оборудования и наступления отказа

Отсюда формируется набор размеченных данных. На основе этих размеченных данных формируются модели для решения двух типов задач: 1) задача классификации заключается в предсказании возможности появления отказа оборудования на некоторый

определенный момент времени в будущем, 2) задача *регрессии* заключается в прогнозировании величины остаточного ресурса оборудования (RUL). Полученные модели могут быть использованы для выработки рекомендаций оптимального использования оборудования в онлайн режиме. Такой подход называется обучением с учителем (*supervised learning*). При этом имеются множество характеристик системы  $X = \{X_1, X_2, \dots, X_k\}$  и множество возможных ответов  $Y = \{y_1, y_2, \dots, y_k\}$ , которые формируют обучающее множество пар  $\{(X_1, y_1), \dots, (X_k, y_k)\}$ . На основе этих данных требуется восстановить зависимость, то есть построить модель, способная для любого объекта  $X$  определить к какому ответу он принадлежит.

Однако на практике часто бывает сложно или невозможно получить множество размеченных данных. Кроме этого, для сложных многообъектных системах исходная информация представляется в виде признаковых описаний, причём число признаков может быть достаточно большим. В таких условиях, целесообразно использовать так называемые методы обучения без учителя (*unsupervised learning*). При этом имеются только множество характеристик системы  $\{X_1, X_2, \dots, X_k\}$ . Требуется обнаружить внутренние взаимосвязи, зависимости, закономерности, существующие между объектами множества. Все задачи, решаемые с помощью обучения без учителя, относятся к одной из следующих категорий: задача кластеризации, задача выявления аномалий, задача уменьшения размерности данных.

### Применение глубоких нейронных сетей для PdM

Глубокие нейронные сети показали превосходные результаты в различных областях, таких как распознавание изображений и речи, обработка естественного языка, компьютерное зрение, медицинская информатика и др. Они также приобрели популярность в других областях, таких как финансы, где данные временных рядов играют важную роль. Аналогичным образом, в PdM, собранные датчиками данные о состоянии оборудования и об условиях его работы имеют решающее значение для целей ТОиР. Исходя из анализа литературы, следует отметить, что сеть долго-краткосрочной памяти (Long Short-Term Memory, LSTM), свёрточная нейронная сеть (convolutional neural network, CNN) и автоэнкодер (autoencoder) являются наиболее часто используемыми методами при реализации стратегии PdM. В Таблице 1 описаны преимущества этих сетей для PdM.

Таблица 1 - Преимущества некоторых глубоких нейронных сетей для PdM

Архитектура	Преимущества для PdM
LSTM	Способны к обучению долгосрочным зависимостям, являются наилучшим выбором для прогнозирования временных рядов и последовательностей.
CNN	Отличаются очень высокой способностью к извлечению информации из больших данных, особенно для изображений. Благодаря операции свертки, фильтры в сверточных слоях могут извлекать локальные паттерны в необработанных данных и далее создавать сложные паттерны для мониторинга состояния оборудования путем комбинирования этих сверточных слоев.
Автоэнкодер	Анализ характеристик неисправностей, извлечение признаков или скрытой информации об отказах из необработанных данных, снижение размерности данных (сжатие данных).

Сети LSTM – особая разновидность архитектуры рекуррентных нейронных сетей (Recurrent neural networks, RNN), разработанная Хохрейтером и Шмидхубером в 1997 году. LSTM имеют цепную структуру повторяющихся модулей (repeating module) нейронной сети как и стандартные RNN. Повторяющиеся модули в стандартных RNN имеют очень простую структуру (только один слой), а в LSTM имеют более сложную структуру (вместо одного нейронного слоя, LSTM имеют четыре взаимодействующих специальных слоя) (Рисунок 4. А). Благодаря этой структуре LSTM способны запоминать информацию в течение длительных периодов времени. По этой причине LSTM является выдающимся инструментом для моделирования данных последовательности, в частности данных многомерных временных рядов. Это подтверждается превосходящими результатами прогнозирования остаточного ресурса турбореактивных авиационных двигателей в [1], остаточного ресурса литий-ионных аккумуляторов в [7], износа станков ЧПУ в [11], итд. Рисунок 4.Б иллюстрирует применение сетей LSTM для моделирования данных многомерных временных рядов.

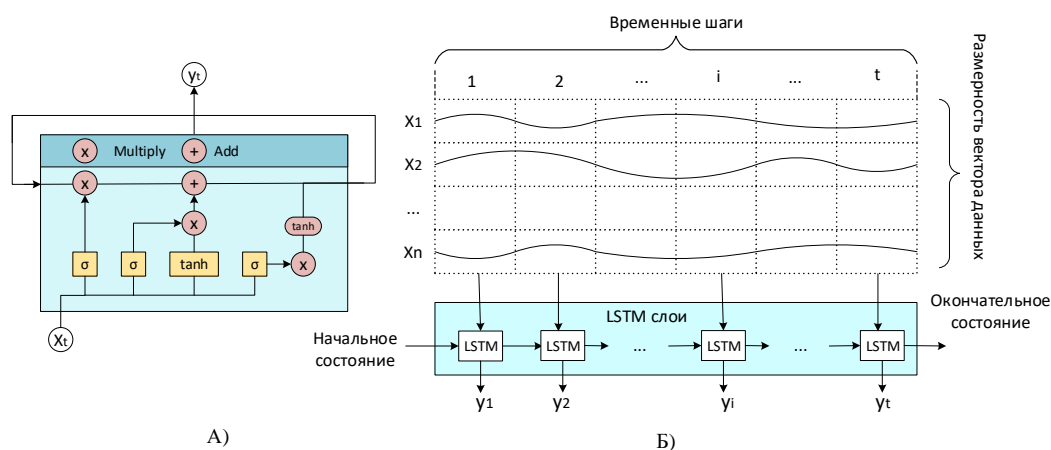


Рисунок 4 - А) Повторяющийся модуль LSTM состоит из 4 слоя, Б) LSTM для моделирования данных многомерных временных рядов

CNN – специальная архитектура искусственных нейронных сетей, предложенная Я. Лекуном в 1988 году, на данный момент является лучшим инструментом для обработки изображений и видео. Основная причина успеха в этом контексте является ее высокой способностью извлечения локальной информации из больших данных путем скольжения сверточных фильтров по двумерным входным данным. Для PdM, в многих сценариях собранные сенсорные данные имеют пространственную структуру, такую как 2D-структура в изображениях, поэтому CNN широко используется в этой области. Простейшая архитектура CNN состоит из сверточных слоев (convolution layers), субдискретизирующих слоев (subsampling layers) и полносвязных слоев (fully-connected layer), которые могут чередоваться в произвольном порядке (Рисунок 5). На сверточных слоях выполняется операция свертки входной карты признаков, слои субдискретизации уменьшают размерность карты признаков и обеспечивают инвариантность к масштабу. После прохождения нескольких слоев карта признаков раскладывается в вектор на полносвязных слоях, который используется в качестве входа для последующего распознавания паттернов. Предложенные модели CNN в работах [1, 12] дают высокую точность при решении задачи прогнозирования остаточного ресурса турбореактивных двигателей на основе данных многомерных рядов.

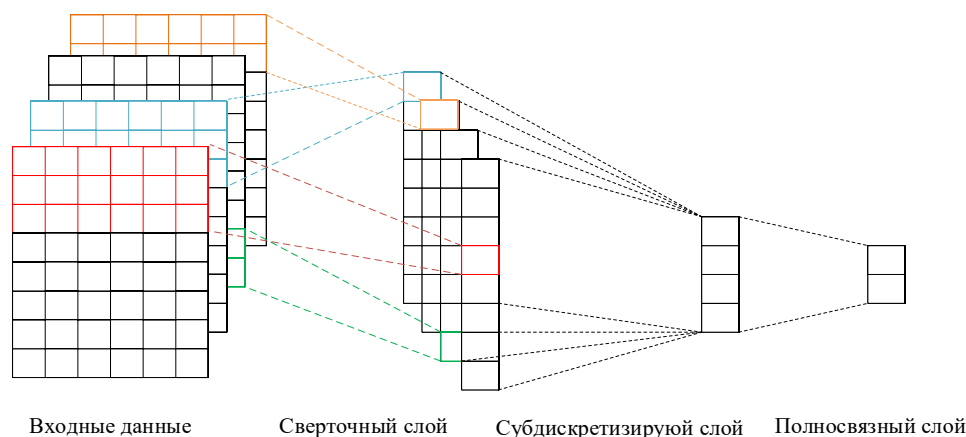


Рисунок 5 - Архитектура сверточной нейронной сети CNN

Автоэнкодер представляет собой искусственную нейронную сеть с специальной симметричной структурой (Рисунок 6). Простейшая архитектура автоэнкодера состоит из входного слоя, скрытого слоя и выходного слоя. Выходной слой содержит столько же нейронов, сколько и входной слой, а количество нейронов скрытого слоя меньше. Так как размерность скрытого пространства меньше, чем пространство исходных данных, нейросеть отбирает только важные признаки. Это позволяет прежде всего осуществить сжатие данных при передаче вектора входного сигнала на выход сети при условиях, если в данных есть скрытые взаимосвязи, корреляция признаков или структура.

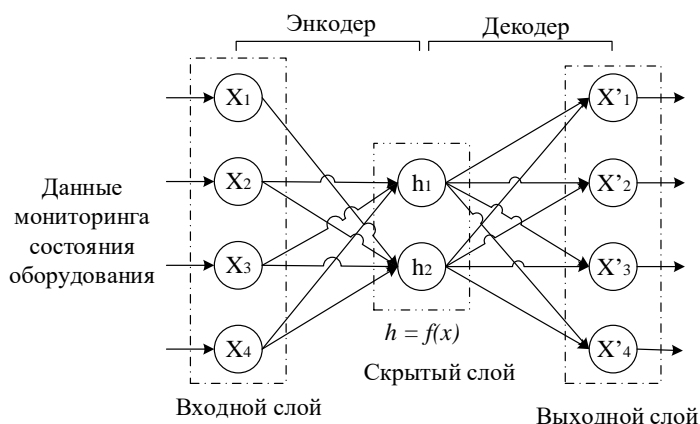


Рисунок 6 - Архитектура автоэнкодера

Кроме этого, автоэнкодеры также широко применяются для обнаружения аномалий в больших многомерных наборах данных. Многослойные автоэнкодеры используются для диагностики неисправностей асинхронного двигателя в работе [13], для диагностики неисправностей компонентов роторных двигателей в работе [14] и для уменьшения размерности данных мониторинга состояния интеллектуального подшипника в работе [15].

В многих задачах повышение точности прогнозирования достигается за счет использования сочетания различных методов. Это обусловлено наличием у каждого отдельного базового метода недостатков и ограничений, которые нейтрализуются при их комплексном использовании. В работе [1] предложенная нами гибридная CNN-LSTM модель дает превосходный результат по сравнению с альтернативными методами при прогнозировании остаточного ресурса авиационных газотурбинных двигателей, где

сеть CNN для извлечения признаков и сеть LSTM для интерпретации функций по временным шагам. Архитектура этой гибридной модели показывается на Рисунке 7.

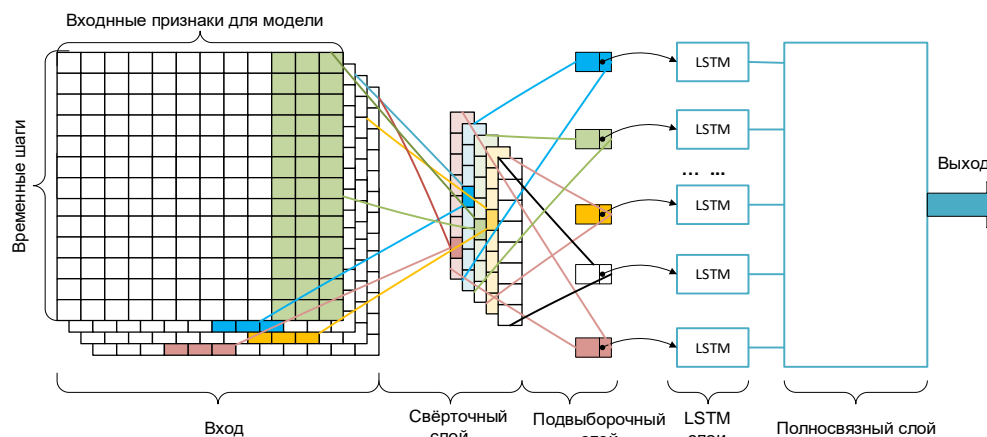


Рисунок 7. Архитектура гибридной CNN-LSTM модели для прогнозирования остаточного ресурса многообъектных сложных систем.

### Инструменты для глубоких нейронных сетей

Одним из ключевых факторов, определяющих широкое распространение практического применения глубоких нейронных сетей, является наличие большого количества различных мощных фреймворков для проектирования и обучения нейронных сетей. Среди наиболее популярных из них в настоящее время можно отметить TensorFlow, Theano, PyTorch, CNTK и Keras.

TensorFlow [16] – открытая программный фреймворк, разработанный компанией Google на языках C++ и Python для проведения высокоскоростных численных вычислений с использованием графов потока данных. Он позволяет легко развертывать вычисления на различных платформах, таких как CPUs, GPUs и TPUs. Tensorflow обеспечивает мощную поддержку машинного обучения и глубокого обучения (нейронных сетей), а также можно использовать для численных вычислений в других научных областях. Основной API для работы с фреймворком реализован для Python, также существуют реализации для C, C++, Java, Go, JavaScript, R, Julia и Swift.

Theano [17] является одной из первых библиотек в Python с открытым исходным кодом для нейронных сетей, которая разработана группой MILA из Монреальского университета. Theano использует NumPy-подобный синтаксис, который затем компилируется для эффективных параллельных вычислений как на обычных CPU, так и на GPU. В последнее время библиотека Theano утратила популярность, в связи с тем что разработчик (Монреальский университет) объявил о прекращении работы над проектом.

PyTorch [20] – современная библиотека глубокого обучения для языка Python с открытым исходным кодом, развивающаяся под крылом Facebook. хорошо работает с Python и создана для апологетов Python. В отличие от TensorFlow, библиотека PyTorch оперирует динамически обновляемым графом, что позволяет вносить изменения в архитектуру в процессе работы.

CNTK (Cognitive Toolkit) – библиотека с открытым исходным кодом для проектирования и развития нейронных сетей разнообразных видов, разработанная Microsoft Research. CNTK реализован с использованием C++ и Python, но он также доступен в C# и Java. На данный момент CNTK обладает одной из самых высоких точностей для обучения моделей нейронных сетей и высокой скоростью обучения. Однако по понятным причинам CNTK в большей мере ориентирован на Windows.



TensorFlow, CNTK, Theano и PyTorch – одни из ведущих платформ для нейронных сетей в настоящее время, и все могут использоваться для ускорения обучения как многоядерные процессоры, так и ускорители вычислений GPU (включая оптимизированную библиотеку cuDNN). Тем не менее, они являются низкоуровневыми инструментами. Это значит, что нужно тщательно продумывать архитектуру нейросети, правильно оценивать размерность и объемы входных и выходных данных. Таким образом, работа с этими фреймворками требует написания значительного количества программного кода.

Наиболее минималистичный подход к использованию TensorFlow, Theano и CNTK в качестве бэкэндов дает открытая высокоуровневая нейросетевая библиотека Keras [19], написанная исследователем Google AI Франсуа Шолле на языке Python с целью ускорения экспериментов. Keras представляет собой библиотеку высокого уровня, предоставляющую высокоуровневые строительные блоки для конструирования моделей глубокого обучения. Она не реализует низкоуровневые операции, такие как манипуляции с тензорами и дифференцирование, для этого используется специализированная и оптимизированная библиотека поддержки тензоров.

### Заключение

В статье рассмотрены различные методы построения предсказательных моделей при реализации проактивной стратегии ТОиР – предсказательного технического обслуживания, нацеленного на минимизацию затрат на ТОиР. Наиболее перспективными представляются методы, основанные на обработке данных с использованием глубоких нейронных сетей. Наиболее эффективными и часто используемыми нейронными сетями являются CNN, LSTM и автоэнкодеры. Одними из ведущих платформ для проектирования и обучения глубоких нейронных сетей в настоящее время являются: TensorFlow, Theano, PyTorch CNTK и Keras.

*Работа выполнена при поддержке РФФИ, проект № 19-47-340010\_r\_a.*

### ЛИТЕРАТУРА

1. Сай Ван Квонг, Щербаков М.В. Метод прогнозирования остаточного ресурса на основе обработки данных многообъектных сложных систем. *Прикаспийский журнал: управление и высокие технологии*. 2019;(1):33-44.
2. Ажмухамедов И.М., Гостюнин Ю.А. Выбор стратегия технического обслуживания и ремонта оборудования сетей связи на предприятиях нефтегазового комплекса. *Электронный научный журнал: Инженерный вестник Дона*. 2017;(2):74-84.
3. Liu J., Wan W., Golnaraghi F. A multi-step predictor with a variable input pattern for system state forecasting // *Mech. Syst. Signal Process.* 2009:1586-1599.
4. Huang Z., Xu Z., Wang W., Sun Y. Remaining useful life prediction for a nonlinear heterogeneous Wiener process model with an adaptive drift. *IEEE Trans. Rel.* 2015;(2):687-700.
5. Dui H., Si S., Zuo M., Sun S. Semi-Markov process-based integrated importance measure for multi-state systems. *IEEE Transactions on Reliability*. 2015;64(2):54-765.
6. Cartella F., Lemeire J., Dimiccoli L., Sahli H. Hidden semi-Markov models for predictive maintenance. *Mathematical Problems in Engineering*. 2015.
7. Liu J., Saxena A., Goebel K., Saha B., Wang W. An adaptive recurrent neural network for remaining useful life prediction of lithiumion batteries. *Annual Conference of the Prognostics and Health Management Society*. 2010:1-9.

8. Lei Y., Li N., Gontarz S., Lin J., Radkowski S., Dybala S. A Model-Based Method for Remaining Useful Life Prediction of Machinery . *IEEE Transactions on Reliability*. 2017.
9. Khoury E., Deloux E., Grall A., Berenguer C. On the Use of Time-Limited Information for Maintenance Decision Support: A Predictive Approach under Maintenance Constraints // *Mathematical Problems in Engineering*, 2013:1-11.
10. De Benedetti M., Leonardi F., Messina F., Santoro C., Vasilakos A. Anomaly detection and predictive maintenance for photovoltaic systems. *Neurocomputing*. 2018:59-68.
11. Zhao R., Wang J., Yan R., Mao K. Machine health monitoring with LSTM networks. *10th International Conference on the Sensing Technology*. 2016.
12. Li X., Ding Q., Sun J. Q. Remaining useful life estimation in prognostics using deep convolution neural networks. *Reliability Engineering & System Safety*/ 2018;(172):1-11.
13. Sun W., Shao S., Zhao R., Yan R., Zhang X., Chen X. A sparse auto- encoder-based deep neural network approach for induction motor faults classification. *Measurement*. 2016;(89):171-178.
14. Lu C., Wang Z., Qin W., Ma, J. Fault diagnosis of rotary machinery components using a stacked denoising autoencoder-based health state. 2017.
15. Guo L., Gao H., Huang H., He X., Li S. Multi-features fusion and nonlinear dimension reduction for intelligent bearing condition monitoring. *Shock and Vibration*. 2016.
16. TensorFlow. Режим доступа: <https://www.tensorflow.org/>, (дата обращения 22.09.2019 на англ).
17. Theano. Режим доступа: <http://deeplearning.net/software/theano/>, (дата обращения 25.09.2019 на англ.).
18. CNTK. Режим доступа: <https://cntk.ai>, свободный. (дата обращения 25.09.2019 на англ).
19. Keras. Режим доступа: <https://keras.io>, (дата обращения 26.09.2019 на англ.).
20. PyTorch. Режим доступа: <https://pytorch.org>, (дата обращения 27.10.2019 на англ.).

## REFERENCES

1. Sai Van Cuong, Shcherbakov M. V. A data-driven method for remaining useful life prediction of multiple-component systems. *Caspian journal: management and high technologies*. 2019;(1):33-44.
2. Azhmuamedov I.M., Gostjunin Ju.A. Vybor strategija tehničkog obsluzhivaniya i remonta oborudovaniya setej svjazi na predpriyatijah neftegazovogo kompleksa. *Jelektronnyj nauchnyj zhurnal: Inzhenernyj vestnik Dona*. 2017;(2):74-84.
3. Liu J., Wan W., Golnaraghi F. A multi-step predictor with a variable input pattern for system state forecasting. *Mech. Syst. Signal Process*. 2009:1586-1599.
4. Huang Z., Xu Z., Wang W., Sun Y. Remaining useful life prediction for a nonlinear heterogeneous Wiener process model with an adaptive drift. *IEEE Trans. Rel*. 2015;(2):687-700.
5. Dui H., Si S., Zuo M., Sun S. Semi-Markov process-based integrated importance measure for multi-state systems. *IEEE Transactions on Reliability*. 2015;64(2):54-765.
6. Cartella F., Lemeire J., Dimiccoli L., Sahli H. Hidden semi-Markov models for predictive maintenance. *Mathematical Problems in Engineering*. 2015.
7. Liu J., Saxena A., Goebel K., Saha B., Wang W. An adaptive recurrent neural network for remaining useful life prediction of lithiumion batteries. *Annual Conference of the Prognostics and Health Management Society*. 2010:1-9.

8. Lei Y., Li N., Gontarz S., Lin J., Radkowski S., Dybala S. A Model-Based Method for Remaining Useful Life Prediction of Machinery. *IEEE Transactions on Reliability*. 2017.
9. Khoury E., Deloux E., Grall A., Berenguer C. On the Use of Time-Limited Information for Maintenance Decision Support: A Predictive Approach under Maintenance Constraints. *Mathematical Problems in Engineering*, 2013:1-11.
10. De Benedetti M., Leonardi F., Messina F., Santoro C., Vasilakos A. Anomaly detection and predictive maintenance for photovoltaic systems. *Neurocomputing*. 2018:59-68.
11. Zhao R., Wang J., Yan R., Mao K. Machine health monitoring with LSTM networks. *10th International Conference on the Sensing Technology*. 2016.
12. Li X., Ding Q., Sun J. Q. Remaining useful life estimation in prognostics using deep convolution neural networks. *Reliability Engineering & System Safety*. 2018;(172):1-11.
13. Sun W., Shao S., Zhao R., Yan R., Zhang X., Chen X. A sparse auto-encoder-based deep neural network approach for induction motor faults classification. *Measurement*. 2016;(89): 171-178.
14. Lu C., Wang Z., Qin W., Ma, J. Fault diagnosis of rotary machinery components using a stacked denoising autoencoder-based health state. 2017.
15. Guo L., Gao H., Huang H., He X., Li S. Multi-features fusion and nonlinear dimension reduction for intelligent bearing condition monitoring. *Shock and Vibration*. 2016.
16. TensorFlow. Available at: <https://www.tensorflow.org/> (accessed 22.09.2019).
17. Theano. Available at: <http://deeplearning.net/software/theano/>, (accessed 25.09.2019).
18. CNTK. Available at: <https://cntk.ai>, (accessed 25.09.2019).
19. Keras. Available at: <https://keras.io>, (accessed 26.09.2019).
20. PyTorch. Available at: <https://pytorch.org>, (accessed 27.10.2019).

#### ИНФОРМАЦИЯ ОБ АВТОРЕ / INFORMATION ABOUT THE AUTHOR

**Сай Ван Квонг**, аспирант, кафедра системы автоматизированного проектирования и поискового конструирования, ФГБОУ ВО "Волгоградский государственный технический университет", Волгоград, Российская Федерация.

**Sai Van Cuong**, PhD Student, CAD Department, Volgograd State Technical University, Volgograd, Russian Federation.