

УДК 004.932

<https://doi.org/10.33619/2414-2948/122/13>

ОСНОВЫ РАБОТЫ С TENSORFLOW: СИНТАКСИС И БАЗОВЫЕ ОПЕРАЦИИ

©Сыдыкова М. Б., ORCID: 0009-0002-0401-0997, SPIN-code: 8614-0079, канд. пед. наук,
Кыргызский национальный университет им. Ж. Баласагына,
г. Бишкек, Кыргызстан, mb_sydykova@mail.ru

©Мамбетова З. Д., SPIN-код: 8121-0721, канд. пед. наук, Кыргызский государственный
университет им. И. Арабаева, г. Бишкек, Кыргызстан, zarya.mambetova@mail.ru

©Убайдылдаева Ж. А., канд. пед. наук, Кыргызский государственный университет
им. И. Арабаева, г. Бишкек, Кыргызстан, uraxima84@mail.ru

FUNDAMENTALS OF WORKING WITH TENSORFLOW: SYNTAX AND BASIC OPERATIONS

©Sydykova M., ORCID: 0009-0002-0401-0997, SPIN-code: 8614-0079, Ph.D., Kyrgyz National
University named after Jusup Balasagyn, Bishkek, Kyrgyzstan, mb_sydykova@mail.ru

©Mambetova Zh., SPIN-code: 8121-0721, Ph.D., Kyrgyz National University named after I.
Arabaev, Bishkek, Kyrgyzstan, zarya.mambetova@mail.ru

©Ubaidyldaeva Zh., Ph.D., Kyrgyz National University named after I. Arabaev,
Bishkek, Kyrgyzstan, uraxima84@mail.ru

Аннотация. Статья посвящена основам работы с библиотекой TensorFlow — одной из наиболее популярных и функциональных платформ для создания и обучения моделей машинного и глубокого обучения. В работе подробно рассматриваются ключевые концепции, лежащие в основе вычислительной архитектуры TensorFlow, включая принципы формирования и обработки тензоров, их свойства, типы данных, размерности и способы представления в вычислительном графе. Особое внимание уделено базовым операциям над тензорами: арифметическим вычислениям, агрегирующими функциям, матричному умножению, а также механизму трансляции (broadcasting), который играет важную роль в оптимизации вычислений и построении моделей. Раскрываются различия между переменными и константами в TensorFlow, их назначение в процессе обучения нейронных сетей и методы изменения или обновления значений тензоров. В статье акцентируется внимание на том, как структуры данных TensorFlow позволяют эффективно выполнять автоматическое дифференцирование и оптимизацию параметров моделей. Материал сопровождается практическими упражнениями и примерами кода, ориентированными на начинающих исследователей и студентов, осваивающих технологии машинного обучения. Представленные примеры демонстрируют создание простейших вычислительных моделей, использование основных операций TensorFlow и связь между теоретическими принципами и их программной реализацией. Статья может служить вводным пособием для тех, кто делает первые шаги в изучении TensorFlow и хочет получить увереные практические навыки работы с этой библиотекой.

Abstract. The article is devoted to the fundamentals of working with the TensorFlow library, one of the most popular and functional platforms for creating and training machine learning and deep learning models. The paper provides a detailed examination of the key concepts underlying the computational architecture of TensorFlow, including the principles of forming and processing tensors, their properties, data types, dimensions, and ways of representing them within the computational

graph. Special attention is given to basic tensor operations such as arithmetic computations, aggregation functions, matrix multiplication, and the broadcasting mechanism, which plays an important role in optimizing computations and building models. The article further explains the differences between variables and constants in TensorFlow, their roles in the process of training neural networks, and the methods used to modify or update tensor values. Emphasis is placed on how TensorFlow's data structures enable efficient automatic differentiation and optimization of model parameters. The material is accompanied by practical exercises and code examples designed for beginning researchers and students learning machine learning technologies. The presented examples demonstrate the creation of simple computational models, the use of core TensorFlow operations, and the connection between theoretical principles and their programmatic implementation. The article can serve as an introductory guide for those taking their first steps in studying TensorFlow and seeking to develop confident practical skills with this library.

Ключевые слова: тензоры, матричное умножение, трансляция, функции активации, обучение модели, машинное обучение.

Keywords: tensors, matrix multiplication, broadcasting, activation functions, model training, machine learning.

TensorFlow — это библиотека с открытым исходным кодом, разработанная компанией Google и предназначенная для построения и обучения моделей машинного обучения и нейронных сетей различной сложности.

Основным объектом работы в TensorFlow является тензор — многомерный массив данных, на котором выполняются все необходимые вычисления. Тензоры могут иметь различную размерность и тип данных, что делает их универсальными для представления как числовой информации, так и структурированных данных, изображений, аудио- и видеопотоков. Библиотека предоставляет широкий спектр инструментов для автоматизации вычислительных процессов, что позволяет ускорять обучение моделей и обрабатывать большие объёмы данных.

TensorFlow поддерживает вычисления на центральных процессорах (CPU), графических процессорах (GPU) и тензорных процессорах (TPU), что делает её эффективным инструментом для работы с как небольшими, так и крупномасштабными задачами глубокого обучения [1, 2].

Одним из значительных преимуществ TensorFlow является высокая интеграция с другими библиотеками и инструментами Python. Например, совместное использование с NumPy позволяет легко преобразовывать и обрабатывать данные, Pandas обеспечивает удобную работу с табличными данными, а Matplotlib и Seaborn используются для визуализации результатов обучения моделей.

Кроме того, TensorFlow поддерживает Keras API — высокоуровневый интерфейс, упрощающий построение нейронных сетей, настройку функций активации, оптимизаторов и функций потерь [3].

TensorFlow широко применяется в разнообразных областях: от анализа изображений и видео с использованием сверточных нейронных сетей (CNN), до обработки текста и речи с применением рекуррентных сетей (RNN) и трансформеров. Библиотека позволяет строить как простые модели линейной регрессии и классификации, так и сложные глубокие нейронные сети для решения реальных практических задач [12].

Помимо базовых возможностей, TensorFlow предоставляет инструменты для оптимизации и мониторинга процессов обучения, такие как TensorBoard, который позволяет отслеживать метрики точности и потерь, визуализировать структуру модели, градиенты и веса. Также существуют расширенные возможности для работы с мобильными устройствами (TensorFlow Lite), веб-приложениями (TensorFlow.js) и построением производственных пайплайнов машинного обучения (TensorFlow Extended, TFX) [6, 13].

Таким образом, TensorFlow является мощной, гибкой и масштабируемой платформой, позволяющей специалистам по машинному обучению и искусственному интеллекту создавать, обучать и внедрять модели различного уровня сложности, а также экспериментировать с новыми архитектурами и методами обучения.

Освоение её базовых концепций — тензоров, операций, переменных, функций активации и оптимизаторов — служит прочной основой для дальнейшего профессионального развития и успешной реализации проектов в области искусственного интеллекта [4].

1. Тензоры. Тензор — это многомерный массив, характеризующийся формой (*shape*), рангом (*rank*) и типом данных (*dtype*). Скаляр (0D) — одно число, например, 5. Вектор (1D) — одномерный массив [1-3]. Матрица (2D) — двумерный массив, [1-4]. Тензоры 3D и выше — используются для изображений, видео и временных рядов. Создание тензоров:

```
import tensorflow as tf
scalar = tf.constant(5)
vector = tf.constant([1-3])
matrix = tf.constant([[1-4]])
zeros = tf.zeros((2, 3))
ones = tf.ones((3, 2))
rand_uniform = tf.random.uniform((2,2), 0, 1)
rand_normal = tf.random.normal((2,2), 0, 1)
```

Манипуляции с формой:

```
tensor = tf.constant([1-6])
reshaped = tf.reshape(tensor, [2, 3])
```

2. Базовые операции с тензорами [11]. TensorFlow поддерживает арифметические операции, линейную алгебру и трансляцию (broadcasting).

Арифметические операции: `a = tf.constant([1-3]) b = tf.constant([4-6]) c = tf.add(a,b) # [5, 7, 9].`

Broadcasting: позволяет выполнять операции над тензорами разной формы: `a = tf.constant([1-4]) b = tf.constant([1, 2]) c = a + b # [2, 4, 6]`

Матричное умножение: `a = tf.constant([[1,2],[3,4]]) b = tf.constant([[5,6],[7,8]]) c = tf.matmul(a,b)`

3. Переменные и константы [10].

Константы (`tf.constant`) — неизменяемые тензоры.

Переменные (`tf.Variable`) — значения которых могут изменяться в процессе обучения модели. `x = tf.Variable([1.0,2.0,3.0]) x.assign([4.0,5.0,6.0])`

4. Математические функции. TensorFlow содержит функции для статистики и линейной алгебры: `tf.reduce_sum()`, `tf.reduce_mean()` — сумма и среднее. `tf.square()`, `tf.sqrt()` — возведение в квадрат и корень. `tf.matmul()` — матричное умножение.

5. Практические упражнения [9]. Создать матрицу 3x3 случайных чисел и найти её транспонированную матрицу. Умножить две матрицы 2x2. Создать переменную и изменить её значение с помощью арифметических операций.

Нормализация тензора:

```
def normalize(tensor):  
    mean = tf.reduce_mean(tensor)  
    std = tf.math.reduce_std(tensor)  
    return (tensor - mean)/std
```

6. Создание моделей TensorFlow позволяет строить модели от простой линейной регрессии до глубоких нейронных сетей.

Пример нейронной сети:

```
from tensorflow.keras.models import Sequential  
from tensorflow.keras.layers import Dense  
model = Sequential([  
    Dense(16, activation='relu', input_shape=(10,)),  
    Dense(8, activation='relu'),  
    Dense(1, activation='sigmoid')  
])  
model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])  
model.fit(X_train, y_train, epochs=20, batch_size=32)
```

Функции активации: ReLU, Sigmoid, Softmax, Tanh.

Оптимизация и функции потерь: SGD, Adam, RMSProp; MSE, Binary/Categorical Crossentropy.

TensorBoard для мониторинга:

```
from tensorflow.keras.callbacks import TensorBoard  
tensorboard_cb = TensorBoard(log_dir='./logs')  
model.fit(X_train, y_train, epochs=20, callbacks=[tensorboard_cb])
```

8. Расширенные возможности TensorFlow

- CNN — сверточные нейронные сети для изображений.
- RNN и LSTM — для последовательных данных.
- TensorFlow Lite и TensorFlow.js — мобильные и веб-приложения.
- TFX — построение промышленного ML-пайплайна.

Пример CNN:

```
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten  
model = Sequential([  
    Conv2D(32,(3,3),activation='relu',input_shape=(28,28,1)),  
    MaxPooling2D((2,2)),  
    Flatten(),  
    Dense(128,activation='relu'),  
    Dense(10,activation='softmax')  
])  
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
```

Выход

Освоение базовых концепций TensorFlow — работы с тензорами, операциями, переменными, функциями активации и оптимизаторами — является первым и необходимым шагом для успешного применения этой библиотеки в задачах машинного и глубокого обучения. Понимание структуры тензоров, их свойств, способов изменения размерностей и типов данных позволяет эффективно управлять вычислительными процессами и подготавливать данные для обучения моделей. Базовые операции с тензорами, такие как

арифметические вычисления, матричное умножение и трансляция, формируют фундамент, необходимый для построения как простых, так и сложных вычислительных моделей [7, 11].

Дальнейшее изучение TensorFlow открывает возможности работы с более сложными архитектурами нейронных сетей, включая сверточные и рекуррентные сети, а также гибридные модели для анализа изображений, текста и временных рядов. Освоение функций оптимизации, методов регуляризации, выбора функций потерь и стратегий обучения на GPU и TPU позволяет значительно ускорять процессы обучения и повышать точность моделей. Интеграция с другими инструментами анализа данных, такими как NumPy, Pandas, Matplotlib, а также с платформами визуализации и мониторинга (TensorBoard) способствует более глубокому пониманию работы моделей и их корректной настройки [8].

Практические навыки, полученные при изучении базовых операций и построении первых моделей, позволяют создавать собственные проекты в области искусственного интеллекта, начиная от простых задач классификации и регрессии до сложных систем прогнозирования и обработки данных в реальном времени. Использование TensorFlow помогает не только автоматизировать вычислительные процессы, но и экспериментировать с различными архитектурами и методами обучения, формируя у специалистов критическое мышление и навыки анализа результатов. Кроме того, приобретённые знания служат прочной основой для дальнейшего профессионального развития: изучение современных методов глубокого обучения, работы с большими данными и создания интеллектуальных систем становится доступным и последовательным. TensorFlow позволяет адаптировать обучение к индивидуальным задачам, обеспечивая гибкость и масштабируемость при разработке проектов в промышленной, научной и образовательной сферах.

Таким образом, освоение базовых понятий и практических навыков работы с TensorFlow является необходимым этапом подготовки специалистов в области искусственного интеллекта, закладывает прочный фундамент для построения сложных вычислительных моделей, способствует развитию аналитических и проектных компетенций и открывает широкие возможности для реализации инновационных проектов на основе машинного и глубокого обучения.

Список литературы:

1. Гудфеллоу И., Бенджио Й., Курвиль А. Глубокое обучение. М., 2018. 652 с.
2. Нильсон Н. Искусственный интеллект: современный подход. М., 2020. 1152 с.
3. Рашка С., Мирджалили В. Python и машинное обучение. СПб.: Питер, 2021. 464 с.
4. Кудинов Д. А. Применение глубокого обучения для анализа медицинских изображений // Информационные технологии и вычислительные системы. 2021. №2. С. 34-41.
5. Блинов А. В., Пастухов А. А. Методы глубокого обучения в интеллектуальных системах управления // Вестник компьютерных и информационных технологий. 2020. №7. С. 12-19.
6. Brown T., Mann B., Ryder N., Subbiah M., Kaplan J. D., Dhariwal P., Amodei D. Language models are few-shot learners // Advances in neural information processing systems. 2020. V. 33. P. 1877-1901. <https://doi.org/10.48550/arXiv.2005.14165>
7. He K., Zhang X., Ren S., Sun J. Deep residual learning for image recognition // Proceedings of the IEEE conference on computer vision and pattern recognition. 2016. P. 770-778. <https://doi.org/10.48550/arXiv.1512.03385>
8. Sutton R. S., Barto A. G. Reinforcement Learning: An Introduction. 2nd ed. Cambridge, MA: MIT Press, 2018. 552 p.

9. Сыдыкова М. Б., Сагыналиева Г. А. Применение информационных и инновационных технологий в устойчивом развитии образования // Тенденции устойчивого развития образования в условиях глобализации: Сборник материалов международной научно-практической конференции. Новосибирск, 2023. С. 309-316.
10. Касымова Т. Д., Сыдыкова М. Б., Жапарова З. А. Применение искусственного интеллекта в математике: научный и социальный аспекты // Бюллетень науки и практики. 2023. Т. 9. №6. С. 32-37. <https://doi.org/10.33619/2414-2948/91/03>
11. Касымова Т. Д., Сыдыкова М. Б., Жапарова З. А. Информационные технологии в базах данных: роль и особенности применения // Бюллетень науки и практики. 2023. Т. 9. №6. С. 483-487. <https://doi.org/10.33619/2414-2948/91/57>
12. Казиева Г. К., Сыдыкова М. Б., Акматов А. М. Совершенствование самостоятельной работы студентов на курсах математики с использованием it-технологий // Современные направления психолого-педагогического сопровождения детства: международный опыт: Сборник материалов всероссийской научно-практической конференции. Новосибирск, 2022. С. 43-50.
13. Турдакунова А. С., Сыдыкова М. Б. Применение дистанционной технологии в образовании // Наука, новые технологии и инновации Кыргызстана. 2021. №10. С. 244-247.

References:

1. Gudfellow, I., Bengio, I., & Courville, A. (2018). Glubokoe obuchenie. Moscow. (in Russian).
2. Nil'son, N. (2020). Iskusstvennyi intellekt: sovremennyi podkhod. Moscow. (in Russian).
3. Rashka, S., & Mirdzhalili, V. (2021). Python i mashinnoe obuchenie. St.Petersburg. (in Russian).
4. Kudinov, D. A. (2021). Primenenie glubokogo obucheniya dlya analiza meditsinskikh izobrazhenii. *Informatsionnye tekhnologii i vychislitel'nye sistemy*, (2), 34-41. (in Russian).
5. Blinov, A. V., & Pastukhov, A. A. (2020). Metody glubokogo obucheniya v intellektual'nykh sistemakh upravleniya. *Vestnik kompyuternykh i informatsionnykh tekhnologii*, (7), 12-19. (in Russian).
6. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33, 1877-1901. <https://doi.org/10.48550/arXiv.2005.14165>
7. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778). <https://doi.org/10.48550/arXiv.1512.03385>
8. Sutton, R. S., & Barto, A. G. (2018). Reinforcement Learning: An Introduction. 2nd ed. Cambridge, MA: MIT Press.
9. Sydykova, M. B., & Sagynalieva, G. A. (2023). Primenenie informatsionnykh i innovatsionnykh tekhnologii v ustoichivom razvitiu obrazovaniya. In *Tendentsii ustoichivogo razvitiya obrazovaniya v usloviyakh globalizatsii: Sbornik materialov mezhdunarodnoi nauchno-prakticheskoi konferentsii*, Novosibirsk, 309-316. (in Russian).
10. Kasymova, T., Sydykova, M., & Zhaparova, Z. (2023). The Use of Artificial Intelligence in Mathematics: Scientific and Social Aspects. *Bulletin of Science and Practice*, 9(6), 32-37. (in Russian). <https://doi.org/10.33619/2414-2948/91/03>
11. Kasymova, T., Sydykova, M., & Zhaparova, Z. (2023). Information Technologies in Databases: Role and Application Features. *Bulletin of Science and Practice*, 9(6), 483-487. (in Russian). <https://doi.org/10.33619/2414-2948/91/57>

12. Kazieva, G. K., Sydykova, M. B., & Akmatov, A. M. (2022). Sovershenstvovanie samostoyatel'noi raboty studentov na kursakh matematiki s ispol'zovaniem it-tehnologii. In *Sovremennye napravleniya psichologo-pedagogicheskogo soprovozhdeniya detstva: mezhdunarodnyi opyt: Sbornik materialov vserossiiskoi nauchno-prakticheskoi konferentsii*, Novosibirsk, 43-50. (in Russian).

13. Turdakunova, A. S., & Sydykova, M. B. (2021). Primenenie distantsionnoi tekhnologii v obrazovanii. *Nauka, novye tekhnologii i innovatsii Kyrgyzstana*, (10), 244-247. (in Russian).

Поступила в редакцию
24.11.2025 г.

Принята к публикации
30.11.2025 г.

Ссылка для цитирования:

Сыдыкова М. Б., Мамбетова З. Д., Убайдылдаева Ж. А. Основы работы с TensorFlow: синтаксис и базовые операции // Бюллетень науки и практики. 2026. Т. 12. №1. С. 104-110. <https://doi.org/10.33619/2414-2948/122/13>

Cite as (APA):

Sydykova, M., Mambetova, Zh., & Ubaidyldaeva, Zh. (2026). Fundamentals of Working with TensorFlow: Syntax and Basic Operations. *Bulletin of Science and Practice*, 12(1), 104-110. (in Russian). <https://doi.org/10.33619/2414-2948/122/13>