

УДК 004.8

## ЭКСТРАКЦИЯ ПРИЗНАКОВ В МОДЕЛЯХ ПОСЛЕДОВАТЕЛЬНОГО ГЛУБОКОГО ОБУЧЕНИЯ

Пылов П.А., студент гр. ИТм-201, 2 курс  
Протодияконов А. В., к.т.н., доцент  
Кузбасский государственный технический университет  
имени Т.Ф. Горбачева

Аннотация на русском языке: Экстракция признаков всегда была и остается особо важной задачей для сепарации внутренних признаков данных.

Правильная экстракция признаков позволяет максимально повысить эффективность решающей модели искусственного интеллекта и значительно утвердить позиции прикладного решения в предметной области.

*Ключевые слова:* искусственный интеллект; машинное обучение; глубокое обучение; искусственная нейронная сеть.

*Key words:* artificial intelligence; machine learning; deep learning; artificial neural network.

Как только последовательная модель построена, она ведет себя как функциональная модель API. Это означает, что каждый слой имеет входной и выходной атрибут.

Эти атрибуты можно использовать для выполнения аккуратных действий, таких как быстрое создание модели, которая извлекает выходные данные всех промежуточных слоев в последовательной модели (рисунок 1):

```
initial_model = keras.Sequential([
    keras.Input(shape=(250, 250, 3)),
    layers.Conv2D(32, 5, strides=2, activation="relu"),
    layers.Conv2D(32, 3, activation="relu"),
    layers.Conv2D(32, 3, activation="relu"),
])
feature_extractor = keras.Model(
    inputs=initial_model.inputs,
    outputs=[layer.output for layer in initial_model.layers],
)

x = tf.ones((1, 250, 250, 3))
features = feature_extractor(x)
```

## Рисунок 1 – Быстрое создание модели на основе Features Extraction

Аналогичный пример, в котором объекты извлекаются только из одного слоя (рисунок 2):

```
initial_model = keras.Sequential([
    keras.Input(shape=(250, 250, 3)),
    layers.Conv2D(32, 5, strides=2, activation="relu"),
    layers.Conv2D(32, 3, activation="relu", name="my_intermediate_layer"),
    layers.Conv2D(32, 3, activation="relu"),
])
feature_extractor = keras.Model(
    inputs=initial_model.inputs,
    outputs=initial_model.get_layer(name="my_intermediate_layer").output,
)
x = tf.ones((1, 250, 250, 3))
features = feature_extractor(x)
```

Рисунок 2 – Извлечение признаков по объектам по одному слою

## Литература:

1. Пылов П.А., Протоdjяконов А.В. Data Cleaning или краеугольная основа General Artificial Intelligence // Инновации. Наука. Образование. 2020. № 23. С. 208 – 212.
2. Пылов П.А., Протоdjяконов А.В. Многомерные нормальные распределения величин наборов данных на примере визуализации частных случаев // Инновации. Наука. Образование. 2020. № 22. С. 352 – 360.
3. Пылов П.А., Протоdjяконов А.В. Масштабирование и нормализация как основа Data Cleaning // Инновации. Наука. Образование. 2020. № 23. С. 225 – 233.
4. Пылов П.А., Протоdjяконов А.В. Способ нечёткого сопоставления в корректировке несогласованного ввода данных алгоритмов artificial intelligence // Инновации. Наука. Образование. 2020. № 23. С. 240 – 246.

5. Пылов П.А., Протодяконов А.В. Особенности построения случайным образом сгенерированных наборов данных с несколькими метками классификации // Инновации. Наука. Образование. 2020. № 22. С. 361 – 363.
6. Пылов П.А., Протодяконов А.В. МНОГОМЕРНЫЕ НОРМАЛЬНЫЕ РАСПРЕДЕЛЕНИЯ ВЕЛИЧИН НАБОРОВ ДАННЫХ НА ПРИМЕРЕ ВИЗУАЛИЗАЦИИ ЧАСТНЫХ СЛУЧАЕВ // Инновации. Наука. Образование. 2020. № 22. С. 352 – 360.
7. Пылов П.А., Протодяконов А.В. МЕТОДОЛОГИИ ЧАСТИЧНОЙ ЗАВИСИМОСТИ В РАСШИРЕННОМ ПОСТРОЕНИИ ГРАФИКОВ // Инновации. Наука. Образование. 2020. № 22. С. 349 – 351.
8. Пылов П.А., Протодяконов А.В. ИССЛЕДОВАНИЕ ПРОБЛЕМЫ "OVERFITTING" В АЛГОРИТМАХ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА НА ПРОТОТИПЕ МОДЕЛИ РЕГРЕССИИ ДЕРЕВА РЕШЕНИЙ // Инновации. Наука. Образование. 2020. № 22. С. 346 – 348.
9. Пылов П.А., Протодяконов А.В. ВЕРОЯТНОСТНЫЕ КАЛИБРОВКИ КЛАССИФИКАТОРОВ В МАШИННОМ ОБУЧЕНИИ // Инновации. Наука. Образование. 2020. № 22. С. 333 – 336.