Модель ML глазами юриста:

1. Правовая природа
2. Охрана прав
3. Ответственность

Мельникова Елена Николаевна
Юрисконсульт и независимый исследователь, магистр права ИТМО

Ереван - 2023

Модель ML – это продукт, нуждающийся в правовой охране

Что мы намерены охранять?

Каким объектом гражданских прав является модель ML?

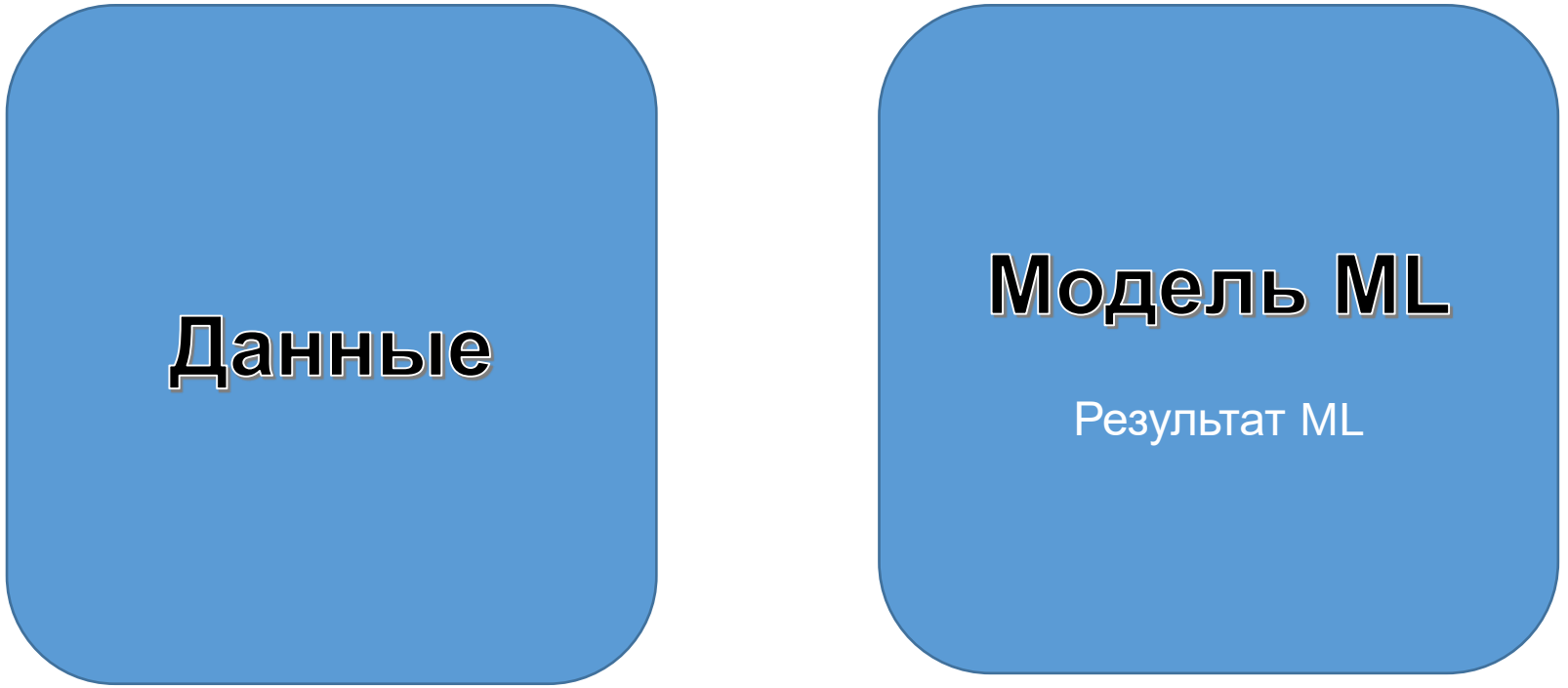
- программа для ЭВМ?

- ноу-хау?

Что влияет на правовую охрану модели ML?

- Подход к пониманию модели
- Способ обучения (контролируемое/неконтролируемое)
- Стадия жизненного цикла модели

Элементы процесса ML



Подходы
Модель ML – это:

Узкий подход
Алгоритм обучения

Сторонниками «узкого» подхода являются: *Барский А. Б.* Нейронные сети: распознавание, управление, принятие решений. М.: Финансы и статистика, 2004. Прикладные информационные технологии. С. 8, 16; *Аникин С. Н., Нажимов Р. А., Позднякова К. Е., Сомов Ю. И.* Актуальные вопросы применения искусственного интеллекта в деятельности таможенных органов // Вестник Российской таможенной академии. 2021. No. 2. Доступ из СПС «Консультант Плюс»

Широкий подход
1. Алгоритм обучения
2. Сценарий обучения
3. Сценарий управления

Brownlee J. Difference Between Algorithm and Model in Machine Learning // Machine Learning Algorithms. August 19, 2020. URL: https://machinelearningmastery.com/difference-between-algorithm-and-model-in-machine-learning. (дата обращения: 07.09.2022); Руководство Microsoft. Учебник по обучению первых моделей машинного обучения (часть 2 из 3) 10.06.2022. URL: https://docs.microsoft.com/ru-ru/azure/machine-learning/tutorial-1st-experiment-sdk-train#create-training-scripts.

Отказ в правовой охране

Правовая охрана

Модель ML:

Алгоритм обучения

Слои нейронной сети

model = k. Sequential ()

Сценарий управления

```
# run-pytorch.py
from azureml.core import Workspace
from azureml.core import Experiment
from azureml.core import Environment
from azureml.core import ScriptRunConfig

if __name__ == "__main__":
    ws = Workspace.from_config()
    experiment = Experiment(workspace=ws, name='day1-experiment-train')
    config = ScriptRunConfig(source_directory='./src',
                             script='train.py',
                             compute_target='cpu-cluster')

    # set up pytorch environment
    env = Environment.from_conda_specification(
        name='pytorch-env',
        file_path='pytorch-env.yml'
    )
    config.run_config.environment = env

    run = experiment.submit(config)

    aml_url = run.get_portal_url()
    print(aml_url)
```

Сценарий обучения

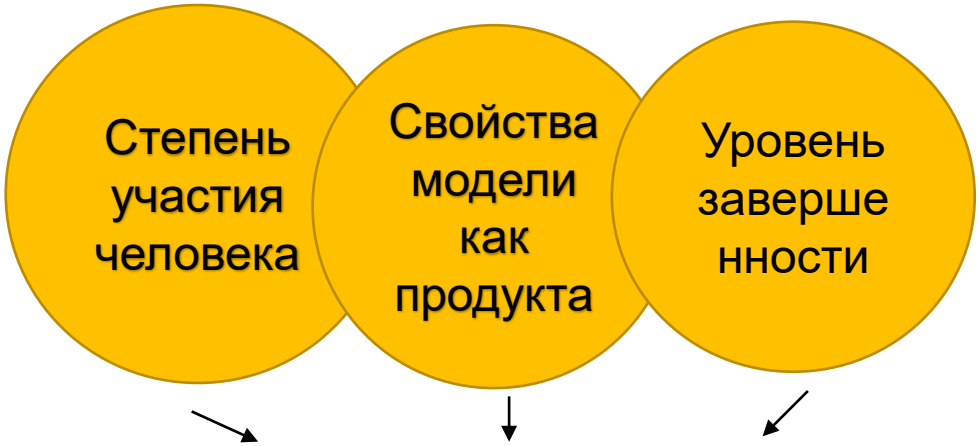
```
    for vector_raw in list(zip(*vectors)):
        vector = []
        for element in vector_raw:
            for e in element:
                vector.append(e)
        formatted.append(vector)
    return formatted


supervised = make_supervised(data_frame)
encoded_inputs = encode(supervised["inputs"])
encoded_outputs = encode(supervised["outputs"])

train_x = encoded_inputs[:600]
train_y = encoded_outputs[:600]

test_x = encoded_inputs[600:]
test_y = encoded_outputs[600:]

model = k.Sequential()
model.add(k.layers.Dense(units=5, activation="relu"))
model.add(k.layers.Dense(units=1, activation="sigmoid"))
model.compile(loss="mse", optimizer="sgd", metrics=["accuracy"])
```

Способы обучения модели ML

«с учителем» (контролируемое)

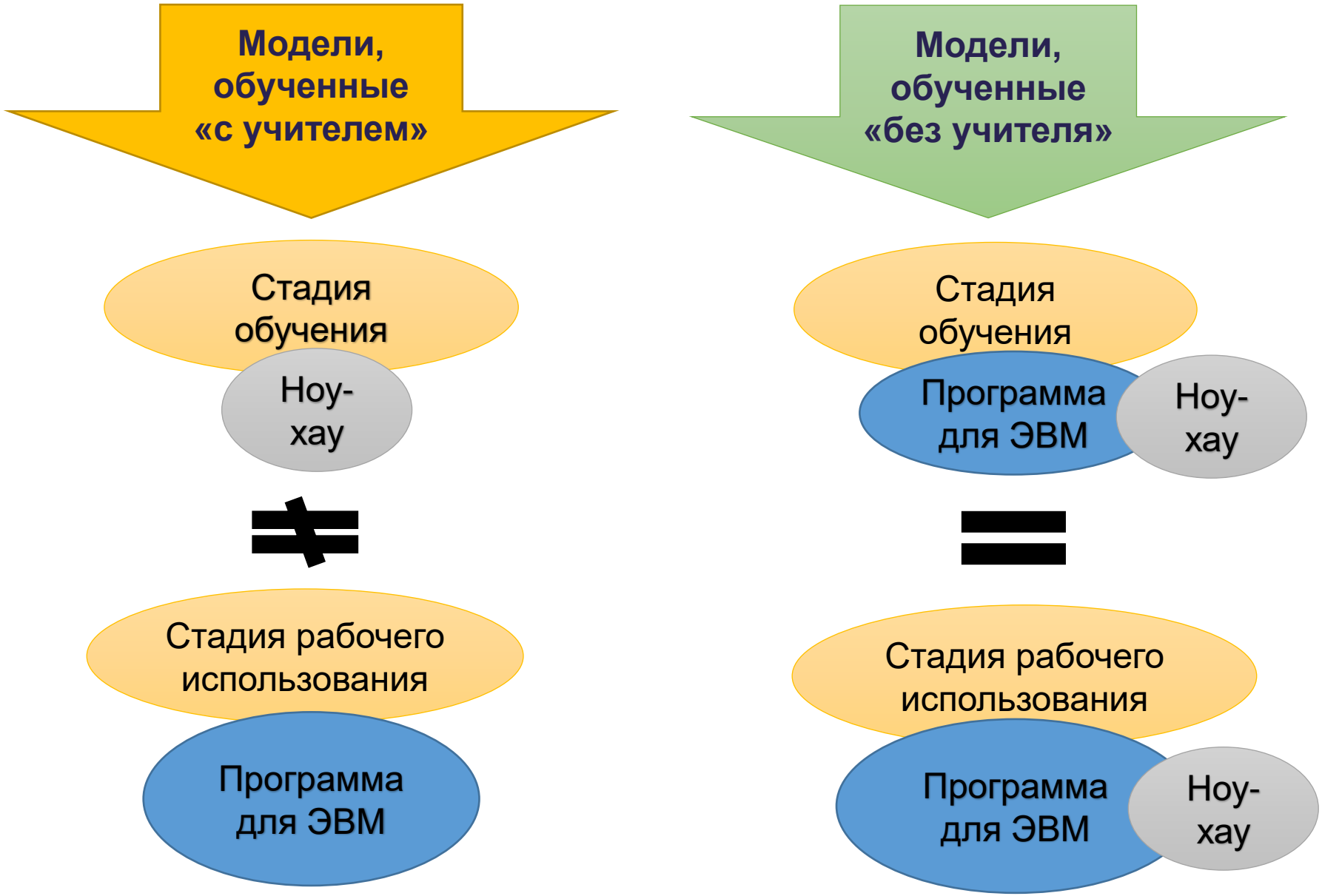
«без учителя» (неконтролируемое)

Способ обучение определяет:



Правовая природа модели ML → Правовая охрана модели ML

Правовая природа модели ML



Патенты на модели ML - изобретения

Способ обучения искусственной нейронной сети

Способ и система обучения алгоритма ML

RU 2504006 C1

RU 2566979 C1

RU 2723270 C1

RU 2649792 C2

Условия патентоспособности технического результата как изобретения:

Технический характер решения

новизна

изобретательский уровень

промышленная применимость

«Технический характер» патентуемого решения

Позиция руководителя отделения физики и прикладной механики ФИПС Сальникова Михаила Юрьевича [Практика патентования IT-технологий // ИС. Промышленная собственность. 2021. N 10. С. 43 – 50]:

«Необходимый для патентной защиты технический результат имеет место тогда, когда он достигается не в силу особенностей вычислительных средств, а в силу особенностей реализуемого этими решениями алгоритма»

Вывод:
контролируемые способы обучения алгоритма не являются техническими решениями и не патентоспособны

Критерий новизны

Алгоритмы обучения относятся к

уровню техники

Новизна достигается за счет сценария обучения

Критерий изобретательского уровня

Изобретение имеет изобретательский уровень, если для специалиста оно явным образом не следует из уровня техники (п. 2 ст. 1350 ГК РФ)

Конвенционный уровень знаний специалиста для определения изобретательского уровня модели МО не определен

Источник: WIPO

Критерий промышленной применимости

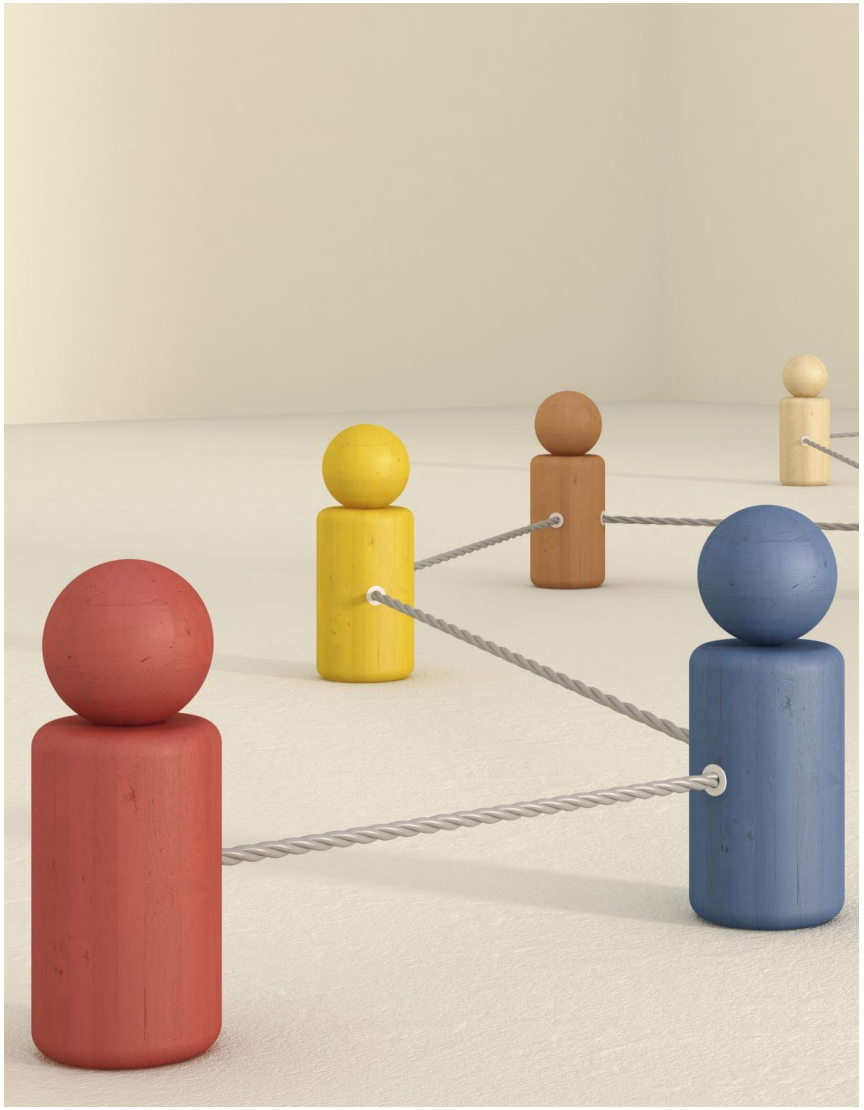
Возможность воспроизводимости модели ML с теми же характеристиками, которые указаны в описании изобретения

Патентование модели ML: общие выводы

~~Модели ML, обученные «с учителем»~~

«Самообучающиеся» модели ML

Способы сбора и обработки информации



ДИСКУССИОННЫЙ ВОПРОС:

распределение вреда, причиненного рабочим использованием модели ML

Что нужно знать для поиска причинителя вреда?

- Кто создал модель?
- Кто осуществляет рабочее использование?
- Кто осуществляет эффективный контроль?

Субъектный состав участников процесса ML

Лицо A владеет **[данными]**

Лицо B создает **[алгоритм обучения]**

Лицо C реализует алгоритм обучения **[создает сценарий обучения]**

Лицо D обучает алгоритм с помощью сценария обучения **[создает модель ML]**

Кто причинитель вреда?

Участники рабочего использования модели ML

лицо A - данные

лицо D - создает модель ML

лицо X – использует модель ML

Кто осуществляет эффективный контроль на стадии рабочего использования модели?

изменение сценария обучения = создание иной модели = модификация

ввод данных = изменение параметров = адаптация программы

Переподготовка обученной / предобученной модели: когда возникает новая модель ML?

Что происходит с моделью при ее «дообучении» «с учителем»?

а) изменение только параметров модели (сценарий обучения остается неизменным) = **адаптация** программы

б) изменением не только параметров, но и сценария обучения = иная модель = **модификация**

Контролируемое обучение

Причинитель вреда - лицо, осуществившее эффективный контроль на стадии рабочего использования модели

Отвечает пользователь модели (лицо Х) при использовании модели способами:

- изменение сценария обучения = модификация модели

- ввод новых данных (изменение параметров, адаптация модели)

При использовании всегда происходит ввод данных

Пользователь отвечает всегда

Неконтролируемое обучение

Причинитель вреда - лицо, осуществившее эффективный контроль на стадии рабочего использования модели

Отвечает пользователь модели (лицо X) если:

- изменение сценария обучения = модификация модели

Отвечает разработчик модели (лицо D) если:

Пользователь не изменял сценарий обучения

Модель при автоматическом вводе данных, согласно заложенному в ней сценарию управления, выдает ошибочные прогнозы

Спасибо за внимание!

Мельникова Елена Николаевна

Юрисконсульт и независимый исследователь, магистр права

melnikova_elena5@mail.ru