

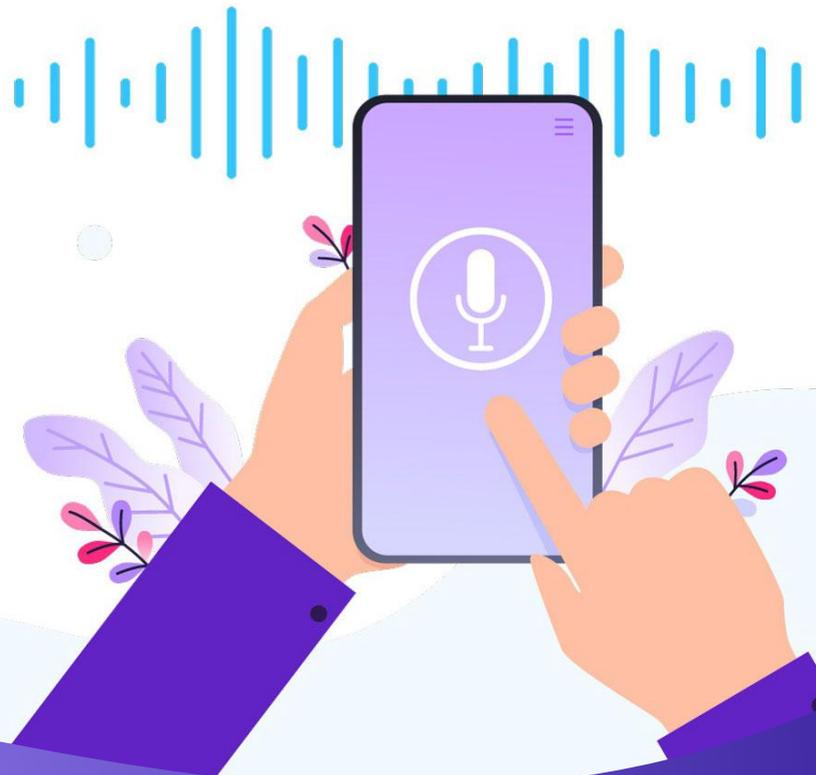
“Смарт - сайт”

Сайты нового поколения на основе нейросетей

- </> Использование нейросетей и машинного обучения
- 💬 Персонализированный контент
- 🎤 Голосовой интерфейс

Представляет собой сочетание систем:

- вопрос - ответная система;
- голосовой чат - бот;
- сайт с системой управления содержимым (CMS);



Принцип работы системы

Архитектура системы - микросервисная, представлена тремя системами, запущенными на разных портах:

API отвечает за взаимодействие с конечным пользователем, выводит HTML интерфейс, получает данные и отправляет текстовые данные в BERT API

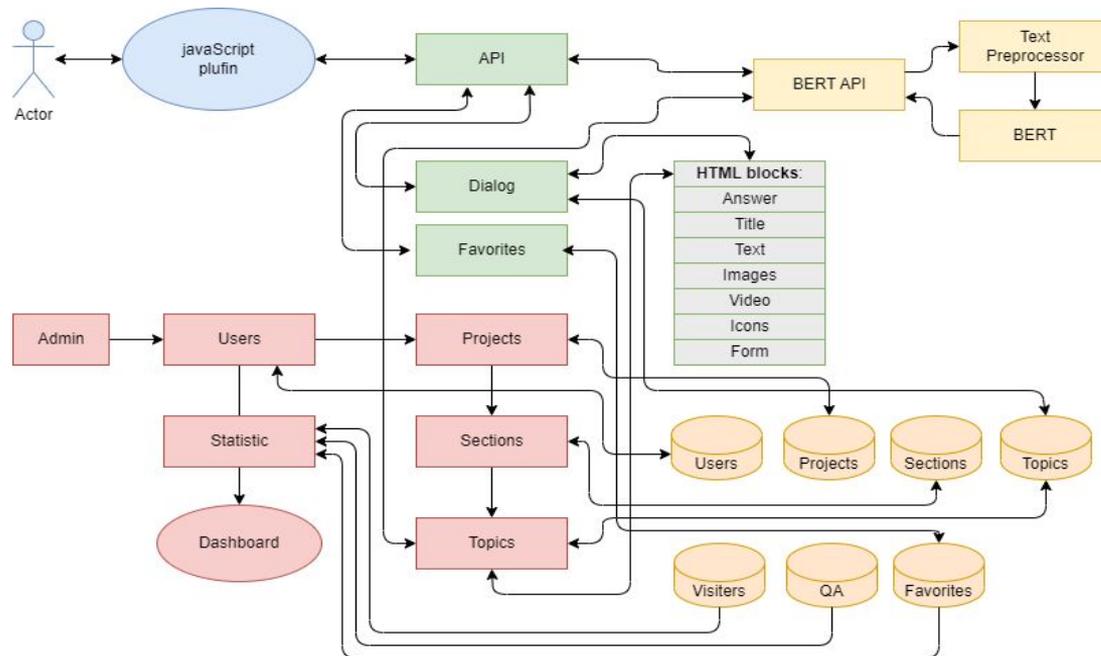
BERT API получает текстовые данные, которые сначала обрабатываются текстовым препроцессором и далее подаются в BERT модель, которая возвращает векторное представление запроса, размерностью (1, 768)

CMS система управления контентом позволяет создавать и управлять контентом сайта и получать информационно - статистические данные.

Производит управление следующими категориями:

- пользователи;
- проекты;
- разделы;
- топики;

Схема работы системы



Пример работы системы

Посмотреть пример работы системы можно на сайте 63s.ru. Система обучена отвечать на вопросы по теме "Создание и продвижение сайтов". На данный момент, количество топиков - около 50. Это очень мало, но мы постоянно работаем над увеличением кругозора системы.

Справа есть кнопка с микрофоном нажмите её.



Разрешите браузеру использовать звук.  Говорите.

Добавляйте нужные ответы в закладки, нажав галочку.



Система запомнит их и при следующем заходе на сайт ваша вкладка "Избранное" сохранится.

Продвижение сайта



Хостинг и домен



Создание сайта



Вывод с внешней стороны

Отключить ответ голосом

Добавить вопрос в закладки

Индикация распознавания речи



Цены на создание сайта

Стоимость создания сайта от 25 000 руб.



Свернуть окно



Ответ голосом

Портфолио



Стоимость интернет-магазина



Цена на создание сайта



Сохраненные вопросы в закладки

Стоимость создания сайта зависит от количества страниц, требованиям к оформлению, наличием индивидуальных компонентов.

Включает опции:

- Волновой дизайн.
- Создание элементов дизайна в photoshop
- Вместо портфолио — кейсы
- 100% валидация кода
- Проверка соответствия качества сайта по 20 параметрам
- 5 пакетов правок.
- Разработка: 3 рабочей недели.

Стоимость разработки: от 25 000 Р до 79 000 Р

Нейросеть

Изначально планировалось использовать предобученную BERT fine-tuning модель с текстовым препроцессором. Данный подход показал хорошую точность и монолитность модели - от обработки текстовых данных, до предсказания класса ответа.

Однако такой подход имеет существенные ограничения. Модель занимает около 3 гигабайт оперативной памяти. И для следующего проекта необходимо ещё столько же места.

Для нашей системы с большим количеством проектов - это nepозволительная роскошь.

Вместо тонкой настройки модели (fine-tuning), пришлось делать систему, в которой вектора мы получаем из предобученной BERT и далее подаём на нейросеть с классификацией. Точность модели существенно упала, однако, теперь у нас только BERT весит около 3 гигабайт, остальные модели до 10 мегабайт. Рассмотрим варианты решения ниже.

Загрузка модели BERT

Мы загрузим две модели, одну для предварительной обработки, а другую для кодирования. Ссылки на модели приведены ниже.

```
1 bert_preprocess = hub.KerasLayer("https://tfhub.dev/tensorflow/bert_multi_cased_preprocess/3")  
2 bert_encoder = hub.KerasLayer("https://tfhub.dev/tensorflow/bert_multi_cased_L-12_H-768_A-12/4", trainable=True)]
```

Инициализация слоев BERT

В приведенном ниже коде мы создаем входной слой, используя `tf.keras.layers.Input` метод. Мы будем использовать `preprocessed_text` качестве входных данных для этого слоя.

Затем `bert_encoder` функция преобразует предварительно обработанный текст в "embedding" векторы. Это будет выход этого слоя. Затем с выхода "outputs" они будут переданы в слой нейронной сети.



```
1 text_input = Input(shape=(), dtype=tf.string, name='text')
2 preprocessed_text = bert_preprocess(text_input)
3 outputs = bert_encoder(preprocessed_text)
```

Текстовый препроцессор, описание

Результатом предварительной обработки является пакет входных последовательностей фиксированной длины для кодера Transformer.

Входная последовательность начинается с одного маркера начала последовательности, за которым следуют сегменты с маркерами, каждый из которых заканчивается одним маркером конца сегмента. Оставшиеся позиции до `seq_length`, если таковые имеются, заполняются токенами заполнения. Если входная последовательность превышает `seq_length`, токенизированные сегменты в ней усекаются до префиксов примерно одинакового размера для точного соответствия.

Это `encoder_inputsdict` `mplex` тензоров `int32`, все с `shape [batch_size, seq_length]`, элементы которых представляют пакет входных последовательностей следующим образом:

- `"input_word_ids"`: содержит идентификаторы токенов входных последовательностей.
- `"input_mask"`: имеет значение 1 в позиции всех входных токенов, присутствующих до заполнения, и значение 0 для токенов заполнения.
- `"input_type_ids"`: имеет индекс входного сегмента, который породил входной токен в соответствующей позиции. Первый входной сегмент (индекс 0) включает маркер начала последовательности и его маркер конца сегмента. Второй сегмент (индекс 1, если он присутствует) включает маркер конца сегмента. Маркеры заполнения снова получают индекс 0.

Пример вектора на выходе энкодера



```
1 bert_encoder(p_text)
```

```
{'default': <tf.Tensor: shape=(1, 768), dtype=float32, numpy=
array([[ 3.32351178e-01, -1.72285944e-01,  3.29328150e-01,
        -3.54896188e-01, -2.01404229e-01,  4.13335651e-01,
         3.43536317e-01,  2.14050919e-01, -4.26788270e-01,
         3.59309286e-01, -2.98414007e-02, -3.05066019e-01,
        -2.86325306e-01, -2.36799479e-01,  1.55280441e-01,
        -4.72096235e-01,  8.08936656e-01,  2.51211345e-01,
         2.60708094e-01, -4.54398930e-01, -9.99970496e-01,
        -5.26440322e-01, -4.04504240e-01, -1.51830748e-01,
        -7.17329204e-01,  6.59313425e-02, -1.61634058e-01,
         2.14907348e-01,  4.19674635e-01, -2.97631681e-01,
        -2.77904645e-02, -9.99972284e-01,  8.55660498e-01,
```

Собираем модель, объединяем все слоу



```
1 text_input = Input(shape=(), dtype=tf.string, name='text')
2 preprocessed_text = bert_preprocess(text_input)
3 outputs = bert_encoder(preprocessed_text)
4
5 x = outputs['pooled_output']
6 x = Dense(768, activation='relu')(x)
7 x = Dropout(0.25)(x)
8 x_output = Dense(50, activation='softmax')(x)
```

Полная модель, fine-tuning



=====
Total params: 178,461,720
Trainable params: 178,461,719
Non-trainable params: 1

None

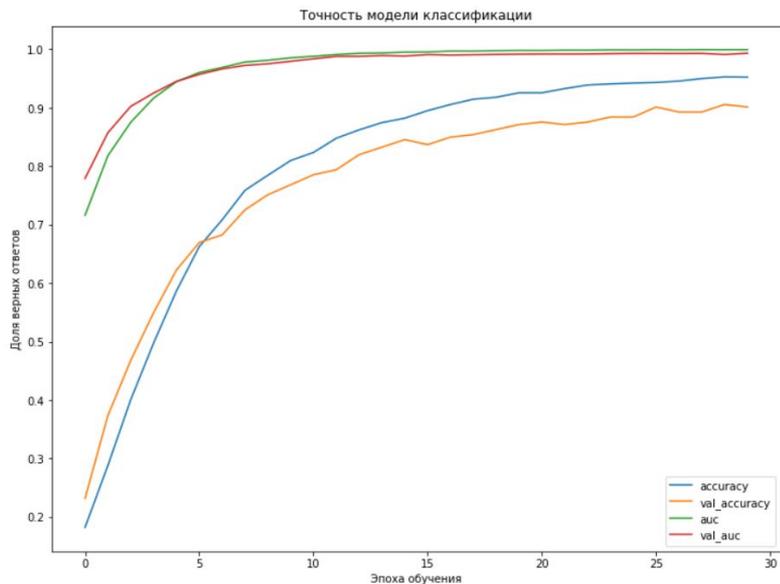
Компиляция модели

Следует обратить внимание на 2 момента:

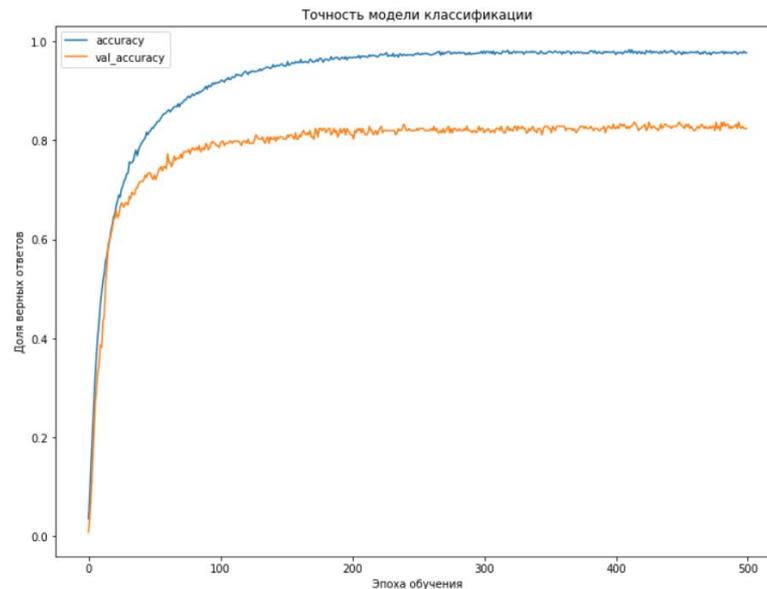
```
1 model.compile(loss='categorical_crossentropy', optimizer=Adam(learning_rate=(0.000005)), metrics=['accuracy', 'AUC'])
```

1. Шаг обучения крайне маленький, при большем значении видно явное переобучение модели
2. Дополнительная метрика "AUC", т.к. классы не сбалансированы

График обучения модели



Fine-tuning model, Learning_rate: 0.000002, 30 epochs
Train data: loss: 0.1510 - accuracy: 0.9657 - auc: 0.9996
Test data: loss: 0.3871 - accuracy: 0.9013 - auc: 0.9932



Classifier model: Learning_rate: 0.00001, 500 epochs
Точность (accuracy): 0.9804 / 0.8240

Сравнение вариантов нейросети

Показатели BERT fine-tuning модели:

Learning_rate: 0.000002, 30 epochs



Train data: loss: 0.1510 - accuracy: 0.9657 - auc: 0.9996
Test data: loss: 0.3871 - accuracy: 0.9013 - auc: 0.9932



Вес дополнительной модели на проект: 3 GB

Показатели модели с классификацией векторов BERT:

Learning_rate: 0.00001, 500 epochs



Train data: accuracy: 0.9804
Test data: accuracy: 0.8240



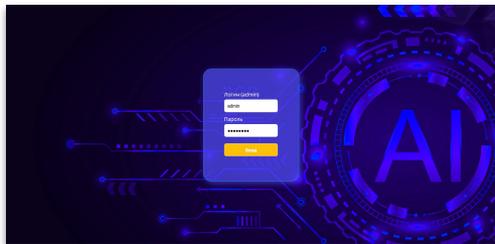
Вес дополнительной модели на проект: 0.01 GB

Система управления содержанием (CMS)

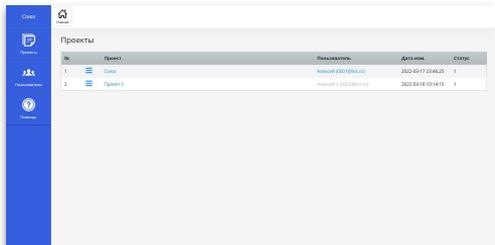
CMS позволяет управлять:

- пользователями;
- проектами;
- разделами;
- топиками;
- получать информационно - статистические данные;

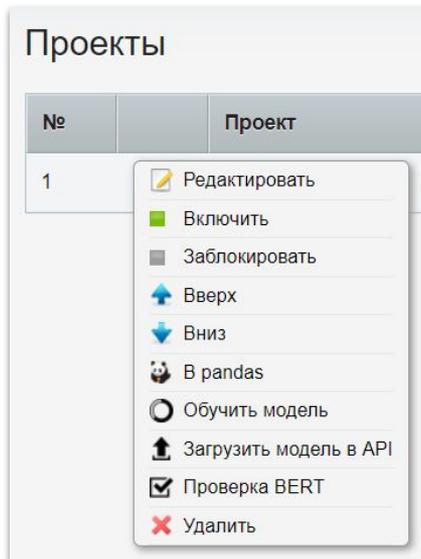
Элементы интерфейса



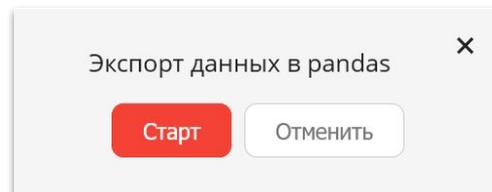
Вход в систему управления



Перечень всех проектов



Контекстное меню управления проектами



Наполненные проекты с классами и контентом можно выгрузить в датафрейм для анализа на pandas

Элементы интерфейса

№	CLS	Тема	Вопросы	Визитки	Даты окончания	Статус
1	0	Создание сайтов			2022-06-15 00:00:00	1
2	0	Разработка сайтов, цена			2022-06-14 11:40:00	1
3	1	Дизайн сайтов			2022-06-15 00:00:00	1
4	2	Платформа для создания сайтов			2022-06-15 00:01:44	1
5	3	Поддержка сайтов			2022-06-15 00:02:00	1
6	4	Портфолио сайтов			2022-06-15 00:14:30	1
7	5	Поддержка и обслуживание сайтов			2022-06-15 00:09:10	1
8	6	Сайт визитка			2022-06-15 00:04:10	1
9	7	Лендинг пейдж			2022-06-15 00:05:10	1
10	8	Корпоративный сайт			2022-06-14 12:12:22	1
11	9	Интернет магазин			2022-06-15 00:02:21	1
12	10	Интернет магазин - дизайн			2022-06-14 11:04:00	1
13	11	Интернет магазин - SEO/PPC			2022-06-14 11:14:00	1

Каждый проект наполняется темами

Редактировать тему

Тема Вопросы Контент

Тема

Порядок следования

Статус

Возможности редактирования

№	CLS	Тема
1	0	Создание сайтов
2	0	Разработка сайтов, цена
3	1	Дизайн сайтов
4	2	Платформа для создания сайта
5	3	Поддержка сайтов
6	4	Портфолио сайтов
7	5	Поддержка и обслуживание сайта
8	6	Сайт визитка
9	7	Лендинг пейдж
10	8	Корпоративный сайт
11	9	Интернет магазин

Контекстное меню управления темой

Редактировать тему

Тема Вопросы Контент

Вопрос

- Создание сайтов?
- Разработка сайтов?
- Изготовление сайтов?
- Сделать сайт?
- Сколько стоит сайт?
- Создание сайтов под ключ?
- Сайт цена?
- Сколько стоит разработка сайта?
- Создать сайт быстро?
- За сколько сделаете сайт?
- Почём сайт?
- Стоимость сайта?
- Сколько стоит сделать сайт?
- Сколько стоит создание сайта?
- Прайс цены на сайт?
- Сделать сайт дорого?

Сохранить Отменить

Каждая тема содержит список вопросов по которым нейросеть будет обучаться. Вопросы можно добавлять и удалять как по отдельности, так и списком. Система проводит проверку вопросов на уникальность и удаляет дубликаты.

Интерфейс управления топиками, закладка “контент”

Управление контентом

Возможности создания контента.
Для редактируемого текста подключается визуальный редактор CKEditor.

Ответ голосом

Редактируемый контент

Стоимость создания сайта от 25 000 руб.

Стоимость создания сайта зависит от количества страниц, требованиям к оформлению, наличию индивидуальных компонентов.

Включает опции:

- Волновой дизайн.
- Создание элементов дизайна в photoshop
- Вместо портфолио — кейсы
- 100% валидация кода
- Проверка соответствия качества сайта по 20 параметрам
- 5 планов дизайна

Обработка данных и вывод информации в терминале

Вывод информации в модальном окне и вывод логов операции в терминале

Домен	Пользователь
63s.ru	Алексей (0001@

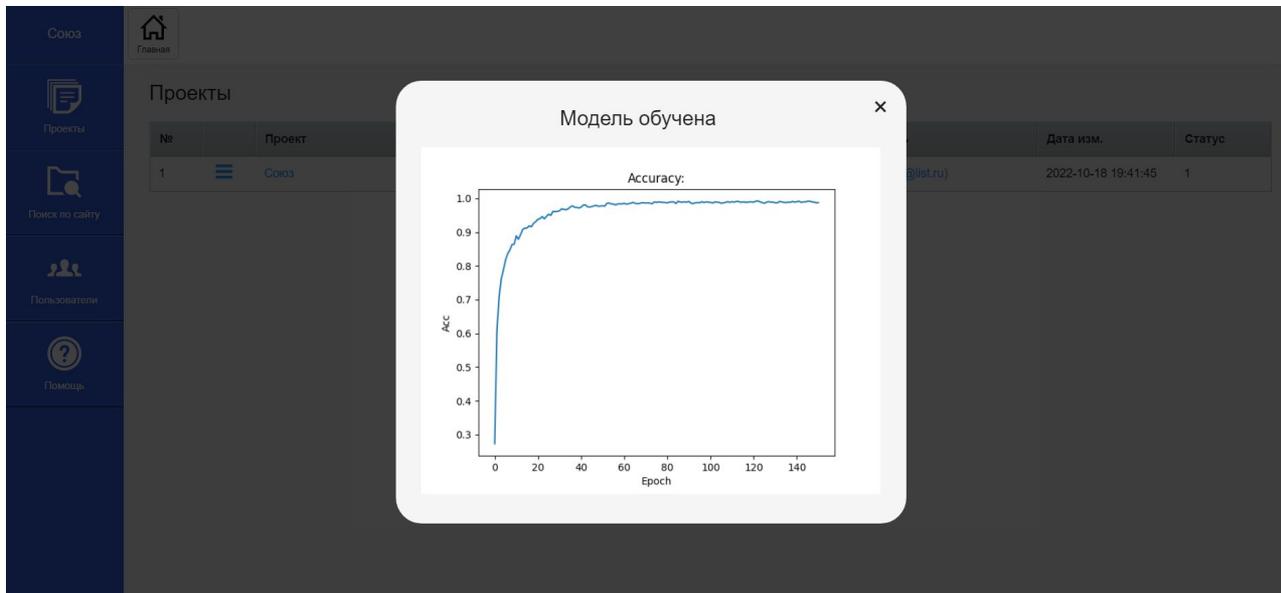
Начата обработка данных проекта 1

Обработано: 11 из 26

Топик: "Тильда создание сайтов", вопросов: 56

```
mc [root@c272]:/var/www/smart_sayt_r_us/...
ие сайтом;создание сайта на битрикс;создать сайт
на битрикс;установить битрикс на сайт;хостинг сай
та битрикс;шаблон сайта битрикс', 'status': 1, 'o
rdering': 12)
TEXT: 1 битрикс сайты
TEXT: 1с битрикс обмен с сайтом
TEXT: 1с битрикс управление сайтом
TEXT: 1с битрикс управление сайтом бизнес
TEXT: битрикс 24 официальный сайт
TEXT: битрикс 24 сайт
TEXT: битрикс 24 создать сайт
TEXT: битрикс добавить сайт
TEXT: битрикс конструктор сайтов
TEXT: битрикс сделать сайт
TEXT: битрикс система управления сайтом
TEXT: битрикс управление сайтом
TEXT: битрикс управление сайтом бизнес
TEXT: выгрузка на сайт битрикс
TEXT: интеграция битрикса с сайтом
TEXT: лицензия 1с битрикс управление сайтом
TEXT: магазин сайтов битрикс
TEXT: настройка сайта 1с битрикс
TEXT: настройка сайта на битрикс
TEXT: перенести сайт битрикс
TEXT: перенос сайта на битрикс
TEXT: подключить битрикс сайту
TEXT: разработка сайта на битрикс 24
TEXT: разработка сайта на битрикс цена
TEXT: разработка сайтов 1с битрикс
TEXT: разработка сайтов на базе 1с битрикс
TEXT: разработка сайтов на битрикс
TEXT: редактирование сайта битрикс
TEXT: сайт каталог битрикс
TEXT: сайт на 1с битрикс
```

Вывод графика обучения модели



Дальнейшее развитие

- **Нейросетевой чат-бот для Telegram**
Возможно реализовать, создав дополнительный микросервис на основе существующего решения
- **Поиск для сайта на основе нейросети**
Автоматический парсинг страниц, создание векторного описания страницы, и поиск ответа по косинусному расстоянию между векторами
- **Диалоговая система**
Извлечение сущностей NLP, разбор теории риторической структуры - Rhetorical Structure Theory (RST)
- **Калькуляторы с голосовым управлением**
Человек задаёт на сайте вопрос в естественном виде, например: "Мне нужен мешок для крупы на 50 кг. 1000 штук."
- **Подключение партнёрских аккаунтов**
Развитие партнёрской сети, после тестирования модели монетизации