

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования

**ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)**

Кафедра автоматизированных систем управления (АСУ)

К ЗАЩИТЕ ДОПУСТИТЬ

Заведующий кафедрой АСУ

д.т.н., профессор

_____А.М. Корилов

«___»_____2018 г.

Аксененко Иван Олегович

**ТЕХНОЛОГИЯ АВТОМАТИЗИРОВАННОГО СОЗДАНИЯ ТЕСТОВЫХ
ЗАДАНИЙ ДЛЯ СИСТЕМЫ ДИСТАНЦИОННОГО ОБУЧЕНИЯ НА
ОСНОВЕ ПРИМЕНЕНИЯ НЕЙРОННОЙ СЕТИ**

Направление магистратуры 01.04.02 – Прикладная математика и информатика

Магистерская программа – Математическое и программное обеспечение вы-
числительных комплексов и компьютерных сетей

Диссертация на соискание степени магистра

Выполнил магистрант гр. 496-М

_____И.О. Аксененко

«___»_____2018 г.

Руководитель диссертации

Доцент каф. АСУ, к.т.н.

_____В.В. Романенко

«___»_____2018 г.

Томск 2018

Реферат

Магистерская диссертация: 113 с., 22 рис., 47 табл., 0 лист., 29 использованных источников, 11 приложений.

КЛЮЧЕВЫЕ СЛОВА: АВТОМАТИЧЕСКАЯ РАЗМЕТКА, ТЕХНОЛОГИЯ СОЗДАНИЯ ТЕСТОВЫХ ЗАДАНИЙ, НЕЙРОННЫЕ СЕТИ, ТЕКСТОВЫЙ КОНТЕНТ, KERAS, RNN, LSTM.

Объект исследования: нейронные сети.

Объект разработки: технология создания тестовых заданий.

Цель работы: Разработать технологию автоматизированного создания тестовых заданий для систем дистанционного обучения на основе применения нейронных сетей.

Метод исследования: моделирование.

Полученные результаты:

1) проведён обзор и анализ технологий создания тестовых заданий, методов алгоритмов и технологий, предназначенных для обработки слабоструктурированных текстовых данных и моделей нейронных сетей, приспособленных для обработки таких данных;

2) спроектированы, реализованы и апробированы модели нейронных сетей для решения задач генерации маски неразмеченного текстового контента, генерации подмасок неразмеченного и размеченного текстового контента и идентификации типа текстового контента;

3) разработано программное обеспечение для решения задач генерации маски размеченного текстового контента и самого размеченного текстового контента; разработано программное обеспечение для генерации данных; сгенерировано необходимое количество данных для обучения разработанных нейронных сетей на примере одного варианта структуры неразмеченного текстового контента;

4) проведено исследование разработанных нейронных сетей и технологии создания тестовых заданий; проанализированы результаты.

Abstract

Master's thesis: 113 pp., 22 figures, 47 tables, 0 worksheets, 29 used sources, 11 applications.

KEY WORDS: AUTOMATIC MARKING, TECHNOLOGY OF CREATION OF TEST TASKS, NEURAL NETWORKS, TEXT CONTENT, KERAS, RNN, LSTM.

Object of investigation: neural networks.

Development object: technology for creating test tasks.

Objective: To develop a technology for automated creation of test tasks for distance learning systems based on the application of neural networks.

Method of research: modeling.

Results:

1) a review and analysis of technologies for creating test tasks, methods of algorithms and technologies intended for processing of weakly structured text data and models of neural networks adapted for processing such data;

2) neural network models were designed, implemented and tested to solve the problems of generating a mask of unmetered text content, generating subpatterns of unplaced and marked text content, and identifying the type of text content;

3) software has been developed to solve the tasks of generating a mask of marked text content and the most marked text content; developed software for data generation; the necessary amount of data for learning the developed neural networks is generated by the example of one version of the structure of unplaced text content;

4) research of the developed neural networks and technology of creation of test tasks is carried out; the results are analyzed.

Министерство образования и науки
Российской Федерации

Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра автоматизированных систем
управления (АСУ)

К ЗАЩИТЕ ДОПУСТИТЬ

Заведующий кафедрой АСУ

д.т.н., профессор

_____ А.М. Кориков

«___» _____ 2018 г.

ЗАДАНИЕ

для магистерской диссертации Аксененко Ивану Олеговичу

группа 496-М факультет систем управления

1. Тема работы: Технология автоматизированного создания тестовых заданий для системы дистанционного обучения на основе применения нейронной сети

(утверждена приказом по ВУЗу от «13» июня 2018 г. № 2556ст)

2. Срок сдачи студентом законченной работы: 26 июня 2018 г.

3. Содержание работы (вопросы, подлежащие разработке):

3.1. Обзор предметной области:

3.1.1. Обзор и анализ технологий создания и публикации тестовых заданий в системах дистанционного обучения;

3.1.2. Обзор и анализ методов, алгоритмов и технологий, предназначенных для обработки слабоструктурированных текстовых данных;

3.1.3. Обзор и анализ моделей нейронных сетей, приспособленных для обработки слабоструктурированных текстовых данных;

3.2. Разработка технологии автоматизированного создания тестовых заданий:

3.2.1. Разработка технологии автоматизированной разметки текстового контента;

3.2.2. Разработка моделей нейронных сетей для автоматизации разметки текстового контента;

3.3. Реализация технологии автоматизированного создания тестовых заданий:

3.3.1. Реализация моделей нейронных сетей для автоматизации разметки текстового контента;

3.3.2. Реализация программного обеспечения для автоматизации разметки текстового контента;

3.4. Исследование разработанной технологии создания тестовых заданий:

3.4.1. Исследование разработанных нейронных сетей для решения задачи автоматизации разметки текстового контента;

3.4.2. Исследование технологии автоматизированной разметки текстового контента;

3.4.3. Анализ результатов.

4. Дата выдачи задания «12» сентября 2017 г.

Руководитель магистерской диссертации

Доцент каф. АСУ, д.т.н.

«12» сентября 2017 г. _____ В.В. Романенко

Задание принял к выполнению

Магистрант гр. 496-М

«12» сентября 2017 г. _____ И.О. Аксененко

Оглавление

1 Введение	8
2 Обзор предметной области	10
2.1 Обзор и анализ технологий создания и публикации тестовых заданий в системах дистанционного обучения	10
2.2 Обзор и анализ методов, алгоритмов и технологий, предназначенных для обработки слабоструктурированных текстовых данных	17
2.3 Обзор и анализ моделей нейронных сетей, приспособленных для обработки слабоструктурированных текстовых данных	18
3 Разработка технологии автоматизированного создания тестовых заданий	23
3.1 Разработка технологии автоматизированной разметки текстового контента	23
3.2 Разработка моделей нейронных сетей для автоматизации разметки текстового контента	24
4 Реализация технологии автоматизированного создания тестовых заданий	32
4.1 Реализация моделей нейронных сетей для автоматизации разметки текстового контента	32
4.2 Реализация программного обеспечения для автоматизации разметки текстового контента	35
5 Исследование разработанной технологии создания тестовых заданий	40
5.1 Исследование разработанных нейронных сетей для решения задачи автоматизации разметки текстового контента	40
5.2 Исследование технологии автоматизированной разметки текстового контента	58
5.3 Анализ результатов	61
6 Заключение	64
Сокращения, обозначения, термины и определения	66
Список использованных источников	67
Приложение А (справочное) Примеры неразмеченного текстового контента ...	70

Приложение Б (справочное) Примеры масок неразмеченного текстового контента	72
Приложение В (справочное) Примеры подмасок неразмеченного текстового контента	73
Приложение Г (справочное) Примеры подмасок размеченного текстового контента	74
Приложение Д (справочное) Примеры масок размеченного текстового контента	77
Приложение Е (справочное) Примеры размеченного текстового контента	79
Приложение Ж (справочное) Пример подготовки данных для нейронной сети, генерирующей подмаски неразмеченного текстового контента	81
Приложение З (справочное) Правила разметки текстового контента.....	84
Приложение И (справочное) Результаты работы нейронной сети для генерации маски неразмеченного текстового контента.....	86
Приложение К (справочное) Результаты работы нейронной сети для генерации подмасок неразмеченного текстового контента	88
Приложение Л (справочное) Результаты работы нейронной сети для генерации подмасок размеченного текстового контента.....	105

1 Введение

В системах дистанционного обучения тестовые задания являются основным способом проверки знаний. Ежедневно специалистам приходится создавать и поддерживать в актуальном состоянии огромное количество тестов, что требует значительных временных и финансовых ресурсов. В целях сокращения этих затрат, необходимо разрабатывать и совершенствовать технологии автоматизированного создания тестовых заданий.

В настоящее время на факультете дистанционного обучения (ФДО) ТУ-СУР создание тестовых заданий осуществляется с помощью технологии создания тестов в системе дистанционного обучения (СДО) Moodle в совокупности с технологией разметки текстового контента и последующей его интерпретации. Технология состоит из четырёх этапов, первые два из которых осуществляются вручную. Работа посвящена автоматизации второго этапа технологии: процесса разметки текстового контента. Применение нейронных сетей обусловлено неоднозначностью формирования структуры неразмеченного текстового контента, что не позволяет решить задачу с помощью классического программирования.

Цель работы:

Разработать технологию автоматизированного создания тестовых заданий для систем дистанционного обучения на основе применения нейронных сетей.

Задачи:

1) провести обзор предметной области:

1.1) провести обзор и анализ технологий создания и публикации тестовых заданий в системах дистанционного обучения;

1.2) провести обзор и анализ методов, алгоритмов и технологий, предназначенных для обработки слабоструктурированных текстовых данных;

1.3) провести обзор и анализ моделей нейронных сетей, приспособленных для обработки слабоструктурированных текстовых данных;

2) разработать технологию автоматизированного создания тестовых заданий:

2.1) разработать технологию автоматизированной разметки текстового контента;

2.2) разработать модели нейронных сетей для автоматизации разметки текстового контента;

3) реализовать технологию автоматизированного создания тестовых заданий:

3.1) реализовать модели нейронных сетей для автоматизации разметки текстового контента;

3.2) реализовать программное обеспечение для автоматизации разметки текстового контента;

4) провести исследование разработанной технологии создания тестовых заданий:

4.1) провести исследование разработанных нейронных сетей для решения задачи автоматизации разметки текстового контента;

4.2) провести исследование технологии автоматизированной разметки текстового контента;

4.3) проанализировать результаты.

2 Обзор предметной области

2.1 Обзор и анализ технологий создания и публикации тестовых заданий в системах дистанционного обучения

Существует большое количество технологий автоматизированного создания тестовых заданий:

- 1) технологии создания тестовых заданий в СДО;
- 2) технологии, основанные на шаблонных моделях;
- 3) технологии, основанные на лингвистическом анализе;
- 4) технологии, основанные на продукционных моделях;
- 5) технологии, основанные на комбинаторных моделях;
- 6) технологии, основанные на формальных грамматиках;
- 7) технологии, основанные на семантических сетях;
- 8) технологии, основанные на нейронных сетях;
- 9) технологии, основанные на онтологиях;
- 10) технологии, основанные на понятийно-тезисных моделях.

Кратко рассмотрим каждую из этих технологий и проведём сравнительный анализ.

2.1.1 Технологии создания тестовых заданий в системах дистанционного обучения

В настоящее время существует два варианта технологий создания тестовых заданий в СДО:

- 1) без привлечения технического специалиста (рис. 2.1);
- 2) с привлечением технического специалиста (рис. 2.2).

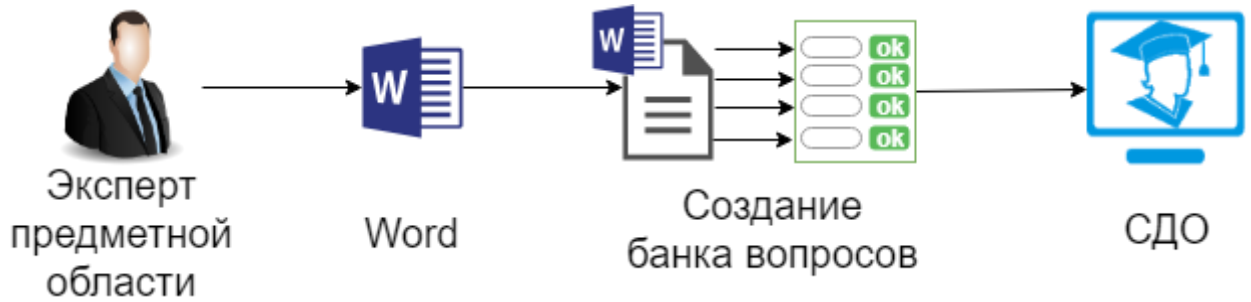


Рисунок 2.1 – Технология создания тестовых заданий в СДО без привлечения технического специалиста

Принцип работы варианта технологии без привлечения технического специалиста:

1) эксперт предметной области на основе теоретического материала осуществляет подготовку текстового контента, который представляет собой текст на естественном языке с вопросами разного типа;

2) эксперт предметной области перепечатывает текстовый контент в СДО, создавая при этом банк вопросов;

3) на основе банка вопросов СДО автоматически генерируется тест.



Рисунок 2.2 – Технология создания тестовых заданий в СДО с привлечением технического специалиста

Принцип работы второго варианта технологии:

1) эксперт предметной области на основе теоретического материала осуществляет подготовку текстового контента, который представляет собой текст на естественном языке с вопросами разного типа;

2) технический специалист производит разметку текстового контента; разметка текстового контента производится в соответствии с правилами разметки, регламентируемыми разработчиками этой разметки; разметка необходима, чтобы избавиться от неоднозначности структуры текстового

контента, привести структуру к единообразному виду; структура размеченного текста представляет собой исходный текст, заключенный в теги специального назначения; на ФДО ТУСУР была разработана собственная разметка и реализована надстройка Test Tagger для Microsoft Word, позволяющая упростить процесс разметки;

3) программа-интерпретатор обрабатывает размеченный текстовый контент и на его основе создает файл Moodle XML;

4) файл Moodle XML загружается в систему Moodle, где скрипт-генератор сформирует тест в виде HTML-страниц.

Второй вариант технологии более эффективный:

1) нет необходимости обучать экспертов предметной области процессу ручного пополнения банка вопросов в СДО;

2) пополнение банка вопросов происходит автоматически, что сокращает временные и финансовые затраты на создание тестов.

Первые два этапа технологии осуществляются вручную. Работа посвящена автоматизации второго этапа технологии: процессу разметки текстового контента. Примеры неразмеченного текстового контента представлены в приложении А.

2.1.2 Технологии, основанные на шаблонных моделях

Принцип работы технологий, основанных на шаблонных моделях в общем случае можно описать следующим образом:

1) эксперт предметной области формирует базу знаний, алгоритм формулировки задачи, алгоритм генерации параметров и алгоритм решения задачи;

2) программа-генератор на основе подготовленных данных генерирует тестовые задания.

Примеры реализации технологий, основанных на шаблонных моделях можно увидеть в источниках [1–7].

2.1.3 Технологии, основанные на лингвистическом анализе

Принцип работы технологий, основанных на лингвистическом анализе в общем случае можно описать следующим образом:

- 1) программа-обработчик проводит графематический, морфологический, синтаксический и семантический анализ, в результате чего формирует базу знаний;
- 2) программа-генератор на основе базы знаний создает набор тестов в требуемом формате.

Примеры реализации технологий, основанных на лингвистическом анализе можно увидеть в источниках [8,9].

2.1.4 Технологии, основанные на продукционных моделях

Принцип работы технологий, основанных на продукционных моделях в общем случае можно описать следующим образом:

- 1) эксперт предметной области формирует продукционную базу знаний;
- 2) программа-генератор на основе базы знаний создает набор тестов в требуемом формате.

Пример реализации технологии, основанной на продукционных моделях можно увидеть в источнике [10].

2.1.5 Технологии, основанные на комбинаторных моделях

Принцип работы технологий, основанных на комбинаторных моделях в общем случае можно описать следующим образом:

- 1) эксперт предметной области формирует дерево И/ИЛИ;
- 2) программа-генератор на основе дерева И/ИЛИ создает набор разных вариантов задания, которое описывает дерево И/ИЛИ;
- 3) итеративно повторяя шаги 1 и 2 составляется набор задач для тестов в требуемом формате.

Примеры реализации технологий, основанных на комбинаторных моделях можно увидеть в источниках [11, 12].

2.1.6 Технологии, основанные на формальных грамматиках

Принцип работы технологий, основанных на формальных грамматиках в общем случае можно описать следующим образом:

1) эксперт предметной области формирует базу знаний предметной области, представленную множеством формальных грамматик, описывающих множество правил вывода тестовых заданий; определяет необходимое количество тестовых заданий;

2) программа-генератор на основе базы знаний формирует нужное количество тестовых заданий разных типов.

Примеры реализации технологий, основанных на формальных грамматиках можно увидеть в источниках [13, 14].

2.1.7 Технологии, основанные на семантических сетях

Принцип работы технологий, основанных на семантических сетях в общем случае можно описать следующим образом:

1) эксперт предметной области формирует семантическую базу знаний;

2) программа-генератор на основе базы знаний формирует необходимое количество тестовых заданий.

Пример реализации технологии, основанной на семантических сетях можно увидеть в источнике [15].

2.1.8 Технологии, основанные на нейронных сетях

Принцип работы технологий, основанных на нейронных сетях в общем случае можно описать следующим образом:

1) формирование тестов происходит на основе лингвистического анализа;

2) программа-анализатор на основе машинного обучения выполняет оценку качества созданных тестовых заданий.

Пример реализации технологии, основанной на нейронных сетях можно увидеть в источнике [16].

2.1.9 Технологии, основанные на онтологиях

Принцип работы технологий, основанных на онтологиях в общем случае можно описать следующим образом:

1) эксперт предметной области создает фреймовую базу знаний, состоящую из формализованных машинных представлений онтологии предметной области;

2) программа-генератор на основе базы знаний формирует необходимое количество тестовых заданий.

Пример реализации технологии, основанной на онтологиях можно увидеть в источнике [17].

2.1.10 Технологии, основанные на понятийно-тезисных моделях

Принцип работы технологий, основанных на понятийно-тезисных моделях в общем случае можно описать следующим образом:

1) эксперт предметной области создает понятийно-тезисную базу знаний;

2) программа-генератор на основе базы знаний формирует необходимое количество тестовых заданий разных типов.

Пример реализации технологии, основанной на понятийно-тезисных моделях можно увидеть в источнике [18].

2.1.11 Сравнительный анализ технологий создания тестовых заданий

Ключевым этапом всех технологий создания тестовых заданий является формирование банка заданий, на основе которого впоследствии генерируется тест. Отличаются технологии по большей части только способами формирова-

ния банка заданий. В таблице 2.1 представлен сравнительный анализ технологий по основным критериям.

Таблица 2.1 – Сравнительный анализ технологий создания тестовых заданий

Критерий	Технологии создания тестовых заданий									
	СДО	ШМ	ЛА	ПМ	КМ	ФГ	СС	НС	ОМ	ПТМ
Автоматизация подготовки текстового контента	0	0	0	0	0	0	0	0	0	0
Автоматизация обработки текстового контента	0	0	1	0	0	0	0	1	0	0
Простота обработки текстового контента	1	1	1	0	0	0	0	1	0	0
Способность создавать разные типы вопросов	1	0	1	0	1	1	1	1	1	1
Автоматизация формирования банка заданий	1	1	1	1	1	1	1	1	1	1
Автоматизация генерации тестов на основе банка заданий	1	1	1	1	1	1	1	1	1	1
Качество тестовых заданий и их лексическая ясность	1	1	0	1	1	1	1	1	1	1
Нет необходимости привлекать специалиста для проверки сгенерированного теста	1	1	0	1	1	1	1	0	1	1
Результат	6	5	5	4	5	5	5	6	5	5

Из таблицы 2.1 видно, что наиболее хорошо критериям удовлетворяют:

- 1) технологии создания тестовых заданий в СДО;
- 2) технологии, основанные на применении нейронных сетей.

Скомбинировав эти технологии, можно автоматизировать этап обработки текстового контента технологии создания тестовых заданий в СДО, что значительно улучшит эту технологию в сравнении с другими.

2.2 Обзор и анализ методов, алгоритмов и технологий, предназначенных для обработки слабоструктурированных текстовых данных

В последнее время активно развиваются технологии обработки больших данных (Big data). В частности, технологии интеллектуального анализа данных (Data mining) и технологии интеллектуального анализа текста (Text mining). Технологии Text mining лежат на пересечении таких областей, как поиск информации, Data mining, машинное самообучение, статистика и компьютерная лингвистика. Эти технологии глубинного анализа текстов дают возможность обрабатывать большие объемы неструктурированной информации и извлекать знания и высококачественную информацию из текстовых массивов. Как правило, Text mining подразумевает процесс структурирования входных текстовых данных, извлечение шаблонов из уже структурированных данных, оценку и интерпретацию полученных результатов. Типичные задачи Text mining включают кластеризацию, классификацию текстов, извлечение концептов и объектов, создание таксономий, смысловой анализ, обобщение документации и моделирование объектов, то есть установление связей между различными известными объектами. Анализ текстов включает себя извлечение информации и лингвистический анализ для выявления частоты вхождений различных слов, выявление шаблонов, расставление тэгов и аннотирование, техники Data mining, включая анализ связей и ассоциаций, визуализацию и прогностический анализ. Подробную информацию можно увидеть в источниках [19–22].

Таким образом, существует четыре основных метода обработки слабоструктурированных текстовых данных:

- 1) лингвистический анализ;

- 2) формальные грамматики;
- 3) статистический анализ;
- 4) нейронные сети.

В силу существенных недостатков первых трех технологий, в настоящее время все большей популярностью в сфере обработки текстовых данных пользуются нейросетевые технологии. Основным недостатком лингвистического анализа является чрезмерная вычислительная сложность. Формальные грамматики, как и статистический анализ, неспособны охватить все аспекты лингвистики естественного языка: например, ограничены задачей семантического анализа языка, без чего большинство современных проблем не могут быть решены. Поэтому сосредоточим свое внимание непосредственно на нейросетевых технологиях.

2.3 Обзор и анализ моделей нейронных сетей, приспособленных для обработки слабоструктурированных текстовых данных

Глубокие нейронные сети [23] (DNN) произвели революцию в области обработки естественного языка (NLP). На сегодняшний день сверточная нейронная сеть [23] (CNN) и рекуррентная нейронная сеть [23, 24] (RNN) – две основные модели нейронных сетей, широко исследуемые и применяемые в области различных задач. Часто возникает спор при выборе той или иной модели нейронной сети, для решения многих современных задачи по NLP.

2.3.1 Рекуррентная нейронная сеть

Рекуррентная нейронная сеть получается из многослойного перцептрона введением обратных связей. Одна из широко распространенных разновидностей рекуррентных нейронных сетей – сеть Элмана (рис. 2.3).

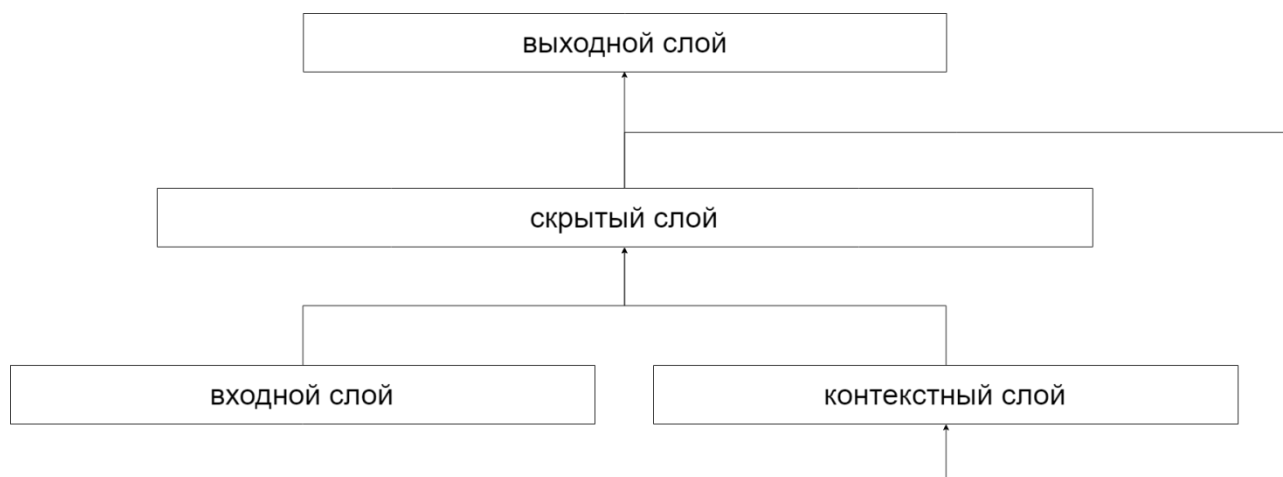


Рисунок 2.3 – Рекуррентная нейронная сеть Элмана

Разворачивать во времени рекуррентные нейронные сети можно различными способами, как показано на рис. 2.4.

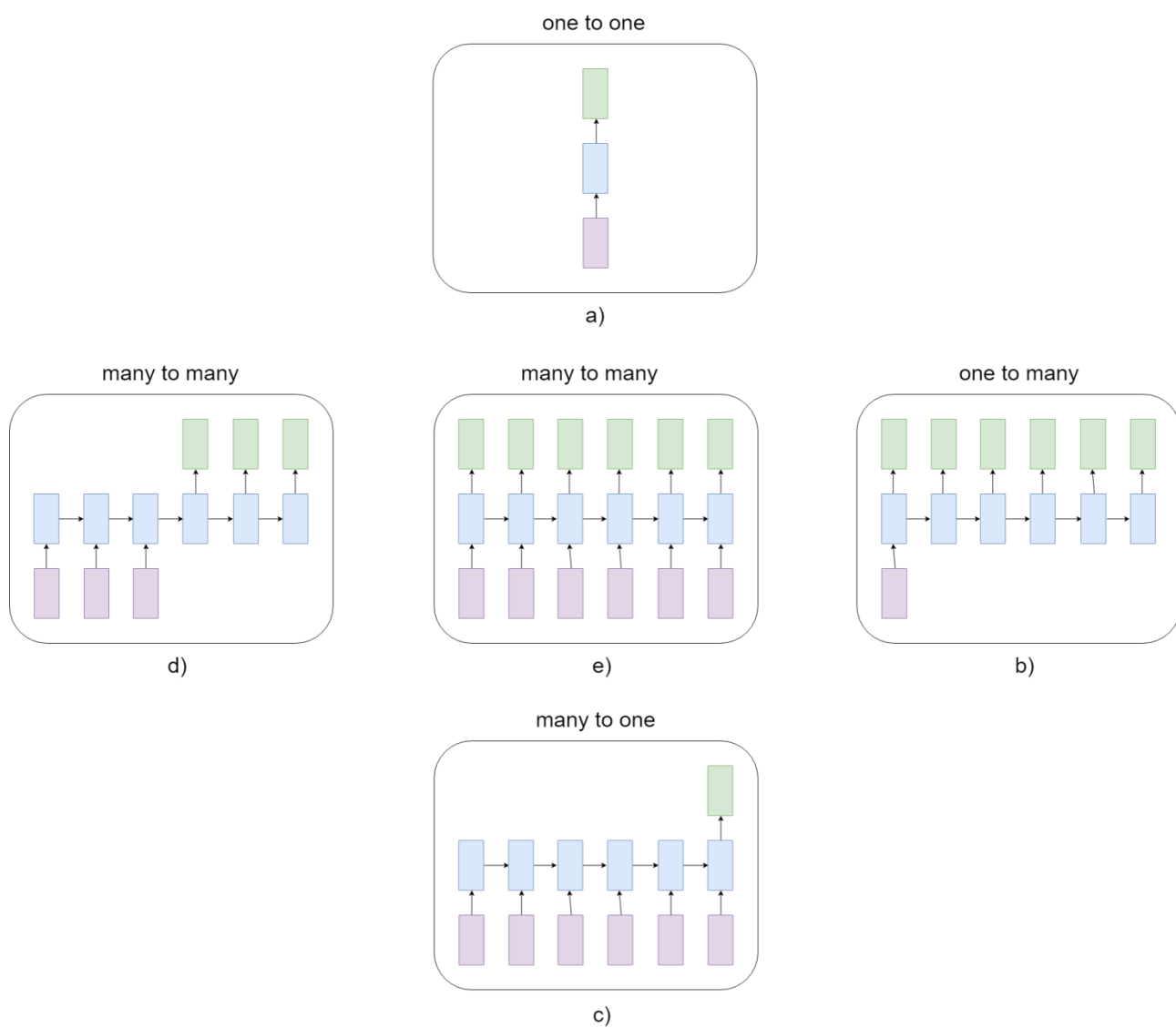


Рисунок 2.4 – Принципы работы рекуррентных нейронных сетей

В RNN обратные связи идут не от выхода сети, а от выходов внутренних нейронов. Это позволяет учесть предысторию наблюдаемых процессов и накопить информацию для выработки правильной стратегии обучения. Главной особенностью рекуррентных нейронных сетей является запоминание последовательностей.

Рекуррентные нейронные сети с долгосрочной краткосрочной памятью [25] (LSTM) решают проблему долгосрочной памяти простых RNN, которая заключается в том, что простые RNN хорошо помнят недавно полученную информацию, но не имеют возможности надолго сохранить в памяти что-то, что обработали много циклов назад, какой бы важной та информация ни была. В LSTM-сетях внутренние нейроны оборудованы сложной системой так называемых ворот, а также концепцией клеточного состояния, которая и представляет собой некий вид долгосрочной памяти. Ворота же определяют, какая информация попадет в клеточное состояние, какая сотрется из него, и какая повлияет на результат, который выдаст RNN на данном шаге. Схема LSTM юнита представлена на рис. 2.5.

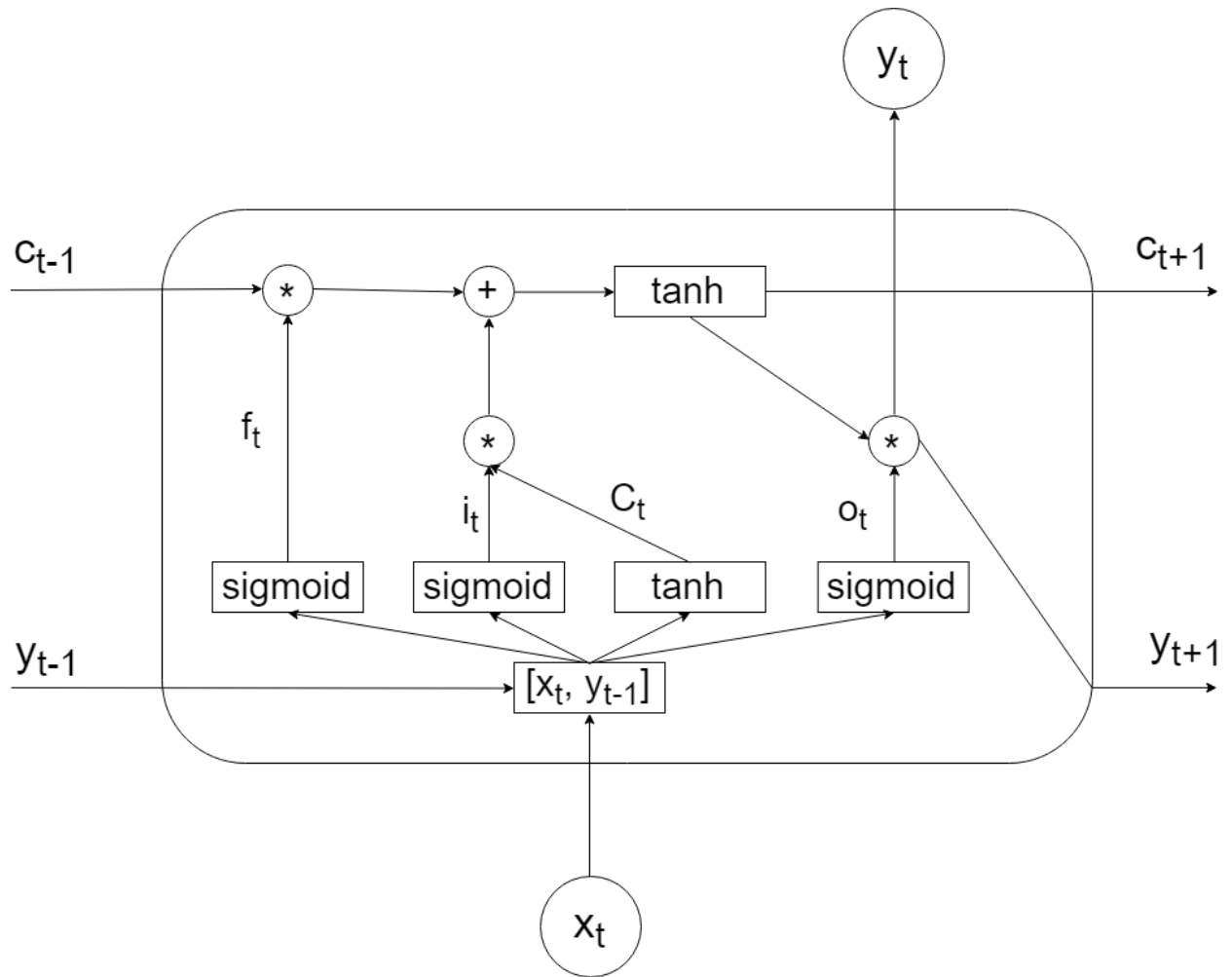


Рисунок 2.5 – Схема нейрона рекуррентных нейронных сетей с долгосрочной краткосрочной памятью

2.3.2 Сверточная нейронная сеть

Сверточная нейронная сеть [23] – однонаправленная многослойная сеть с применением операции свертки, при которой каждый фрагмент входных данных умножается на матрицу (ядро) свертки поэлементно, а результат суммируется и записывается в аналогичную позицию выходных данных. Обобщенная схема CNN приведена на рис. 2.6 – 2.7.

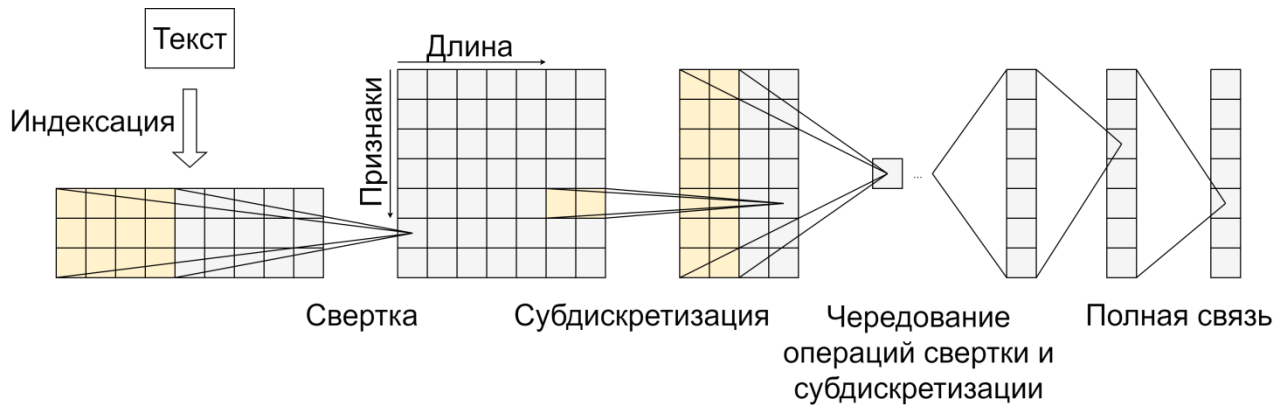


Рисунок 2.6 – Обобщенная схема сверточной нейронной сети

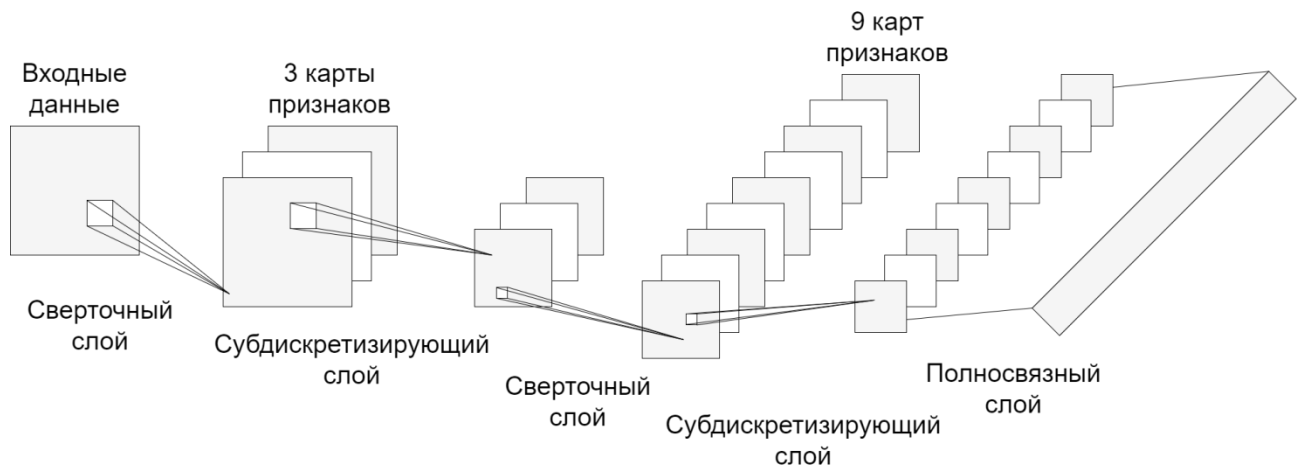


Рисунок 2.7 – Обобщенная схема сверточной нейронной сети

CNN состоит из трех основных видов слоев: сверточный слой (convolution layer), субдискретизирующий слой (subsampling layer) и выходной слой (чаще всего полносвязный (fully-connected layer)). Слои CNN расположены друг за другом: сначала сверточный слой, затем субдискретизирующий, за последним сверточным слоем следует выходной слой. Сверточный и субдискретизирующий слои считаются слоями двумерной размерности, а выходной слой, как правило, представляет собой вектор из пространства \mathbb{R}^1 . В CNN каждый двумерный слой имеет несколько уровней. Каждый уровень представляет собой двумерный массив. Выход каждого уровня называется картой признаков (feature map).

3 Разработка технологии автоматизированного создания тестовых заданий

3.1 Разработка технологии автоматизированной разметки текстового контента

Задачу разметки текстового контента можно декомпозировать на пять более простых задач (рис. 3.1):

1) задача генерации маски неразмеченного текстового контента (см. приложение Б): удаление из неразмеченного текстового контента всех неструктурных символов;

2) задача генерации подмасок неразмеченного текстового контента (см. приложение В): разделение маски неразмеченного текстового контента на фрагменты, которые соответствуют определенным фрагментам маски размеченного текстового контента;

3) задача генерации подмасок размеченного текстового контента (см. приложение Г): отображение подмасок неразмеченного текстового контента на соответствующие им фрагменты маски размеченного текстового контента;

4) задача генерации маски размеченного текстового контента (см. приложение Д): соединение подмасок размеченного текстового контента согласно правилам разметки текстового контента (см. приложение З);

5) задача генерации размеченного текстового контента (см. приложение Е): возвращение в маску размеченного текстового контента неструктурных символов, удаленных при решении первой задачи.

Первые три задачи необходимо решить с помощью нейронных сетей, поскольку неизвестна структура неразмеченного текстового контента. Четвёртую задачу можно решить классическим программированием, т.к. известны правила размещения тегов и атрибутов (приложение З). При решении последней задачи, структура текстового контента уже приведена к единообразному виду, что позволяет решить эту задачу также без применения нейронных сетей.

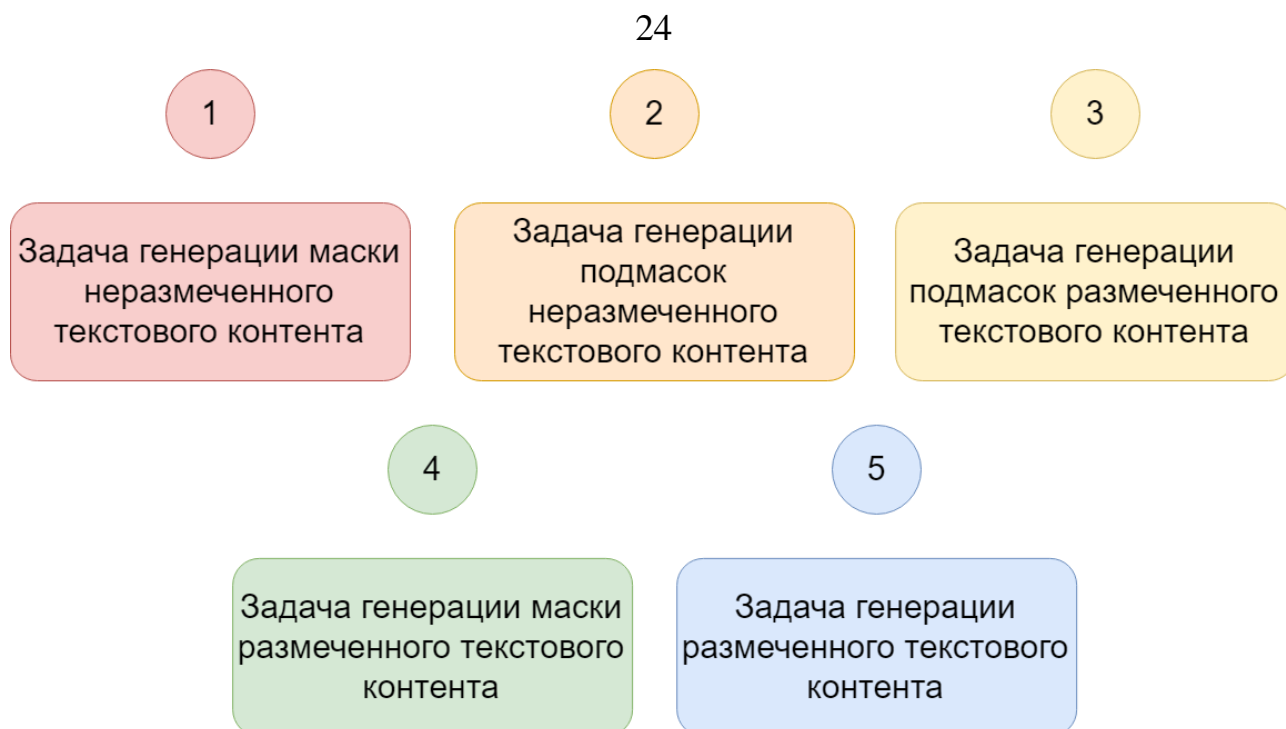


Рисунок 3.1 – Декомпозиция задачи разметки текстового контента

3.2 Разработка моделей нейронных сетей для автоматизации разметки текстового контента

3.2.1 Модель нейронной сети для решения задачи генерации маски неразмеченного текстового контента

Примеры входных и выходных данных для этой нейронной сети представлены в приложении Б.

Решение данной задачи предполагает, что нейронная сеть выделит из исходного текстового контента структурные символы, которые влияют на разметку текстового контента.

Для решения этой задачи будем использовать рекуррентную нейронную сеть с долгосрочной краткосрочной памятью (RNN LSTM) и принципом работы много во много (many-to-many) (см. рис. 2.4, е). Такой выбор обусловлен особенностями входных и выходных данных для нейронной сети. Входные и выходные данные не имеют фиксированную длину и по своей природе представляют собой последовательность элементов, где каждый текущий элемент зависит от предыдущих.

Модель такой нейронной сети представлена на рис. 3.2.

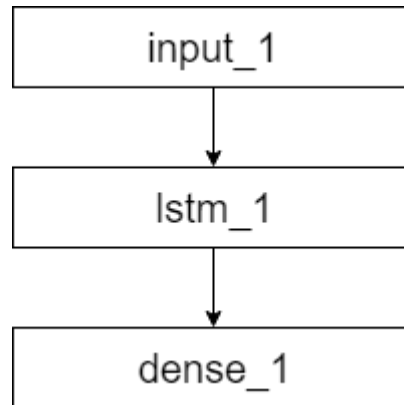


Рисунок 3.2 – Модель нейронной сети для решения задачи генерации маски неразмеченного текстового контента

На рис. 3.2 input_1 – это входной слой нейронной сети, lstm_1 – это рекуррентный слой с долгосрочной краткосрочной памятью и dense_1 – полносвязный выходной слой нейронной сети.

Входные и выходные данные для LSTM слоя представляют собой трёхмерные тензоры. Первая размерность – размер порции данных, вторая – размер временных шагов, и третья – размер самих данных (рис. 3.3).

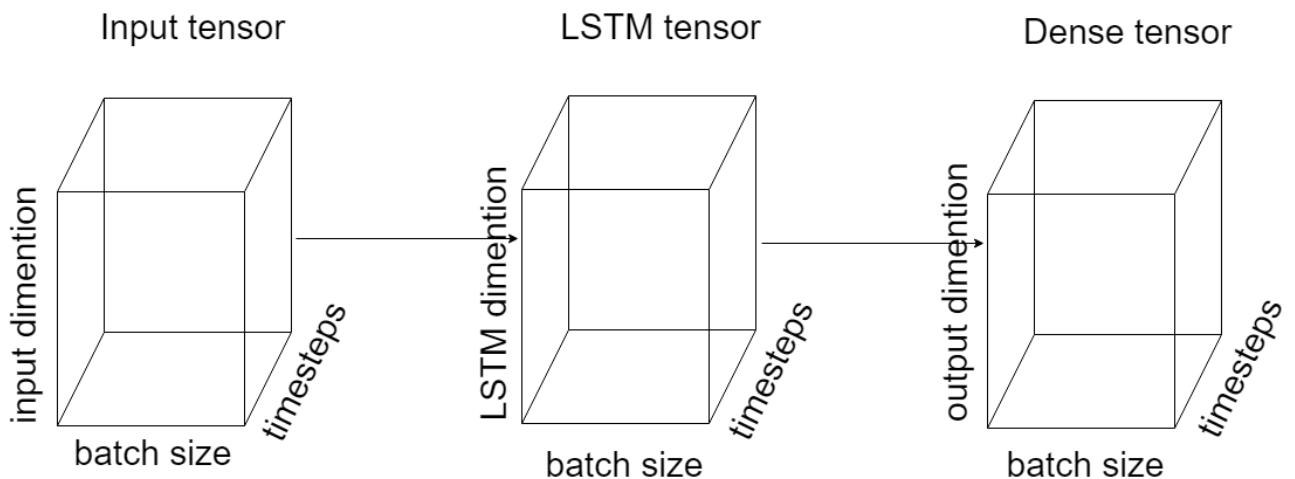


Рисунок 3.3 – Тензорное представление данных LSTM слоя

В момент времени t обработка данных происходит по принципам рекуррентной нейронной сети, как показано на рис. 3.4.

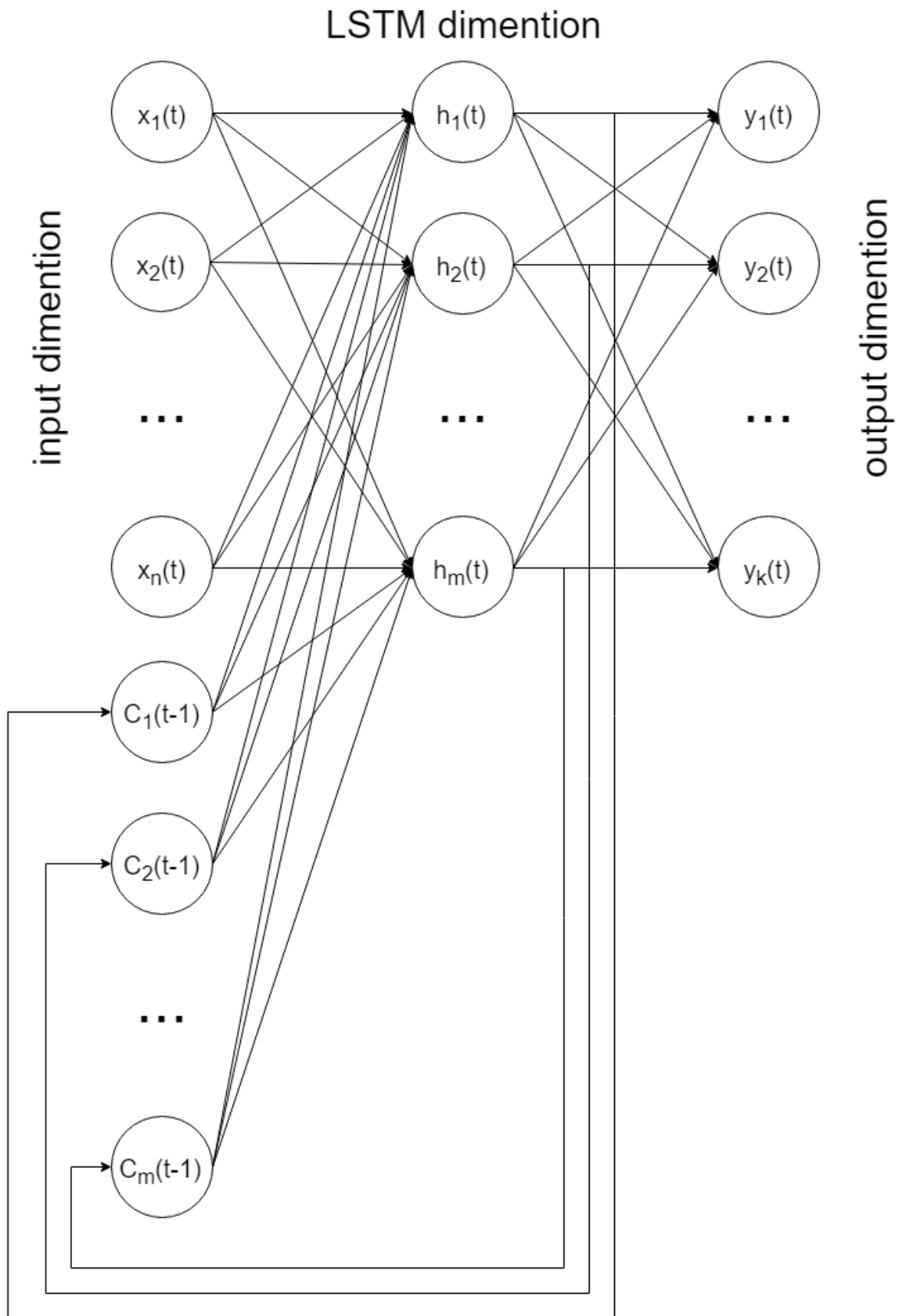


Рисунок 3.4 – Схема работы LSTM юнита в момент времени t

Разворачивание LSTM юнита во времени в модели, представленной на рис. 3.2, показано на рис. 2.4, е.

Гиперпараметры для этой модели нейронной сети подбирались эмпирическим путём и представлены в разделе исследования нейронных сетей.

Объёмы данных для модели нейронной сети, решающей задачу генерации маски неразмеченного текстового контента составляют:

- 1) 114 примеров тренировочного набора данных;
- 2) 30 примеров верификационного набора данных;
- 3) 30 примеров тестового набора данных.

Процент правильных ответов на тестовом наборе данных достиг значения 100.0%.

3.2.2 Модель нейронной сети для решения задачи генерации подмасок неразмеченного текстового контента

Примеры входных и выходных данных для этой нейронной сети можно увидеть в приложении В.

Решение данной задачи предполагает, что из маски неразмеченного текстового контента нейронная сеть сможет сгенерировать необходимое количество подмасок, где каждая подмаска состоит только из тех структурных элементов, которые отвечают за определённый фрагмент размеченного текстового контента.

В соответствии с входными и выходными данными, принцип работы такой нейронной сети аналогичен принципу работы нейронной сети для решения задачи генерации маски неразмеченного текстового контента (рис. 2.4, е). Однако чтобы на один пример входных данных сгенерировать несколько примеров выходных данных, необходимо на вход помимо маски неразмеченного текстового контента отправлять также закодированный номер подмаски, которую необходимо сгенерировать. Кроме того, для исключения неоднозначности, на вход нейронной сети отправляется закодированный идентификатор типа вопро-

са, который предсказывает отдельная нейронная сеть, идентифицирующая тип вопроса. Пример формирования данных см. в приложении Ж.

Модель такой нейронной сети представлена на рис. 3.5.

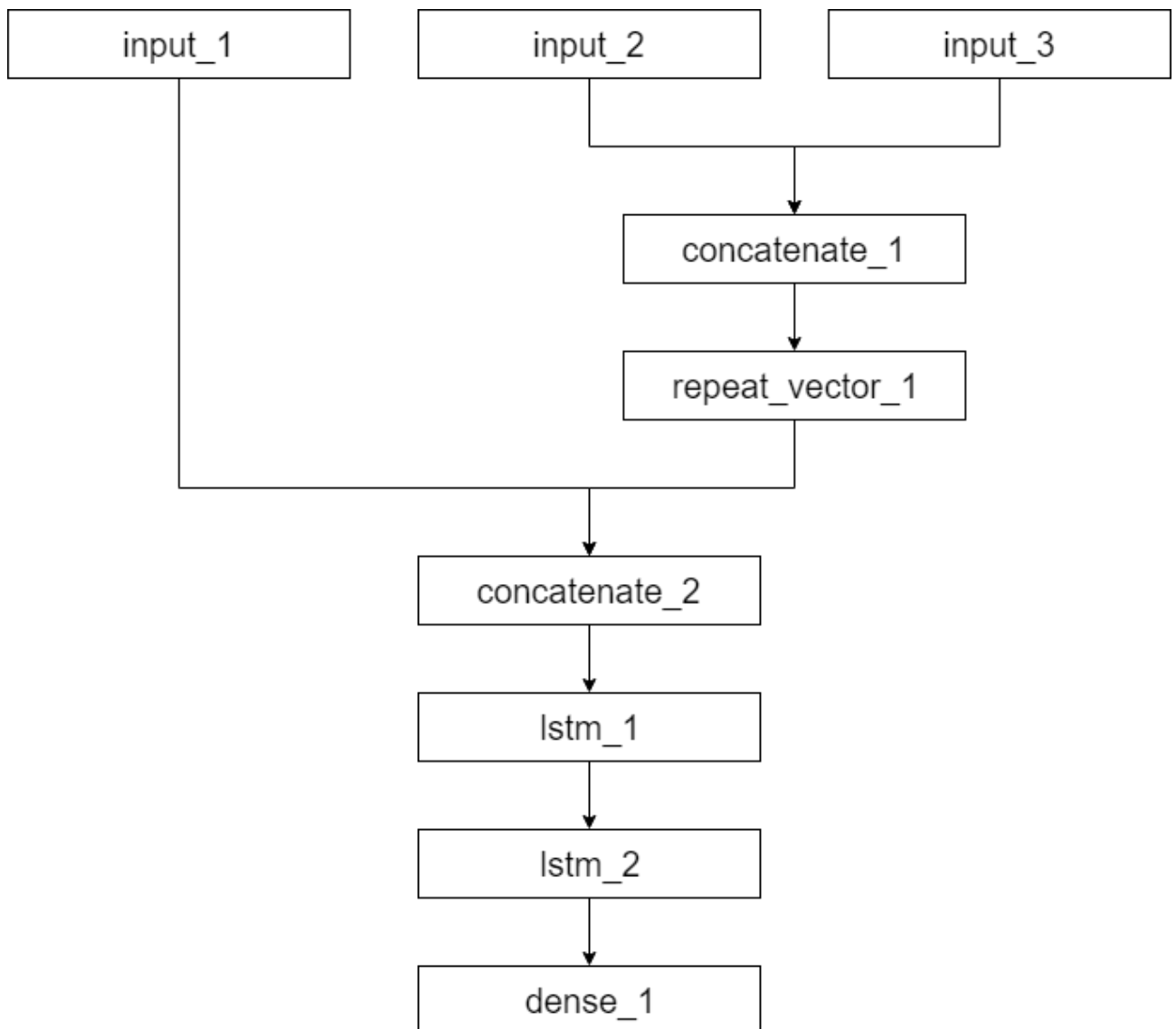


Рисунок 3.5 – Модель нейронной сети для решения задачи генерации подмасок неразмеченного текстового контента

Гиперпараметры для этой модели нейронной сети подбирались эмпирическим путём и представлены в разделе исследования нейронных сетей.

Объёмы данных для модели нейронной сети, решающей задачу генерации подмасок неразмеченного текстового контента составляют:

- 1) 30 192 примера тренировочного набора данных;
- 2) 480 примеров верификационного набора данных;
- 3) 480 примеров тестового набора данных.

Процент правильных ответов на тестовом наборе данных достиг значения 99.9%.

3.2.3 Модель нейронной сети для решения задачи генерации подмасок размеченного текстового контента

Примеры входных и выходных данных для нейронной сети, генерирующей подмаски размеченного текстового контента можно увидеть в приложении Г.

Решение данной задачи предполагает, что каждой подмаске неразмеченного текстового контента нейронная сеть поставит в соответствие определённый фрагмент размеченного текстового контента, который назовём маской размеченного текстового контента.

В соответствии с входными и выходными данными принцип работы такой сети many-to-one (см. рис. 2.4, с). Для исключения неоднозначности данных, на вход, аналогично модели нейронной сети для генерации подмасок неразмеченного текстового контента, дополнительно подаётся идентифицирующий вектор, сгенерированный отдельной нейронной сетью, идентифицирующей тип контента.

Таким образом, модель нейронной сети для решения этой задачи выглядит, как показано на рис. 3.6.

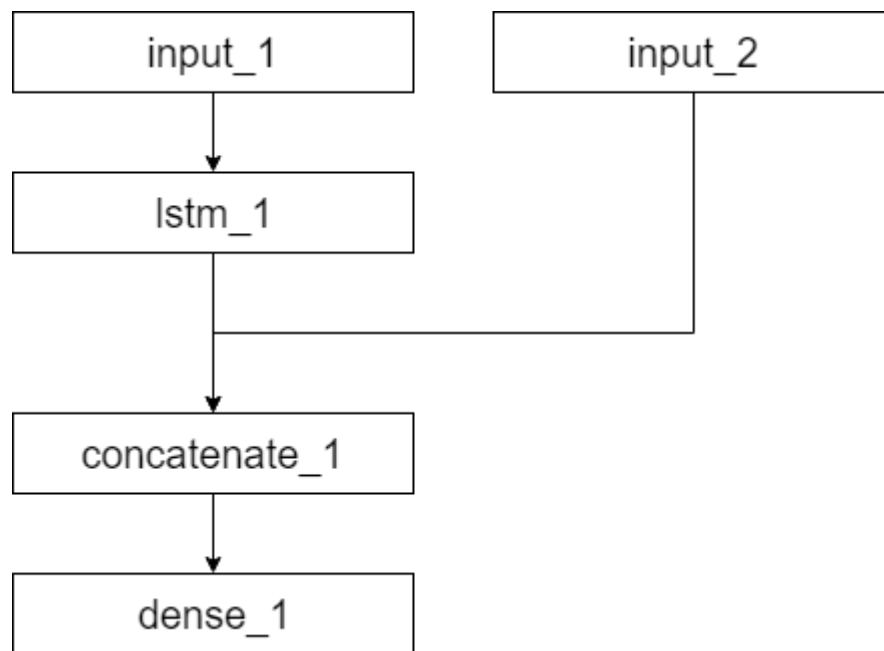


Рисунок 3.6 – Модель нейронной сети для решения задачи генерации подмасок размеченного текстового контента

Гиперпараметры для этой модели нейронной сети подбирались эмпирическим путём и представлены в разделе исследования нейронных сетей.

Объёмы данных для модели нейронной сети, решающей задачу генерации подмасок размеченного текстового контента составляют:

- 1) 17 015 примеров тренировочного набора данных;
- 2) 202 примера верификационного набора данных;
- 3) 202 примера тестового набора данных.

Процент правильных ответов на тестовом наборе данных достиг значения 99.91%.

3.2.4 Модель нейронной сети для решения задачи идентификации типа неразмеченного текстового контента

Входными данными для данной нейронной сети является маска неразмеченного текстового контента, выходные данные представляют собой идентифицирующие one-hot вектора.

В соответствии с входными и выходными данными, принцип работы нейронной сети, идентифицирующей тип контента many-to-one (см. рис. 2.4, с).

Модель такой нейронной сети представлена на рис. 3.7.

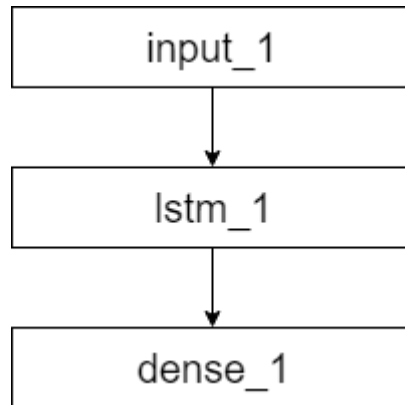


Рисунок 3.7 – Модель нейронной сети для решения задачи идентификации типа
неразмеченного текстового контента

Объёмы данных для модели нейронной сети, решающей задачу генерации подмасок размеченного текстового контента составляют:

- 1) 1887 примеров тренировочного набора данных;
- 2) 144 примера верификационного набора данных;
- 3) 144 примера тестового набора данных.

Процент правильных ответов на тестовом наборе данных достиг значения 100.0%.

4 Реализация технологии автоматизированного создания тестовых заданий

4.1 Реализация моделей нейронных сетей для автоматизации разметки текстового контента

В настоящее время нет необходимости разрабатывать собственные реализации нейронных сетей, если вы пытаетесь решить задачу с помощью существующих топологий. Существует огромное количество фреймворков [26], позволяющих быстро перейти от идеи к её реализации: Apache Spark, H2O, Apache Singa, Caffe2, Google TensorFlow, Amazon Machine Learning, Microsoft Azure ML Studio, Microsoft Distributed Machine Learning Toolkit, Microsoft Cognitive Toolkit, Apache Mahout, Vels, mlpack2, Neon, Massive Online Analysis, scikit-learn, Shogun, Theano, Torch, Accord.NET, Weka, PrismaLabs.ai, Core ML, Neural Processing Engine. Наиболее популярными являются Tensor Flow [27] и Theano [28]. Фреймворк Keras [23, 29] изначально разрабатывался как надстройка над Theano, для упрощения реализации, но в дальнейшем стал поддерживать Tensor Flow и даже стал его частью. Сейчас, Keras является наиболее популярным фреймворком для моделирования и реализации нейронных сетей.

Реализация моделей нейронных сетей с помощью фреймворка Keras представлена в таб. 4.1.

Таблица 4.1 – Реализация моделей нейронных сетей с помощью фреймворка Keras

Название нейронной сети	Реализация модели
Нейронная сеть для решения задачи генерации маски неразмеченного текстового контента	<pre> input_1 = Input(shape=(None, len(data.input_dict))) lstm_1 = LSTM(64, return_sequences=True)(input_1) dense_1 = Dense(len(data.output_dict), activation='softmax')(lstm_1) model = Model(inputs=input_1, outputs=dense_1) model.compile(optimizer=RMSprop(lr=0.01, decay=1E-5), loss='categorical_crossentropy', metrics=['accuracy']) </pre>
Нейронная сеть для решения задачи генерации подмасок	<pre> input_1 = Input(shape=(data.input_seq_max_len, len(data.input_dict))) input_2 = Input(shape=(data.out_sub_seq_code_len,)) input_3 = Input(shape=(7,)) </pre>

неразмеченного текстового контента	<pre> concatenate_1 = Concatenate()([input_2, input_3]) repeat_vector_1 = RepeatVector(data.input_seq_max_len)(concatenate_1) concatenate_2 = Concatenate()([input_1, repeat_vector_1]) lstm_1 = LSTM(64, return_sequences=True)(concatenate_2) lstm_1 = LSTM(256, return_sequences=True, dropout=0.7)(lstm_1) dense_1 = Dense(len(data.output_dict), activation='softmax')(lstm_1) model = Model(inputs=[input_1, input_2, input_3], outputs=dense_1) model.compile(optimizer=RMSprop(lr=0.01, decay=1E-4), loss='categorical_crossentropy', metrics=['accuracy']) </pre>
Нейронная сеть для решения задачи генерации подмасок размеченного тек- стового контента	<pre> input_1 = Input(shape=(None, len(data.input_dict))) input_2 = Input(shape=(7,)) lstm_1 = LSTM(128, dropout=0.1, recurrent_dropout=0.1)(input_1) concatenate_1 = Concatenate()([lstm_1, input_2]) dense_1 = Dense(len(data.output_dict), activa- tion='softmax')(concatenate_1) model = Model(inputs=[input_1, input_2], outputs=dense_1) model.compile(optimizer=RMSprop(lr=0.01, decay=1E-5), loss='categorical_crossentropy', metrics=['accuracy']) </pre>
Нейронная сеть для решения задачи идентификации типа неразмеченно- го текстового кон- тента	<pre> model = Sequential() model.add(LSTM(8, input_shape=(None, len(data.input_dict)))) model.add(Dense(7, activation='softmax')) model.compile(optimizer=RMSprop(lr=0.01), loss='categorical_crossentropy', metrics=['accuracy']) </pre>

Для подготовки данных для нейронных сетей на языке Python был реализован специальный модуль DataPreparation.

Для каждой нейронной сети, реализован собственный генератор данных.

Для нейронной сети, генерирующей маску неразмеченного текстового контента реализовать генератор входных данных классическим программированием невозможно, поскольку входными данными здесь является неразмеченный текстовый контент, который составляет эксперт предметной области на первом этапе технологии создания тестовых заданий. Автоматизация этого этапа технологии является отдельной достаточно сложной задачей. Однако для фиксированной структуры неразмеченного текстового контента, существует способ автоматической генерации выходных данных. В качестве примера был реализован генератор, получающий на вход неразмеченный текстовый контент со структурой первого типа (см. приложение А) и возвращающий маску нераз-

меченного текстового контента. Реализованный генератор основан на рекуррентной обработке текста по регулярным выражениям.

Генераторы для остальных нейронных сетей также основаны на фиксированной структуре неразмеченного текстового контента и переборе возможных вариантов её формирования. Рассмотрим, например, маску неразмеченного текстового контента для вопроса с одиночным выбором ответа и структурой первого типа. В общем случае, она состоит из следующих элементов:

- 1) структурный элемент текста: $\langle p \rangle T \langle /p \rangle$;
- 2) структурный элемент верного варианта ответа: $\langle p \rangle = T \langle /p \rangle$, который может быть пронумерован, например, так $\langle p \rangle 1. = T \langle /p \rangle$;
- 3) структурный элемент неверного варианта ответа: $\langle p \rangle \sim T \langle /p \rangle$, который тоже может быть пронумерован: $\langle p \rangle 1. \sim T \langle /p \rangle$.

Маски неразмеченного текстового контента для данного типа вопроса, отличаются количеством структурных элементов текста и неверного варианта ответа, а также некоторой перестановкой структурных элементов. Структурный элемент верного варианта ответа для данного типа вопроса всегда один. Например, при количестве структурных элементов вариантов ответов $n = 1$ и количестве структурных элементов текста $m = 1$, получаем один вариант маски неразмеченного текстового контента:

- 1) $\langle p \rangle T \langle /p \rangle \langle p \rangle = T \langle /p \rangle$

В случае $n = 2$, получим два варианта масок:

- 1) $\langle p \rangle T \langle /p \rangle \langle p \rangle = T \langle /p \rangle \langle p \rangle \sim T \langle /p \rangle$
- 2) $\langle p \rangle T \langle /p \rangle \langle p \rangle \sim T \langle /p \rangle \langle p \rangle = T \langle /p \rangle$

Таким образом, для любого n можно сформировать n различных масок неразмеченного текстового контента для вопроса с одиночным выбором ответа. Структурные элементы текста могут находиться либо слева от всех структурных элементов вариантов ответов, либо справа от них, но не между ними. Кроме того, слева всегда должен находиться хотя бы один структурный элемент текста, который отвечает за вопрос. Например, для $n = 1$, $m = 2$, получаем два варианта масок:

1) $\langle p \rangle T \langle /p \rangle \langle p \rangle T \langle /p \rangle \langle p \rangle = T \langle /p \rangle$

2) $\langle p \rangle T \langle /p \rangle \langle p \rangle = T \langle /p \rangle \langle p \rangle T \langle /p \rangle$

Для $n = 1$, $m = 3$, получаем три варианта масок:

1) $\langle p \rangle T \langle /p \rangle \langle p \rangle T \langle /p \rangle \langle p \rangle T \langle /p \rangle \langle p \rangle = T \langle /p \rangle$

2) $\langle p \rangle T \langle /p \rangle \langle p \rangle T \langle /p \rangle \langle p \rangle = T \langle /p \rangle \langle p \rangle T \langle /p \rangle$

3) $\langle p \rangle T \langle /p \rangle \langle p \rangle = T \langle /p \rangle \langle p \rangle T \langle /p \rangle \langle p \rangle T \langle /p \rangle$

Таким образом, при увеличении m количество вариантов масок также увеличивается в m раз. Следовательно, общее количество вариантов масок неразмеченного текстового контента для вопроса с одиночным вариантом ответа равно nm . Для других типов вопросов генерация происходит аналогичным образом.

4.2 Реализация программного обеспечения для автоматизации разметки текстового контента

4.2.1 Задача генерации маски размеченного текстового контента

Решив задачу генерации подмасок размеченного текстового контента, получим фрагменты, из которых состоит маска размеченного текстового контента. Воспользовавшись правилами разметки текстового контента, представленными в приложении 3, несложно на основе подмасок размеченного текстового контента получить маску размеченного текстового контента. Алгоритм программы, генерирующей маску размеченного текстового контента представлен на рис. 4.1.

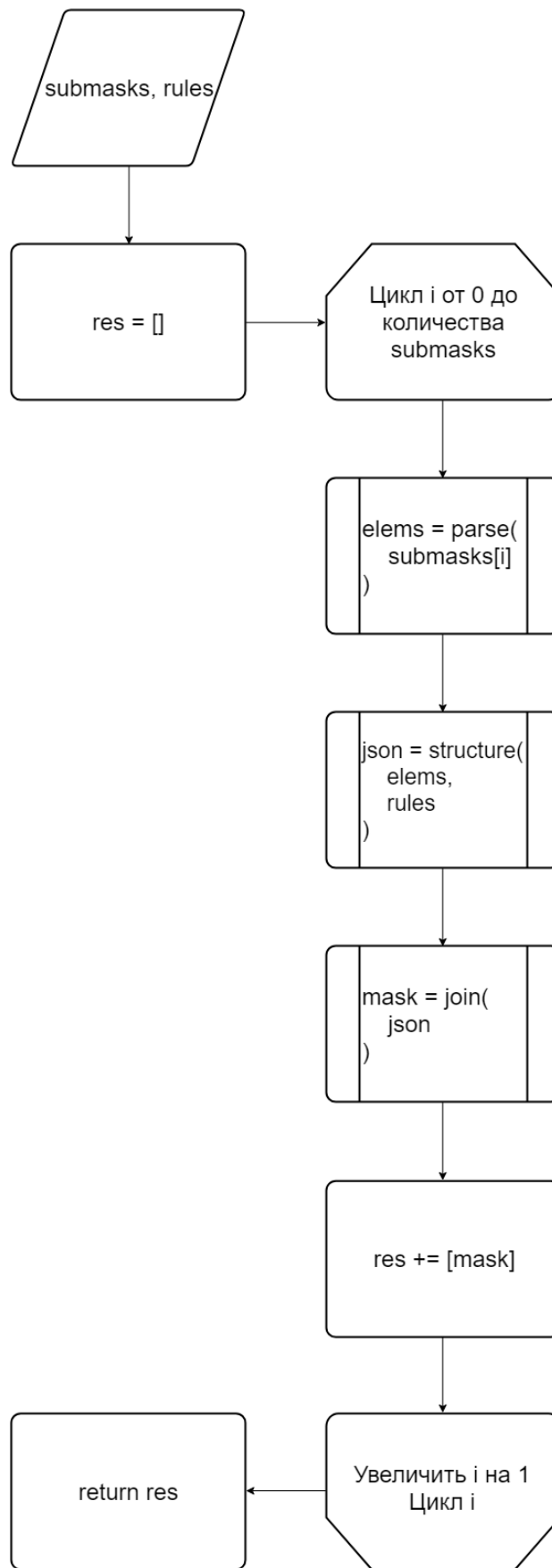


Рисунок 4.1 – Алгоритм программы для генерации маски размеченного текстового контента

Здесь `submasks` – набор массивов с подмасками размеченного текстового контента, `rules` – правила разметки, представленные в `json` формате.

Функция `parse(submasks[i])` разбирает подмаски, представленные в строковом виде и возвращает список структур `elems` вида:

```
{ 'type': 'tag', 'name': str, 'content': str, 'attrs': [] }
```

или

```
{ 'type': 'attr', 'name': str, 'val': str },
```

в зависимости от того, к какому типу принадлежит данная подмаска: к тегу или атрибуту соответственно. Вместо `str` подставляются разобранные фрагменты подмаски в строковом виде, а вместо `[]` – разобранный список атрибутов.

Функция `structure(elems, rules)` представляет собой рекурсивную процедуру, которая формирует дерево в формате `json` из разобранных элементов `elems`, подобное дереву `rules`, которое хранит информацию о всех тегах, их атрибутах и контенте.

Функция `join(json)` на основе сформированного дерева в формате `json` генерирует строковое его представление, что и является маской размеченного текстового контента.

4.2.2 Задача генерации размеченного текстового контента

После того, как сгенерирована маска размеченного текстового контента необходимо вернуть текст, удалённый на первом этапе решения задачи. Этот текст легко получить, сопоставив исходный неразмеченный текстовый контент и маску неразмеченного текстового контента (см. таб. 4.2).

Таблица 4.2 – Неразмеченный текстовый контент сопоставленный маске неразмеченного текстового контента

<r>Укажите административный центр Томской области:</r><r>1. ~ Асино;</r><r>2. ~ Северск;</r> <r>_____</r><r>1. ~ ____;</r><r>2. ~ ____;</r>
<r>3. ~ Стрежевой;</r><r>4. = Томск;</r> <r>3. ~ ____;</r><r>4. = ____;</r>

Таким образом, получаем следующие фрагменты текста:

1) Укажите административный центр Томской области:

- 2) Асино
- 3) Северск
- 4) Стрежевой
- 5) Томск

Заменяв каждый символ Т маски размеченного текстового контента, на полученные фрагменты текста в прямом порядке следования, получим размеченный текстовый контент:

Таблица 4.3 – Размеченный текстовый контент

```
[item]
  Укажите административный центр Томской области:
  [singlechoice numbering=123.; view=vertical]
    [choice] Асино [/choice]
    [choice] Северск [/choice]
    [choice] Стрежевой [/choice]
    [choice value=1] Томск [/choice]
  [/singlechoice]
[/item]
```

Алгоритм решения этой задачи представлен на рис. 4.2. Здесь txtcont – неразмеченный текстовый контент, masks – маски неразмеченного текстового контента, tmask – маски размеченного текстового контента.

Функция get_content(tmask) получает из масок размеченного текстового контента содержимое их тегов, которое нужно будет заменить текстовым содержимым.

Функция get_text(txtcont, masks, content) на основе неразмеченного текстового контента и масок неразмеченного текстового контента получает текст, который необходимо поместить в маску размеченного текстового контента. Содержимое масок размеченного текстового контента content необходимо для этой функции, чтобы определить правильную последовательность возвращаемых фрагментов.

Функция replace(tmask, content, text) заменяет в масках размеченного текстового контента tmask элементы content фрагментами текста text, в результате чего получается размеченный текстовый контент.

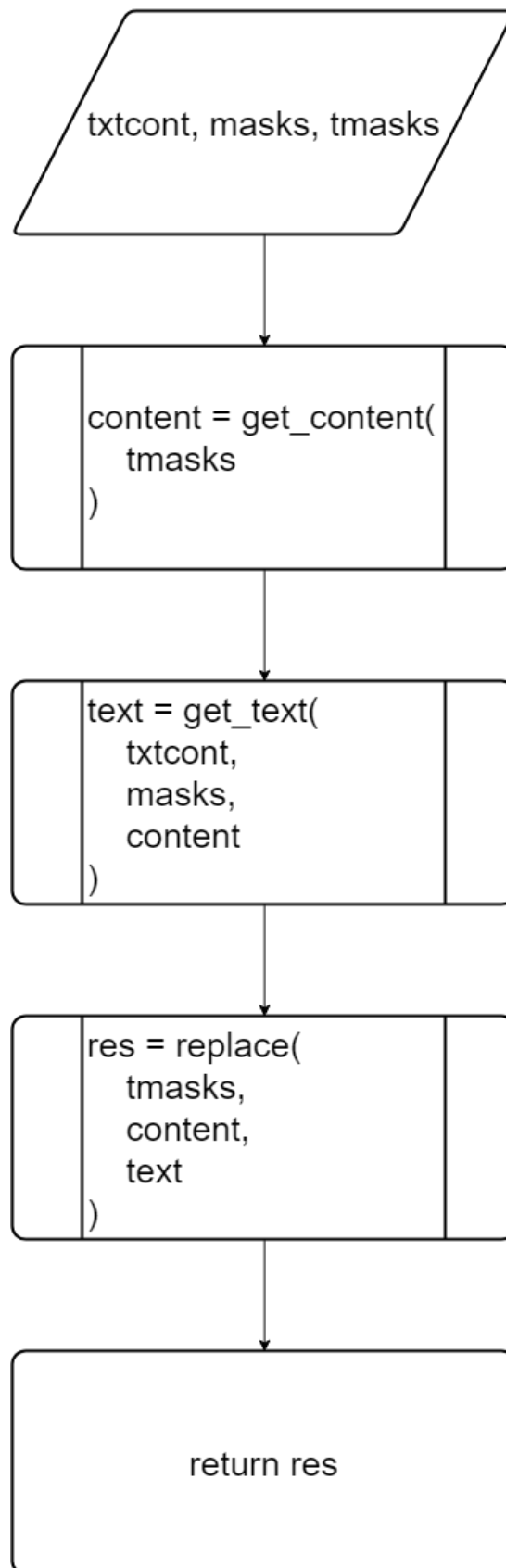


Рисунок 4.2 – Алгоритм решения задачи генерации размеченного текстового контента

5 Исследование разработанной технологии создания тестовых заданий

5.1 Исследование разработанных нейронных сетей для решения задачи автоматизации разметки текстового контента

Любая нейронная сеть зависит от некоторого количества гиперпараметров, таких как:

- 1) количество слоёв (layers);
- 2) количество нейронов в слое (neurons);
- 3) количество эпох обучения (epochs);
- 4) скорость обучения (learning rate);
- 5) величина понижения скорости обучения (decay);
- 6) величина dropout регуляризации (dropout);
- 7) величина рекуррентной dropout регуляризации (recurrent_dropout);
- 8) величина порции данных (batches);

и представляет собой некоторую функцию, зависящую от них. Поскольку алгоритма выбора этих параметров не существует, для выбора оптимальных значений необходимо проводить исследование. Исследование заключается в переборе различных значений гиперпараметров нейронной сети и выборе оптимальных среди них. Пример исследования нейронной сети представлен в источнике [23].

5.1.1 Исследование нейронной сети для решения задачи генерации маски неразмеченного текстового контента

Стартовые значения гиперпараметров представлены в таб. 5.1.

Таблица 5.1 – Стартовые значения гиперпараметров

Гиперпараметр	Значение гиперпараметра
Количество lstm слоёв	1
Количество нейронов в слое	2
Количество эпох обучения	1000

Скорость обучения	0.01
Величина decay	0
Величина dropout	0
Величина recurrent_dropout	0
Величина batches	2

В таблице 5.2 представлены оптимальные значения функции потерь и точности на обучающем наборе данных (loss, acc) и на валидационном наборе данных (val_loss, val_acc) при разных значениях количества нейронов (neurons), а также показано, на какой эпохе (epochs) нейронная сеть достигла таких значений функций потерь и точности.

Таблица 5.2 – Исследование влияния количества нейронов на точность нейронной сети

neurons	epochs	loss	acc	val_loss	val_acc
2	930	1,13E-01	9,65E-01	1,21E-01	9,62E-01
4	465	1,37E-04	1,00E+00	6,10E-03	9,99E-01
8	23	1,90E-03	1,00E+00	4,60E-03	9,99E-01
16	41	1,01E-04	1,00E+00	3,20E-03	9,99E-01
32	23	2,40E-04	1,00E+00	8,92E-04	1,00E+00
64	22	3,42E-05	1,00E+00	4,57E-05	1,00E+00
128	27	1,38E-05	1,00E+00	3,05E-04	1,00E+00
256	70	5,06E-05	1,00E+00	7,79E-05	1,00E+00
512	56	1,90E-04	1,00E+00	3,43E-04	1,00E+00

Из таблицы 5.2 видно, что оптимальное значение достигается уже на 22 эпохе обучения нейронной сети с количеством нейронов равном 64, следовательно количество эпох обучения можно снизить до 100.

В следующей таблице представлены оптимальные значения функций потерь и точности при разных значениях скорости обучения нейронной сети.

Таблица 5.3 – Исследование влияния скорости обучения на точность нейронной сети

lr	epochs	loss	acc	val_loss	val_acc
0.1	10	1,54E-01	1,60E-01	1,19E-07	2,12E-02
0.01	20	1,17E-05	1,00E+00	5,48E-05	1,00E+00
0.001	77	8,08E-05	1,00E+00	3,09E-04	1,00E+00
0.0001	100	2,79E-02	9,96E-01	3,08E-02	9,95E-01

Из таблицы 5.3 видно, что минимальное значение функции потерь на валидационной выборке достигается при скорости обучения равной 0.1, однако, если посмотреть на точность (val_acc), достигнутую при этом значении параметра, становится ясно, что данное значение на самом деле не является оптимальным. Таким образом, оптимальной скоростью обучения является 0.01.

В следующей таблице представлены оптимальные значения функций потерь и точности при разных значениях dropout нейронной сети.

Таблица 5.4 – Исследование влияния dropout регуляризации на точность нейронной сети

dropout	epochs	loss	acc	val_loss	val_acc
0	20	1,17E-05	1,00E+00	5,48E-05	1,00E+00
0.1	40	1,56E-04	1,00E+00	3,17E-04	1,00E+00
0.2	67	6,74E-04	1,00E+00	7,08E-04	1,00E+00
0.3	66	7,61E-04	1,00E+00	9,91E-04	1,00E+00

Из таблицы 5.4 видно, что оптимальным значением dropout регуляризации является 0, то есть её отсутствие. Это значит, что количество нейронов в слое подобрано оптимально.

В следующей таблице представлены оптимальные значения функций потерь и точности при разных значениях decay нейронной сети.

Таблица 5.5 – Исследование влияния decay на точность нейронной сети

decay	epochs	loss	acc	val_loss	val_acc
0,00E+00	20	1,17E-05	1,00E+00	5,48E-05	1,00E+00

1,00E-01	100	9,11E-02	9,87E-01	9,18E-02	9,87E-01
1,00E-02	18	6,70E-04	1,00E+00	2,30E-03	1,00E+00
1,00E-03	18	8,24E-06	1,00E+00	9,32E-05	1,00E+00
1,00E-04	17	6,14E-05	1,00E+00	4,45E-04	1,00E+00
1,00E-05	20	1,24E-06	1,00E+00	4,78E-06	1,00E+00
1,00E-06	24	2,50E-07	1,00E+00	5,52E-06	1,00E+00
1,00E-07	19	1,16E-04	1,00E+00	6,57E-04	1,00E+00
1,00E-08	31	1,76E-06	1,00E+00	8,44E-05	1,00E+00
1,00E-09	19	9,39E-05	1,00E+00	5,93E-04	1,00E+00

Из таблицы 5.5 видно, что оптимальным значением `decau` является величина $1E-05$.

В следующей таблице представлены оптимальные значения функций потерь и точности при разных значениях `recurrent_dropout` регуляризации нейронной сети.

Таблица 5.6 – Исследование влияния `recurrent_dropout` регуляризации на точность нейронной сети

rec_drop	epochs	loss	acc	val_loss	val_acc
0	20	1,24E-06	1,00E+00	4,78E-06	1,00E+00
0,1	35	3,28E-05	1,00E+00	1,34E-04	1,00E+00
0,2	27	2,82E-04	1,00E+00	5,06E-04	1,00E+00
0,3	79	5,64E-06	1,00E+00	7,69E-06	1,00E+00
0,4	54	1,34E-05	1,00E+00	2,33E-05	1,00E+00
0,5	79	3,55E-06	1,00E+00	5,53E-06	1,00E+00
0,6	47	5,36E-05	1,00E+00	1,33E-04	1,00E+00
0,7	44	3,80E-04	1,00E+00	1,10E-03	1,00E+00
0,8	100	7,67E-05	1,00E+00	1,20E-03	1,00E+00
0,9	37	3,87E-02	7,79E-01	1,19E-07	2,12E-02

Из таблицы 5.6 видно, что оптимальным значением recurrent_dropout регуляризации является 0.9, однако точность модели при этом очень низкая. Поэтому в действительности оптимальным значением recurrent_dropout является 0.

В следующей таблице представлены оптимальные значения функций потерь и точности при разных значениях batches нейронной сети. Поскольку практически все данные разной длины, максимальное количество объектов в одной порции – 2.

Таблица 5.7 – Исследование влияния величины batches на точность нейронной сети

batches	epochs	loss	acc	val_loss	val_acc
1	16	1,13E-05	1,00E+00	1,25E-04	1,00E+00
2	20	1,24E-06	1,00E+00	4,78E-06	1,00E+00

Из таблицы 5.7 видно, что оптимальным значением величины batches является 2.

В результате исследования, выяснилось, что оптимальными гиперпараметрами нейронной сети являются значения, представленные в таблице 5.8.

Таблица 5.8 – Оптимальные значения гиперпараметров нейронной сети для решения задачи генерации маски неразмеченного текстового контента

Гиперпараметр	Значение гиперпараметра
Количество lstm слоёв	1
Количество нейронов в слое	64
Количество эпох обучения	20
Скорость обучения	0.01
Величина decay	1E-5
Величина dropout	0
Величина recurrent_dropout	0
Величина batches	2

В результате обучения нейронной сети с гиперпараметрами, представленными в таб. 5.8, нейронная сеть достигла ошибки на обучающем наборе данных 1.24E-06, а на валидационном наборе данных 4.78E-06, что говорит о

сто процентной сходимости нейронной сети. На рис. 5.1 представлен график сходимости нейронной сети на обучающей и валидационной выборке при заданных гиперпараметрах.

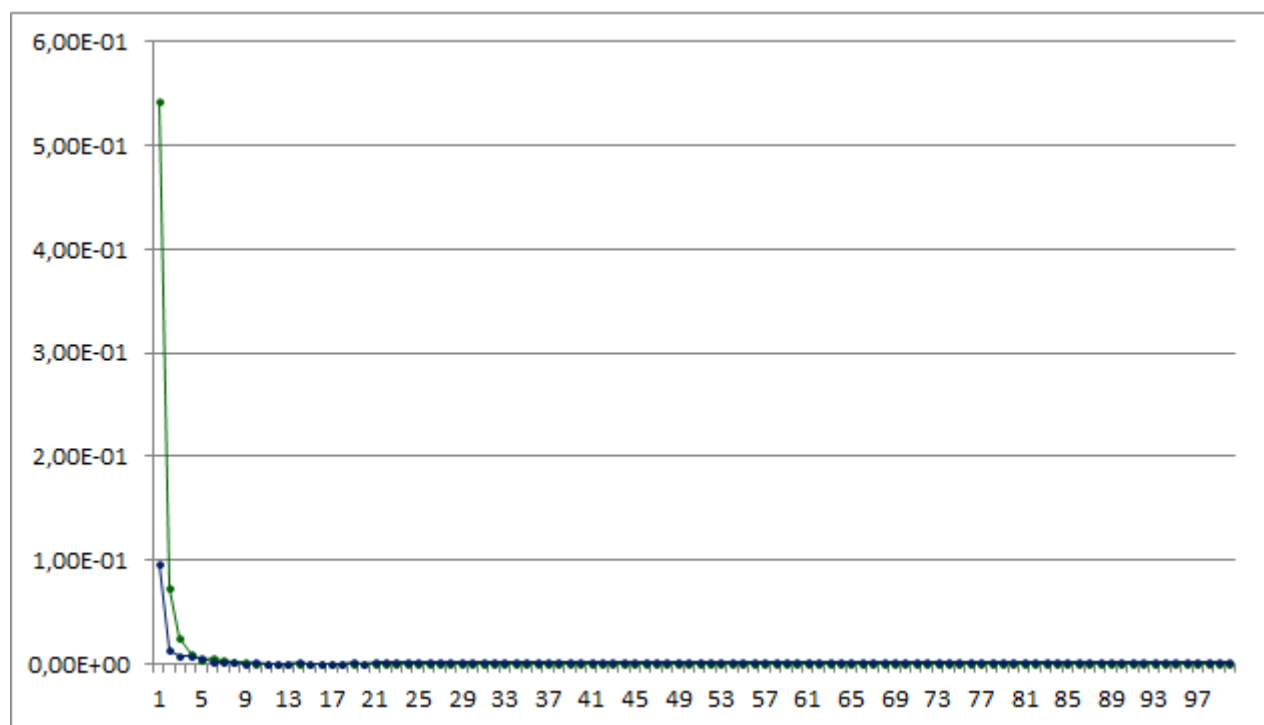


Рисунок 5.1 – График зависимости значений функции потерь на обучающем наборе данных (loss, зелёная линия) и на валидационном наборе данных (val_loss, синяя линия) от количества эпох

На рис. 5.2 представлен график изменения точности нейронной сети на обучающей и валидационной выборке в зависимости от количества эпох при заданных гиперпараметрах.

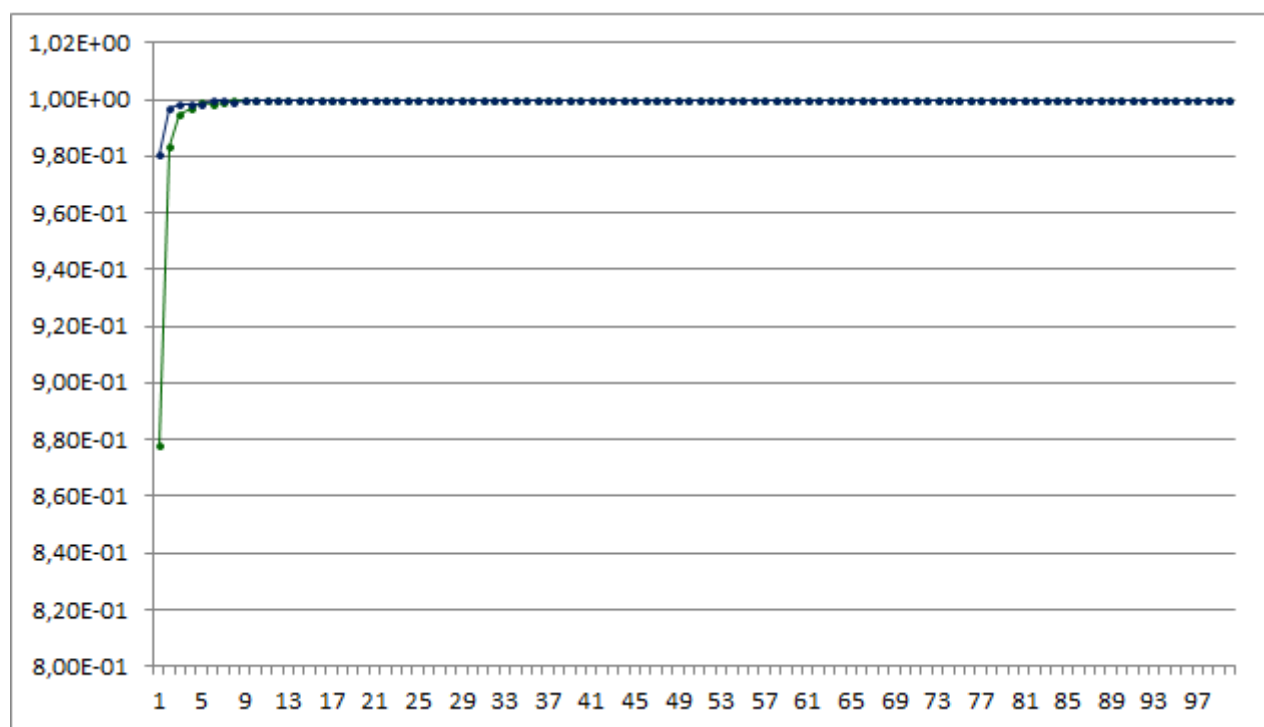


Рисунок 5.2 – График зависимости точности на обучающем наборе данных (асс, зелёная линия) и на валидационном наборе данных (val_асс, синяя линия) от количества эпох

5.1.2 Исследование нейронной сети для решения задачи генерации подмасок неразмеченного текстового контента

Стартовые значения гиперпараметров представлены в таб. 5.9.

Таблица 5.9 – Стартовые значения гиперпараметров

Гиперпараметр	Значение гиперпараметра
Количество lstm слоёв	1
Количество нейронов в слое	2
Количество эпох обучения	100
Скорость обучения	0.001
Величина decay	0
Величина dropout	0
Величина recurrent_dropout	0
Величина batches	512

В таблице 5.10 представлены оптимальные значения функции потерь и точности на обучающем наборе данных и на валидационном наборе данных при разных значениях количества нейронов, а также показано, на какой эпохе нейронная сеть достигла таких значений функций потерь и точности.

Таблица 5.10 – Исследование влияния количества нейронов на точность нейронной сети на 100 эпохах обучения

neurons	epochs	loss	acc	val_loss	val_acc
2	100	6,92E-02	9,84E-01	8,07E-02	9,87E-01
4	100	6,65E-02	9,84E-01	7,83E-02	9,87E-01
8	100	1,49E-02	9,96E-01	4,27E-02	9,90E-01
16	100	6,40E-03	9,98E-01	1,84E-02	9,94E-01
32	97	1,70E-03	9,99E-01	6,40E-03	9,98E-01
64	98	1,00E-03	9,99E-01	3,20E-03	9,99E-01
128	91	9,27E-04	1,00E+00	2,20E-03	9,99E-01
256	100	8,73E-04	1,00E+00	1,50E-03	9,99E-01
512	92	6,17E-04	1,00E+00	1,20E-03	9,99E-01

Из таблицы 5.10 видно, что оптимальное значение количества нейронов на 100 эпохах обучения – 512, однако, видно, что значения достигаются на эпохах довольно близких к 100. Это свидетельствует о том, что нейронная сеть недоучилась. Проведём ещё 100 эпох обучения. Результаты представлены в таб. 5.11.

Таблица 5.11 – Исследование влияния количества нейронов на точность нейронной сети на 200 эпохах обучения

neurons	epochs	loss	acc	val_loss	val_acc
2	100	5,30E-02	9,86E-01	7,07E-02	9,87E-01
4	100	3,41E-02	9,88E-01	4,30E-02	9,88E-01
8	100	5,90E-03	9,98E-01	1,63E-02	9,95E-01
16	99	1,60E-03	9,99E-01	5,50E-03	9,98E-01
32	97	1,30E-03	9,99E-01	2,40E-03	9,99E-01

64	93	7,15E-04	1,00E+00	1,90E-03	9,99E-01
128	63	4,98E-04	1,00E+00	1,20E-03	9,99E-01
256	19	4,59E-04	1,00E+00	1,10E-03	1,00E+00
512	28	4,81E-04	1,00E+00	1,10E-03	1,00E+00

Из таблицы 5.11 видно, что на 200 эпохах оптимальное значение количества нейронов уже стало 256, но тем не менее есть ещё значения, которые достигаются при максимальном количестве эпох, следовательно, обучение нужно продолжить. Проведём ещё 100 эпох обучения. Результаты представлены в таб. 5.12.

Таблица 5.12 – Исследование влияния количества нейронов на точность нейронной сети на 300 эпохах обучения

neurons	epochs	loss	acc	val_loss	neurons
2	95	4,89E-02	9,86E-01	6,84E-02	2
4	100	2,11E-02	9,93E-01	3,15E-02	4
8	100	3,70E-03	9,99E-01	9,90E-03	8
16	94	1,20E-03	9,99E-01	3,10E-03	16
32	92	9,14E-04	1,00E+00	2,10E-03	32
64	70	9,48E-04	1,00E+00	1,10E-03	64
128	82	6,86E-04	1,00E+00	1,10E-03	128
256	43	5,88E-04	1,00E+00	1,10E-03	256
512	3	4,72E-04	1,00E+00	1,20E-03	512

Из таблицы 5.12 видно, что на 300 эпохах оптимальное значение количества нейронов уже стало 64, но тем не менее есть ещё значения, которые достигаются при максимальном количестве эпох, следовательно, обучение нужно продолжить. Проведём ещё 100 эпох обучения. Результаты представлены в таб. 5.13.

Таблица 5.13 – Исследование влияния количества нейронов на точность нейронной сети на 400 эпохах обучения

neurons	epochs	loss	acc	val_loss	neurons
---------	--------	------	-----	----------	---------

2	96	4,65E-02	9,86E-01	6,69E-02	2
4	99	1,49E-02	9,96E-01	2,63E-02	4
8	95	2,60E-03	9,99E-01	6,80E-03	8
16	96	9,61E-04	1,00E+00	2,30E-03	16
32	70	7,29E-04	1,00E+00	1,60E-03	32
64	50	4,36E-04	1,00E+00	1,10E-03	64
128	17	6,12E-04	1,00E+00	1,10E-03	128
256	55	5,10E-04	1,00E+00	1,10E-03	256
512	29	5,34E-04	1,00E+00	1,40E-03	512

Из таблицы 5.13 видно, что оптимальное значение количества нейронов не поменялось относительно 300 эпох обучения, а оптимальные значения функций потерь и точности при других количествах нейронов уже достигаются на эпохах, меньших максимально заданного количества, что говорит о том, что нейронная сеть достаточно обучилась. Но для того, чтобы окончательно в этом убедиться проведём ещё 100 эпох обучения. Результаты представлены в таблице 5.14.

Таблица 5.14 – Исследование влияния количества нейронов на точность нейронной сети на 500 эпохах обучения

neurons	epochs	loss	acc	val_loss	neurons
2	97	4,43E-02	9,86E-01	6,50E-02	2
4	97	9,70E-03	9,98E-01	2,25E-02	4
8	97	1,80E-03	9,99E-01	4,90E-03	8
16	77	9,06E-04	1,00E+00	2,20E-03	16
32	67	7,10E-04	1,00E+00	1,50E-03	32
64	38	4,98E-04	1,00E+00	9,98E-04	64
128	16	4,71E-04	1,00E+00	1,10E-03	128
256	14	6,08E-04	1,00E+00	1,20E-03	256
512	62	4,91E-04	1,00E+00	1,30E-03	512

Действительно, оптимальное количество нейронов не меняется, по-прежнему 64. Тогда перейдём к исследованию следующего гиперпараметра – скорости обучения нейронной сети. Исследовать этот гиперпараметр будем на 500 эпохах обучения, т.к. из таблицы 5.14 видно, что оптимальное значение достигается на 438 эпохе обучения. Результаты представлены в таблице 5.15.

Таблица 5.15 – Исследование влияния скорости обучения на точность нейронной сети

lr	epochs	loss	acc	val_loss	val_acc
1,00E-04	498	2,00E-03	9,99E-01	1,21E-02	9,96E-01
1,00E-03	468	5,15E-04	1,00E+00	1,00E-03	9,99E-01
1,00E-02	81	5,52E-04	1,00E+00	1,00E-03	1,00E+00
1,00E-01	87	7,05E-02	9,86E-01	8,77E-02	9,87E-01

Из таблицы 5.15 видно, что оптимальное значение скорости обучения 0.01. Установим этот параметр и перейдём к исследованию следующего. В таблице 5.16 представлены результаты исследования влияния параметра decay на точность нейронной сети.

Таблица 5.16 – Исследование влияния параметра decay на точность нейронной сети

decay	epochs	loss	acc	val_loss	val_acc
0,00E+00	54	6,52E-04	1,00E+00	1,10E-03	1,00E+00
1,00E-01	100	2,43E-02	9,95E-01	5,43E-02	9,90E-01
1,00E-02	100	1,00E-03	1,00E+00	3,20E-03	9,98E-01
1,00E-03	90	4,57E-04	1,00E+00	1,20E-03	9,99E-01
1,00E-04	75	5,25E-04	1,00E+00	1,00E-03	9,99E-01
1,00E-05	96	5,48E-04	1,00E+00	1,00E-03	9,99E-01
1,00E-06	47	6,11E-04	1,00E+00	1,10E-03	1,00E+00
1,00E-07	55	7,89E-04	1,00E+00	1,10E-03	9,99E-01

Из таблицы 5.16 видно, что оптимальным значением параметра decay является 1E-04. Установим этот параметр и перейдём к исследованию следующего

го. В таблице 5.17 представлены результаты исследования влияния параметра dropout на точность нейронной сети.

Таблица 5.17 – Исследование влияния параметра dropout на точность нейронной сети

dropout	epochs	loss	acc	val_loss	val_acc
0.0	75	5,25E-04	1,00E+00	1,00E-03	9,99E-01
0.1	99	8,51E-04	1,00E+00	1,70E-03	9,99E-01
0.2	87	9,74E-04	1,00E+00	1,20E-03	9,99E-01
0.3	92	1,80E-03	9,99E-01	2,10E-03	9,99E-01

Из таблицы 5.17 видно, что оптимальным значением параметра dropout является 0. Установим этот параметр и перейдём к исследованию следующего. В таблице 5.18 представлены результаты исследования влияния параметра recurrent_dropout на точность нейронной сети.

Таблица 5.18 – Исследование влияния параметра recurrent_dropout на точность нейронной сети

r_drop	epochs	loss	acc	val_loss	val_acc
0.0	75	5,25E-04	1,00E+00	1,00E-03	9,99E-01
0.1	60	6,09E-04	1,00E+00	1,10E-03	9,99E-01
0.2	55	6,89E-04	1,00E+00	1,10E-03	9,99E-01
0.3	99	6,95E-04	1,00E+00	1,50E-03	9,99E-01

Из таблицы 5.18 видно, что оптимальным значением параметра recurrent_dropout является 0. Установим этот параметр и перейдём к исследованию следующего. В таблице 5.19 представлены результаты исследования влияния параметра batches на точность нейронной сети.

Таблица 5.19 – Исследование влияния параметра batches на точность нейронной сети

batches	epochs	loss	acc	val_loss	val_acc
16	37	4,69E-04	1,00E+00	2,50E-03	9,99E-01
32	75	3,99E-04	1,00E+00	1,40E-03	9,99E-01

64	34	4,97E-04	1,00E+00	1,30E-03	9,99E-01
128	61	5,84E-04	1,00E+00	1,30E-03	9,99E-01
256	71	6,59E-04	1,00E+00	1,20E-03	9,99E-01
512	75	5,25E-04	1,00E+00	1,00E-03	9,99E-01
1024	100	7,27E-04	1,00E+00	1,30E-03	9,99E-01
2048	86	1,00E-03	9,99E-01	2,30E-03	9,99E-01

Из таблицы 5.19 видно, что оптимальным значением параметра *batches* является 512. Установим этот параметр и перейдём к исследованию следующего. Попробуем увеличить количество слоёв нейронной сети на одну единицу и повaryировать количество нейронов во втором слое. Результаты представлены в таблице 5.20.

Таблица 5.20 – Исследование влияния количества нейронов во втором слое нейронной сети на её точность

neurons	epochs	loss	acc	val_loss	val_acc
	75	5,25E-04	1,00E+00	1,00E-03	9,99E-01
2	100	2,08E-02	9,93E-01	4,85E-02	9,90E-01
4	100	7,82E-04	1,00E+00	3,10E-03	9,99E-01
8	93	5,10E-04	1,00E+00	1,30E-03	9,99E-01
16	79	5,56E-04	1,00E+00	1,00E-03	1,00E+00
32	79	8,04E-04	1,00E+00	1,10E-03	9,99E-01
64	46	1,30E-03	1,00E+00	1,10E-03	9,99E-01
128	68	4,99E-04	1,00E+00	1,00E-03	1,00E+00
256	80	5,21E-04	1,00E+00	9,93E-04	9,99E-01
512	100	2,56E-01	9,84E-01	1,71E-01	9,89E-01

В первой строке таблицы 5.20 представлены наилучшие результаты с одним слоем нейронной сети, а дальше уже данные для второго слоя. Видно, что второй слой, действительно, немного улучшил точность нейронной сети и оптимальное количество нейронов во втором слое – 256.

В таблице 5.21 представлены результаты исследования величины dropout регуляризации второго слоя нейронной сети.

Таблица 5.21 – Исследование влияния dropout регуляризации второго слоя нейронной сети на её точность

dropout	epochs	loss	acc	val_loss	val_acc
0.0	80	5,21E-04	1,00E+00	9,93E-04	9,99E-01
0.1	59	7,95E-04	1,00E+00	1,10E-03	1,00E+00
0.2	37	6,16E-04	1,00E+00	1,10E-03	1,00E+00
0.3	45	6,42E-04	1,00E+00	1,00E-03	9,99E-01
0.4	53	5,93E-04	1,00E+00	1,10E-03	1,00E+00
0.5	38	6,79E-04	1,00E+00	1,00E-03	9,99E-01
0.6	57	5,72E-04	1,00E+00	9,96E-04	9,99E-01
0.7	91	4,73E-04	1,00E+00	9,86E-04	1,00E+00
0.8	40	6,85E-04	1,00E+00	1,10E-03	9,99E-01
0.9	72	6,66E-04	1,00E+00	1,00E-03	1,00E+00

Из таблицы 5.21 видно, что оптимальное значение достигается при значении dropout равном 0.7. Установим это значение и исследуем влияние recurrent_dropout регуляризации второго слоя нейронной сети.

Таблица 5.22 – Исследование влияния recurrent_dropout регуляризации второго слоя нейронной сети на её точность

rec_drop	epochs	loss	acc	val_loss	val_acc
0.0	91	4,73E-04	1,00E+00	9,86E-04	1,00E+00
0.1	80	4,86E-04	1,00E+00	1,00E-03	1,00E+00
0.2	43	1,30E-03	1,00E+00	1,20E-03	9,99E-01
0.3	50	5,93E-04	1,00E+00	1,10E-03	9,99E-01
0.4	42	6,68E-04	1,00E+00	1,10E-03	1,00E+00
0.5	84	5,32E-04	1,00E+00	1,10E-03	9,99E-01
0.6	82	5,37E-04	1,00E+00	1,20E-03	9,99E-01
0.7	25	3,20E-03	9,99E-01	4,20E-03	9,98E-01

0.8	17	4,50E-03	9,99E-01	1,05E-02	9,96E-01
0.9	96	1,26E-01	9,84E-01	1,51E-01	9,87E-01

Из таблицы 5.22 видно, что оптимальной величиной recurrent_dropout второго слоя нейронной сети является 0.0. Попробуем добавить третий lstm слой нейронной сети и исследовать влияние количества нейронов на точность в нём на точность нейронной сети. Результаты представлены в таб. 5.23.

Таблица 5.23 – Исследование количества нейронов третьего lstm слоя нейронной сети на её точность

neurons	epochs	loss	acc	val_loss	val_acc
	91	4,73E-04	1,00E+00	9,86E-04	1,00E+00
2	6	1,27E-01	9,84E-01	1,28E-01	9,87E-01
4	4	1,27E-01	9,84E-01	1,30E-01	9,87E-01
8	2	1,27E-01	9,84E-01	1,30E-01	9,87E-01
16	2	1,26E-01	9,84E-01	1,38E-01	9,87E-01
32	70	7,26E-04	1,00E+00	1,00E-03	1,00E+00
64	89	6,01E-04	1,00E+00	1,10E-03	9,99E-01
128	99	1,10E-03	9,99E-01	2,30E-03	9,99E-01
256	92	5,39E-04	1,00E+00	1,10E-03	1,00E+00
512	100	3,10E-03	9,99E-01	3,90E-03	9,98E-01

В первой строке таб. 5.23 представлен лучший результат при двух слоях нейронной сети. Далее, показано изменение точности нейронной сети при трёх слоях и разном количестве нейронов. Видно, что третий слой не позволяет улучшить точность нейронной сети. Таким образом, наиболее оптимальные параметры для нейронной сети, генерирующей подмаски неразмеченного текстового контента определены и представлены в таб. 5.24.

Таблица 5.24 – Оптимальные гиперпараметры нейронной сети для решения задачи генерации подмасок неразмеченного текстового контента

Гиперпараметр	Значение гиперпараметра
Количество lstm слоёв	2

Количество нейронов в первом слое	64
Количество нейронов во втором слое	256
Количество эпох обучения	91
Скорость обучения	0.01
Величина decay	1E-4
Величина dropout первого слоя	0
Величина dropout второго слоя	0.7
Величина recurrent_dropout первого слоя	0
Величина recurrent_dropout второго слоя	0
Величина batches	512

В результате обучения нейронной сети с гиперпараметрами, представленными в таб. 5.24, нейронная сеть достигла ошибки на обучающем наборе данных $4.73\text{E-}04$, а на валидационном наборе данных $9.86\text{E-}04$. На рис. 5.3 представлен график сходимости нейронной сети на обучающей и валидационной выборке при заданных гиперпараметрах.

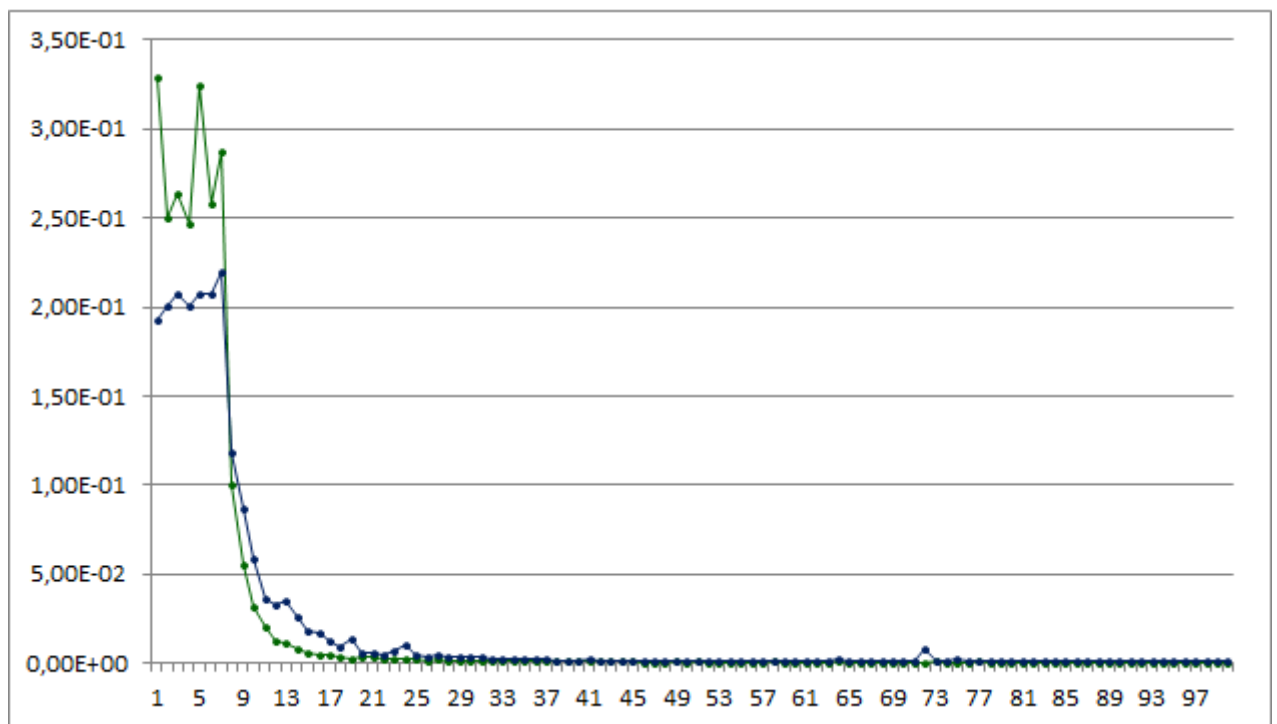


Рисунок 5.3 – График зависимости значений функции потерь на обучающем наборе данных (loss, зелёная линия) и на валидационном наборе данных (val_loss, синяя линия) от количества эпох

На рис. 5.4 представлен график изменения точности нейронной сети на обучающей и валидационной выборке в зависимости от количества эпох при заданных гиперпараметрах.

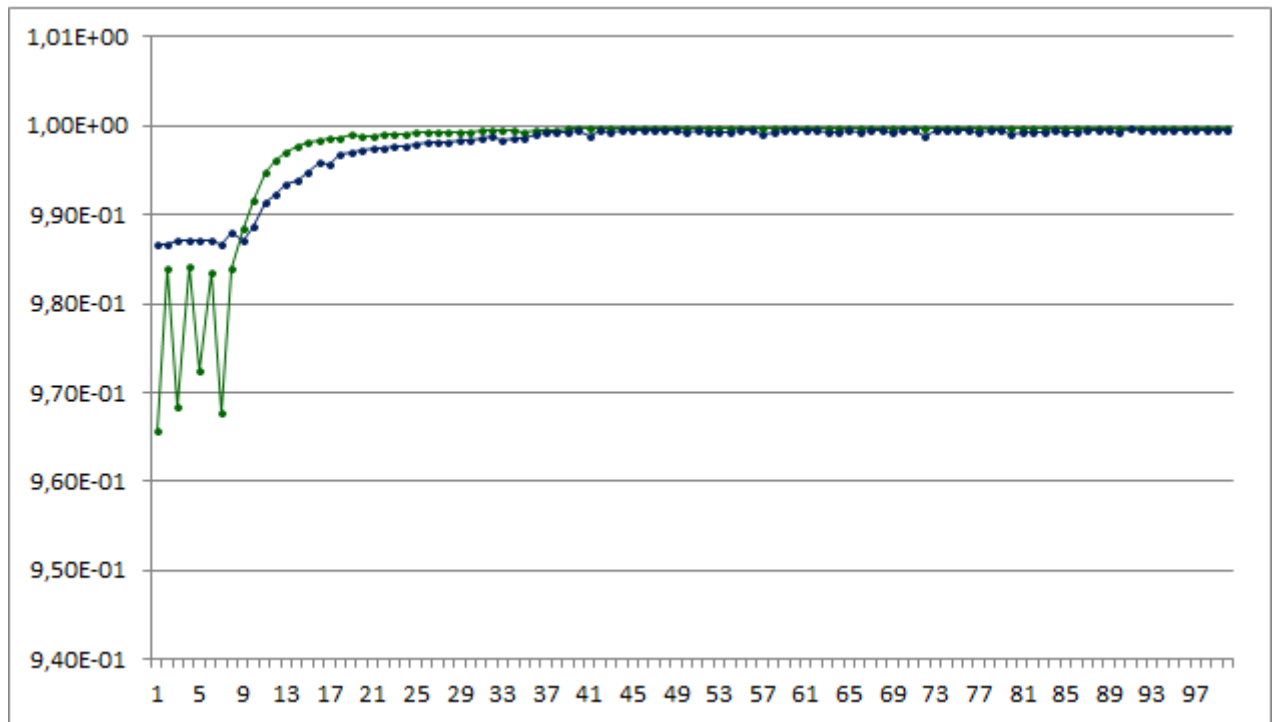


Рисунок 5.4 – График зависимости точности на обучающем наборе данных (асс, зелёная линия) и на валидационном наборе данных (val_асс, синяя линия) от количества эпох

5.1.3 Исследование нейронной сети для решения задачи генерации подмасок размеченного текстового контента

Исследование проводилось аналогично предыдущим двум нейронным сетям. В таблице 5.25 представлены оптимальные параметры для этой нейронной сети.

Таблица 5.25 – Оптимальные гиперпараметры нейронной сети для решения задачи генерации подмасок размеченного текстового контента

Гиперпараметр	Значение гиперпараметра
Количество lstm слоёв	1
Количество нейронов в слое	128
Количество эпох обучения	945

Скорость обучения	0.001
Величина decay	1E-5
Величина dropout	0.1
Величина recurrent_dropout	0.1
Величина batches	32

В результате обучения нейронной сети с гиперпараметрами, представленными в таб. 5.25, нейронная сеть достигла ошибки на обучающем наборе данных $8.60\text{E-}04$, а на валидационном наборе данных $8.62\text{E-}04$.

5.1.4 Исследование нейронной сети для решения задачи идентификации типа неразмеченного текстового контента

В таблице 5.26 представлены оптимальные параметры для нейронной сети, идентифицирующей тип неразмеченного текстового контента.

Таблица 5.26 – Оптимальные гиперпараметры нейронной сети для решения задачи идентификации типа неразмеченного текстового контента

Гиперпараметр	Значение гиперпараметра
Количество lstm слоёв	1
Количество нейронов в слое	8
Количество эпох обучения	114
Скорость обучения	0.01
Величина decay	0
Величина dropout	0
Величина recurrent_dropout	0
Величина batches	1024

В результате обучения нейронной сети с гиперпараметрами, представленными в таб. 5.26, нейронная сеть достигла ошибки на обучающем наборе данных $1.19\text{E-}07$, а на валидационном наборе данных $1.19\text{E-}07$.

5.2 Исследование технологии автоматизированной разметки текстового контента

Декомпозиция задачи разметки текстового контента позволяет разбить сложную задачу на несколько более простых подзадач, решить которые становится уже значительно легче. Однако минусы такого подхода к решению сложной задачи заключаются в том, что решение подзадачи на очередном этапе производится на данных, полученных в результате решения подзадачи предыдущего этапа. Это означает, что если на предыдущем этапе подзадача решена с некоторой ошибкой, то она распространится и усилится на следующем этапе решения задачи. Так, например, во втором этапе решения задачи, при генерации подмасок неразмеченного текстового контента, нейронная сеть ошибается с вероятностью 0.1 %, а на третьем этапе, при решении задачи генерации подмасок размеченного текстового контента, нейронная сеть ошибается уже с вероятностью 5.04%, в то время, как на идеальных входных данных на третьем этапе нейронная сеть ошибается с вероятностью 0.09%. Впрочем, подзадачи следующих этапов решаются уже классическим программированием, где появляется возможность снизить ошибку, полученную на предыдущем этапе. Так, например, в результате решения третьей подзадачи, наиболее часто возникают ошибки при генерации подмасок размеченного текстового контента для типа вопроса с множественным выбором ответа (таб. 5.27).

Таблица 5.27 – Результаты работы нейронной сети для генерации подмасок размеченного текстового контента на данных, полученных от нейронной сети для генерации подмасок неразмеченного текстового контента для вопроса с множественным выбором ответа

Пример	Целевые данные	Предсказанные данные
1	[item]T[/item]	[item]T[/item]
	[multichoice][multichoice]	[multichoice][multichoice]
	numbering=123.;	numbering=123.;
	[choice]T[/choice]	[choice]T[/choice]

	value=-50%	value=-50%
	[choice]T[/choice]	[choice]T[/choice]
	value=33%	value=33%
	[choice]T[/choice]	[choice]T[/choice]
	value=-50%	value=-100%
	[choice]T[/choice]	[choice]T[/choice]
	value=33%	value=33%
	[choice]T[/choice]	[choice]T[/choice]
	value=33%	value=33%
2	[item]T[/item]	[item]T[/item]
	[multichoice][[/multichoice]	[multichoice][[/multichoice]
	numbering=123.;	numbering=123.;
	[choice]T[/choice]	[choice]T[/choice]
	value=-33%	value=-33%
	[choice]T[/choice]	[choice]T[/choice]
	value=-33%	value=-50%
	[choice]T[/choice]	[choice]T[/choice]
	value=50%	value=50%
	[choice]T[/choice]	[choice]T[/choice]
	value=-33%	[choice]T[/choice]
	[choice]T[/choice]	[choice]T[/choice]
	value=50%	value=50%
3	[item]T[/item]	[item]T[/item]
	[multichoice][[/multichoice]	[multichoice][[/multichoice]
	numbering=123.;	numbering=123.;
	[choice]T[/choice]	[choice]T[/choice]
	value=-33%	value=-33%
	[choice]T[/choice]	[choice]T[/choice]
	value=50%	value=50%

	[choice]T[/choice]	[choice]T[/choice]
	value=-33%	[multichoice][multichoice]
	[choice]T[/choice]	[choice]T[/choice]
	value=-33%	value=-50%
	[choice]T[/choice]	[choice]T[/choice]
	value=50%	value=50%
4	[item]T[/item]	[item]T[/item]
	[multichoice][multichoice]	[multichoice][multichoice]
	numbering=123.;	numbering=123.;
	[choice]T[/choice]	[choice]T[/choice]
	value=-33%	value=-33%
	[choice]T[/choice]	[choice]T[/choice]
	value=-33%	value=-50%
	[choice]T[/choice]	[choice]T[/choice]
	value=50%	value=50%
	[choice]T[/choice]	[choice]T[/choice]
	value=-33%	[choice]T[/choice]
	[choice]T[/choice]	[choice]T[/choice]
	value=50%	value=50%
5	[item]T[/item]	[item]T[/item]
	[multichoice][multichoice]	[multichoice][multichoice]
	numbering=123.;	numbering=123.;
	[choice]T[/choice]	[choice]T[/choice]
	value=50%	value=50%
	[choice]T[/choice]	[choice]T[/choice]
	value=-33%	[multichoice][multichoice]
	[choice]T[/choice]	[choice]T[/choice]
	value=-33%	value=-33%
	[choice]T[/choice]	[choice]T[/choice]

	value=-33%	[multichoice][multichoice]
	[choice]T[/choice]	[choice]T[/choice]
	value=50%	value=50%

Анализируя фрагменты каждого примера несложно восстановить некорректно сгенерированный элемент. Например, в первом варианте видно ошибку значения атрибута value. Согласно правилам разметки для данного типа вопроса значение атрибута value у тега choice рассчитывается исходя из количества правильных и неправильных ответов. Среди сгенерированных фрагментов имеются три тега choice с атрибутами value, значение которых 33%. Поскольку число 33 положительно, то эти теги определяют правильные варианты ответов. Если просуммировать значения этих атрибутов и округлить результат, получим 100%, следовательно, среди оставшихся тегов choice больше нет тех, которые определяют правильные варианты ответов. Согласно количеству оставшихся тегов choice неправильных вариантов ответов должно быть два, следовательно, атрибут value у них должен принимать значение -50%, откуда легко вычисляется ошибочное значение атрибута и ликвидируется ошибка. Аналогично ликвидируются ошибки для других вариантов в таблице 5.27.

5.3 Анализ результатов

Таким образом, технологию создания тестовых заданий теперь можно представить восьмью этапами (рис. 5.5). Первый и последние два этапа остались неизменными. Второй этап разбился на 5 дополнительных этапов, первые три из которых выполняются с помощью нейронных сетей, а последние два классическим программированием.

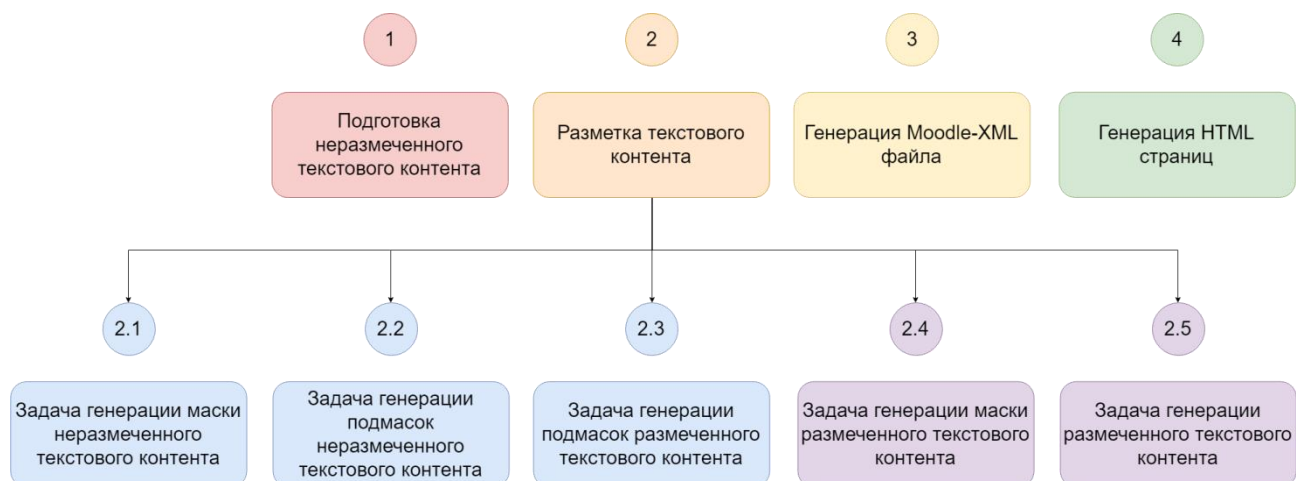


Рисунок 5.5 – Технология создания тестовых заданий для систем дистанционного обучения на основе применения нейронных сетей

Как показало исследование нейронных сетей, на идеальных данных точность первой модели 100 %, второй – 99.9% и третьей – 99.91%. Однако, в действительности, на вход очередной нейронной сети подаются не идеальные данные, а данные сгенерированные предыдущей нейронной сетью. В этом случае точность первых двух моделей остаётся неизменной, а точность третьей модели становится равной 94.96%. Поскольку процент правильных ответов по-прежнему остаётся достаточно большим, на четвёртом этапе удаётся частично устранить ошибки нейронных сетей и отправить на следующий этап уже полностью корректные данные.

Таким образом, появляется ещё один вариант технологии разработки образовательного контента (рис. 5.6), в котором рутинная работа технического специалиста, связанная с разметкой текстового контента автоматизируется с помощью искусственного интеллекта, что значительно уменьшает затраты на создание образовательного контента.

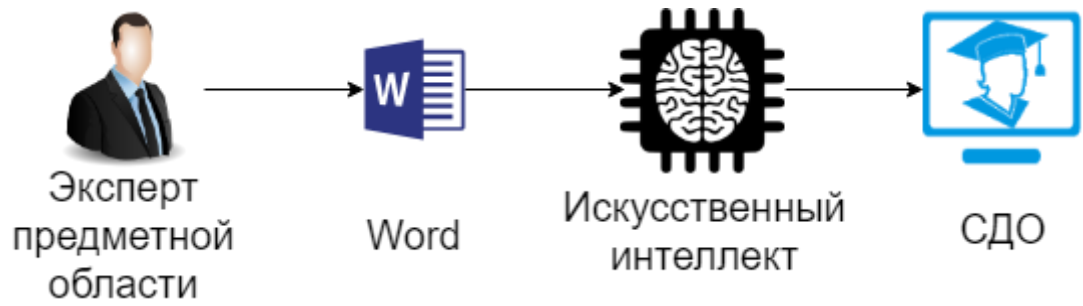


Рисунок 5.6 – Схематичное представления технологии разработки образовательного контента на основе применения нейронных сетей

6 Заключение

В ходе выполнения работы были получены следующие результаты:

1) проведён обзор предметной области:

1.1) в лучшей степени критериям удовлетворяют технологии создания тестовых заданий в СДО и технологии, основанные на применении нейронных сетей; комбинация этих технологий, т.е. применение нейросетей к технологии СДО позволяет получить технологию, которая лучше всего удовлетворяет критериям.

1.2) для обработки слабоструктурированных текстовых данных наиболее целесообразно применять нейронные сети;

1.3) для обработки слабоструктурированных текстовых данных применяются два вида нейронных сетей: рекуррентные и свёрточные; для решения задачи были выбраны рекуррентные нейронные сети;

2) разработана технология автоматизированного создания тестовых заданий:

2.1) задача разметки текстового контента декомпозирована на пять более простых задач, первые три из которых решаются с помощью нейронных сетей, а последние две классическим программированием;

2.2) разработаны модели нейронных сетей для решения первых трёх задач технологии разметки текстового контента; модели содержат не более двух lstm слоёв с принципами работы many-to-many или many-to-one, в зависимости от формы входных и выходных данных;

3) реализована технология автоматизированного создания тестовых заданий:

3.1) реализованы модели нейронных сетей для решения первых трёх задач технологии разметки текстового контента; для реализации моделей нейронных сетей использован фреймворк Keras;

3.2) реализованы программы для решения последних двух задач технологии разметки текстового контента и генераторы данных для обучения нейронных сетей;

4) проведено исследование разработанной технологии создания тестовых заданий:

4.1) исследование нейронных сетей позволило выявить оптимальные гиперпараметры, в результате чего процент ошибок нейронных сетей на валидационном наборе данных достиг величин: $4.78E-06$, $9.86E-04$, $8,62E-04$ и $1.19E-07$.

4.2) исследование технологии показало, что благодаря малому проценту ошибок нейронных сетей, можно восстановить правильные ответы на четвёртом этапе технологии разметки текстового контента;

План дальнейшей работы включает в себя:

1) подготовку большего количества данных для разработанных нейронных сетей;

2) внедрение разработанной технологии в эксплуатацию на ФДО ТУСУР.

Сокращения, обозначения, термины и определения

ТУСУР – Томский университет систем управления и радиоэлектроники.

АСУ – Автоматизированные системы управления.

ФДО – Факультет дистанционного обучения.

СДО – Система дистанционного обучения.

DNN – Deep Neural Networks.

NLP – Natural Language Processing.

CNN – Convolutional Neural Network.

RNN – Recurrent Neural Network.

LSTM – Long Short-Term Memory.

Список использованных источников

1 Кручинин В.В., Магазинников Л.И., Морозова Ю.В. Модели и алгоритмы компьютерных самостоятельных работ на основе генерации тестовых заданий // Известия Томского политехнического университета. Инжиниринг георесурсов. 2006. Т. 309. № 8. С. 258-263.

2 Мисюра В.В., Бубякин К.И., Лучинский Е.О. Программный модуль генерации тестовых заданий по учебным дисциплинам // Известия Ростовского государственного строительного университета. 2015. Т. 2. № 20. С. 163-168.

3 Посов И.А. Образовательные технологии и общество. Обзор генераторов и методов генерации учебных заданий // 2014. Т. 17. № 4. С. 593-609.

4 Дударев О.К. Тесты с автоматической генерацией заданий // Образование и наука в современных условиях. 2015. № 3. С. 209-210.

5 Осокин А. Е. Использование вычисляемого типа тестовых вопросов в системе Moodle // Информация и образование: границы коммуникаций. 2013. № 5 (13). С. 385-386.

6 Кручинин В.В., Морозова Ю.В. Модели и алгоритмы генерации задач в компьютерном тестировании // Известия Томского политехнического университета. Инжиниринг георесурсов. 2004. Т. 307. № 5. С. 127-131.

7 Паркалова О.В. Проектирование генератора тестовых заданий для системы компьютерного тестирования // В сборнике: Международный конгресс по информатике: информационные системы и технологии материалы международного научного конгресса. С. В. Абламейко (гл. редактор). 2016. С. 720-723.

8 Куртасов А.М., Швецов А.Н. Метод автоматизированной генерации заданий для тестов контроля знаний из текстов учебных пособий // Современные информационные технологии и ИТ-образование. 2013. № 9. С. 218-228.

9 Алсынбаева Л.Г. Об одном подходе к автоматизированной генерации тестовых заданий для контента инструктивного ТИПА // Информационно-управляющие системы. 2009. № 5. С. 79-83.

10 Братчиков И.Л. Генерация тестовых заданий в экспертно-обучающих системах // Вестник Российского университета дружбы народов. Серия: Информатизация образования. 2012. № 2. С. 47-60.

11 Кручинин В.В. Использование деревьев и/или для генерации вопросов и задач // Вестник Томского государственного университета. 2004. № 284. С. 182-186.

12 Зорин Ю.А. Интерпретатор языка построения генераторов тестовых заданий на основе деревьев и/или // Доклады Томского государственного университета систем управления и радиоэлектроники. 2013. № 1 (27). С. 75-79.

13 Сергушичева А.П., Швецов А.Н. Алгоритм реализации метода автоматической генерации тестовых заданий // Алгоритмы, методы и системы обработки данных. 2006. № 11. С. 118-125.

14 Личаргин Д.В., Усова А.А., Сотникова В.В., Липман С.А., Бутовченко В.В. Разработка приложения по генерации учебных заданий к тексту на естественном языке на основе порождаемых шаблонов // Современные проблемы науки и образования. 2015. № 2-2. С. 120.

15 Использование семантических сетей в генераторах заданий [Электронный ресурс]. – Режим доступа: http://www.rusnauka.com/20_AND_2014/Informatica/3_174474.doc.htm

16 Куртасов А.М., Швецов А.Н. Метод автоматизированной генерации контрольно-тестовых заданий из текста учебных материалов // Вестник Череповецкого государственного университета. 2014. № 7 (60). С. 7-11.

17 Дерябкин В.П., Пиявский С.А., Пузанков Н.М. Интеллектуальная информационная система тестирования знаний // Перспективные информационные технологии (ПИТ 2015), Том 2: труды Международной научно-тех. конф. Самара: Изд-во Самарского научного центра РАН, - 2015. С. 141-145. 4. М

18 Титенко С.В. Автоматизации построения тестовых заданий в системах дистанционного обучения на основе понятийно-тезисной модели // Образовательные технологии и общество. 2013. Т. 16. № 1. С. 463-481.

19 Кутукова Е.С. Технология text mining // Научные труды SWorld. 2013. Т. 30. № 4. С. 33-36.

20 Гершевич Е.К., Кукарцев В.В. Технологии text mining и web mining // Молодежный научный форум: технические и математические науки. 2016. № 4 (33). С. 25-28.

21 Алексеев А.А., Катасёв А.С., Кириллов А.Е., Кирпичников А.П. Классификация текстовых документов на основе технологии text mining // Вестник Казанского технологического университета. 2016. Т. 19. № 18. С. 116-119.

22 Федюшкин Н.А., Федосин С.А. Основные технологии интеллектуального анализа текста // В сборнике: Развитие технических наук в современном мире сборник научных трудов по итогам международной научно-практической конференции. 2016. С. 21-26.

23 Джулли А., Пал С. Библиотека Keras - инструмент глубокого обучения. Реализация нейронных сетей с помощью библиотек Theano и TensorFlow / пер. с англ. Слинкин А.А. - М.: ДМК Пресс, 2018. - 294 с.: ил.

24 Азбука ИИ: «Рекуррентные нейросети» [Электронный ресурс]. – Режим доступа: <https://nplus1.ru/material/2016/11/04/recurrent-networks> (дата обращения: 17.03.2018)

25 Hochreiter, S., Schmidhuber, J. Long short-term memory / S. Hochreiter, J. Schmidhuber // Neural Computation 9(8):1735-1780, 1997 г., 32 с.

26 Машинное обучение. Фреймворки и платформы для разработчиков [Электронный ресурс]. – Режим доступа: <https://appttractor.ru/ML/> (дата обращения: 03.03.2018)

27 TensorFlow [Электронный ресурс]. – Режим доступа: <https://www.tensorflow.org/> (дата обращения: 03.03.2018)

28 Download Theano [Электронный ресурс]. – Режим доступа: <https://github.com/Theano/Theano> (дата обращения: 03.03.2018)

29 Keras: The Python Deep Learning library [Электронный ресурс]. – Режим доступа: <https://keras.io/> (дата обращения: 03.03.2018)

Приложение А

(справочное)

Примеры неразмеченного текстового контента

Таблица А.1 – Примеры неразмеченного текстового контента

Тип вопроса	Неразмеченный текстовый контент		
Вопрос с оди- ночным выбо- ром ответа	Укажите административный центр Томской области: 1. ~ Асино; 2. ~ Северск; 3. ~ Стрежевой; 4. = Томск;	Укажите административный центр Томской области: 1. Асино; 2. Северск; 3. Стрежевой; 4. Томск;	Укажите административный центр Томской области: 1. Асино; 2. Северск; 3. Стрежевой; 4. Томск;
Вопрос с множе- ственным выбо- ром ответа	Укажите города Томской области: 1. = Томск; 2. ~ Кожевниково; 3. = Северск; 4. = Асино; 5. ~ Юрга;	Укажите города Томской области: 1. Томск; 2. Кожевниково; 3. Северск; 4. Асино; 5. Юрга;	Укажите города Томской области: 1. Томск; 2. Кожевниково; 3. Северск; 4. Асино; 5. Юрга;
Вопрос с вводом текстового отве- та	Какой город является административным центром Томской области? {= Томск}	Какой город является административным центром Томской области? [Томск]	Какой город является административным центром Томской области? <u>Томск</u>
Вопрос с вводом числового отве- та	Укажите численность населения г. Томска в 2016 году. Ответ укажите в тыс. чел. {# 569 293}	Укажите численность населения г. Томска в 2016 году. Ответ укажите в тыс. чел. <i>[569 293]</i>	Укажите численность населения г. Томска в 2016 году. Ответ укажите в тыс. чел. 569 293
Вопрос на со- ответствие	Сопоставьте населённые пункты Томской области и год их основания. 1) Томск 2) Северск 3) Каргасок 4) Кривошеино а) 1640 б) 1604 в) 1826 г) 1949 Ответ: 1-б, 2-г, 3-а, 4-в	Сопоставьте населённые пункты Томской области и год их основания. Ответ: <i>Томск – 1604, Северск 1949, Каргасок – 1640, Кривошеино – 1826</i>	Сопоставьте населённые пункты Томской области и год их основания. Населённые пункты: Томск, Северск, Каргасок, Кривоше- ино Года основания: 1640, 1604, 1826, 1949
Вопрос на упо- рядочивание	Расставьте населённые пункты Том- ской области по порядку согласно году их основания. 1) Каргасок 2) Северск 3) Томск 4) Кривошеино Ответ: 3, 1, 4, 2	Расставьте населённые пункты Том- ской области по порядку согласно году их основания. Ответ: <i>Томск, Каргасок, Кривошеино, Северск</i>	Расставьте населённые пункты Том- ской области по порядку согласно году их основания. Населённые пункты: Кривошеино, Каргасок, Томск, Северск
Составной во- прос	Укажите административный центр Томской области: 1. ~ Асино; 2. ~ Северск; 3. ~ Стрежевой;	Укажите административный центр Томской области: 1. Асино; 2. Северск; 3. Стрежевой;	Укажите административный центр Томской области: 1. Асино; 2. Северск; 3. Стрежевой;

	4. = Томск; Он был основан в {# 1604} году.	4. Томск; Он был основан в [1604] году.	4. Томск; Он был основан в 1604 году.
--	--	---	---

Приложение Б

(справочное)

Примеры масок неразмеченного текстового контента

Таблица Б.1 – Примеры масок неразмеченного текстового контента

Тип вопроса	Неразмеченный текстовый контент / Маска неразмеченного текстового контента
Вопрос с оди- ночным выбо- ром ответа	<p><r>Укажите административный центр Томской области:</r><r>1. ~ Асино;</r><r>2. ~ Северск;</r> <r>_____ </r><r>1. ~ ____;</r><r>2. ~ ____;</r></p> <p><r>3. ~ Стрежевой;</r><r>4. = Томск;</r> <r>3. ~ ____;</r><r>4. = ____;</r></p>
Вопрос с мно- жественным выбором ответа	<p><r>Укажите города Томской области:</r><r>1. = Томск;</r><r>2. ~ Кожевниково;</r> <r>_____ </r><r>1. = ____;</r><r>2. ~ ____;</r></p> <p><r>3. = Северск;</r><r>4. = Асино;</r><r>5. ~ Юрга;</r> <r>3. = ____;</r><r>4. = ____;</r><r>5. ~ ____;</r></p>
Вопрос с вво- дом текстового ответа	<p><r>Какой город является административным центром Томской области?</r><r>{= Томск}</r> <r>_____ </r><r>{= ____}</r></p>
Вопрос с вво- дом числового ответа	<p><r>Укажите численность населения г. Томска в 2016 году. Ответ укажите в тыс. чел. <r>_____ </r></p> <p>{# 569 293}</r> {# ____}</r></p>
Вопрос на со- ответствие	<p><r>Сопоставьте населённые пункты Томской области и год их основания.</r><r>1) Томск</r> <r>1) ____</r><r>_____ </r></p> <p><r>2) Северск</r><r>3) Каргасок</r><r>4) Кривошеино</r><r>a) 1640</r><r>b) 1604</r> <r>2) ____</r><r>3) ____</r><r>4) ____</r><r>a) ____</r><r>b) ____</r></p> <p><r>в) 1826</r><r>г) 1949</r><r>0ответ: 1-б, 2-г, 3-а, 4-в</r> <r>в) ____</r><r>г) ____</r><r>0ответ: 1-б, 2-г, 3-а, 4-в</r></p>
Вопрос на упо- рядочивание	<p><r>Расставьте населённые пункты Томской области по порядку согласно году их основания.</r> <r>_____ </r></p> <p><r>1) Каргасок</r><r>2) Северск</r><r>3) Томск</r><r>4) Кривошеино</r><r>0ответ: 3, 1, 4, 2</r> <r>1) ____</r><r>2) ____</r><r>3) ____</r><r>4) ____</r><r>0ответ: 3, 1, 4, 2</r></p>
Составной вопрос	<p><r>Укажите административный центр Томской области:</r><r>1. ~ Асино;</r><r>2. ~ Северск;</r> <r>_____ </r><r>1. ~ ____;</r><r>2. ~ ____;</r></p> <p><r>3. ~ Стрежевой;</r><r>4. = Томск;</r><r>Он был основан в {# 1604} году.</r> <r>3. ~ ____;</r><r>4. = ____;</r><r>____ {# ____} ____</r></p>

Приложение В

(справочное)

Примеры подмасок неразмеченного текстового контента

Таблица В.1 – Примеры подмасок неразмеченного текстового контента

Тип вопроса	Маска неразмеченного текстового контента / Подмаски неразмеченного текстового контента
Вопрос с единственным выбором ответа	<p><p>Т</p><p>1.~Т;</p><p>2.~Т;</p><p>3.~Т;</p><p>4.=Т;</p></p> <p>____Т____</p> <p>____~Т____~Т____~Т____=Т____</p> <p>____1.____;____2.____;____3.____;____4.____;</p> <p><p>_</p><p>____</p><p>____</p><p>____</p><p>____</p></p> <p>____~Т____</p> <p>____~Т____</p> <p>____~Т____</p> <p>____=Т____</p>
Вопрос с множественным выбором ответа	<p><p>Т</p><p>1.=Т;</p><p>2.~Т;</p><p>3.=Т;</p><p>4.=Т;</p><p>5.~Т;</p></p> <p>____Т____</p> <p>____=Т____~Т____=Т____=Т____~Т____</p> <p>____1.____;____2.____;____3.____;____4.____;____5.____;</p> <p>____=Т____</p> <p>____=Т____=Т____=Т____</p> <p>____~Т____</p> <p>____~Т____~Т____</p> <p>____=Т____</p> <p>____=Т____=Т____=Т____</p> <p>____=Т____</p> <p>____~Т____</p> <p>____~Т____~Т____</p>
Вопрос с вводом текстового ответа	<p><p>Т</p><p><{=Т}</p></p> <p>____Т____</p> <p>____{=Т}____</p>
Вопрос с вводом числового ответа	<p><p>Т{#Т}</p></p> <p>____Т____</p> <p>____{#Т}____</p>
Вопрос на соответствие	<p><p>Т</p><p><p>1)Т</p><p>2)Т</p><p>3)Т</p><p>4)Т</p><p>а)Т</p><p>б)Т</p><p>в)Т</p><p>г)Т</p><p>Ответ: 1-б, 2-г, 3-а, 4-в</p></p> <p>____Т____</p> <p>____1-б, 2-г, 3-а, 4-в____</p> <p>____1)____2)____3)____4)____</p> <p>____1-б____</p> <p>____2-г____</p> <p>____3-а____</p> <p>____4-в____</p>
Вопрос на упорядочивание	<p><p>Т</p><p><p>1)Т</p><p>2)Т</p><p>3)Т</p><p>4)Т</p><p>Ответ: 3, 1, 4, 2</p></p> <p>____Т____</p> <p>____3, 1, 4, 2____</p> <p>____1)____2)____3)____4)____</p> <p>____3____</p> <p>____1____</p> <p>____4____</p> <p>____2____</p>
Составной вопрос	<p><p>Т</p><p><p>1.~Т;</p><p>2.~Т;</p><p>3.~Т;</p><p>4.=Т;</p><p>Т{#Т}Т</p></p> <p>____Т____</p> <p>____~Т____~Т____~Т____=Т____</p> <p>____1.____;____2.____;____3.____;____4.____;</p> <p><p>_</p><p>____</p><p>____</p><p>____</p><p>____</p><p>____</p></p> <p>____~Т____</p> <p>____~Т____</p> <p>____~Т____</p> <p>____=Т____</p> <p>____Т{#Т}Т____</p>

Приложение Г

(справочное)

Примеры подмасок размеченного текстового контента

Таблица Г.1 – Примеры подмасок размеченного текстового контента

Тип вопроса	Подмаска неразмеченного текстового контента / Подмаска размеченного текстового контента
Вопрос с одиночным выбором ответа	<p>in: __T__</p> <p>out: [item]T[/item]</p> <p>in: _____~T_____~T_____~T_____ =T_____</p> <p>out: [singlechoice][singlechoice]</p> <p>in: _____1.____;_____2.____;_____3.____;_____4.____;_____</p> <p>out: numbering=123.;</p> <p>in: <p>_</p><p>_____</p><p>_____</p><p>_____</p><p>_____</p></p> <p>out: view=vertical</p> <p>in: _____~T_____</p> <p>out: [choice]T[/choice]</p> <p>in: _____~T_____</p> <p>out: [choice]T[/choice]</p> <p>in: _____~T_____</p> <p>out: [choice]T[/choice]</p> <p>in: _____ =T_____</p> <p>out: [choice value=1]T[/choice]</p>
Вопрос с множественным выбором ответа	<p>in: __T__</p> <p>out: [item]T[/item]</p> <p>in: _____=T_____~T_____ =T_____ =T_____~T_____</p> <p>out: [multichoice][multichoice]</p> <p>in: _____1.____;_____2.____;_____3.____;_____4.____;_____5.____;_____</p> <p>out: numbering=123.;</p> <p>in: _____ =T_____</p> <p>out: [choice]T[/choice]</p> <p>in: _____=T_____ =T_____ =T_____</p> <p>out: value=33%</p> <p>in: _____~T_____</p> <p>out: [choice]T[/choice]</p> <p>in: _____~T_____~T_____</p> <p>out: value=-50%</p> <p>in: _____ =T_____</p> <p>out: [choice]T[/choice]</p> <p>in: _____=T_____ =T_____ =T_____</p> <p>out: value=33%</p> <p>in: _____ =T_____</p> <p>out: [choice]T[/choice]</p> <p>in: _____=T_____ =T_____ =T_____</p> <p>out: value=33%</p> <p>in: _____~T_____</p> <p>out: [choice]T[/choice]</p> <p>in: _____~T_____~T_____</p> <p>out: value=-50%</p>

[illegible]

	<div>out: [choice value=1]T[/choice]</div> <div>in: _____T{#T}T_____</div> <div>out: T[numberinput][answer]T[/answer][numberinput]T</div>
--	---

Приложение Д

(справочное)

Примеры масок размеченного текстового контента

Таблица Д.1 – Примеры масок размеченного текстового контента

Тип вопроса	Маска размеченного текстового контента
Вопрос с оди- ночным выбо- ром ответа	<pre>[item] Т [singlechoice numbering=123.; view=vertical] [choice]Т[/choice] [choice]Т[/choice] [choice]Т[/choice] [choice value=1]Т[/choice] [/singlechoice] [/item]</pre>
Вопрос с мно- жественным выбором ответа	<pre>[item] Т [multichoice numbering=123.;] [choice value=33%]Т[/choice] [choice value=-50%]Т[/choice] [choice value=33%]Т[/choice] [choice value=33%]Т[/choice] [choice value=-50%]Т[/choice] [/multichoice] [/item]</pre>
Вопрос с вводом текстового отве- та	<pre>[item] Т [textinput] [answer]Т[/answer] [/textinput] [/item]</pre>
Вопрос с вводом числового отве- та	<pre>[item] Т [numberinput] [answer]Т[/answer] [/numberinput] [/item]</pre>
Вопрос на соот- ветствие	<pre>[item] Т [matching numbering=123)] [subitem]1[answer]6[/answer][[/subitem] [subitem]2[answer]r[/answer][[/subitem] [subitem]3[answer]a[/answer][[/subitem] [subitem]4[answer]B[/answer][[/subitem] [/matching] [/item]</pre>
Вопрос на упо- рядочивание	<pre>[item] Т</pre>

	<pre> [ordering numbering=123] [subitem]3[/subitem] [subitem]1[/subitem] [subitem]4[/subitem] [subitem]2[/subitem] [/ordering] [/item] </pre>
Составной во-прос	<pre> [cloze] T [singlechoice numbering=123.; view=vertical] [choice]T[/choice] [choice]T[/choice] [choice]T[/choice] [choice value=1]T[/choice] [/singlechoice] T [numberinput] [answer]T[/answer] [/numberinput] T [/cloze] </pre>

Приложение Е

(справочное)

Примеры размеченного текстового контента

Таблица Е.1 – Примеры размеченного текстового контента

Тип вопроса	Размеченный текстовый контент
Вопрос с оди- ночным выбо- ром ответа	<pre>[item] Укажите административный центр Томской области: [singlechoice numbering=123.; view=vertical] [choice]Асино[/choice] [choice]Северск[/choice] [choice]Стрежевой[/choice] [choice value=1]Томск[/choice] [/singlechoice] [/item]</pre>
Вопрос с мно- жественным выбором ответа	<pre>[item] Укажите города Томской области: [multichoice numbering=123.;] [choice value=33%]Томск[/choice] [choice value=-50%]Кожевниково[/choice] [choice value=33%]Северск[/choice] [choice value=33%]Асино[/choice] [choice value=-50%]Юпра[/choice] [/multichoice] [/item]</pre>
Вопрос с вводом текстового отве- та	<pre>[item] Какой город является административным центром Томской области? [textinput] [answer]Томск[/answer] [/textinput] [/item]</pre>
Вопрос с вводом числового отве- та	<pre>[item] Укажите численность населения г. Томска в 2016 году. Ответ укажите в тыс. чел. [numberinput] [answer]569 293[/answer] [/numberinput] [/item]</pre>
Вопрос на соот- ветствие	<pre>[item] Сопоставьте населённые пункты Томской области и год их основания. [matching numbering=123)] [subitem]Томск[answer]1604[/answer][/subitem] [subitem]Северск[answer]1949[/answer][/subitem] [subitem]Капрасок[answer]1640[/answer][/subitem] [subitem]Кривошеино[answer]1826[/answer][/subitem] [/matching] [/item]</pre>
Вопрос на упо- рядочивание	<pre>[item] Расставьте населённые пункты Томской области по порядку согласно году их основания. [ordering numbering=123)] [subitem]Томск[answer]1604[/answer][/subitem] [subitem]Северск[answer]1949[/answer][/subitem] [subitem]Капрасок[answer]1640[/answer][/subitem] [subitem]Кривошеино[answer]1826[/answer][/subitem] [/ordering] [/item]</pre>

	<pre> [ordering numbering=123)] [subitem]Томск[/subitem] [subitem]Капрасок[/subitem] [subitem]Кривошеино[/subitem] [subitem]Северск[/subitem] [/ordering] [/item] </pre>
Составной во-прос	<pre> [cloze] Укажите административный центр Томской области: [singlechoice numbering=123.; view=vertical] [choice]Асино[/choice] [choice]Северск[/choice] [choice]Стрежевой[/choice] [choice value=1]Томск[/choice] [/singlechoice] Он был основан в [numberinput] [answer]1604[/answer] [/numberinput] году. [/cloze] </pre>

Приложение Ж

(справочное)

Пример подготовки данных для нейронной сети, генерирующей подмаски неразмеченного текстового контента

Пусть имеется маска неразмеченного текстового контента для вопроса с
одиноким выбором ответа (таб. Ж.1).

Таблица Ж.1 – Маска неразмеченного текстового контента

Маска неразмеченного текстового контента для вопроса с одноким выбором ответа
<p>T</p><p>1.~T;</p><p>2.~T;</p><p>3.~T;</p><p>4.=T;</p>

Эта маска отображается на 8 подмасок неразмеченного текстового контента (таб. Ж.2).

Таблица Ж.2 – Подмаски неразмеченного текстового контента

Подмаски неразмеченного текстового контента для вопроса с одноким выбором ответа
<p>__T__</p> <p>____~T____~T____~T____=T____</p> <p>____1.____;____2.____;____3.____;____4.____;</p> <p><p>_</p><p>____</p><p>____</p><p>____</p><p>____</p></p> <p>____~T____</p> <p>____~T____</p> <p>____~T____</p> <p>____=T____</p>

Словарь маски неразмеченного текстового контента состоит из 13 элементов (таб. Ж. 3).

Таблица Ж.3 – Словарь маски неразмеченного текстового контента

Элементы словаря входных данных
['~', '>', '=', 'p', '/', 'T', '.', '<', ';', '4', '1', '3', '2']

Сопоставим каждому элементу словаря one-hot вектор, как показано в таб. Ж.4.

Таблица Ж.4 – Векторизация элементов словаря

Элемент словаря	One-hot вектор
~	[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
>	[0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
=	[0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
p	[0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0]
/	[0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0]
T	[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0]

.	[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]
<	[0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0]
;	[0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0]
4	[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0]
1	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0]
3	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]
2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]

Теперь, каждый символ в маске неразмеченного текстового контента можно заменить соответствующим ему one-hot вектором, как показано в таб. Ж.2.

Таблица Ж.5 – Векторизация входных данных

Символ входных данных	One-hot вектор	Символ входных данных	One-hot вектор	Символ входных данных	One-hot вектор	Символ входных данных	One-hot вектор
<	[0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0]	Т	[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0]	<	[0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0]	р	[0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0]
р	[0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0]	;	[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0]	/	[0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0]	>	[0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
>	[0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]	<	[0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0]	р	[0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0]	<	[0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0]
Т	[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0]	/	[0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0]	>	[0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]	р	[0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0]
<	[0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0]	р	[0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0]	<	[0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0]	>	[0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
/	[0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0]	>	[0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]	р	[0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0]	4	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0]
р	[0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0]	<	[0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0]	>	[0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]	.	[0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]
>	[0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]	р	[0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0]	3	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0]	=	[0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
<	[0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0]	>	[0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]	.	[0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]	Т	[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0]
р	[0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0]	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0]	~	[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]	;	[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0]
>	[0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]	.	[0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]	Т	[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0]	<	[0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0]
1	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0]	~	[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]	;	[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0]	/	[0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0]
.	[0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]	Т	[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0]	<	[0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0]	р	[0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0]
~	[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]	;	[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0]	/	[0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0]	>	[0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

Номер подмаски можно закодировать, как показано в таб. Ж.6.

Таблица Ж.6 – Векторизация номеров подмасок

Номер подмаски	One-hot вектор
----------------	----------------

1	[1, 0, 0, 0, 0, 0, 0, 0]
2	[0, 1, 0, 0, 0, 0, 0, 0]
3	[0, 0, 1, 0, 0, 0, 0, 0]
4	[0, 0, 0, 1, 0, 0, 0, 0]
5	[0, 0, 0, 0, 1, 0, 0, 0]
6	[0, 0, 0, 0, 0, 1, 0, 0]
7	[0, 0, 0, 0, 0, 0, 1, 0]
8	[0, 0, 0, 0, 0, 0, 0, 1]

Нейронная сеть, идентифицирующая тип текстового контента вернёт вектор [1, 0, 0, 0, 0, 0, 0, 0], соответствующий типу вопроса с одиночным выбором ответа.

Далее необходимо конкатенировать вектора таб. Ж.5 с каждым вектором из таб. Ж.6 и с вектором [1, 0, 0, 0, 0, 0, 0, 0], идентифицирующим тип текстового контента. В результате получим тензор данных с формой (8, 56, 28), где последняя размерность складывается из размерностей конкатенируемых векторов $13 + 8 + 7 = 28$.

Приложение 3

(справочное)

Правила разметки текстового контента

Таблица 3.1 – Правила разметки текстового контента

Тег	Описание	
[test]	Содержит описание теста. В одном документе может быть описано несколько тестов.	
[id]	Идентификатор Лоцмана для теста.	
[discipline]	Наименование дисциплины. Необходимо использовать внутри тега [test].	
[title]	Наименование теста. Если не указано – для теста будет оставлено название по умолчанию. Необходимо использовать внутри тега [test].	
[author]	Автор теста. Необходимо использовать внутри тега [test].	
[date]	Дата подготовки теста. Необходимо использовать внутри тега [test].	
[category]	Наименование категории вопросов. Все последующие вопросы будут помещены в эту категорию до встречи следующего тега [category]. Если категория не указана, вопросы помещаются в категорию по умолчанию. Необходимо использовать внутри тега [test].	
	power	Количество вопросов из данной категории, попадающих в тест.
[item]	Описывает один вопрос теста. Необходимо использовать внутри тега [test].	
[name]	Наименование вопроса. Необходимо использовать внутри тега [item] или [close].	
[hint]	Подсказка к вводу ответа. Если подсказка не указана, используется подсказка по умолчанию (в зависимости от типа вопроса). Необходимо использовать внутри тега [singlechoice], [multichoice], [textinput], [numberinput], [matching], [item] или [close].	
[singlechoice]	Ввод ответа на вопрос в виде выбора одной из представленных альтернатив. Необходимо использовать внутри тега [test], [item] или [close]. Параметры:	
	numbering	Нумерация вариантов ответа. Если нумерация задана одним символом, то список ответов будет маркированным («–», «•» и т.п.). Если символов несколько, то нумерованным списком. Можно указать любые три символа, следующие по порядку в выбранной кодировке. Затем можно указать символ-разделитель (по умолчанию – точка) и символ, добавляемый после варианта ответа (по умолчанию – ничего). Например: «123..», «abc.», «абв);», «456).», «pqr» и т.п. Если shuffle=true, допустимы лишь варианты «123», «abc», «ABC». По умолчанию нумерация отсутствует.
	shuffle	Определяет, нужно ли перемешивать варианты ответа. Задается целой (0/1) или логической (true/false) константой. По умолчанию – true. Игнорируется в составных вопросах.
	view	Способ отображения альтернатив ответа (только при использовании в составных вопросах). Может принимать значения vertical/horizontal/select. Значение по умолчанию – select.
[multichoice]	Ввод ответа на вопрос в виде выбора одной или нескольких из представленных альтернатив. Необходимо использовать внутри тега [test] или [item]. Параметры:	
	numbering	См. выше.
	shuffle	См. выше.
[choice]	Описание одного варианта ответа [singlechoice] или [multichoice]. Параметры:	

	value	Оценка за данный вариант ответа. Может задаваться в виде чисел в диапазоне от 0 до 1 (0.33, -0.5, ...), процентов (33%, -50%, ...) или дробей (1/3, -1/2). Значение по умолчанию равно 0.
[@], [#], [+], [=]	Правильный вариант ответа. Аналог [choice value=1].	
[textinput]	Ввод простого текстового ответа. Необходимо использовать внутри тега [test], [item] или [close].	
	ignorecase	Определяет, игнорировать ли регистр символов при анализе ответов. Задается целой (0/1) или логической (true/false) константой. Значение по умолчанию – true.
	width	Ширина поля ввода в символах. Если задать 0, используется ширина самого длинного ответа. По умолчанию используются настройки Moodle.
	regex	Определяет, задан ли ответ в виде регулярного выражения. Задается целой (0/1) или логической (true/false) константой. Значение по умолчанию – false.
[numberinput]	Ввод числового ответа. Необходимо использовать внутри тега [test], [item] или [close].	
	width	См. выше.
	tolerance	Абсолютная погрешность ввода числового ответа. По умолчанию равна 0.
[unit]	Единицы измерения числового ответа. Необходимо использовать внутри тега [numberinput]. Игнорируется в составных вопросах.	
	name	Имя единицы измерения ответа. По умолчанию отсутствует.
	multiplier	Множитель ответа. По умолчанию равен 1.
[answer]	Ответ на вопрос [textinput] или [numberinput].	
	value	См. выше. Значение по умолчанию равно 1.
	regex	См. выше (только для тега [textinput]).
	tolerance	См. выше (только для тега [numberinput]).
[@], [#], [+], [=]	Правильный вариант ответа. Аналог [answer value=1]. Если вариант ответа один, можно вообще не указывать.	
[matching]	Вопрос на соответствие. Необходимо использовать внутри тега [test] или [item].	
	width	См. выше.
	shuffle	См. выше.
	numbering	См. выше. Символ, добавляемый после варианта ответа, не используется.
[ordering]	Вопрос на упорядочение. Необходимо использовать внутри тега [test] или [item].	
	width	См. выше.
	numbering	См. выше. Символ, добавляемый после варианта ответа, не используется.
[subitem]	Описывает один пункт соответствия или упорядочения. Необходимо использовать внутри тега [matching] или [ordering].	
[answer]	Правильный вариант соответствия. Необходимо использовать внутри тега [subitem].	
[close]	Описывает составной вопрос. Необходимо использовать внутри тега [test].	

Приложение II

(справочное)

Результаты работы нейронной сети для генерации маски неразмеченного текстового контента

В таблице И.1 представлены результаты работы нейронной сети для генерации маски неразмеченного текстового контента. Здесь символом Т (target) отмечены целевые значения, а Р (predict) – предсказанные нейронной сетью.

Таблица И.1 – Результаты работы нейронной сети для генерации маски неразмеченного текстового контента

1	T	<p>Т</p><p>1.~Т;</p><p>2.~Т;</p><p>3.=Т;</p><p>4.~Т;</p><p>5.~Т;</p>
	P	<p>Т</p><p>1.~Т;</p><p>2.~Т;</p><p>3.=Т;</p><p>4.~Т;</p><p>5.~Т;</p>
2	T	<p>Т</p><p>1.~Т;</p><p>2.=Т;</p><p>3.~Т;</p><p>4.=Т;</p><p>5.=Т;</p>
	P	<p>Т</p><p>1.~Т;</p><p>2.=Т;</p><p>3.~Т;</p><p>4.=Т;</p><p>5.=Т;</p>
3	T	<p>Т</p><p>{=Т}</p>
	P	<p>Т</p><p>{=Т}</p>
4	T	<p>Т</p><p>{#Т}</p>
	P	<p>Т</p><p>{#Т}</p>
5	T	<p>Т</p><p>1)Т</p><p>2)Т</p><p>3)Т</p><p>4)Т</p><p>а)Т</p><p>б)Т</p><p>в)Т</p><p>г)Т</p><p>ответ:1-6,2-г,3-в,4-а</p>
	P	<p>Т</p><p>1)Т</p><p>2)Т</p><p>3)Т</p><p>4)Т</p><p>а)Т</p><p>б)Т</p><p>в)Т</p><p>г)Т</p><p>ответ:1-6,2-г,3-в,4-а</p>
6	T	<p>Т</p><p>1)Т</p><p>2)Т</p><p>3)Т</p><p>4)Т</p><p>ответ:1,2,3,4</p>
	P	<p>Т</p><p>1)Т</p><p>2)Т</p><p>3)Т</p><p>4)Т</p><p>ответ:1,2,3,4</p>
7	T	<p>Т</p><p>1.~Т;</p><p>2.~Т;</p><p>3.~Т;</p><p>4.=Т;</p><p>5.~Т;</p>
	P	<p>Т</p><p>1.~Т;</p><p>2.~Т;</p><p>3.~Т;</p><p>4.=Т;</p><p>5.~Т;</p>
8	T	<p>Т</p><p>1.~Т;</p><p>2.~Т;</p><p>3.=Т;</p><p>4.~Т;</p><p>5.=Т;</p>
	P	<p>Т</p><p>1.~Т;</p><p>2.~Т;</p><p>3.=Т;</p><p>4.~Т;</p><p>5.=Т;</p>
9	T	<p>Т</p><p>{=Т}</p>
	P	<p>Т</p><p>{=Т}</p>
10	T	<p>Т</p><p>{#Т}</p>
	P	<p>Т</p><p>{#Т}</p>
11	T	<p>Т</p><p>1)Т</p><p>2)Т</p><p>3)Т</p><p>4)Т</p><p>а)Т</p><p>б)Т</p><p>в)Т</p><p>г)Т</p><p>ответ:1-а,2-в,3-б,4-г</p>
	P	<p>Т</p><p>1)Т</p><p>2)Т</p><p>3)Т</p><p>4)Т</p><p>а)Т</p><p>б)Т</p><p>в)Т</p><p>г)Т</p><p>ответ:1-а,2-в,3-б,4-г</p>
12	T	<p>Т</p><p>1)Т</p><p>2)Т</p><p>3)Т</p><p>4)Т</p><p>ответ:1,2,3,4</p>
	P	<p>Т</p><p>1)Т</p><p>2)Т</p><p>3)Т</p><p>4)Т</p><p>ответ:1,2,3,4</p>
13	T	<p>Т</p><p>1.~Т;</p><p>2.=Т;</p><p>3.~Т;</p><p>4.~Т;</p><p>5.~Т;</p>
	P	<p>Т</p><p>1.~Т;</p><p>2.=Т;</p><p>3.~Т;</p><p>4.~Т;</p><p>5.~Т;</p>
14	T	<p>Т</p><p>1.~Т;</p><p>2.=Т;</p><p>3.~Т;</p><p>4.~Т;</p><p>5.=Т;</p>
	P	<p>Т</p><p>1.~Т;</p><p>2.=Т;</p><p>3.~Т;</p><p>4.~Т;</p><p>5.=Т;</p>
15	T	<p>Т</p><p>{=Т}</p>
	P	<p>Т</p><p>{=Т}</p>

[illegible]

Приложение К

(справочное)

Результаты работы нейронной сети для генерации подмасок неразмеченного текстового контента

В таблице К.1 представлены результаты работы нейронной сети для генерации подмасок неразмеченного текстового контента. Здесь символом Т (target) отмечены целевые значения, а Р (predict) – предсказанные нейронной сетью.

Таблица К.1 – Результаты работы нейронной сети для генерации подмасок неразмеченного текстового контента

1	Т	__Т__
	Р	__Т__
	Т	__~Т__~Т__=Т__~Т__~Т__
	Р	__~Т__~Т__=Т__~Т__~Т__
	Т	__1.__;__2.__;__3.__;__4.__;__5.__;
	Р	__1.__;__2.__;__3.__;__4.__;__5.__;
	Т	<p>_</p><p>__</p><p>__</p><p>__</p><p>__</p><p>__</p>
	Р	<p>_</p><p>__</p><p>__</p><p>__</p><p>__</p><p>__</p>
	Т	__~Т__
	Р	__~Т__
	Т	__~Т__
	Р	__~Т__
	Т	__=Т__
	Р	__=Т__
	Т	__~Т__
	Р	__~Т__
	Т	__~Т__

	P	$\sim T$
2	T	T
	P	T
	T	$\sim T = T \sim T = T = T$
	P	$\sim T = T \sim T = T = T$
	T	1. $_$; 2. $_$; 3. $_$; 4. $_$; 5. $_$;
	P	1. $_$; 2. $_$; 3. $_$; 4. $_$; 5. $_$;
	T	$\sim T$
	P	$\sim T$
	T	$\sim T \sim T$
	P	$\sim T \sim T$
	T	$= T$
	P	$= T$
	T	$= T = T = T$
	P	$= T = T = T$
	T	$\sim T$
	P	$\sim T$
	T	$\sim T \sim T$
	P	$= T \sim T$
	T	$= T$
	P	$= T$
	T	$= T = T = T$
	P	$= T = T = T$
	T	$= T$
	P	$= T$

	T	_____ = T _____ = T _____ = T _____ _____
	P	_____ = T _____ = T _____ = T _____ _____
3	T	___ T _____ _____
	P	___ T _____ _____
	T	_____ {=T} _____ _____
	P	_____ {=T} _____ _____
4	T	___ T _____ _____
	P	___ T _____ _____
	T	_____ {#T} _____ _____
	P	_____ {#T} _____ _____
5	T	___ T _____ _____
	P	___ T _____ _____
	T	_____ 1-6, 2-г, 3-в, 4-а _____ _____
	P	_____ 1-6, 2-г, 3-в, 4-а _____ _____
	T	_____ 1) _____ 2) _____ 3) _____ 4) _____ _____
	P	_____ 1) _____ 2) _____ 3) _____ 4) _____ _____
	T	_____ 1-6 _____ _____
	P	_____ 1-6 _____ _____
	T	_____ 2-г _____ _____
	P	_____ 2-г _____ _____
	T	_____ 3-в _____ _____
	P	_____ 3-в _____ _____
	T	_____ 4-а _____ _____
	P	_____ 4-а _____ _____
	T	___ T _____ _____
6	T	___ T _____ _____

	P	___ T _____
	T	_____ 1,2,3,4 _____
	P	_____ 1,2,3,4 _____
	T	_____ 1) _____ 2) _____ 3) _____ 4) _____
	P	_____ 1) _____ 2) _____ 3) _____ 4) _____
	T	_____ 1 _____
	P	_____ 1 _____
	T	_____ 2 _____
	P	_____ 2 _____
	T	_____ 3 _____
	P	_____ 3 _____
	T	_____ 4 _____
	P	_____ 4 _____
7	T	___ T _____
	P	___ T _____
	T	_____ ~T _____ ~T _____ ~T _____ =T _____ ~T _____
	P	_____ ~T _____ ~T _____ ~T _____ =T _____ ~T _____
	T	_____ 1. ___ ; _____ 2. ___ ; _____ 3. ___ ; _____ 4. ___ ; _____ 5. ___ ; _____
	P	_____ 1. ___ ; _____ 2. ___ ; _____ 3. ___ ; _____ 4. ___ ; _____ 5. ___ ; _____
	T	<p>_</p><p>_____</p><p>_____</p><p>_____</p><p>_____</p><p>_____</p>_____
	P	<p>_</p><p>_____</p><p>_____</p><p>_____</p><p>_____</p><p>_____</p>_____
	T	_____ ~T _____
	P	_____ ~T _____
	T	_____ ~T _____
P	_____ ~T _____	

	T	<div><div>~T</div><div></div></div>
	P	<div><div>~T</div><div></div></div>
	T	<div><div>=T</div><div></div></div>
	P	<div><div>=T</div><div></div></div>
	T	<div><div>~T</div><div></div></div>
	P	<div><div>~T</div><div></div></div>
8	T	<div><div>T</div><div></div></div>
	P	<div><div>T</div><div></div></div>
	T	<div><div>~T</div><div>~T</div><div>=T</div><div>~T</div><div>=T</div></div>
	P	<div><div>~T</div><div>~T</div><div>=T</div><div>~T</div><div>=T</div></div>
	T	<div><div>1. ; 2. ; 3. ; 4. ; 5. ;</div><div></div></div>
	P	<div><div>1. ; 2. ; 3. ; 4. ; 5. ;</div><div></div></div>
	T	<div><div>~T</div><div></div></div>
	P	<div><div>~T</div><div></div></div>
	T	<div><div>~T</div><div>~T</div><div>~T</div></div>
	P	<div><div>~T</div><div>~T</div><div>~T</div></div>
	T	<div><div>~T</div><div></div></div>
	P	<div><div>~T</div><div></div></div>
	T	<div><div>~T</div><div>~T</div><div>~T</div></div>
	P	<div><div>~T</div><div>~T</div></div>
	T	<div><div>=T</div><div></div></div>
	P	<div><div>=T</div><div></div></div>
	T	<div><div>=T</div><div>=T</div></div>
	P	<div><div>=T</div><div>=T</div></div>
	T	<div><div>~T</div><div></div></div>

	P	$\sim T$
	T	$\sim T \quad \sim T \quad \sim T$
	P	$=T \quad \sim T$
	T	$=T$
	P	$=T$
	T	$=T \quad =T$
	P	$=T \quad =T$
9	T	T
	P	T
	T	$\{=T\}$
	P	$\{=T\}$
10	T	T
	P	T
	T	$\{\#T\}$
	P	$\{\#T\}$
11	T	T
	P	T
	T	$1-a, 2-b, 3-6, 4-r$
	P	$1-a, 2-b, 3-6, 4-r$
	T	$1) \quad 2) \quad 3) \quad 4)$
	P	$1) \quad 2) \quad 3) \quad 4)$
	T	$1-a$
	P	$1-a$
	T	$2-b$
	P	$2-b$

	T	_____
	P	_____
	T	_____
	P	_____
12	T	_____T_____
	P	_____T_____
	T	_____1,2,3,4_____
	P	_____1,2,3,4_____
	T	_____1)_____2)_____3)_____4)_____
	P	_____1)_____2)_____3)_____4)_____
	T	_____1_____
	P	_____1_____
	T	_____2_____
	P	_____2_____
	T	_____3_____
	P	_____3_____
	T	_____4_____
	P	_____4_____
13	T	_____T_____
	P	_____T_____
	T	_____~T_____ = T_____ ~T_____ ~T_____ ~T_____
	P	_____~T_____ = T_____ ~T_____ ~T_____ ~T_____
	T	_____1._____; _____2._____; _____3._____; _____4._____; _____5._____;
	P	_____1._____; _____2._____; _____3._____; _____4._____; _____5._____;
	T	<p>_</p><p>_____</p><p>_____</p><p>_____</p><p>_____</p><p>_____</p>_____

14

	T	$\sim T$
	P	$\sim T$
	T	$\sim T \quad \sim T \quad \sim T$
	P	$=T \quad \sim T \quad \sim T$
	T	$\sim T$
	P	$\sim T$
	T	$\sim T \quad \sim T \quad \sim T$
	P	$=T \quad \sim T$
	T	$=T$
	P	$=T$
	T	$=T \quad =T$
	P	$=T \quad =T$
15	T	T
	P	T
	T	$\{=T\}$
	P	$\{=T\}$
16	T	T
	P	T
	T	$\{ \#T \}$
	P	$\{ \#T \}$
17	T	T
	P	T
	T	$1-6, 2-a, 3-г, 4-в$
	P	$1-6, 2-a, 3-г, 4-в$
	T	$1) \quad 2) \quad 3) \quad 4)$

	P	_____ 1) _____ 2) _____ 3) _____ 4) _____
	T	_____ 1-6 _____
	P	_____ 1-6 _____
	T	_____ 2-a _____
	P	_____ 2-a _____
	T	_____ 3-г _____
	P	_____ 3-г _____
	T	_____ 4-в _____
	P	_____ 4-в _____
18	T	_____ Т _____
	P	_____ Т _____
	T	_____ 1,2,3,4 _____
	P	_____ 1,2,3,4 _____
	T	_____ 1) _____ 2) _____ 3) _____ 4) _____
	P	_____ 1) _____ 2) _____ 3) _____ 4) _____
	T	_____ 1 _____
	P	_____ 1 _____
	T	_____ 2 _____
	P	_____ 2 _____
	T	_____ 3 _____
	P	_____ 3 _____
	T	_____ 4 _____
	P	_____ 4 _____
19	T	_____ Т _____
	P	_____ Т _____

	T	_____~T_____~T_____~T_____ =T_____~T_____
	P	_____~T_____~T_____~T_____ =T_____~T_____
	T	_____1.____;_____2.____;_____3.____;_____4.____;_____5.____;
	P	_____1.____;_____2.____;_____3.____;_____4.____;_____5.____;
	T	<p>_</p><p>_____</p><p>_____</p><p>_____</p><p>_____</p><p>_____</p>_____
	P	<p>_</p><p>_____</p><p>_____</p><p>_____</p><p>_____</p><p>_____</p>_____
	T	_____~T_____
	P	_____~T_____
	T	_____~T_____
	P	_____~T_____
	T	_____~T_____
	P	_____~T_____
	T	_____~T_____
	P	_____~T_____
	T	_____~T_____
	P	_____~T_____
20	T	____T_____
	P	____T_____
	T	_____~T_____~T_____ =T_____~T_____ =T_____
	P	_____~T_____~T_____ =T_____~T_____ =T_____
	T	_____1.____;_____2.____;_____3.____;_____4.____;_____5.____;
	P	_____1.____;_____2.____;_____3.____;_____4.____;_____5.____;
	T	_____~T_____
	P	_____~T_____
	T	_____~T_____~T_____~T_____
	P	_____~T_____~T_____~T_____

	P	$\sim T$ $\sim T$ $\sim T$
	T	$\sim T$
	P	$\sim T$
	T	$\sim T$ $\sim T$ $\sim T$
	P	$\sim T$ $\sim T$
	T	$=T$
	P	$=T$
	T	$=T$ $=T$
	P	$=T$ $=T$
	T	$\sim T$
	P	$\sim T$
	T	$\sim T$ $\sim T$ $\sim T$
	P	$=T$ $\sim T$
	T	$=T$
	P	$=T$
	T	$=T$ $=T$
21	T	T
	P	T
	T	$\{=T\}$
	P	$\{=T\}$
22	T	T
	P	T
	T	$\{\#T\}$
	P	$\{\#T\}$

23	T	___T___
	P	___T___
	T	_____1-6,2-д,3-в,4-г,5-а_____
	P	_____1-6,2-д,3-в,4-г,5-а_____
	T	_____1)_____2)_____3)_____4)_____5)_____
	P	_____1)_____2)_____3)_____4)_____5)_____
	T	_____1-6_____
	P	_____1-6_____
	T	_____2-д_____
	P	_____2-д_____
	T	_____3-в_____
	P	_____3-в_____
	T	_____4-г_____
	P	_____4-г_____
	T	_____5-а_____
	P	_____5-а_____
24	T	___T___
	P	___T___
	T	_____1,2,3,4_____
	P	_____1,2,3,4_____
	T	_____1)_____2)_____3)_____4)_____
	P	_____1)_____2)_____3)_____4)_____
	T	_____1_____
	P	_____1_____
	T	_____2_____

	P	_____ 2 _____
	T	_____ 3 _____
	P	_____ 3 _____
	T	_____ 4 _____
	P	_____ 4 _____
25	T	___ T _____
	P	___ T _____
	T	_____ ~T _____ ~T _____ ~T _____ =T _____ ~T _____
	P	_____ ~T _____ ~T _____ ~T _____ =T _____ ~T _____
	T	_____ 1. ___ ; _____ 2. ___ ; _____ 3. ___ ; _____ 4. ___ ; _____ 5. ___ ; _____
	P	_____ 1. ___ ; _____ 2. ___ ; _____ 3. ___ ; _____ 4. ___ ; _____ 5. ___ ; _____
	T	<p>_</p><p>_____</p><p>_____</p><p>_____</p><p>_____</p><p>_____</p>_____
	P	<p>_</p><p>_____</p><p>_____</p><p>_____</p><p>_____</p><p>_____</p>_____
	T	_____ ~T _____
	P	_____ ~T _____
	T	_____ ~T _____
	P	_____ ~T _____
	T	_____ ~T _____
	P	_____ ~T _____
	T	_____ =T _____
	P	_____ =T _____
	T	_____ ~T _____
P	_____ ~T _____	
26	T	___ T _____
	P	___ T _____

	T	_____ =T_____ ~T_____ ~T_____ ~T_____ =T_____
	P	_____ =T_____ ~T_____ ~T_____ ~T_____ =T_____
	T	_____ 1. ____; _____ 2. ____; _____ 3. ____; _____ 4. ____; _____ 5. ____;
	P	_____ 1. ____; _____ 2. ____; _____ 3. ____; _____ 4. ____; _____ 5. ____;
	T	_____ =T_____
	P	_____ =T_____
	T	_____ =T_____ =T_____
	P	_____ =T_____ =T_____
	T	_____ ~T_____
	P	_____ ~T_____
	T	_____ ~T_____ ~T_____ ~T_____
	P	_____ =T_____ ~T_____ ~T_____ ~T_____
	T	_____ ~T_____
	P	_____ ~T_____
	T	_____ ~T_____ ~T_____ ~T_____
	P	_____ =T_____ ~T_____ ~T_____
	T	_____ ~T_____
	P	_____ ~T_____
	T	_____ ~T_____ ~T_____ ~T_____
	P	_____ =T_____ ~T_____ ~T_____
	T	_____ ~T_____
	P	_____ ~T_____
	T	_____ ~T_____ ~T_____ ~T_____
	P	_____ =T_____ ~T_____
	T	_____ =T_____
	P	_____ =T_____
	T	_____ =T_____ =T_____
	P	_____ =T_____ =T_____
27	T	_____ T_____

	P	___ T _____
	T	_____ {=T} _____
	P	_____ {=T} _____
28	T	___ T _____
	P	___ T _____
	T	_____ {#T} _____
	P	_____ {#T} _____
29	T	___ T _____
	P	___ T _____
	T	___ 1-г, 2-а, 3-б, 4-в _____
	P	___ 1-г, 2-а, 3-б, 4-в _____
	T	_____ 1) _____ 2) _____ 3) _____ 4) _____
	P	_____ 1) _____ 2) _____ 3) _____ 4) _____
	T	___ 1-г _____
	P	___ 1-г _____
	T	___ 2-а _____
	P	___ 2-а _____
	T	___ 3-б _____
	P	___ 3-б _____
	T	___ 4-в _____
	P	___ 4-в _____
30	T	___ T _____
	P	___ T _____
	T	_____ 1, 2, 3, 4, 5 _____
	P	_____ 1, 2, 3, 4, 5 _____

	T	_____ 1) _____ 2) _____ 3) _____ 4) _____ 5) _____ _____
	P	_____ 1) _____ 2) _____ 3) _____ 4) _____ 5) _____ _____
	T	_____ 1 _____ _____
	P	_____ 1 _____ _____
	T	_____ 2 _____ _____
	P	_____ 2 _____ _____
	T	_____ 3 _____ _____
	P	_____ 3 _____ _____
	T	_____ 4 _____ _____
	P	_____ 4 _____ _____
T	_____ 5 _____ _____	
P	_____ 5 _____ _____	

Приложение Л

(справочное)

Результаты работы нейронной сети для генерации подмасок размеченного текстового контента

В таблице Л.1 представлены результаты работы нейронной сети для генерации подмасок размеченного текстового контента. Здесь символом Т (target) отмечены целевые значения, а Р (predict) – предсказанные нейронной сетью.

Таблица Л.1 – Результаты работы нейронной сети для генерации подмасок размеченного текстового контента

1	T	[item]_[/item]
	P	[item]_[/item]
	T	[singlechoice][/]singlechoice]
	P	[singlechoice][/]singlechoice]
	T	numbering=123.;
	P	numbering=123.;
	T	view=vertical
	P	view=vertical
	T	[choice]_[/choice]
	P	[choice]_[/choice]
	T	[choice]_[/choice]
	P	[choice]_[/choice]
	T	[choice value=1]T[/choice]
	P	[choice value=1]T[/choice]
	T	[choice]_[/choice]
	P	[choice]_[/choice]
2	T	[item]_[/item]
	P	[item]_[/item]
	T	[multichoice][/]multichoice]
	P	[multichoice][/]multichoice]
	T	numbering=123.;
	P	numbering=123.;
	T	[choice]_[/choice]
	P	[choice]_[/choice]
	T	value=-50%
	P	value=-50%
	T	[choice]_[/choice]
	P	[choice]_[/choice]
	T	value=33%
	P	value=33%
	T	[choice]_[/choice]

	P	[choice]_[/choice]
	T	value=-50%
	P	value=-50%
	T	[choice]_[/choice]
	P	[choice]_[/choice]
	T	value=33%
	P	value=33%
	T	[choice]_[/choice]
	P	[choice]_[/choice]
	T	value=33%
	P	value=33%
3	T	[item]_[/item]
	P	[item]_[/item]
	T	[textinput][answer]_[/answer][[/textinput]
	P	[textinput][answer]_[/answer][[/textinput]
4	T	[item]_[/item]
	P	[item]_[/item]
	T	[numberinput][answer]_[/answer][[/numberinput]
	P	[numberinput][answer]_[/answer][[/numberinput]
5	T	[item]_[/item]
	P	[item]_[/item]
	T	[matching]_[/matching]
	P	[matching]_[/matching]
	T	numbering=123)
	P	numbering=123)
	T	[subitem]1[answer]6[/answer][[/subitem]
	P	[subitem]1[answer]6[/answer][[/subitem]
	T	[subitem]2[answer]r[/answer][[/subitem]
	P	[subitem]2[answer]r[/answer][[/subitem]
	T	[subitem]3[answer]B[/answer][[/subitem]
	P	[subitem]3[answer]B[/answer][[/subitem]
6	T	[item]_[/item]
	P	[item]_[/item]
	T	[ordering]_[/ordering]
	P	[ordering]_[/ordering]
	T	numbering=123)
	P	numbering=123)
	T	[subitem]1[/subitem]
	P	[subitem]1[/subitem]
	T	[subitem]2[/subitem]
	P	[subitem]2[/subitem]
	T	[subitem]3[/subitem]
	P	[subitem]3[/subitem]
7	T	[item]_[/item]
	P	[item]_[/item]

	T	[singlechoice][singlechoice]
	P	[singlechoice][singlechoice]
	T	numbering=123.;
	P	numbering=123.;
	T	view=vertical
	P	view=vertical
	T	[choice][_choice]
	P	[choice][_choice]
	T	[choice][_choice]
	P	[choice][_choice]
	T	[choice][_choice]
	P	[choice][_choice]
	T	[choice value=1][_choice]
	P	[choice value=1][_choice]
8	T	[choice][_choice]
	P	[choice][_choice]
	T	[choice value=1][_choice]
	P	[choice value=1][_choice]
	T	[choice][_choice]
	P	[choice][_choice]
	T	[choice value=1][_choice]
	P	[choice value=1][_choice]
	T	[choice][_choice]
	P	[choice][_choice]
	T	[choice value=1][_choice]
	P	[choice value=1][_choice]
	T	[choice][_choice]
	P	[choice][_choice]
	T	[choice value=1][_choice]
	P	[choice value=1][_choice]
	T	[choice][_choice]
	P	[choice][_choice]
	T	[choice value=1][_choice]
	P	[choice value=1][_choice]
	T	[choice][_choice]
	P	[choice][_choice]
	T	[choice value=1][_choice]
	P	[choice value=1][_choice]
	T	[choice][_choice]
	P	[choice][_choice]
	T	[choice value=1][_choice]
	P	[choice value=1][_choice]
9	T	[item][_item]
	P	[item][_item]
	T	[textinput][answer][_answer][textinput]
	P	[textinput][answer][_answer][textinput]
10	T	[item][_item]
	P	[item][_item]
	T	[numberinput][answer][_answer][numberinput]

	P	[numberinput][answer]_[/answer][/numberinput]
11	T	[item]_[/item]
	P	[item]_[/item]
	T	[matching]_[/matching]
	P	[matching]_[/matching]
	T	numbering=123)
	P	numbering=123)
	T	[subitem]1[answer]a[/answer][/subitem]
	P	[subitem]1[answer]a[/answer][/subitem]
	T	[subitem]2[answer]b[/answer][/subitem]
	P	[subitem]2[answer]b[/answer][/subitem]
	T	[subitem]3[answer]6[/answer][/subitem]
	P	[subitem]3[answer]6[/answer][/subitem]
	T	[subitem]4[answer]r[/answer][/subitem]
	P	[subitem]4[answer]r[/answer][/subitem]
12	T	[item]_[/item]
	P	[item]_[/item]
	T	[ordering]_[/ordering]
	P	[ordering]_[/ordering]
	T	numbering=123)
	P	numbering=123)
	T	[subitem]1[/subitem]
	P	[subitem]1[/subitem]
	T	[subitem]2[/subitem]
	P	[subitem]2[/subitem]
	T	[subitem]3[/subitem]
	P	[subitem]3[/subitem]
	T	[subitem]4[/subitem]
	P	[subitem]4[/subitem]
13	T	[item]_[/item]
	P	[item]_[/item]
	T	[singlechoice][[/singlechoice]
	P	[singlechoice][[/singlechoice]
	T	numbering=123.;
	P	numbering=123.;
	T	view=vertical
	P	view=vertical
	T	[choice]_[/choice]
	P	[choice]_[/choice]
	T	[choice value=1]_[/choice]
	P	[choice value=1]_[/choice]
	T	[choice]_[/choice]
	P	[choice]_[/choice]
	T	[choice]_[/choice]
	P	[choice]_[/choice]
	T	[choice]_[/choice]
	P	[choice]_[/choice]
14	T	[item]_[/item]
	P	[item]_[/item]

	T	[multichoice][/multichoice]
	P	[multichoice][/multichoice]
	T	numbering=123.;
	P	numbering=123.;
	T	[choice][_ /choice]
	P	[choice][_ /choice]
	T	value=-33%
	P	value=-33%
	T	[choice][_ /choice]
	P	[choice][_ /choice]
	T	value=50%
	P	value=50%
	T	[choice][_ /choice]
	P	[choice][_ /choice]
	T	value=-33%
	P	value=-33%
	T	[choice][_ /choice]
	P	[choice][_ /choice]
	T	value=-33%
	P	value=-33%
	T	[choice][_ /choice]
	P	[choice][_ /choice]
	T	value=50%
	P	value=50%
15	T	[item][_ /item]
	P	[item][_ /item]
	T	[textinput][answer][_ /answer][/textinput]
	P	[textinput][answer][_ /answer][/textinput]
16	T	[item][_ /item]
	P	[item][_ /item]
	T	[numberinput][answer][_ /answer][/numberinput]
	P	[numberinput][answer][_ /answer][/numberinput]
17	T	[item][_ /item]
	P	[item][_ /item]
	T	[matching][_ /matching]
	P	[matching][_ /matching]
	T	numbering=123)
	P	numbering=123)
	T	[subitem]1[answer]6[/answer][/subitem]
	P	[subitem]1[answer]6[/answer][/subitem]
	T	[subitem]2[answer]a[/answer][/subitem]
	P	[subitem]2[answer]a[/answer][/subitem]
	T	[subitem]3[answer]r[/answer][/subitem]
	P	[subitem]3[answer]r[/answer][/subitem]
	T	[subitem]4[answer]B[/answer][/subitem]
	P	[subitem]4[answer]B[/answer][/subitem]
18	T	[item][_ /item]
	P	[item][_ /item]
	T	[ordering][_ /ordering]

	P	[ordering]_[/ordering]
	T	numbering=123)
	P	numbering=123)
	T	[subitem]1[/subitem]
	P	[subitem]1[/subitem]
	T	[subitem]2[/subitem]
	P	[subitem]2[/subitem]
	T	[subitem]3[/subitem]
	P	[subitem]3[/subitem]
	T	[subitem]4[/subitem]
	P	[subitem]4[/subitem]
19	T	[item]_[/item]
	P	[item]_[/item]
	T	[singlechoice][[/singlechoice]
	P	[singlechoice][[/singlechoice]
	T	numbering=123.;
	P	numbering=123.;
	T	view=vertical
	P	view=vertical
	T	[choice]_[/choice]
	P	[choice]_[/choice]
	T	[choice]_[/choice]
	P	[choice]_[/choice]
	T	[choice]_[/choice]
	P	[choice]_[/choice]
	T	[choice value=1]_[/choice]
	P	[choice value=1]_[/choice]
	T	[choice]_[/choice]
	P	[choice]_[/choice]
20	T	[item]_[/item]
	P	[item]_[/item]
	T	[multichoice][[/multichoice]
	P	[multichoice][[/multichoice]
	T	numbering=123.;
	P	numbering=123.;
	T	[choice]_[/choice]
	P	[choice]_[/choice]
	T	value=-33%
	P	value=-33%
	T	[choice]_[/choice]
	P	[choice]_[/choice]
	T	value=-33%
	P	value=-33%
	T	[choice]_[/choice]
	P	[choice]_[/choice]
	T	value=50%
	P	value=50%
	T	[choice]_[/choice]
	P	[choice]_[/choice]

	T	value=-33%
	P	value=-33%
	T	[choice]_[/choice]
	P	[choice]_[/choice]
	T	value=50%
	P	value=50%
21	T	[item]_[/item]
	P	[item]_[/item]
	T	[textinput][answer]_[/answer][[/textinput]
	P	[textinput][answer]_[/answer][[/textinput]
22	T	[item]_[/item]
	P	[item]_[/item]
	T	[numberinput][answer]_[/answer][[/numberinput]
	P	[numberinput][answer]_[/answer][[/numberinput]
23	T	[item]_[/item]
	P	[item]_[/item]
	T	[matching]_[/matching]
	P	[matching]_[/matching]
	T	numbering=123)
	P	numbering=123)
	T	[subitem]1[answer]6[/answer][[/subitem]
	P	[subitem]1[answer]6[/answer][[/subitem]
	T	[subitem]2[answer]d[/answer][[/subitem]
	P	[subitem]2[answer]d[/answer][[/subitem]
	T	[subitem]3[answer]b[/answer][[/subitem]
	P	[subitem]3[answer]b[/answer][[/subitem]
	T	[subitem]4[answer]r[/answer][[/subitem]
	P	[subitem]4[answer]r[/answer][[/subitem]
	T	[subitem]5[answer]a[/answer][[/subitem]
	P	[subitem]5[answer]a[/answer][[/subitem]
24	T	[item]_[/item]
	P	[item]_[/item]
	T	[ordering]_[/ordering]
	P	[ordering]_[/ordering]
	T	numbering=123)
	P	numbering=123)
	T	[subitem]1[/subitem]
	P	[subitem]1[/subitem]
	T	[subitem]2[/subitem]
	P	[subitem]2[/subitem]
	T	[subitem]3[/subitem]
	P	[subitem]3[/subitem]
	T	[subitem]4[/subitem]
	P	[subitem]4[/subitem]
25	T	[item]T[/item]
	P	[item]T[/item]
	T	[singlechoice][[/singlechoice]
	P	[singlechoice][[/singlechoice]
	T	numbering=123.;

	P	numbering=123.;
	T	view=vertical
	P	view=vertical
	T	[choice]T[/choice]
	P	[choice]T[/choice]
	T	[choice]T[/choice]
	P	[choice]T[/choice]
	T	[choice]T[/choice]
	P	[choice]T[/choice]
	T	[choice value=1]T[/choice]
	P	[choice value=1]T[/choice]
	T	[choice]T[/choice]
	P	[choice]T[/choice]
26	T	[item]T[/item]
	P	[item]T[/item]
	T	[multichoice][multichoice]
	P	[multichoice][multichoice]
	T	numbering=123.;
	P	numbering=123.;
	T	[choice]T[/choice]
	P	[choice]T[/choice]
	T	value=50%
	P	value=50%
	T	[choice]T[/choice]
	P	[choice]T[/choice]
	T	value=-33%
	P	value=-33%
	T	[choice]T[/choice]
	P	[choice]T[/choice]
	T	value=-33%
	P	value=-33%
	T	[choice]T[/choice]
	P	[choice]T[/choice]
	T	value=-33%
	P	value=-33%
	T	[choice]T[/choice]
	P	[choice]T[/choice]
	T	value=50%
	P	value=50%
27	T	[item]T[/item]
	P	[item]T[/item]
	T	[textinput][answer]T[/answer][textinput]
	P	[textinput][answer]T[/answer][textinput]
28	T	[item]T[/item]
	P	[item]T[/item]
	T	[numberinput][answer]T[/answer][numberinput]
	P	[numberinput][answer]T[/answer][numberinput]
29	T	[item]T[/item]
	P	[item]T[/item]

	T	[matching][/ matching]
	P	[matching][/ matching]
	T	numbering=123)
	P	numbering=123)
	T	[subitem]1[answer]r[/ answer]/subitem]
	P	[subitem]1[answer]r[/ answer]/subitem]
	T	[subitem]2[answer]a[/ answer]/subitem]
	P	[subitem]2[answer]a[/ answer]/subitem]
	T	[subitem]3[answer]6[/ answer]/subitem]
	P	[subitem]3[answer]6[/ answer]/subitem]
	T	[subitem]4[answer]B[/ answer]/subitem]
	P	[subitem]4[answer]B[/ answer]/subitem]
30	T	[item]T[/item]
	P	[item]T[/item]
	T	[ordering][/ ordering]
	P	[ordering][/ ordering]
	T	numbering=123)
	P	numbering=123)
	T	[subitem]1[/subitem]
	P	[subitem]1[/subitem]
	T	[subitem]2[/subitem]
	P	[subitem]2[/subitem]
	T	[subitem]3[/subitem]
	P	[subitem]3[/subitem]
	T	[subitem]4[/subitem]
	P	[subitem]4[/subitem]
	T	[subitem]5[/subitem]
	P	[subitem]5[/subitem]