**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра вычислительной техники**

**ОТЧЕТ**

**по лабораторной работе №2**

**по дисциплине «Нейронные сети»**

**Тема: Рекуррентные нейронные сети**

|  |  |  |
| --- | --- | --- |
| Студенты гр. 8308 |  | Сергеев В.С. |
| Преподаватель |  | Беляев П.Ю. |

Санкт-Петербург

2023

Оглавление

[1. Цель работы 3](#_Toc129211692)

[2. Ход работы 4](#_Toc129211693)

[3. Вывод 12](#_Toc129211694)

# Цель работы

Целью данной лабораторной работы является изучение рекуррентных нейронных сетей и их применение. В процессе работы будет рассмотрено несколько различных архитектур рекуррентных нейронных сетей, включая RNN, LSTM и GRU.

Задачи:

* Изучить основные принципы работы рекуррентных нейронных сетей.
* Изучить архитектуры рекуррентных нейронных сетей, такие как RNN, LSTM, GRU и их различия.
* Составить набор данных для обучения и тестирования модели.
* Разработать и обучить модель рекуррентной нейронной сети.
* Проанализировать результаты обучения и определить точность модели.
* Провести эксперименты, варьируя параметры модели, и оценить их влияние на качество классификации.
* Сравнить результаты работы различных архитектур рекуррентных нейронных сетей.
* Сделать выводы о применимости и эффективности рекуррентных нейронных сетей.

Задание 1: Подготовка данных

В задании необходимо продемонстрировать методы подготовки и предварительной обработки данных, таких как увеличение и нормализация данных. Данные - временные последовательности. Усложненное задание\* - последовательности слов.

Для формирования набора данных – можете найти существующий.

Задание 2: Архитектура RNN

В этом задании необходимо написать и уметь объяснить как работает разработанная архитектура. Что поможет в этом: изучение различных слоев, используемых в RNN.

Разработка собственной архитектуры (и последующее обучение с 0) – 3-5 различных архитектур с разной комбинацией слоев/функций активации.

Задание 3: Обучение RNN

В данном задании студентам необходимо продемонстрировать знания, как обучать RNN, используя функции потерь, оптимизаторы и методы регуляризации. Студенты также должны понимать, как предотвратить переобучение и недообучение в RNN.

Задание 4: Настройка гиперпараметров и выбор модели

В этом задании студенты должны объяснить, как настраивать гиперпараметры в RNN и как выполнять выбор модели.

# Ход работы

Задание 1.

Для выполнения лабораторной работы был выбран набор данных AG\_NEWS Dataset, содержащий заголовки новостей. В данном датасете представлены 4 класса:

1. Новости мира;
2. Спортивные новости;
3. Новости бизнеса;
4. Новости науки и техники.

Данные предразбиты на тестовую (120 тыс. записей) и обучающую (7.6 тыс. записей) выборки.

Для представления текстовых данных используется метод Bag-of-Words, эмбеддинги для кодирования слов были обучены локально на тренировочной выборке датасета. Средняя длинна текста в записи составляет примерно 40 эмбеддингов. Получаемая длинна словаря составляет 40708 записи. Применяемые настройки обучения эмбеддингов представленные на рисунке 1.

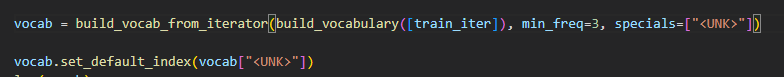


Рисунок 1. Обучение эмбеддингов

Также ранее был опробован датасет IMDB Review Sentiment, содержащий записи двух классов, однако, на данном датасете не удалось достигнуть значимых результатов. Предполагаемой причиной является большая длинна текста в записи — около 1200 эмбеддингов. При этом длинна словаря при схожих настройках превышает 60 тыс. записей. Для его сокращения длинны обрабатываемого текста была опробована лемматизация, а также были отброшены малозначимые знаки препинания как “/” и прочие. Среднюю длину текста удалось сократить до 127 эмбеддингов, однако даже в таком случае при обучении RNN не удалось значительно превысить точность в 50%. LSTM для данного датасета не был опробован, однако в целях экономии времени, основной фокус был переключен на описанный раннее датасет.

Задание 2.

В ходе работы созданы 2 RNN и 2 LSTM архитектуры.

Архитектура RNN\_1 представлена на рисунке 2.



Рисунок 2. Архитектура RNN\_1

Сеть состоит из слоя эмбеддингов, одного рекуррентного слоя размером 50, за которым следует линейный слой.

Архитектура RNN\_2 представлена на рисунке 3.

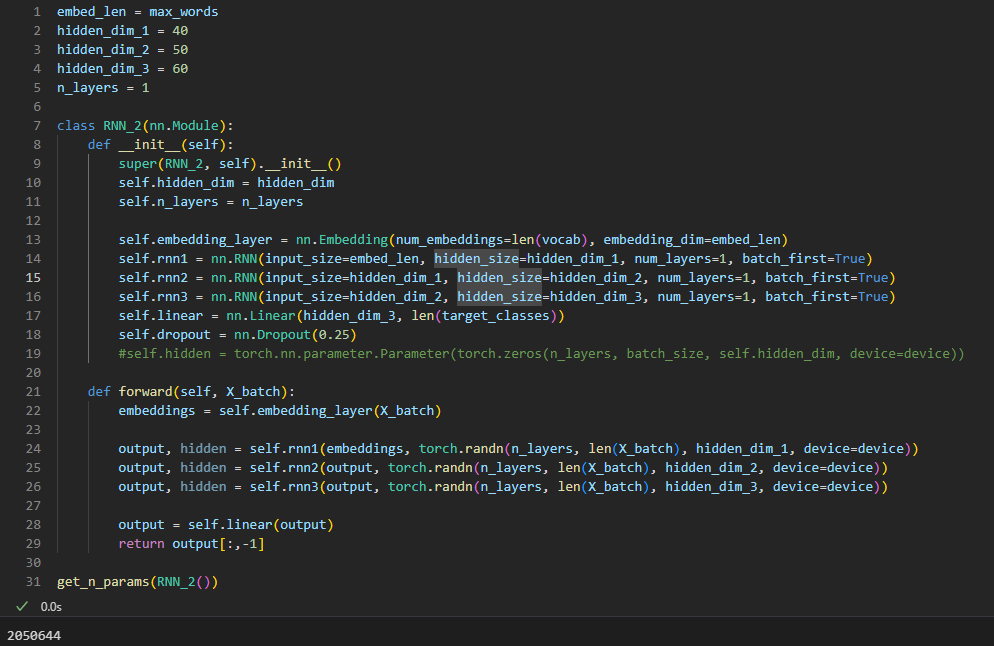


Рисунок 3. Архитектура RNN\_1

Сеть состоит из слоя эмбеддингов, трёх последовательных рекуррентных слоёв размерами 40, 50 и 60 соответственно за которыми следует линейный слой.

Архитектура LSTM\_1 представлена на рисунке 4.

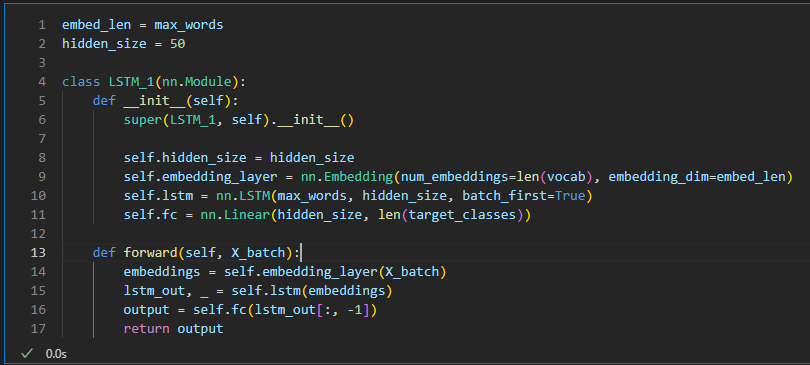


Рисунок 4. Архитектура LSTM\_1

Сеть состоит из слоя эмбеддингов, одной LSTM ячейки за которой следует линейный слой.

Архитектура LSTM\_2 представлена на рисунке 5.

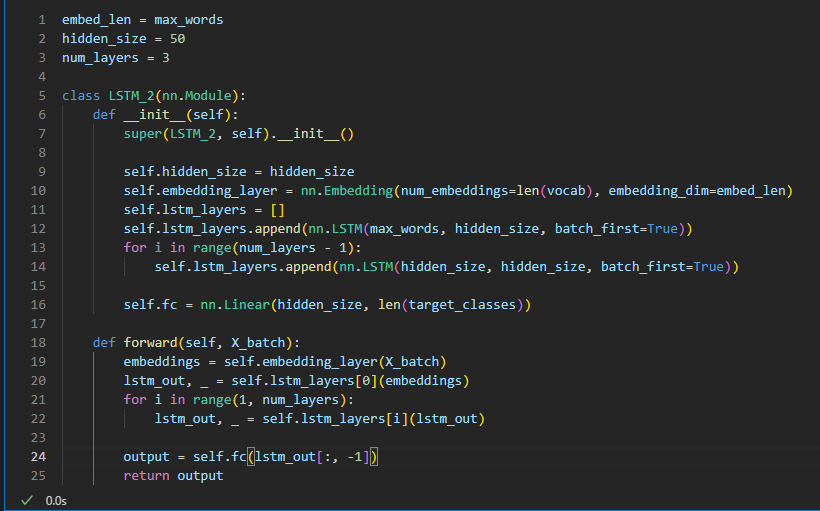


Рисунок 5. Архитектура LSTM\_2

Сеть состоит из слоя эмбеддингов, трёх последовательно идущих LSTM ячеек, за которыми следует линейный слой.

Задание 3 и 4.

Для обучения CNN в качестве функции потерь используется CrossEntropyLoss. Используется оптимизатор Adam. Оптимально подобранный коэффициент скорости обучения равен 0.001. На коэффициенте обучения, равном 0.01, функция потерь не сходится на разумном количестве эпох обучения, а на 0.0001 скорость обучения крайне мала.

Размер батча при обучении равен 512. Также в ходе экспериментов было замечено, что большие (512 или 256 против 16 или 64) размеры батчей положительно влияют на генерализацию сети. В отличии от лабораторной работы №1, нестабильности при обучении большими батчами не наблюдалось.

Далее приведёны результаты обучения моделей. На рисунках 6-9 представлены графики функций потерь сетей.

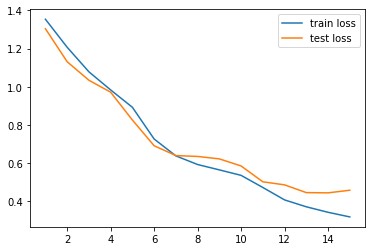


Рисунок 6. Функция потерь RNN\_1

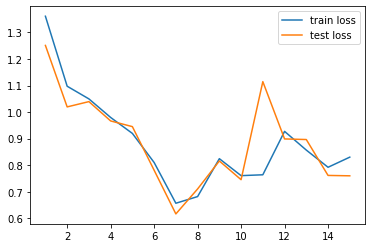


Рисунок 7. Функция потерь RNN\_2

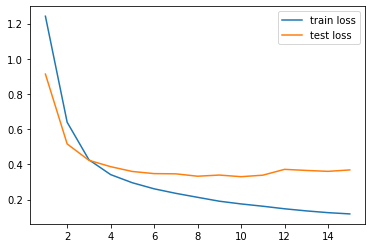


Рисунок 8. Функция потерь LSTM\_1

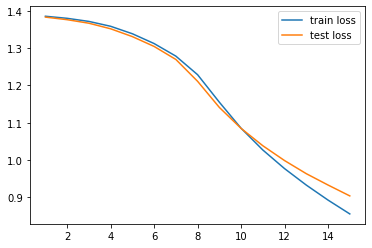


Рисунок 9. Функция потерь LSTM\_2

Таблица 1. Точность моделей

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Модель | RNN\_1 | RNN\_2 | LSTM\_3 | LSTM\_4 |
| Точность модели | 0.875 | 0.877 | 0.898 | 0.702 |

На рисунках 10-13 представлены метрики обученных сетей.

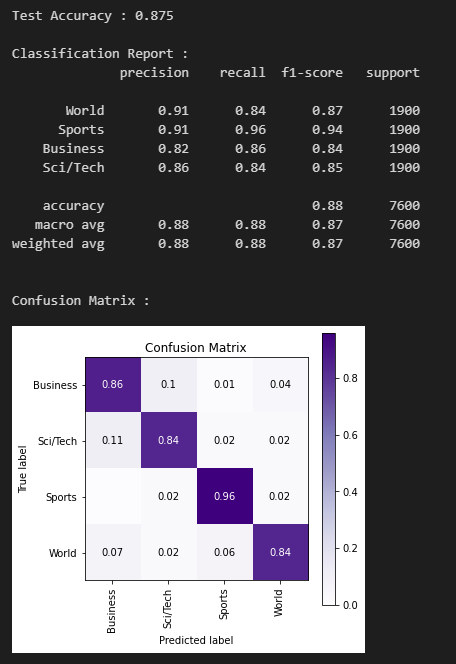


Рисунок 10. Метрики RNN\_1

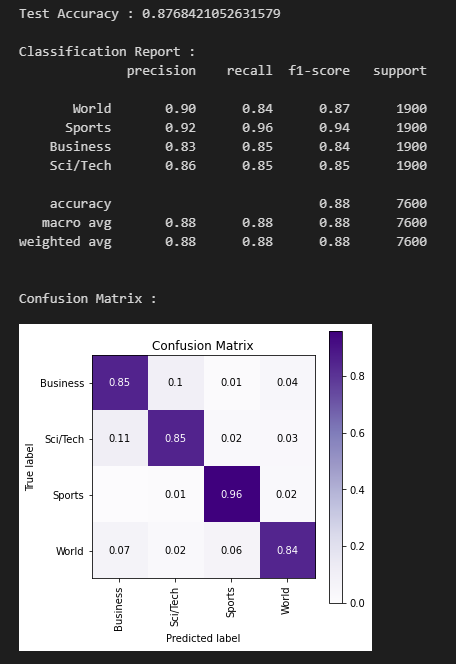


Рисунок 11. Метрики RNN\_2

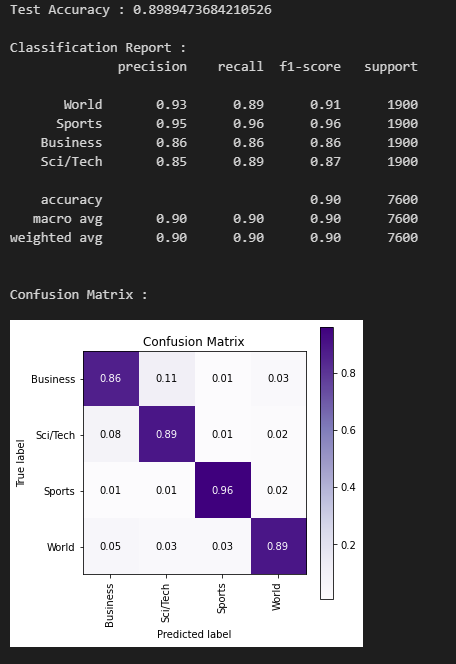


Рисунок 12. Метрики LSTM\_1

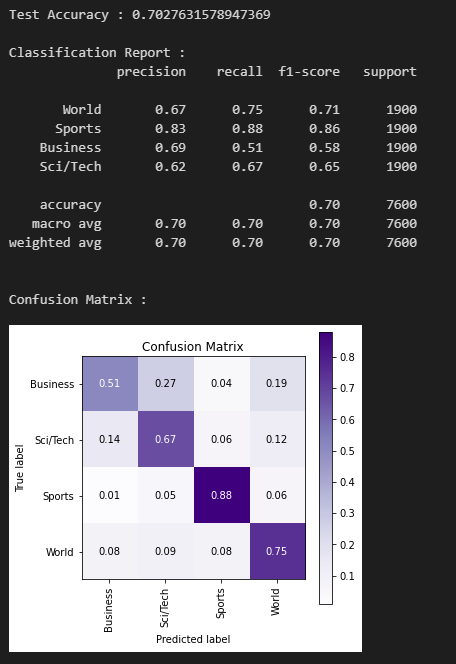


Рисунок 13. Метрики LSTM\_2

В итоговой версии общая точность каждой из моделей составила более 65%.

Модели были сохранены на диск и вновь загружены с него.Далее произведено тестирование моделей на конкретном примере. Был написан пример новостного заголовка, после чего с помощью моделей было сделано предсказание, к какому классу он относится.

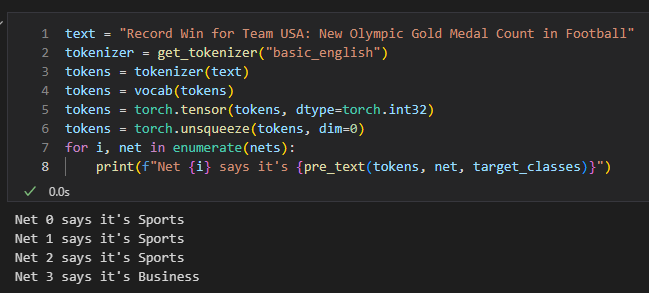
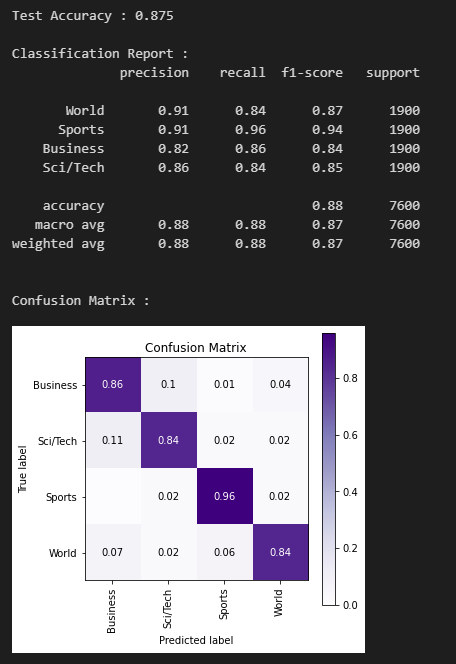


Рисунок 14. Результаты работы моделей на загруженном примере

Модели 1, 3 и 4 корректно определили класс изображения, модель 2 определяет класс ошибочно.

# Вывод

В ходе выполнения лабораторной работы был самостоятельно собран набор данных. Были созданы две рекуррентных и две LSTM архитектуры нейронных сетей, после чего модели были обучены на подготовленном. Полученные модели достигают необходимый уровень точности.

На данном датасете наращивание слоев рекуррентных сетей на даёт значимого преимущества, так как уже одного рекуррентного слоя достаточно для хорошей точности. Нейросеть с LSTM ячейкой показывает ещё большую эффективность, однако установлено, что архитектура с несколькими последовательными LSTM ячейками значительно медленнее обучается как по скорости падения функции потерь, так и по процессорному времени, затрачиваемому на одну эпоху.

Стоит отметить, что в библиотеке Pytorch обучение LSTM сети с эмбеддингами, имеющими целочисленный тип, пришлось проводить на cpu, а не cuda, из-за отсутствия поддержки перемножения необходимого типа матриц на cuda.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Cat Dataset [Электронный ресурс]. // Kaggle: Your Home for Data Science. URL: https://www.kaggle.com/datasets/crawford/cat-dataset. (Дата обращения 26.02.2023).