МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«МИРЭА – Российский технологический университет»

РТУ МИРЭА

Институт информационных технологий

Кафедра прикладной математики

КУРСОВАЯ РАБОТА

по дисциплине

Технологии организации, обработки и хранения статистических данных.

Тема курсовой работы: Задача классификации в обработке естественного языка на примере определения авторства.

Студент группы ИМБО-01-19 Балобин Дмитрий Юрьевич

_____

(подпись студента)

Руководитель курсовой работы Митина Ольга Алексеевна

_____

(подпись руководителя)

Работа представлена к защите «___» _____ 2020 г.

Допущен к защите «__» _____2020 г.

Москва 2020

Оглавление

Введение

Определение авторства – это задача определения личности, написавшей заданный текст или его отрывок. Она стоит перед исследователями различных областей культуры, такими как, например, литературоведы и историки, ещё с XVIII века. Примером можно привести вставший у людей в середине XVIII века вопрос об авторстве приписываемых У. Шекспиру работ.

Наиболее явно показывать необходимость своего решения эта задача начала во второй половине XX века после применения различных, ещё только зарождающихся и не устоявшихся, лингвистических методов в юриспруденции и криминалистике. Самыми известными прецедентами использования *судебной лингвистики* можно считать исследование в 1968 году шведским лингвистом Я. Свартвиком дела убийцы Дж. Р. Кристи, в котором впервые упоминается этот термин, и анализ манифеста Унабомбера в 1995-1996 годах. Многочисленные применения анализа текста на авторство к показали эффективность некоторых методов, после чего во многих правовых системах мира лингвистическая экспертиза стала полноценным аргументом в делах. Однако атрибуция текста авторством не исчерпывает себя лишь в применении к расследованию судебных процессов, а лишь является людям в этих ситуациях как наиболее эффективный инструмент в некоторых случаях.

В начале развития этой области основным принимающим решения лицом являлся эксперт, чаще всего изучивший стилистические и не только особенности нужного текста лингвист [12]. Также было принято считать, что для анализа подобного анализа текста размер исходного материала должен быть не менее 250 слов. С развитием методов теории принятия решений, приложений теории вероятностей и методов анализа данных стало ясно, что проблему определения авторства вполне можно доверить вычислительной системе и алгоритму, иначе говоря - автоматизировать. И, как показали практические примеры, это может быть эффективно. А применение к анализу

твитов - коротких по длине и часто скудных по содержанию высказываний в интернете - продемонстрировало возможность продуктивного анализа текстов намного меньшей длины, в среднем около 25 слов на текст [13].

Применению методов машинного обучения к задаче определения авторства частей лингвистических данных и посвящена данная работа. Более конкретными её целями являются:

- Изучение литературы по теме обработки естественного языка в целом и по теме определения авторства в частности, определение актуальных проблем и решений;
- Изучение и применение методов обработки “сырых” данных;
- Изучение типовых архитектур нейронных сетей, используемых в рассматриваемой области, выбор и применение наиболее эффективной из них для решения задачи определения авторства;
- Получение навыков работы с инструментами, de facto являющимися стандартом для решения задач подобного типа и подходящих для него.

Теоретическая часть

Особенности задачи

Первая из двух проблем, стоящих при исследовании определения авторства, — это выбор параметров сравнения текста. Что имеется ввиду? Описать “авторский почерк” или “бардовский отпечаток пальца” возможно с использованием различных уровней особенностей [1][7]:

1. *Лексические особенности* определяются словарным запасом автора. В них входит использование необычных в употреблении слов и предложений (неологизмы или архаизмы, диалектизмы или профессионализмы, пословицы или поговорки, и т.п.). В самом тексте это выражается последовательностью определённых токенов, основанных на символьных или на словарных последовательностях;
2. *Синтаксические особенности* выражаются в сочетаниях слов и синтаксических конструкций, в склонности предпочитать одни иным;
3. *Стилистические особенности* – характерные для данного текста, автора или контекста стилистические приёмы, сюжеты или темы, структура текста или общепринятые шаблоны.

Из куда большего числа возможных особенностей выделены лишь выше представленные три по той причине, что их, относительно остальных, намного сложнее подделать. Они чаще всего выражаются несознательно при написании, а подражание этим особенностям конкретного автора требует, как минимум, большого количества сведений из его биографии.

Упор на ту или иную особенность анализируемого текста, вообще говоря, определяется его особенностями или гипотезами о доминирующем количестве выражений этой самой особенности для выбранного набора авторов. Отбирать ограниченное множество авторов, иными словами, создавать *закрытый класс решений*, приходится из-за материальных и вполне понятных причин: *открытый класс* решений невозможен в силу ограниченности ресурсов памяти и вычислительной мощности систем, а также неравномерности

представленности авторов в наборе данных или вовсе их несравнимости. Последнее относится не только к жанровым различиям, но, даже в большей степени, к языковым. В настоящее время сложность возникает уже на стадии адаптации алгоритмов обработки одного естественного языка к другому, не говоря уже об использовании в одной модели материалов на разных языках.

Из преобладания конкретной особенности в свою очередь может следовать приоритет одного класса методов анализа над другим. К примеру, если различия могут быть выражены статистически, то, возможно, наиболее эффективно будет использовать модель на основе теоремы Байеса.

Вторая проблема – формализация и подход к работе с текстом. Необходимо понять, в каком виде стоит представлять текстовые данные и как правильно обобщить работу с ними.

Модель предсказания

В терминах машинного обучения стоящая перед нами задача атрибуции текста может быть рассмотрена как *мультиклассовая классификация с одной меткой*, в которой *метками*, или классами, являются имена авторов текста, а *экземплярами* мы будем считать предложения [3].

В машинном обучении задача классификации является одной из самых исследуемых, потому для большинства, если не для всех, типов информации имеется проверенное решение. Не является исключением и классификация текстов. Хотя классификация коротких по длине символьных последовательностей, вполне ожидаемо, является более сложной по сравнению с классификацией длинных задачей, всё же, как показывают проведённые на твитах и предложениях из литературных произведений исследования, она выполнима.

Конкретных моделей и архитектур для этого существует много; в данной работе используется одна из популярных архитектур трёхканальной свёрточной нейронной сети. Выбор типа архитектуры обусловлен двумя соображениями:

1. Успешным его применением во многих работах, к примеру [3][4][5][13], в некоторых из которых показано превосходство этого типа над другими (например, над SVM моделью или простой перцептронной). Итоговое решение, исходя из этого, будет лучше классифицировать;
2. Перечисленные в предыдущем разделе проблемы частично решаются итоговым способом представления данных (см. следующий раздел), который чаще всего используется в комбинации с выбранным типом нейронной сети (те же работы [5][13]);
3. Архитектура со свёрточным слоем является универсальной в том плане, что её использование эффективно и в классификации иных типов данных [11]. Опыт в её применении может стать полезен в будущих исследованиях и работах.

Ниже приводится объяснение принципа работы выбранной нейронной сети на основе [10][11][9].

Нейронная сеть – упрощённая модель биологической нейронной сети, одновременно представляющая собой удобное и более понятное представление сложной функции, состоящей из, в общем случае рекурсивных, но чаще всего прямых, аппликаций более простых функций. Такие модели повсеместно начали использовать в машинном обучении с 2010х годов, а уже к концу десятилетия они стали самым эффективным методом анализа больших данных.

Структурной единицей выступает *нейрон* – функция активации от нескольких переменных, обычно взвешенных, возвращающая одно значение. Нейроны же в свою очередь объединены в *слои*, своего рода логические блоки сети, отвечающие за конкретный этап обработки входных данных. К примеру, *входной слой* $\vec{X}$ ответственен за получение входных данных для обучения или предсказания, а *выходной* $\vec{Y}$ – за представление результата вычислений. *Многослойными* же называют те сети, которые состоят из большего количества слоёв. Количество и особенности слоёв и их сочетаний определяют сложность

моделирования и работоспособность в конкретной ситуации нейронной сети, поэтому конструирование архитектуры и её настройка являются самыми важными этапами разработки модели.

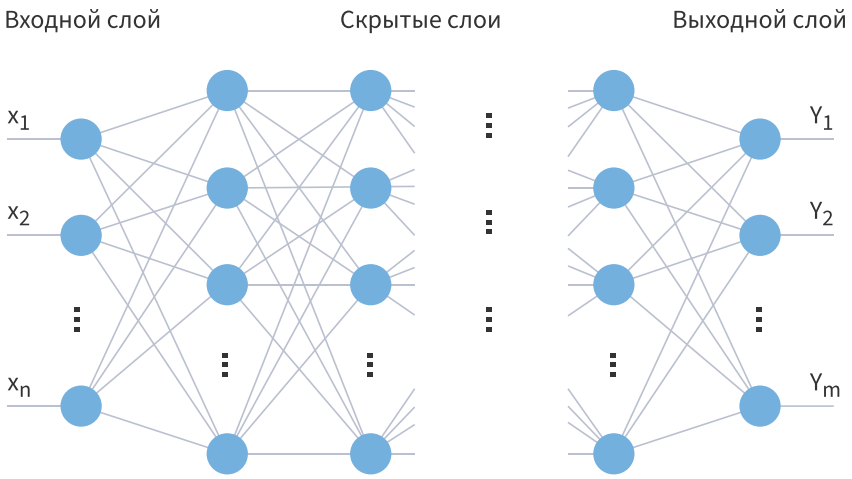


Рис. 1 – Схема многослойной нейронной сети

Обучить нейронную сеть значит найти или приблизить значение неизвестных коэффициентов (весов) внутри неё так, чтобы она решала поставленную задачу. Следует помнить, что на практике сведение ошибки определения ответов к минимуму приводит к *переобучению*, к умению вычислять правильные выводы лишь для обучающей выборки. Подхода к этому процессу обучения существует два:

1. *Обучение с учителем* построено на “подгонке” весов таким образом, чтобы на выходе ответ минимально отличался от заданного в тестовой выборке. В этом подходе самым популярным методом принято считать *метод обратного распространения ошибки*, основанный на градиентном спуске. Кратко этот метод можно описать так:

Пусть для обучения сети с рёбрами (весами) $\{w_{i,j}\}$ между $i-$ым и $j-$тым нейроном заданы: параметры η скорости движения и α параметр функции активации, обучающая выборка из d векторов входа $\{x_{i=1\dots n}^{d=1\dots m}\}$ и векторов правильных ответов $\{t_{i=1\dots k}^{d=1\dots m}\}$, количество повторений обучающей процедуры Q. Тогда алгоритм будет таков:

a) Инициализировать веса случайными значениями;

b) Повторить указанные процедуры Q раз для каждого $d = 1 \dots m$:

a. Получить выходы сети для каждого узла o_i^d, подав на вход обучающие наборы $\{x_i^d\}$;

b. Для всех выходных $i-$тых нейронов, количество которых k, посчитать

$$\delta_i = -o_i(1 - o_i)(t_i - o_k)$$

c. Для каждого $j-$ого уровня I, начиная с предпоследнего, вычислить

$$\delta_j = o_j(1 - o_j) \sum_{k \in \text{Выходы}(j)} \delta_k w_{j,k}$$

d. И для каждого ребра $\{i, j\}$ сети посчитать итоговые значения весов

$$\Delta w_{i,j}(n) = \alpha \Delta w_{i,j}(n-1) + (1 - \alpha)\eta \delta_i o_i$$

$$w_{i,j}(n) = w_{i,j}(n-1) + \Delta w_{i,j}(n)$$

c) Получены итоговые значения весов сети. Её можно считать обученной.

Данный подход, несмотря на недостатки основных его методов, используется в подавляющем большинстве случаев, т.к. он, во-первых, создаёт уверенность в правильном направлении обучения и, во-вторых, более подвержен оценке правильности полученной модели.

2. *Обучение без учителя*. Этот подход отличен от предыдущего тем, что обучающая выборка не классифицирована заранее, сеть сама должна выдать метку входной последовательности или для себя “понять” как работать с похожими сигналами. Его используют не только лишь все, мало кто использует его.

По очевидным причинам в ходе выполнения работы модель обучалась с уже размеченными предложениями.

Свёрточная нейронная сеть [8] – нейронная сеть, имеющая в своём составе *свёрточный слой*, отвечающий за одну или несколько операций свёртки пары матриц предыдущих слоёв $A^{n_x \times n_y}$ и $B^{m_x \times m_y}$ в следующий слой $C^{(\mathrm{n_x}-\mathrm{m_x}+1)\times(\mathrm{n}_y-m_y+1)}$:

$$C_{i,j} = \sum_{u=0}^{m_x-1} \sum_{v=0}^{m_y-1} A_{i+u,j+v} \cdot B_{u,v}$$

Свёртываемые матрицы, вообще говоря, могут быть многомерными, однако обыденной является свёртка двумерных ввиду первоначального применения этого приёма.

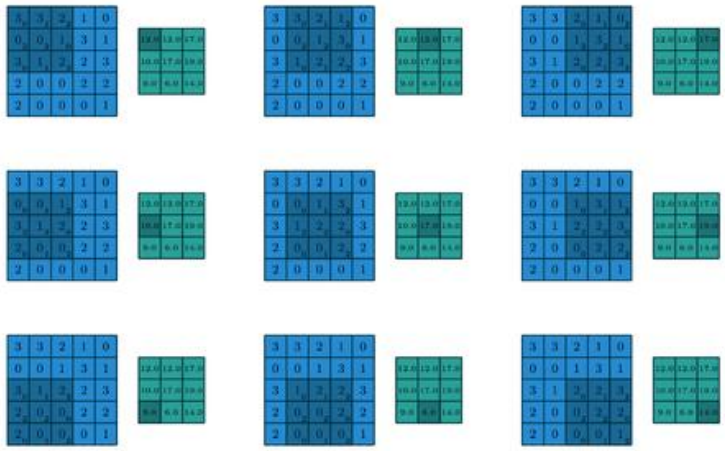


Рис. 2 – Иллюстрация свёртки матриц размера 5×5 и 3×3

Многоканальность нейронной сети в нашем случае означает условную “параллельность” нескольких входных слоёв до определённого момента вычислений:

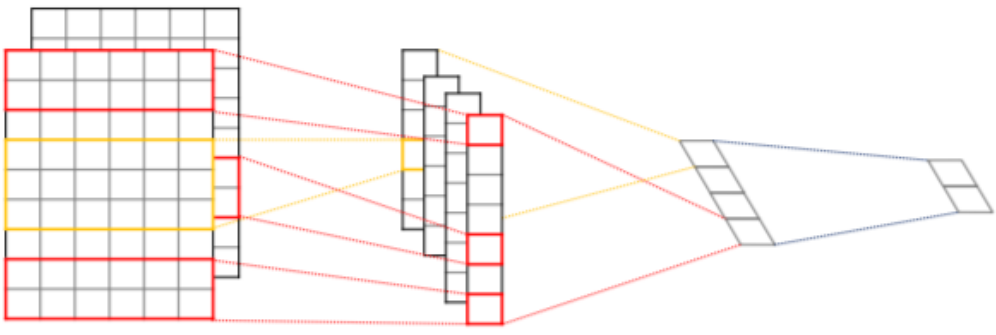


Рис. 3 – Демонстрация многоканальности (двуканальности)

И обуславливается методом конечного представления данных перед непосредственной обработкой.

Предобработка данных для анализа

Обучающая выборка будет состоять из отдельных предложений сочинений некоторого набора авторов. Эти предложения необходимо представить в векторном виде, ведь все математические модели классификации, так или иначе, используют методы сравнения поступающих данных друг с другом. Способ, которым это будет делаться в данной работе, называется *разбиение на n-граммы*.

N-граммы [5][13] – последовательности состоящие из n-элементов, разбитые на основании какого-либо признака, к примеру, по слогам, буквам или словам. В случае представления данных набором n-грамм их сравнение производится подсчётом количества одинаковых последовательностей. Иначе говоря, любая информация представляется векторно в пространстве, размер которого определён количеством уникальных n-грамм в обучающей выборке, а координаты – инцидентностью n-граммы по отношению к конкретному набору.

Для корректного представления в этом виде каждый элемент выборки следует привести к более обобщённому виду: перевести все буквы слов к строчным, удалить пунктуацию. Если первое изменение не устраняет проявления авторского стиля, то второе, вообще говоря, стирает довольно важную часть личностного отпечатка в тексте, что, очевидно, скажется на качестве распознавания. Однако, это не единственное пренебрежение в предобработке. Векторное представление предложений является разновидностью *bag-of-words* модели – такой модели, в которой невозможно определить порядок следования признаков (n-грамм) относительно других, информация в ней сохраняется лишь о наличии и степени проявленности признака.

Часто используемые в языке слова могут сильно мешать в вычленении особенностей письма; для решения этой проблемы в задачах обработки естественного языка принято удалять их из выборки, а такие слова называть *стоп-словами*.

Как показывают исследования в этой области [12], n-граммы способны улавливать лексические, синтаксические и структурные нюансы. Надёжным решением, как показано в работах, является использование символьных триграмм, что и будет применено. Однако в некоторых статьях говорится и об эффективности 1- и биграмм, которые после нескольких экспериментов на взятом наборе данных окажутся наименее полезными, и об эффективности тетраграмм.

Таким образом, данной работе анализу будут подвергаться лексические и, в меньшей степени, синтаксические особенности. Обеспечено же это будет использованием символьных триграмм.

Метрики и оценка результатов

В ходе обучения моделей будут изменяться следующие метрики:

- *Точность* (*accuracy*) показывает процент правильно предсказанных моделью меток на обучающих данных;
- *Ошибка* (*loss*) – значение параметра ошибки на обучающих данных;
- *Точность и ошибка на кросс-валидационных данных* (*val_accuracy* и *val_loss* соответственно) – те же значения, но полученные при использовании *кросс-валидационных данных*. Это случайно выбранные из обучающих не использованные данные;
- Точность определения классов на тестовой выборке. Будет главным критерием эффективности модели. В похожих работах достигалось значение 70%.

Практическая часть

Сбор данных и их формат

Как уже было сказано ранее, адаптация решений одной и той же задачи на разных языках крайне трудна. Это находит своё отражение в ситуации с рассматриваемой темой на русском.

Вполне ожидаемо, что подавляющее количество работ и исследований по определению авторства текстов и их отрывков написано и использует исходные материалы на английском. Соответственно, на русском языке материалов по данной теме крайне мало (или же автор этой работы плохо искал), да и не только лишь на русском. Потому сложности возникают уже на этапе сбора данных.

Корпусы текста, которые можно найти, ограничиваются новостными статьями или же собранными европейскими организациями собраниями сочинений одного автора, однако чаще всего хранящимися не в текстовом формате, а в виде сканированных изображений. Появляется потребность в наборе данных, удовлетворяющем следующие требования:

- текстовый формат (*txt*);
- общая для всего корпуса сочинений кодировка (была выбрана UTF-8 ввиду повсеместного использования некоторыми авторами символов разных алфавитов);
- свободная для использования лицензия (находящиеся в общественном достоянии произведения подходят идеально);
- относительная “чистота” материала – отсутствие какой-либо сторонней информации в файлах, что позволит обойтись более лёгкой предобработкой.

Ссылка на собранный и покаместь регулярно обновляемый корпус русской классической литературы указана в блоке **Ресурсы**.

При использовании данных для обучения классификационной модели следует помнить о принципе равнопредставленности примеров для каждого

класса. В связи с чем в работе выбирается фиксированное количество предложений для каждого автора. Принцип этот является не более чем рекомендацией, но пропорциональность количества примеров обеспечивает более лучшую категоризацию; в ранних работах по выявлению особенностей текстовых данных конкретного автора использовался корпус литературы, в котором каждый автор был представлен разным количеством материала [6].

Ограничения в количестве использованных произведений связаны также и с техническими проблемами. Дело в том, что время просчёта коэффициентов и затраты вычислительных ресурсов сильно зависят от величины обучающей выборки. Также, после нескольких экспериментов с ней оказалось, что количество классов сильно влияет на точность конечной модели в худшую сторону, после чего было принято решение использовать всего 3 класса. Стоит заметить, что разработанное решение свободно работает и с намного большим их количеством, но, т.к. решения приведённой проблемы найдено не было, ограничение на количество авторов всё ещё не преодолено. Вполне возможен вариант, при котором причины качественного ухудшения работоспособности крылись в другом, феномен исследован слишком мало.

Итоговый размер файла с размеченными данными равен 10.4 MB, данные в нём выглядят примерно так:

	text	author
1	Вы не знаете, как вы для меня важны и как вы много для меня сделали!..	Толстой
2	мемуаров не была пропущена цензурой при их первой публикации, совпадающей с периодом работы Толстого над «Войной и миром» (см.	Толстой
3	рибежал из леса к опушке и, бледный, с расширенными зрачками, хотел что-то сказать, но одышка и волнение долго мешали ему говорить.	Чехов
4	— сказала мать, притворно сердито отталкивая дочь.	Толстой
5	Вечер проектировался, однако же, запросто; ожидались одни только «друзья дома», в самом малом числе.	Достоевский
6	И вот князь судорожно устремляется к тому дому, и что же в том, что действительно он там встречает Рогожина?	Достоевский
7	За месяц вперед возьми.	Чехов
8	в и документов то, что можно «прочесть между строк», увидеть под спудом пристрастных и противоречивых суждений очевидцев событий.	Толстой
9	— Будьте так любезны, отпустите меня!	Чехов
10	Это была мысль о том: кто, кто же, наконец, приговорил его к казни.	Толстой
11	Это согласие – ведь это опять одно из народных начал, вот тех самых, которые в нас до сих пор еще Потугины отрицают.	Достоевский
12	снял собравшимся вокруг него солдатам, что ватерпас не что иное есть, как происходит, что атмосферическая ртуть свое движение имеет.	Толстой
13	Не она первая, не она последняя.	Чехов
14	го только три месяца двести рублей отцовских проиграл, с тем и умер старик, что не успел узнать; ты меня затащил, а Книф передергивал.	Достоевский
15	Или, быть может, ее тщеславие ожидало еще большей помпы?.	Чехов
16	Муж ничего не должен знать, и жена должна быть перед ним чистой, как ангельчик!	Чехов
17	Что рассказать, в чем признаться?	Достоевский
18	А я могу ему только завидовать.	Чехов

Рис. 4 – Вид размеченных предложений

А файла с предобработанными данными – 9.9MB.

	text	author
1	вы не знаете как вы для меня важны и как вы много для меня сделали	Толстой
2	мемуаров не была пропущена цензурой при их первой публикации совпадающей с периодом работы толстого над войной и миром см	Толстой
3	ибежал из леса к опушке и бледный с расширенными зрачками хотел чтото сказать но одышка и волнение долго мешали ему говорить	Чехов
4	сказала мать притворно сердито отталкивая дочь	Толстой
5	вечер проектировался однако же запросто ожидались одни только друзья дома в самом малом числе	Достоевский
6	и вот князь судорожно устремляется к тому дому и что же в том что действительно он там встречает рогожина	Достоевский
7	за месяц вперед возьми	Чехов
8	з и документов то что можно прочесть между строк увидеть под слудом пристрастных и противоречивых суждений очевидцев событий	Толстой
9	будьте так любезны отпустите меня	Чехов
10	это была мысль о том кто кто же наконец приговорил его к казни	Толстой
11	это согласие ведь это опять одно из народных начал вот тех самых которые в нас до сих пор еще потугины отрицают	Достоевский
12	ал собравшимся вокруг него солдатам что ватерпас не что иное есть как происходит что атмосферическая ртуть свое движение имеет	Толстой
13	не она первая не она последняя	Чехов
14	только три месяца двести рублей отцовских проиграл с тем и умер старик что не успел узнать ты меня затащил а книф передергивал	Достоевский
15	или быть может ее тщеславие ожидало еще большей помпы	Чехов
16	муж ничего не должен знать и жена должна быть перед ним чистой как ангельчик	Чехов
17	что рассказать в чем признаться	Достоевский
18	а я могу ему только завидовать	Чехов

Рис. 5 – Предобработанные данные

Немного статистики по материалам:

- Всего в выборке участвует 54000 записей, по 18000 на каждого из трёх русских классиков: А.П. Чехова, Л.Н. Толстого и Ф.М. Достоевского;
- Статистика по количеству слов в предложениях следующая:
    - Минимальное количество слов среди всех предложений 1, максимальное – 323, а среднее равно примерно 16.359;
    - Распределение перцентилей (процент – количество слов):
        - 50% - 11;
        - 1% - 1;
        - 95% - 49;
        - 99% - 82;
        - 99.5% - 96;

- 99.9% - 136;

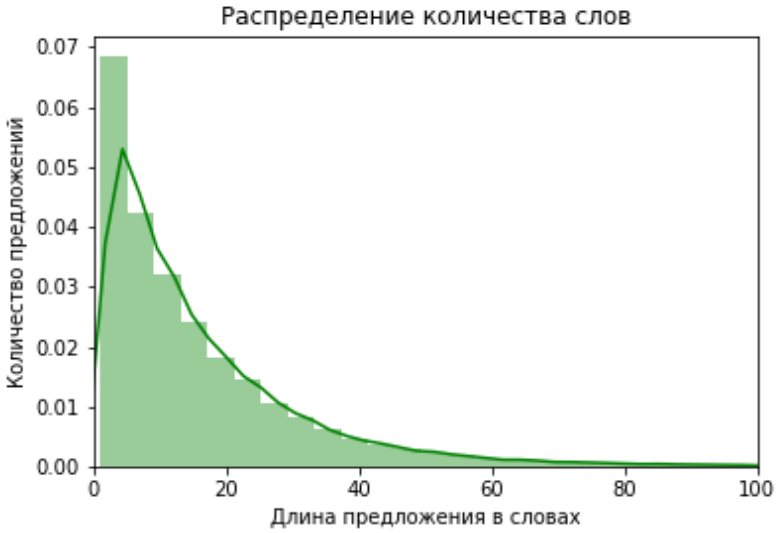


Рис. 6 – График распределения слов в выборке

Из статистики можно сделать вывод, что длина входных в модель последовательностей должна быть не менее 323. Использовано будет число 350 как подходящее ещё и для более расширенного набора произведений.

Занимателен факт того, что из всех предложений с длиной более 150 слов на Льва Николаевича приходится 28, когда на остальных – по 3 на каждого.

- Статистика по количеству символов:
  - Минимальное количество символов среди всех предложений 5, максимальное – 2042, а среднее равно примерно 101.249;
  - Распределение перцентилей:
    - 50% - 67;
    - 1% - 6;
    - 95% - 309;
    - 99% - 523;
    - 99.5% - 618;

- 99.9% - 893;

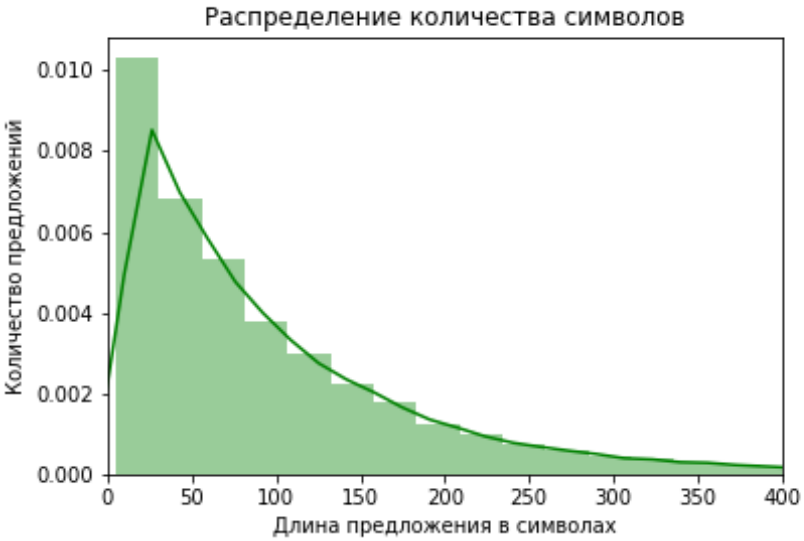


Рис. 7 – График распределения символов в выборке

Из всего множества символов возможно выделить 45 “необычных”: символов кириллистического алфавита, символов с диактрическими знаками и символов из дореволюционного русского алфавита; и из множества записей 1755 предложений, в которых эти символы встречаются. Из необычной пунктуации можно заметить несколько необычных для современного текста на русском видов кавычек: французские “лапки” и английские двойные.

- Статистика по средней длине слов в предложении:
  - Минимальная средняя длина слов в предложении среди всех 2.3, максимальная – 18, а среднее равно примерно 6.112;
  - Распределение перцентилей:
    - 50% - 6;
    - 1% - 3;
    - 95% - 8.125;
    - 99% - 10;
    - 99.5% - 11;

- 99.9% - 13.

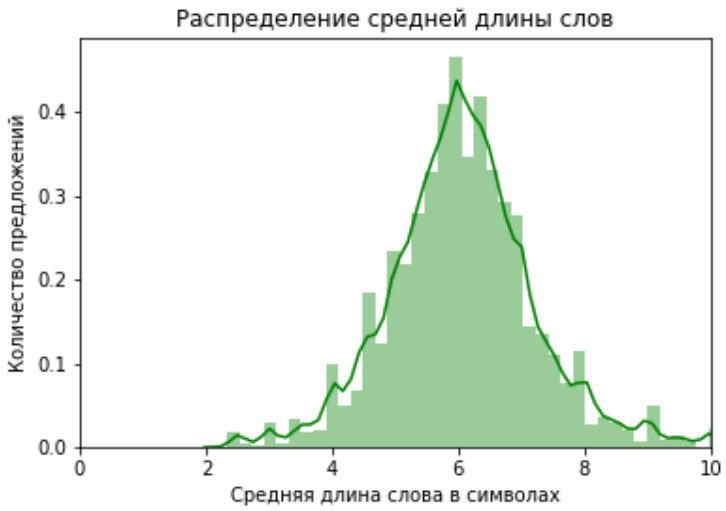


Рис. 8 – График распределения средней длины слов

Последний пункт статистики, средняя длина слов, может показаться ненужным и неважным; но до 60х годов прошлого века гипотеза о том, что средняя длина слов для каждого автора текста уникальна, была одной из самых ходовых [].

Стоит, как мне кажется, привести пример преобразования конкретного предложения в n-граммы для n от 1 до 3 включительно на предложении:

1) [30 35 25 38 30 38 25 111 25 38 111 20 121 25 101 30 58 25 58 72 22 35 20 81 30 5 101 20 110 25 40 30 35 20 38 30 35 30 101 25 40 121 70 30 20 101 5 35 5 38 35 30 30 70 35 20 20 35 30 110 45 70 25 111 25 30 38 30 87 111 25 110 22 38 30 70 28 30 38 30 121 20 70 22 28 20 121 25 45 25 30 99 28 25 20 70 20 35 25 28 30 40 111 20 99 30 110 30 28 30 111 30 35 30 81 40 10 25 35 20 1 38 30 70 30 101 45 30 81 30 70 20 45 101 25 101 30 1 30 35 20 70 35 30 58 25 58 30 1 110 30 10 101 30 38 35 30 45 70 30 38 30 99 30 110 30 28 30 70 81 45 110 25 101 30 <0…0 175 раз>].

Для $n = 2,3$ см. ссылку на работу в разделе **Ресурсы**. Текстовые данные преобразуются в вектор, каждая компонента которого подаётся на соответствующий нейрон входного слоя сети.

Использованные библиотеки и инструменты

Прикладная сфера машинного обучения в настоящий момент переживает всё ещё стремительный рост, в этой области уже имеются устоявшиеся практики и инструменты, сильно упрощающие разработку. Исчезает необходимость реализовывать подавляющее число алгоритмов и структур данных. Для нужд предобработки и исследования текстовых данных самой используемой является библиотека *NLTK*, для создания и обучения моделей – Keras и SciKit-Learn, для работы с большими наборами данных – Pandas, а для визуализации – Matplotlib. В приведённом наборе библиотек имеются все необходимые функции и методы, включая различные метрики и статистические оценки.

Помимо этих библиотек для языка *Python*, также будет использоваться оболочка *Jupyter*, которая позволит разделить вычисления на блоки по логическому принципу и по потребности в вычислительных мощностях, что очень сильно ускорит проведение экспериментов с конфигурациями сети.

Почему же не используется аналитическая платформа *Loginom*? На этот счёт есть четыре довода, имеющие своей причиной в основном специфику задачи:

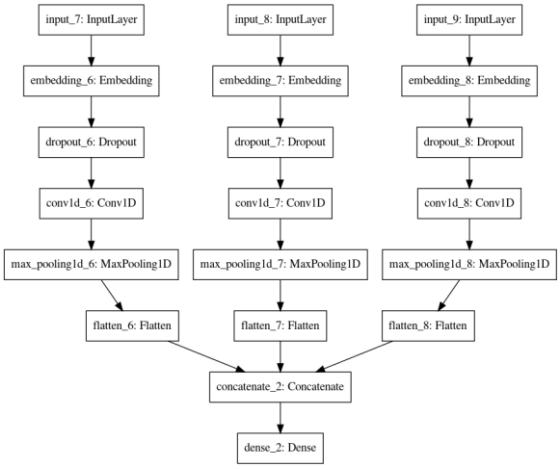
1. Первый и основной – реализация встроенной в Loginom нейронной сети, вне зависимости от назначения. Её архитектурой является “простой” многослойный перцептрон. Решение, построенное на этом компоненте, хоть и подходит для решения профильных для данной программы бизнес-задач, не является сколь-нибудь удовлетворительным для выбранной, что уже было продемонстрировано выше в разделе **Модель предсказания.** Таким образом становится невыполнимым этап создания и обучения модели на этой платформе;
2. Отсутствие встроенных компонентов и функций для обработки строковых данных: вся поддержка работы со строками ограничивается возможностью использования регулярных выражений. Это делает

почти или полностью невозможным предобработку и подготовку данных к обучению модели;

3. Невозможность в доступной версии программы запустить вычисления на удалённой машине. Обозначенный недостаток особенно выделялся при выполнении этой работы: обучение моделей даже на относительно мощных конфигурациях занимало часы, а на локальной машине в системе Loginom подобные эксперименты были бы не осуществимы. Это также делает бессмысленной недавно появившеюся (9.11.2020) возможность использовать компоненты выполнения Python кода;
4. Визуализация на платформе общепринято является неудовлетворительной, хоть и улучшается с каждым обновлением.

Ход анализа

В ходе работы используется одна архитектура нейронной сети, но с небольшими изменениями. После того, как стало ясно, что триграммное представление данных показывает более лучшие результаты, было принято решение сравнить её со своей модифицированной версией с четырьмя свёрточными слоями:



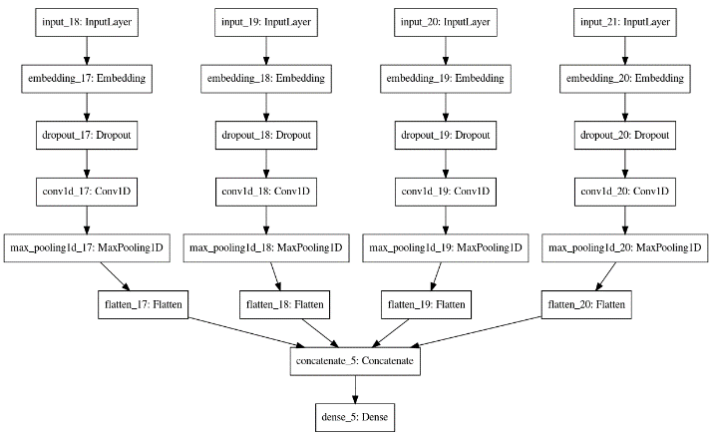
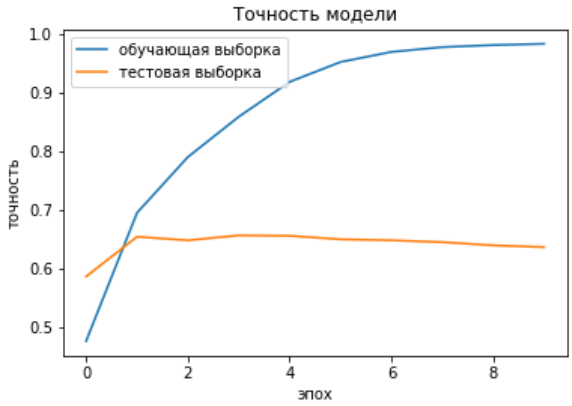


Рис. 9 и 10 – Архитектуры исходная и улучшенная соответственно

Ввиду большого времени обучения исходной триграммной сети (10437 секунд для 10ти эпох или в среднем 1043 секунды на эпоху) и изучения особенностей хода её обучения был сделан вывод о том, что стоит улучшенную модель обучать меньшее количество эпох. Для сравнения, время обучения

второй модели составило 6699 секунд для 5ти эпох или в среднем 1350 секунд на эпоху.

Сравним полученные в ходе обучения метрики. Слева всегда будет график для первой модели, справа, соответственно, для второй:



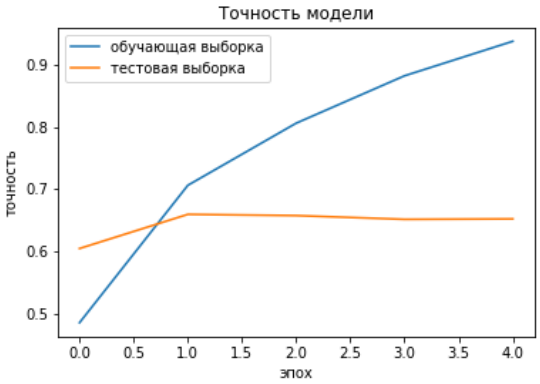
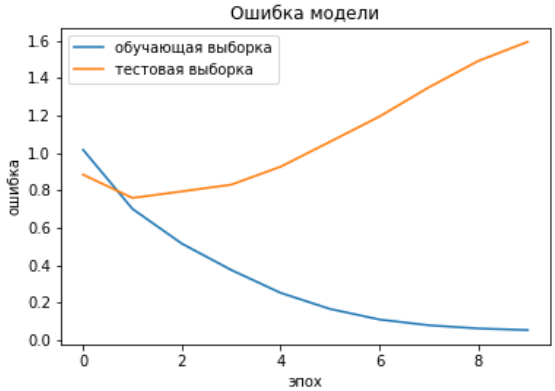


Рис. 11 и 12 – Точность (*accuracy* и *val_accuracy*) для моделей



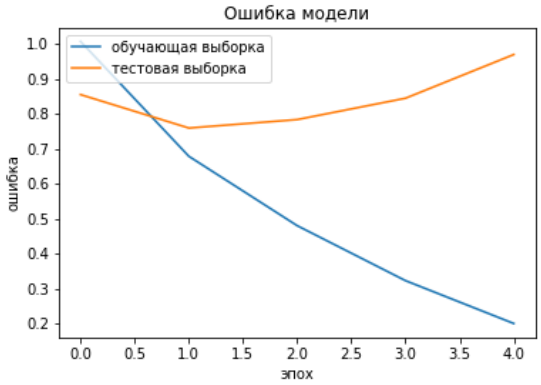
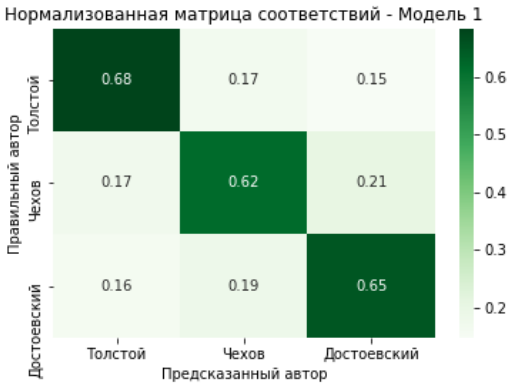


Рис. 13 и 14 – Ошибка (*loss* и *val_loss*) для моделей

Средняя длина правильно предсказанных предложений в символах для первой модели равна примерно 104.613, для второй – 104.197. Средняя длина правильно предсказанных предложений в словах для первой модели равна примерно 17.39, для второй – 17.34. Из этого можно сделать заключение, что эффективный анализ коротких отрывков текста возможен.



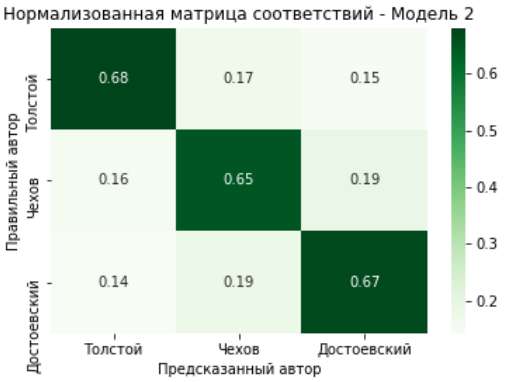


Рис. 15 и 16 – Матрица соответствий

Ресурсы

Исходный код решения доступен по ссылке https://github.com/d0rj/AuthorshipAttribution. Репозиторий содержит отдельные документы для трёх этапов: создания набора “сырых” данных, предобработки с исследованием и непосредственного обучения моделей. Также в нём будут находиться исходники получившихся графиков.

Используемый *набор данных* доступен по ссылке https://github.com/d0rj/RusLit. Все находящиеся в этом репозитории произведения сгруппированы по авторам и имеют лицензию общественного достояния.

Заключение

В ходе работы были изучены материалы по теме выявления особенностей текста, в частности наибольший интерес, в связи с выбранной задачей, представляли работы по определению авторства отрывка произведения. Типовые решения были рассмотрены, а наиболее используемое было изучено и реализовано на практике. В ходе реализации использовались общепринятые в среде машинного обучения инструменты создания моделей и манипуляции с данными, что позволило получить опыт работы с этими инструментами и представление о ходе работы в сфере обработки данных.

Выводы

Нейронные сети, в частности свёрточные, являются одним из самых действенных решений в области классификации. Их применение не ограничивается каким-либо форматом поступающих данных, а алгоритм их работы не изменяется исходя из предметной области, в которой их прикладывают.

Задача классификации текстов на русском языке по признаку авторства на сегодняшний момент не имеет конвенционального решения, или хотя бы общепризнанного подхода к своему решению. Многие алгоритмы из области обработки естественного языка также не адаптированы в популярных инструментах разработки. Готовых наборов текстов на русском языке в открытом доступе меньше, чем на английском, в разы.

Получившиеся свёрточные нейронные сети выполняют свою задачу не более чем удовлетворительно. Средний процент правильных предсказаний – 67, что всё же меньше, чем результаты в похожих работах (около 70 при определении авторства предложений из рассказов и повестей и до 90% у твитов).

Возможные улучшения

Путей улучшения видится три:

1) Улучшить разбитие на n-граммы исходных записей или векторизовать их как-то по-другому так, чтобы учитывалась пунктуация и особенности употребления знаков препинания;
2) Добавить семантическую обработку поступающих предложений;
3) Улучшить архитектуру сети и увеличить количество определяющихся авторов.

Список литературы

1. Батура Татьяна Викторовна Формальные методы определения авторства текстов // Вестник НГУ. Серия: Информационные технологии. 2012. №4;
2. How to Develop a Multichannel CNN Model for Text Classification // Machine Learning Mastery URL: https://machinelearningmastery.com/develop-n-gram-multichannel-convolutional-neural-network-sentiment-analysis/ (дата обращения: 14.12.2020);
3. Ahmed M. Mohsen, Nagwa M. El-Makky, Nagia Ghanem Author Identification using Deep Learning // 15th IEEE International Conference on Machine Learning and Applications. 2016;
4. Kim Y. Convolutional neural networks for sentence classification //arXiv preprint arXiv:1408.5882. – 2014;
5. Shrestha P. et al. Convolutional neural networks for authorship attribution of short texts //Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers. – 2017. – С. 669-674;
6. Forsyth R. S., Holmes D. I. Feature-finding for text classification //Literary and Linguistic Computing. – 1996. – Т. 11. – №. 4. – С. 163-174;
7. Гилемшина Айсылу Габдулзямилевна Своеобразие экспрессивной лексики в разновременных переводах Корана // Учен. зап. Казан. ун-та. Сер. Гуманит. науки. 2007. №4;
8. Dumoulin V., Visin F. A guide to convolution arithmetic for deep learning //arXiv preprint arXiv:1603.07285. – 2016;
9. Collobert R. et al. Natural language processing (almost) from scratch //Journal of machine learning research. – 2011. – Т. 12. – №. ARTICLE. – С. 2493− 2537;

10. Нейрокомпьютерная техника: Теория и практика // Эврика! URL: http://evrika.tsi.lv/index.php?name=texts&file=show&f=410 (дата обращения: 14.12.2020);
11. Машинное обучение // neerc.ifmo.ru URL: http://neerc.ifmo.ru/wiki/index.php?title=%D0%97%D0%B0%D0%B3%D0%BB%D0%B0%D0%B2%D0%BD%D0%B0%D1%8F_%D1%81%D1%82%D1%80%D0%B0%D0%BD%D0%B8%D1%86%D0%B0 (дата обращения: 14.12.2020);
12. Stamatatos E. A survey of modern authorship attribution methods //Journal of the American Society for information Science and Technology. – 2009. – Т. 60. – №. 3. – С. 538-556;
13. Schwartz R. et al. Authorship attribution of micro-messages //Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. – 2013. – С. 1880-1891.