Построение графа последовательности текстовых единиц для создания системы генерации предложений

М.П. Каминский[1] И.А. Рыцарев[1,2], А.В. Куприянов[1,2]

[1]Самарский национальный исследовательский университет им. академика С.П. Королева, Московское шоссе 34А, Самара, Россия, 443086
[2]Институт систем обработки изображений РАН - филиал ФНИЦ «Кристаллография и фотоника» РАН, Молодогвардейская 151, Самара, Россия, 443001

Аннотация. Статья посвящена разработке системы анализа текстовых данных. Рассмотрен подход представления текста из постов отдельно взятой страницы в виде графа ключевых словосочетаний, при помощи которого в последствии будут строиться цепочки слов, схожие по стилистике написания с текстами автора. В рамках работы реализованы: сбор, фильтрация и обработка данных с использованием технологии Big Data.

1. Введение

Понятие «социальная сеть» было использовано социологами ещё в 1920-х годах для исследования взаимосвязей между участниками различных сообществ. Психологом Якобом Морено были предложены социограммы, предсталяющие из себя графы, на которых точками были представлены отдельные индивиды, а линиями – взаимосвязи между ними. Идею использования аппарата теории графов для изучения взаимоотношений людей подхватили специалисты в области социологии, психологии, антропологии, политологии, экономики — так было сформировано направление Social Network Analysis, занимающееся изучением структурных свойств социальных взаимоотношений, моделируемых в виде графов и сетей. Значимым, но довольно трудоемким этапом такого исследования являлось построение модели на основе различных данных из печатных источников, дополнительных опросов и анкетирования [1].

Современные социальные сети значительно упростили жизнь для исследователей, предоставив им бурно развивающийся и легкодоступный источник больших данных. Каждый день пользователи социальных сетей генерируют большие объёмы данных различного рода. Результаты анализа этой информации могут стать отличным материалом для исследований в самых различных отраслях. Поэтому так важно уметь эти данные представить в удобном для эффективного анализа виде [2].

Методы аннотирования текста можно разбить на две группы: извлекающие и генерирующие. Среди извлекающих методов автоматического аннотирования можно выделить метод на основе теории графов, где текст представляется в виде графа, узлы которого – фрагменты текста, а рёбра – отношения между ними [3].

2. Постановка задачи

Современный мир динамичен, компьютеризирован, от работника требуется максимально оперативное и качественное выполнение задания. Программное средство использующее в своей работе разработанный алгоритм может быть применено работниками профессий, где необходимо часто набирать текст для составления схожих по содержанию документов, позволяя сократить время, затраченное на эту задачу, или в организациях, осуществляющих приём граждан с ограниченными возможностями здоровья (статодинамические нарушения верхних конечностей, зрительные нарушения) на квотируемые места, должностные обязанности которых связаны непосредственно с работой на компьютерах.

3. Сбор и работа с данными

Разработанный в рамках исследования алгоритм сначала осуществляет сбор данных, затем производится их фильтрация, с целью получения значимой текстовой информации, далее строится граф ключевых слов, при проходе по которому происходит построение цепочек слов. В дальнейшем, при необходимости, систему можно дообучать, добавляя новые тексты, принадлежащие другим авторам, для комбинирования их стилей [4].

В качестве источника данных была выбрана одна из самых известных блог-платформ «LiveJournal» предоставляющая возможность публиковать свои и комментировать чужие записи. Этот крупный ресурс изобилует блогами на самые различные тематики и является отличным источником крупных объёмов текстовой информации. Вся полученная информация сохраняется в текстовый файл, с которым производится дальнейшая работа.

Для последующей работы с текстом, данные необходимо подготовить. Отфильтровываются веб-ссылки и эмодзи, знаки препинания и спецсимволы, оставшиеся буквы переводятся в строчные. Также отфильтровываются слова, длина которых меньше четырёх символов, с целью исключения большинства служебных частей речи. Затем производится разбиение текста на отдельные ключевые слова. Далее производится лемматизация токенов, то есть приведение слов в их первоначальную форму. При лемматизации части речи преобразуются по следующему типу: существительные – единственное число, именительный падеж; прилагательное – единственное число, мужской род, именительный падеж; глагол – неопределённая форма (инфинитив). Пример лемматизации можно увидеть на рисунке 1.

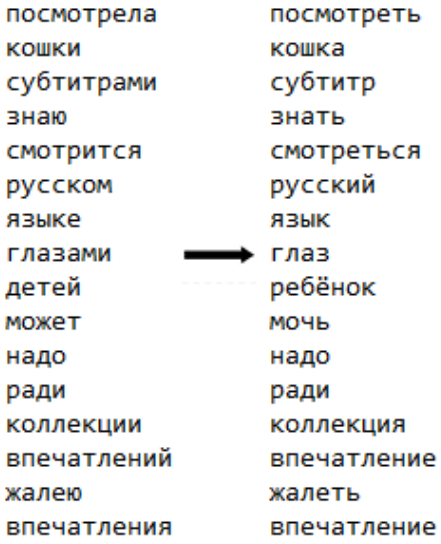

Рисунок 1. Пример преобразования слов в лемму.

После этих преобразований создаётся словарь, упорядоченный по частоте употребления ключевых слов в тексте, по которому строится матрица словосочетаний, значениями которой будет количество встреч между словами в тексте. Далее, имея матрицу словосочетаний и словарь $W = w_1 w_2 w_3 \dots w_n$, мы можем строить граф. Узлами графа будут ключевые слова w_i

из словаря W, рёбра соединяют их в словосочетания, встречаемые в тексте. В качестве веса рёбер указывается количество встреч между словами.

При дообучении в граф вносится новая порция обработанной информации. Но так как вес у новой связи изначально будет ниже чем у тех, которые уже были в графе, то для компенсации вводится новая структура у каждого узла, представленная в виде стека слов ($K = k_1 k_2 k_3 \dots k_m$, где k_j – отдельно взятое слово из стека). В ней хранятся последние связи, которые были после узла. Приоритет вывода будет отдаваться новым данным и компенсация малого веса связи будет производиться при помощи введения зависящего от положения слова в стеке коэффициента s, подобрав который сможем выстраивать логические цепочки сразу для двух наборов данных, но второй будет иметь некоторое преимущество так как им дообучали систему. На рисунке 2 представлена обобщённая схема работы описанного алгоритма.

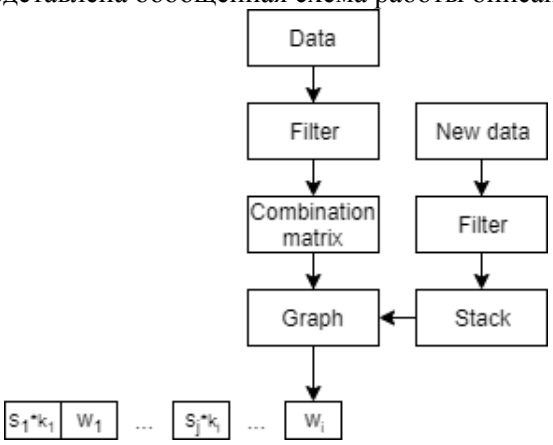

Рисунок 2. Схематическое представление работы алгоритма.

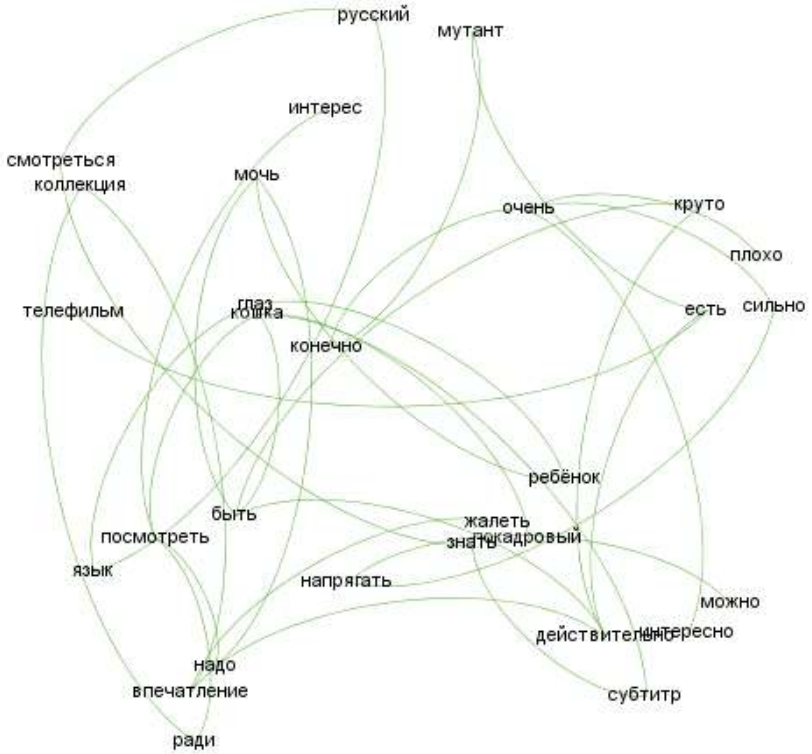

Рисунок 3. Упрощенное изображение графа, составленного по посту «Мюзикл «Кошки» в кино: мутанты, не умеющие петь» под авторством пользователя shakko_kitsune.

4. Сравнение стилей разных авторов

Для проведения исследования были взяты две поста посвященные киномюзиклу «Кошки» двух различных авторов. После получения текста, данные были отфильтрованы и обработаны для

составления словаря ключевых слов и матрицы словосочетаний. Затем были построены два взвешенных графа, увидеть которые можно на рисунке 3 и 4.

У первого графа 297 узлов и 385 рёбер, у второго 296 узлов и 384 рёбра. Сравнив их между собой было обнаружено всего 49 одноимённых узлов. Также для этих 49 совпадений можно пронаблюдать большую разницу между количеством соседних узлов, из чего можно понять, что и частота употребления совпавших слов у авторов различается.

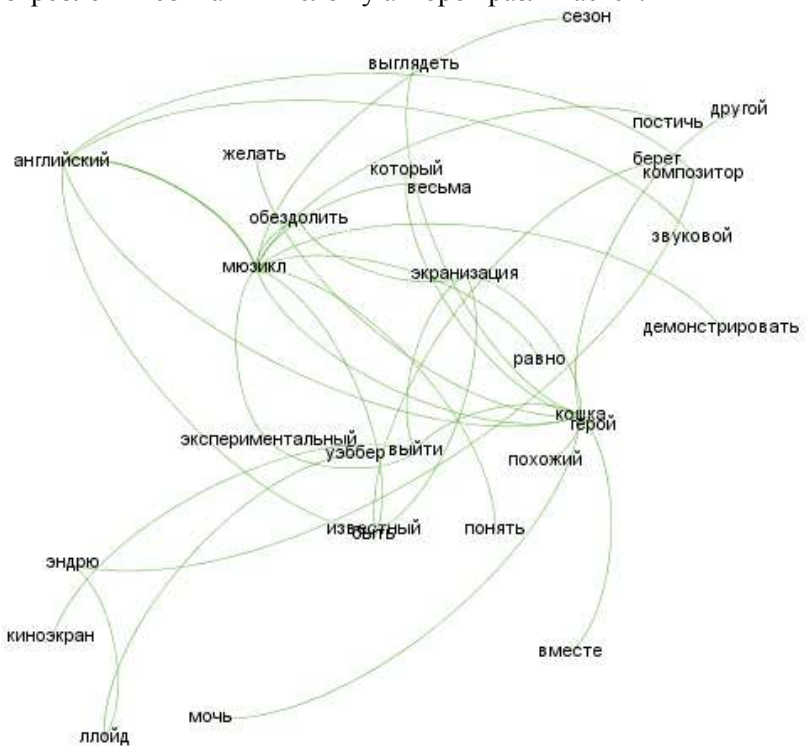

Рисунок 4. Упрощенное изображение графа, составленного по посту «Кошки: мурчащий мюзикл» под авторством пользователя carabaas.

Далее сравним суммарную мощность для каждого узла из графа. На рис 5 и 6 видно, что первый автор часто применяет определенные слова (например вариации слова быть), в то время как у другого использование слов более равномерно.

В результате данных сравнений можно сделать вывод о том, что лексикон авторов сильно разнится даже несмотря на написание статей по схожей теме.

быть	27
который	14
фильм	12
танцевать	11
вообще	10
реально	8
только	8
человек	8
большой	6
даже	6
кошка	6
очень	6
роль	6
сейчас	6
экранизация	6
посмотреть	5

Рисунок 5. Мощность узлов первого графа.

который		18
мюзикл		18
кошка		14
актёр		10
выглядеть		10
поэтому		10
фильм		10
английский		7
выступление		6
есть		6
однако		6
понять		6
потому		6
похожий		5
автор		4
быть		4

Рисунок 6. Мощность узлов первого графа.

5. Вывод

Мы представили обучаемую систему аннотирования на основе теории графов, позволяющую строить цепочки слов схожие по стилю с текстами автора, которую при необходимости можно дообучить, загрузив тексты другого авторства или другой тематики. При дальнейшем развитии алгоритма работы разработанной системы считаем целесообразным ее применение в наборе объемных текстов, позволяя увеличить скорость их написания. Также программа может найти применение в организациях, предоставляющих рабочие места людям с ограниченными возможностями, упрощая ввод текста, позволяя более рационально использовать рабочее время.

6. Благодарности

Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (№ 18-37-00418, № 19-29-01135, № 19-31-90160) и Министерства науки и высшего образования Российской Федерации в рамках выполнения государственного задания Самарского университета и ФНИЦ «Кристаллография и фотоника» РАН.

7. Литература

[1] Тан, В. Аналитика Больших Данных и социальные сети / В. Тан, Б. Блейк, И. Салех // Открытые системы. СУБД. – 2013. – № 8. – С. 37-41.

[2] Rytsarev, I.A. Clustering of media content from social networks using bigdata technology / I.A Rytsarev, D.V. Kirish, A.V. Kupriyanov // Computer Optics. – 2018. – Vol. 42(5). – P. 921-927. DOI: 10.18287/2412-6179-2018-42-5- 921-927.

[3] Осминин, П.Г. Современные подходы к автоматическому реферированию и аннотированию / П.Г. Осминин // Вестник ЮУрГУ. Серия: Лингвистика. – 2012. – № 25 – С. 134-135.

[4] Рыцарев, И.А. Разработка и реализация сервисов по сбору данных социальных сетей в целях улучшения среды обитания человека / И.А. Рыцарев, А.В. Благов, М.И. Хотилин // Сборник трудов V международной конференции и молодежной школы «Информационные технологии и нанотехнологии» (ИТНТ) – Самара: Новая техника, 2018. – С. 2452-2457.

Building a graph of a sequence of text units to create a sentence generation system

M.P. Kaminskiy[1], I.A. Rytsarev[1,2], A.V. Kupriyanov[1,2]

[1]Samara National Research University, Moskovskoe Shosse 34A, Samara, Russia, 443086
[2]Image Processing Systems Institute of RAS - Branch of the FSRC "Crystallography and Photonics" RAS, Molodogvardejskaya street 151, Samara, Russia, 443001

Abstract. The article is devoted to the development of a text data analysis system. The approaches to the presentation of text from the posts of a single page in the form of a dictionary of phrases are considered. Within the framework of the work, data collection, filtering and processing using Big Data technologies were implemented.