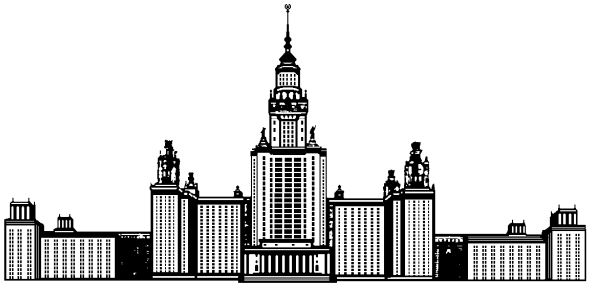

Московский государственный университет имени М.В. Ломоносова

Факультет вычислительной математики и кибернетики

Кафедра математических методов прогнозирования

Тематическое моделирование текстовых коллекций в диалоговых системах

Научный руководитель:

д.ф.-м.н., доцент, профессор РАН

К. В. Воронцов

Выполнил:

студент 417 группы

Н. Н. Кругликов

Москва, 2018

Содержание

1 Введение

Задача построения виртуального собеседника является центральной в области обработки естественного языка. В последнее время диалоговые системы снова набирают популярность. Многие крупные IT-системы создают диалоговые версии, такие как Siri (Apple), Cortana (Microsoft), Alexa (Amazon) и Алиса (Яндекс). [4]

Диалоговые системы бывают принято делить на (goal-oriented) и нецелевые (non-goal-oriented). Целевые диалоговые системы решают конкретную задачу пользователя и, как правило, взаимодействуют с внешним хранилищем данных. Нецелевые предназначены для развлечения пользователя с помощью имитации разговора с реальным собеседником. Нецелевые диалоговые системы могут быть порождающими (generative) и поисковыми (retrieval-based). Порождающие системы генерируют ответ с помощью некоторой порождающей модели текста, а поисковые выбирают ответ из набора готовых ответов.

Тематическое моделирование — способ извлечения скрытой структуры из коллекции текстовых документов. Тематическая структура помогает решать различные задачи из области обработки естественного языка, в том числе задачи информационного поиска. [13]

В настоящее время становится актуальной задача диалогового поиска [9], совмещающая подходы диалоговых систем и информационного поиска. Несмотря на большой потенциал тематических моделей в информационном поиске, тематические модели до сих пор практически не применялись в системах диалогового поиска. В данной работе исследуется возможность применения тематических моделей в этой области.

2 Диалоговые системы

2.1 Целевые диалоговые системы

Целевые диалоговые системы позволяют пользователю решить какую-нибудь конкретную задачу, например, заказать билет в кино. Целевые диалоговые системы, как правило, состоят из следующих компонент: [1]

- анализатор естественного языка (natural language understanding) преобразует высказывания пользователя в машинно-читаемую структуру данных;
- трекер состояния диалога (dialogue state tracker) определяет текущее состояние диалога, которое используется для выбора следующего действия;
- система определения действия (dialogue policy learning) выбирает следующее действие диалоговой системы в зависимости от текущего состояния;
- генератор естественного языка (natural language generator) преобразует результат действия из машинно-читаемой структуры в человеческий язык.

Каждая целевая диалоговая система создаётся и настраивается под конкретную задачу. При этом часто некоторые или все компоненты создаются инженерным методом, с большим числом вручную созданных правил. Такие системы невозможно перенастроить под другие задачи. В настоящие время создание универсальной диалоговой системой является открытой проблемой.

2.2 Нецелевые диалоговые системы

Нецелевые диалоговые системы, также называемые чатботами, не пытаются решить конкретную задачу. Их цель в том, чтобы поддержать разговор с пользователем на произвольную тему.

Существует два основных подхода к построению нецелевых диалоговых систем — генеративный и поисковый.

В генеративном подходе ответ на сообщение пользователя порождается с помощью некоторой модели. В настоящее время наиболее популярный способ построения таких систем — seq2seq-модели.

В поисковом подходе ответ на сообщение пользователя выбирается из большого набора готовых ответов. Как правило, для сообщения и всех ответов строятся векторные представления одной размерности, после чего ответ выбирается в соответствии с некоторой метрикой.

2.3 Оценка диалоговых систем

Целевые диалоговые системы можно оценивать как по откликам пользователей, так и по критериями, соответствующим завершению цели — например, получилось ли у пользователя купить билет в кинотеатр, и сколько сообщений ему для этого потребовалось.

Нецелевые диалоговые системы можно оценивать по метрикам перекрытия, которые обычно используются для оценки автоматического перевода и суммаризации, таким как BLEU и ROUGE. Однако было показано [7], что эти метрики практически не коррелируют с пользовательскими откликами, которые, таким образом, остаются единственным разумным способом оценки таких систем.

2.4 Практические аспекты реализации диалоговых систем

Требования к диалоговым системам

Чтобы диалоговая система была готова к применению на практике, она должна удовлетворять следующих требований:

- Быстрое время ответа. Пользователь ожидает от диалоговой системы той же скорости реакции, что и от обычного человека, поэтому обрабатывать запрос в течение минуты недопустимо.

- Надёжность. Пользователь общается с диалоговой системой в удобное для него время. Если однажды диалоговая система не ответит из-за перегрузки или профилактических работ, пользователь может больше никогда к ней не обратиться.

- Лёгкость масштабирования. Диалоговая система за которткий промежуток времени может быстро стать популярной, при этом также возможны значительные снижения нагрузки, например, в ночное время. Необходимо эффективно использовать вычислительные ресурсы, чтобы не допускать ни замедления работы под нагрузкой, ни простоя большого количества мощностей.

Диалоговые программные интерфейсы

В последнее время многие популярные веб-сервисы предоставляют сторонним разработчикам программные интерфейсы (API) для разработки диалоговых систем, интегрированных в этот сервис. В частности, такие интерфейсы предоставляют социальные сети ВКонтакте и Facebook, а также мессенджеры WhatsApp, Telegram и Viber.

Как правило, программные интерфейсы для диалоговых систем поддерживают два режима работы. В одном режиме работы, называемом long polling, стороннее приложение переодически запрашивает у сервиса новые события, такие, как сообщения от пользователей. В другом режиме, называемом callback или webhook, сервис самостоятельно оповещает стороннее приложение о новых событиях по заранее определённому URL-адресу.

Первый режим проще в использовании, но у него есть ряд недостатков. Во-первых, он нерационально расходует ресурсы процессора, поддерживая сетевое соединение, по которому сравнительно редко передаются данные. Во-вторых, в таком режиме заметно ограничена скорость реагирования диалоговой системы на сообщения пользователей.

Бессерверная архитектура

Облачные платформы, такие как Amazon Web Services [2] или Google Cloud Platform, позволили разработчикам веб-приложений гибко управлять вычислительными ресурсами. Программные интерфейсы облачных платформ дают возможность определять серверы, базы данных и сетевые соединения в терминах кода (Infrastructure as Code).

Новая парадигма в облачных вычислениях — бессерверная архитектура (serverless) — является логичным продолжением Infrastructure as Code. В этой парадигме понятие сервера полностью абстрагируется от разработчика, которому теперь достаточно реализовать функцию без побочных эффектов. Загруженная в облако функция вызывается в ответ на определённые события, а стоимость вычислительных ресурсов вычисляется на миллисекундной основе. При этом горизонтальное масштабирование полностью реализуется облачной платформой, прозрачно для разработчика.

Бессерверная архитектура позволяет разрабатывать быстрые, надёжные и легко масштабируемые диалоговые системы [3]. Время ответа такой диалоговой системы практически не

зависит от количества пользователей.

3 Вероятностное тематическое моделирование

Тематическое моделирование — метод выявления скрытой семантической структуры в корпусе текстов.

Формально, пусть D — множество документов в корпусе, W — множество слов. Предположим, что существует множество тем T и автор текста, пиша слово w в документе d, задумывал некую тему t. Тогда можно считать, что коллекция текстов представляет собой набор троек (d, w, t), взятых из дискретного распределения $p(d, w, t)$ на множестве $D \times W \times T$.

Добавим ещё одно условие — гипотезу условной независимости. Будем считать, что появление слова с темой t не зависит от документа, в котором появилось это слово:

$$p(w|t, d) = p(w|t)$$

Обозначим $p(w|t) = \phi_{wt}$ и $p(t|d) = \theta_{td}$. Тогда тематическую модель можно записать в виде

$$p(w|d) = \sum_{t \in T} p(w|t)p(t \mid d) = \sum_{t \in T} \phi_{wt}\theta_{td}$$

Построение тематической модели — задача восстановления распредений $p(w|t)$ и $p(t|d)$.

3.1 PLSA

PLSA [6] — простейшая вероятностная тематическая модель.

Для восстановления распределений максимизируется логарифм правдоподобия

$$L(\Phi, \Theta) = \sum_{d \in D} \sum_{w \in W} n_{dw} \ln \sum_{t \in T} \phi_{wt}\theta_{td} \to \max_{\Phi, \Theta}$$

при ограничениях

$$\sum_{w \in W} \phi_{wt} = 1, \phi_{wt} \geq 0; \quad \sum_{t \in T} \theta_{td} = 1, \theta_{td} \geq 0$$

Эта задача решается с помощью EM-алгоритма. На E-шаге по ϕ_{wt} и θ_{td} вычисляются вероятности $p(t|d, w)$ для всех $t \in T$, $d \in D$, $w \in W$. На M-шаге по вероятностям $p(t|d, w)$ вычисляются параметры ϕ_{wt} и θ_{td}. Изначальное приближение параметров ϕ_{wt} и θ_{td} генерируется случайно.

3.2 ARTM

Модель ARTM [13] является обобщением модели PLSA. К логарифму правдоподобия прибавляются регуляризаторы $R_i(\Phi, \Theta)$ с коэффициентами регуляризации τ_i.

Эта задача также решается с помощью EM-алгоритма, с небольшими модификациями M-шага.

Модель ARTM предоставляет широкие возможности по описанию требований к тематической модели с помощью регуляризаторов. Большинство тематических моделей, основанных на байесовском подходе, можно выразить с помощью ARTM, в том числе наиболее часто использующуюся на практике модель LDA.

3.3 Мультимодальные тематические модели

Более обобщённым подходом к вероятностному тематическому моделированию являются мультимодальные тематические модели. В мультимодальных моделях предполагается, что каждому документу документ описывается не только своим мешком слов, но и набором дополнительной информации, в качестве которой могут выступать автор текста или набор вручную проставленных тэгов.

При построении мультимодальной тематической модели оптимизируемый функционал представляет собой сумму логарифмов правдоподобия (с регуляризаторами, в случае ARTM) для каждой модальности.

4 Методы суммаризации текстов

Задача суммаризации заключается в уменьшении объёма текста без потери смысла. Методы суммаризации делятся на извлекающие (extractive) и порождающие (abstractive) [5]. Извлекающие методы строят суммаризации из фрагментов изначального текста, а порождающие используют модели порождения текста.

Как правило, извлекающие методы делят текст на фрагменты, ранжируют их согласно некоторому критерию и строят суммаризацию из N лучших предложений, где N — параметр алгоритма. Графовые методы суммаризации используют идеи, вдохновлённые алгоритмом PageRank, придуманным основателями Google для ранжирования веб-страниц. В качестве вершин графа вместо страниц используются фрагменты текста (например, LexRank использует слова, а TextRank — предложения), а в качестве рёбер — определённые каким-либо образом связи между предложениями.

Алгоритм TextRank [8] строит на предложениях текста взвешенный граф, где вес ребра определяется количеством общих слов в предложении, нормированных на их длину:

$$Similarity(S_i, S_j) = \frac{|\{w_k \mid w_k \in S_i, w_k \in S_j\}|}{\log|S_i| + \log|S_j|}$$

Затем вершинам присваиваются случайные веса, которые пересчитываются до сходимости по формуле:

$$WS(V_i) = (1-d) + d \star \sum_{V_j \in In(V_i)} \frac{w_{ji}}{\sum_{V_k \in Out(V_j)} w_{jk}} WS(V_j)$$

Здесь через V_i, V_j обозначены вершины графа, через w_{ij} — веса рёбер, а d — параметр, лежащий между 0 и 1 и устанавливаемый авторами на 0.85. Это формула является адаптацией формулы, предложенной в методе PageRank для ориентированных невзвешенных графов.

5 Тематическое моделирование в диалоговых системах

Практика применения тематического моделирования в диалоговых системах достаточно невелика. Одной из преград к применению тематических моделей является так называемая проблема коротких текстов. Проблема заключается в том, что задача тематического моделирования достаточно устойчиво решается на коллекциях, состоящих из достаточно длинных документов. Если документы сопоставимы по размеру с типичным сообщением в мессенджере, строить тематические модели традиционными способами не получается. Для этих случаев разработаны тематические модели с дополнительными ограничениями, например, модель TwitterLDA добавляет предположение, что каждый документ содержит только одну тему.

В существующей литературе по нецелевым диалоговым системам тематическое моделирование используется только в качестве вспомогательной информации для порождающих seq2seq сетей [10,11]. Как правило, авторы используют TwitterLDA для моделирования большой коллекции коротких сообщений.

В данной работе выбран иной подход. В качестве источника ответов на реплики пользователя рассматривается коллекция текстовых документов, длина которых позволяет построить полноценную тематическую модель. Для того, чтобы извлечь из длинного документа ответ подходящей длины, используются методы суммаризации.

6 Тематическое моделирование в поисковых диалоговых системах

Пусть имеется коллекция текстов X, по которой построена тематическая модель (Φ, Θ). Пользователь присылает реплику r. Задача системы — подобрать максимально релевантный фрагмент текста s для реплики r.

Пусть для каждой реплики r определён тематический вектор $q(r)$. Кроме того, пусть для каждого текста x определена суммаризация σ_x, а для каждой суммаризации определён тематический вектор $b(\sigma_x)$.

Тогда ответ s будем искать как

$$\operatorname*{argmin}_{\sigma} d(q(r), b(\sigma)),$$

где d — некоторая метрика в пространстве тематических векторов.

Определим тематический вектор $q(r)$ как среднее тематических векторов по словам реплики:

$$q(r) = \frac{\sum_w p(t|w)}{|r|}$$

Оценим тематический вектор $b(\sigma_x)$ каждой суммаризации как среднее векторов $p(t|x,w)$ по словам суммаризации.

$$b(\sigma_x) = \frac{\sum_w p(t|x,w)}{|\sigma_x|}$$

В качестве расстояния $d(\cdot,\cdot)$ будем использовать косинусное расстояние.

Релевантность ответа реплике будем определять по пользовательскому отклику.

7 Эксперименты

7.1 Установка

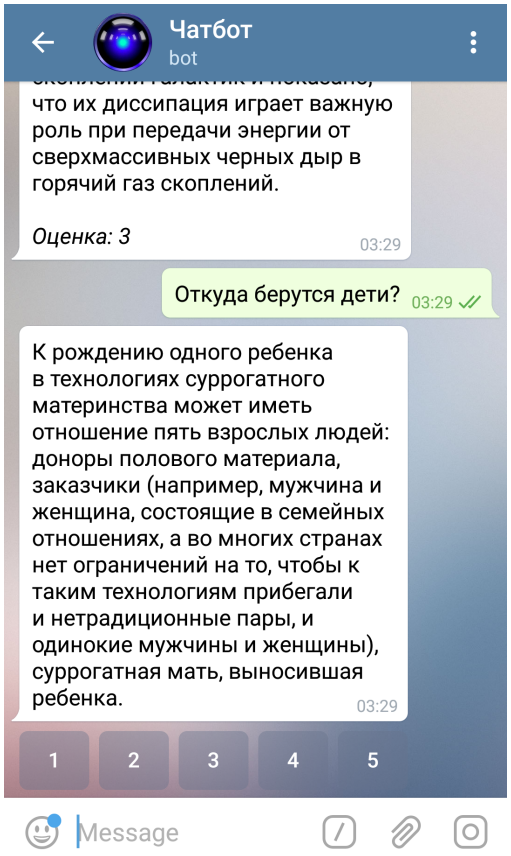

Рис. 1: Пример работы системы

Для проведения экспериментов необходимо тестировать диалоговую систему на реальных пользователях. Была разработано приложение для тестирования разных диалоговых систем, позволяющее собирать оценки пользователей.

В качестве платформы для реализации диалоговой системы был выбран мессенджер Telegram. Преимущество этой платформы состоит, во-первых, в большом количестве активных русскоязычных пользователей, а во-вторых, в богатых возможностях программного интерфейса для диалоговых систем, позволяющего, в частности, легко добавить функциональность оценки сообщений системы пользователями.

В качестве облачного провайдера был выбран Amazon Web Services [2]. Этот провайдер одним из первых запустил поддержку бессерверных приложений и предоставляет богатый набор инструментов, помогающих интегрировать бессерверные приложения с базами данных и другими ресурсами.

Архитектура разработанной системы состоит из двух основных независимых частей. Первая часть, названная диспетчером, отвечает за взаимодействие с программным интерфейсом Telegram, а также хранение сообщений и оценок пользователей. Каждое сообщение, полученное от пользователя, отправляется во вторую часть, названную экспериментом. Каждый эксперимент

— это метод, принимающий на вход идентификатор пользователя и сообщение, и возвращающий ответ. Экспериментов несколько, что позволяет проводить сравнение с помощью A/B тестирования разных методов построения нецелевых диалоговых систем.

Разработанная архитектура обеспечивает медианное время ответа меньше одной секунды вне зависимости от количества пользователей.

7.2 Коллекция

В качестве коллекции текстов был взят набор научно-популярных статей с сайта «Постнаука». Собранный датасет содержит 2976 документов. Одна из особенностей портала — богатая коллекция тэгов, вручную проставленная редакторами под каждой статьёй. Всего, за исплючением стопслов, в коллекции 43196 уникальных леммы, а также 1799 тэгов.

7.3 Тематическая модель

По коллекции статей с «Постнауки» была построена тематическая модель с 20 темами (одна из них — фоновая). Для улучшения качества модели была использована модальность тэгов.

В качестве метода суммаризации была выбрана открытая реализация метода TextRank с небольшими модификациями [12]. Количество слов в суммаризации N_{sum} является гиперпа-

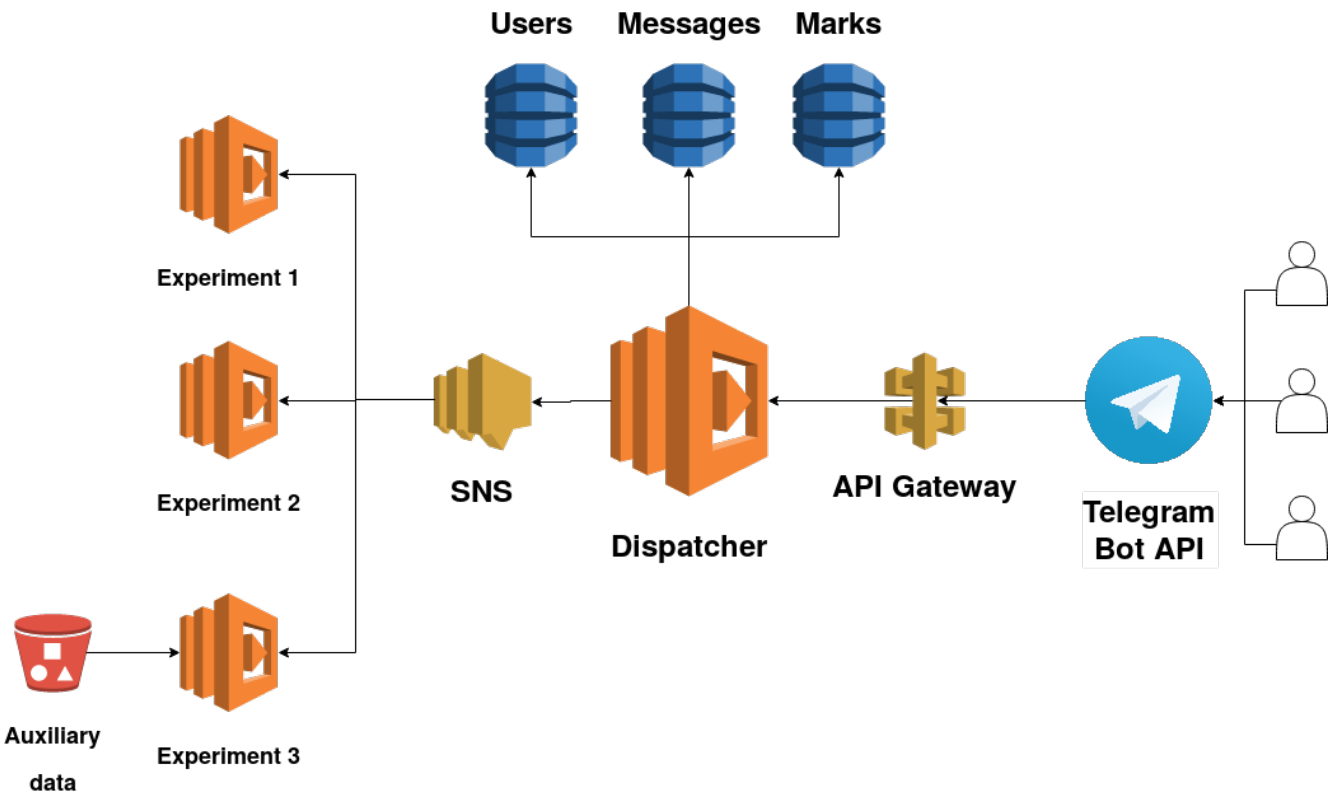

Рис. 2: Архитектура системы

топ-токены	топ-тэги
вид	эволюция
эволюция	биология
группа	антропология
организм	человек
ген	палеонтология
эволюционный	антропогенез
раса	физиология
возникать	происхождение_человека
отбор	эволюционная_биология
часть	иммунитет

Таблица 1: Топ-токены и топ-тэги первой темы модели

раметром. Было проведено три варианта экспериментов с $N_{sum} = 50$, 100 и 150.

7.4 Оценки пользователей

Пользователь:	Можно ли вылечить рак содой
Система:	Существует взаимосвязь между увеличением употребления в пищу жиров и повышением уровня холестерина в крови, и оба эти фактора наряду с ростом уровня женских гормонов, в свою очередь, положительно коррелируют с увеличением частоты возникновения рака молочной железы и наступлением менархе в более раннем возрасте. Очень важное наблюдение — значительная корреляция между высоким содержанием жиров и животных белков в питании и уровнем репродуктивных гормонов и наступлением менархе в раннем возрасте, ведь оба последних фактора повышают риск возникновения рака молочной железы. Мы смогли не только обнаружить простую взаимосвязь между потреблением в пищу жиров и раком молочной железы, но и выявить гораздо более сложный = комплекс факторов, позволяющих понять, как питание влияет на риск возникновения рака.

Таблица 2: Пример ответа, оценённого на 5

В ходе эксперимента было собрано 250 оценок от 17 пользователей. Оценки были равномерно распределены по вариантам алгоритма с разными длинами суммаризаций. В среднем, пользователи выше оценили алгоритм, выбирающий ответ из 100-словных суммаризаций.

Пользователь:	Когда я зеваю, у меня открывается рот. Многие говорят, что это потому, что я хочу спать, но на самом деле потому что я хочу проснуться. Почему человек зевает?
Система:	Пытаться продолжить все виды деятельности, которые прямо не связаны с приобретением и собственностью, а это очень сложный вопрос, потому что в основном мы сейчас переживаем в развитых странах эпоху глобального капитализма, которая означает, что каждый человек вовлечен в некоторый процесс конкуренции, касающейся личной собственности и личного заработка. А кризис, в котором находится человечество, за пределами собственно науки — я просто упомяну, что человечество вступило в такой период, когда возникают серьезные вопросы, не может ли все человечество погибнуть отчасти по собственной вине, отчасти потому, что гуманитарные и социальные науки недостаточно развиты и не предлагают нам достаточно квалифицированных выходов из этого кризиса, который гораздо важнее, чем кризис познания.

Таблица 3: Пример ответа, оценённого на 1

8 Заключение

В работе предложен способ применения полноценных тематических моделей в задаче построения нецелевой диалоговой системе.

Кроме того, разработана готовая к применению в реальных приложениях установка для запуска и оценки диалоговых систем.

Дальнейшее улучшение работы данного метода возможно за счёт увеличения текстовой коллекции или внедрения дополнительных нетематических метрик для ранжирования фрагментов текста.

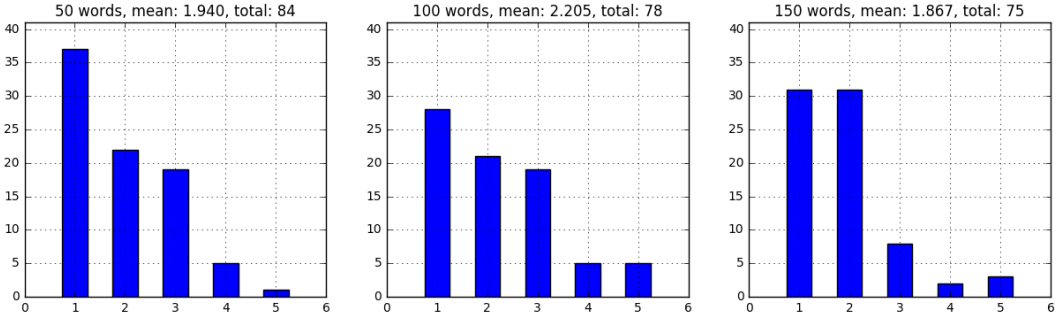

Рис. 3: Распределение оценок пользователей в зависимости от длины суммаризации

Список литературы

[1] A Survey on Dialogue Systems: Recent Advances and New Frontiers / H. Chen, X. Liu, D. Yin, J. Tang. — 2017. — no. 1.
http://http://arxiv.org/abs/1711.01731.

[2] *Amazon Web Services Inc.* Overview of Amazon Web Services // *White Paper.* — 2015. — no. December.
http://https://docs.aws.amazon.com/aws-technical-content/latest/aws-overview/aws-overview.pdf?ic

[3] Building a Chatbot with Serverless Computing / M. Yan, P. Castro, P. Cheng, V. Ishakian // *Proceedings of the 1st International Workshop on Mashups of Things and APIs - MOTA '16.* — 2016. — Pp. 1–4.
http://http://dl.acm.org/citation.cfm?doid=3007203.3007217.

[4] *DALE R.* The return of the chatbots // *Natural Language Engineering.* — 2016. — Vol. 22, no. 5. — P. 811–817.

[5] *Gambhir M., Gupta V.* Recent automatic text summarization techniques: a survey // *Artificial Intelligence Review.* — 2017. — jan. — Vol. 47, no. 1. — Pp. 1–66.
http://http://link.springer.com/10.1007/s10462-016-9475-9.

[6] *Hofmann T.* Probabilistic latent semantic analysis // *Proc. of Uncertainty in Artificial Intelligence, UAI'99.* — 1999. — P. 21.
http://http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.84.1137&rep=rep1&type=pdf.

[7] How NOT to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation / C. Liu, R. Lowe, I. V. Serban et al. // *CoRR.* — 2016. — Vol. abs/1603.08023.
http://http://arxiv.org/abs/1603.08023.

[8] *Mihalcea R., Tarau P.* TextRank: Bringing order into texts // *Proceedings of EMNLP.* — 2004. — Vol. 85. — Pp. 404–411.
http://http://acl.ldc.upenn.edu/acl2004/emnlp/pdf/Mihalcea.pdf.

[9] *Radlinski F., Craswell N.* A Theoretical Framework for Conversational Search // *Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval - CHIIR '17.* — 2017. — Pp. 117–126.
http://http://dl.acm.org/citation.cfm?doid=3020165.3020183.

[10] Response Selection with Topic Clues for Retrieval-based Chatbots / Y. Wu, W. Wu, Z. Li, M. Zhou. — 2016.
http://http://arxiv.org/abs/1605.00090.

[11] Topic Aware Neural Response Generation / C. Xing, W. Wu, Y. Wu et al. — 2016. — Pp. 3351–3357.
http://http://arxiv.org/abs/1606.08340.

[12] Variations of the Similarity Function of TextRank for Automated Summarization / F. Barrios, F. López, L. Argerich, R. Wachenchauzer. — 2016.
http://http://arxiv.org/abs/1602.03606.

[13] *Vorontsov K., Potapenko A.* Additive regularization of topic models // *Machine Learning.* — 2015. — Vol. 101, no. 1-3. — Pp. 303–323.
http://http://machinelearning.ru/wiki/images/4/47/Voron14mlj.pdf.