А.В. Заболеева-Зотова, Ю.А. Орлова, В.Л. Розалиев

Комплексный семантический анализ потока новостных текстов[1]

Аннотация. Работа посвящена адаптации текстовой информации для лиц с ограниченными возможностями здоровья по зрению. Рассматривается извлечение ключевых сущностей из текста новостной статьи и их визуализация. Кратко рассмотрены и проанализированы существующие методы и алгоритмы определения нечетких дубликатов текстов, такие как TF-IDF и его модификации, Long Sent, Shingles, Lex Rand. Для решения задачи разделения новостей по тематикам разработан алгоритм, включающий метод шинглов. Представлены несколько вариантов параллельной реализации алгоритма с использованием технологий CUDA, Open CL и Google App Engine. Оценены параметры алгоритма (время работы, ускорение по сравнению с последовательной обработкой) применительно к задаче анализа новостных текстов. Дан пример с программной реализацией комплексного анализа новостного текста, основанный на комбинации смыслового анализа и последующего аннотирования текста статьи с представлением ее в сжатом виде в формате так называемой mind map (интеллект-карты).

Ключевые слова: новостной текст, нечеткие дубликаты, шинглы, TF-IDF, CUDA, Open CL, Google App Engine, аннотирование, mind map.

Введение

Создание безбарьерной среды для лиц с ограниченными возможностями здоровья (ОВЗ) – это одна из важнейших задач государства. Каждый человек должен иметь равные возможности для получения образования и коммуникации с внешним миром. Многое делается в данном направлении, однако сложнее всего адаптировать современный образовательный процесс для людей с ОВЗ. В данной работе рассматриваются модели и методы адаптации текстовой информации для людей с ОВЗ по зрению, которые могут применяться для автоматизации сбора и визуализации новостной информации с официальных сайтов образовательной организации и образовательных web-ресурсов.

Сбором новостной информации в Интернете занимаются различные программы, так называемые агрегаторы. Эти программы собирают по заданным признакам информацию из Интернет-ресурсов в одно хранилище. Новостные агрегаторы - одни из самых востребованных ресурсов в Интернете. Новости читают все, однако для людей с ОВЗ по зрению практически отсутствуют удобные средства и возможности работы с новостными ресурсами и лентами новостей (кроме масштабирования изображения на экране дисплея). Например, ленты новостей с образовательных сайтов становятся для них малодоступными, так как разработчики дизайна сайтов заинтересованы в размещении большего объема отображаемой информации за счет уменьшения размера шрифта. На решение этой проблемы направлены последние издаваемые законы и нормативные акты министерства образования РФ (в том числе закон № 273-ФЗ), но большинство Интернет-ресурсов пока вообще не пытается сделать доступным изображение для слабовидящих людей, в лучшем случае предоставляется возможность увеличить шрифт основного текста.

В данной работе представлен подход, позволяющий расширить возможности коммуника-

[1] Работа выполнена при финансовой поддержке РФФИ (проекты №13-07-00351, 14-07-97017, 15-07-07519, 15-07-05440).

ции с окружающим миром для лиц с ОВЗ по зрению, основанный на методах отображения текстовой информации в адаптированной форме – аннотирование и представление в виде графической записи MindMap. Основной идеей работы является объединение различных источников новостей за счет алгоритма поиска нечетких дубликатов фрагментов текста. Далее осуществляется аннотирование текстов и их визуальное представление. Альтернативой зрительному представлению может быть устное воспроизведение новостной информации (или аннотаций к ней) на основе звукосинтезирующих программ. Однако такое решение имеет ряд недостатков, главные из которых это необходимость установки дополнительных программ или постоянный доступа в Интернет, достаточно низкая скорость воспроизведения, а соответственно, и восприятия информации.

1. Обзор существующих систем аннотирования и анализа текстов

В настоящее время имеется много систем семантического анализа текстов и различных новостных агрегаторов [3, 6]. Среди отечественных – это TextAnalyst, Content Analyzer, технологии АОТ, RCO, МедиаЛингва Аннотатор, система Яндекс Новостей. Среди зарубежных – Extractor, QDA Miner (пакеты WordStat и Simstat), системы Inxight Summarizer (компонент поискового механизма AltaVista), Intelligent Text Miner (IBM), MEAD, NetSum, Newblaster и пр. Кроме того, в США уже существуют программы (роботы), сами пишущие новости на основе анализа данных с различных сайтов. Для примера, проект QuakeBot, создающий заметки о землетрясениях, проект Mapping LA, публикующий сообщения по мотивам криминальной хроники. Однако роботы умеют работать только с наборами конкретных данных: результатами соревнований, показателями коммерческой деятельности, биржевыми индексами, заполняя пустые места в уже готовых фразах. В большинстве таких систем отсутствует возможность обработки русскоязычных текстов.

В данной работе мы представляем программный продукт для обработки текстов на русском языке. Предлагается метод для анализа новостных текстов, сочетающий агрегацию новостных статей методом шинглов с комплексным анализом текста.

2. Методы установления тематического подобия текстов

Существует несколько основных методов установления тематической близости документов: TF-IDF, Long Sent, Shingles, Lex Rand и т.п. [2, 4]. Рассмотрим наиболее распространенные из них.

Идея алгоритма TF-IDF и ряда его модификаций похожа: для всей коллекции документов строится словарь, ставящий каждому слову в соответствие число документов, в которых оно встречается хотя бы один раз и определяется средняя длина документа. Затем строится частотный словарь документа и для каждого слова вычисляется его «вес». Затем выбираются и сцепляются в алфавитном порядке в строку шесть слов с наибольшими значениями «веса». В качестве сигнатуры документа используется контрольная сумма CRC32 полученной строки.

Метод Long Sent основан на том, что документ разбивается на предложения, которые упорядочиваются по убыванию длины, выраженной количеством слов, а при равенстве длин – в алфавитном порядке. Затем выбираются и сцепляются в строку в алфавитном порядке два самых длинных предложения. В качестве сигнатуры документа используют контрольную сумму CRC32 полученной строки.

В методе Lex Rand вначале для всей коллекции документов строится словарь, из которого удаляются слова с наибольшими и наименьшими значениями обратной частоты документа относительно запроса (IDF). Затем на основании этого словаря генерируются 10 дополнительных словарей, содержащих примерно на 30% меньше слов, чем в исходном. Слова удаляются случайным образом. Для каждого документа строится 11 сигнатур. Дубликатами считаются документы, в которых совпадает хотя бы одна сигнатура.

Метод шинглов и его модификации (*Shingles algorithm*) был предложен в 1997 году Бродером [12]. Он основан на представлении документа в виде последовательностей фиксированной длины N, состоящих из слов, расположенных в тексте рядом. При этом на последовательности могут накладываться различные ограничения. Например, ограничение,

требующее, чтобы слова находились в одном предложении. Такие последовательности в одних источниках называют "шинглами", в других "N-граммами" [1]. Два документа считаются похожими, если множества их N-грамм существенно пересекаются. Аналогично можно определить похожесть двух предложений, или же предложения и текста.

Д. Фетерли была предложена модификация алгоритма шинглов, в которой документ представлялся 84 цепочками слов [4]. Выбор из всего множества шинглов происходит по следующей схеме: для всех шинглов документа рассчитывается значение 84 хеш-функций. Для каждой хеш-функции выбирается шингл с максимальным значением хеш-функции. Затем эти 84 шингла разбиваются на 6 групп по 14 шинглов. Такие группы называются "супершинглами". Далее документ представляется всевозможными попарными сочетаниями из шести супершинглов, которые называются "мегашинглами". Число таких мегашинглов равно 15 (число сочетаний из 6 по 2). Два документа считают сходными по содержанию, если у них совпадает хотя бы один мегашингл. Данный алгоритм является неустойчивым при модификации текста в процессе заимствований. Кроме того, этот алгоритм не обнаруживает заимствования в случае малого совпадения документов. Описанный алгоритм применим для отбора документов для поиска, если критерием отбора документов являются масштабные заимствования. Для поиска конкретных заимствованных фрагментов алгоритм не применим.

3. Алгоритм выделения тематически близких новостных статей

В результате экспериментального сравнения качества работы каждого из методов, было принято решение дополнить алгоритм шинглов собственным алгоритмом для выделения ключевых сущностей и наиболее важных предложений.

Алгоритм состоит из нескольких шагов [6, 7].

1) Сначала производится канонизация текста. Из оригинального текста удаляются предлоги, союзы, знаки препинания, теги разметки веб-страниц. Существительные в текстах приводятся к именительному падежу, единственному числу [3].

2) Далее находятся кандидаты в ключевые слова. Кандидатами в ключевые слова являются существительные в именительном падеже и уникальные слова (персоны, организации, места и прочие). Также в кандидаты добавляются слова, которые не удалось определить при помощи предварительного морфологического анализа.

3) Затем вычисляем «вес» wt_{kw} для каждого кандидата по формуле:

$$wt_{kw} = tf_{kw} \times idf_{kw}, \quad (1)$$

где

$$tf_{kw} = \frac{n_i}{\sum_k n_k}, \quad (2)$$

откуда tf_{kw} – частота слова в статье; n_i – число вхождений слова в статью; $\sum_k n_k$ – число слов в статье;

$$idf_{kw} = \log \frac{D - df + 0.5}{df + 0.5}, \quad (3)$$

где D – число статей; df – число статей, содержащих данное слово.

Пороговое значение веса wt_{kw} для отнесения кандидата к ключевым словам экспериментально установлено равным 0,8×максимальный вес кандидатов. В ключевые слова включаются кандидаты, превышающие это значение.

4) Учитывая структуру новостного текста, вычисляем вес предложений. Новостной текст состоит из лид и контекста. Лид – это заголовок новости и основные факты, содержащиеся в тексте. Заголовок Новости отражает ее тему и содержит не более 10 слов. Основные факты отражены в первом и втором абзацах. Контекстом считают третий абзац и все следующие, в которых раскрывают детали происходящего. Вес предложений

$$W_s = N_{kw} \cdot tf_{kw} \cdot ParagraphWeight \cdot k, \quad (4)$$

где $N_{kw} \cdot tf_{kw}$ – вес ключевого слова в предложении;

N_{kw} – количество вхождений ключевого слова в предложение;

tf_{kw} – частота ключевого слова в документе;

$ParagraphWeight$ – относительный вес параграфа в тексте, равен 0.35 для первого параграфа (лид), 0.2 – для второго, 0.1 – для остальных (контекст), вес заголовка равен 3;

k – коэффициент значимости предложения внутри параграфа; для первого предложения в абзаце равен 1, для остальных 0.8.

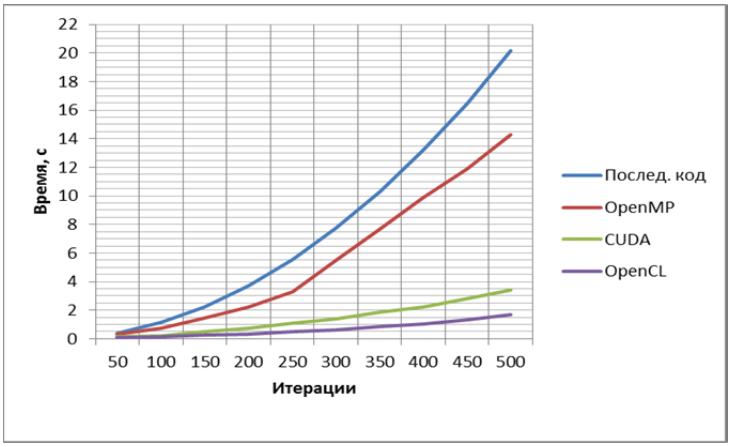

Рис. 1. Зависимость времени обработки от числа текстов

В дальнейшем, для корректировки полноты и точности передаваемой информации, мы планируем ввести дополнительный коэффициент, учитывающий время появления новости.

В ключевые предложения включаются предложения, имеющие вес равный 0,8×максимальный вес предложений для каждой статьи. При этом учитывается, что общее число слов в таких предложениях не должно быть меньше 30. Если слов меньше, то порог снижается.

5) В выбранных предложениях выделяются шинглы длиной 10 слов. Выбор происходит внахлест на одно слово. Таким образом, получается набор шинглов, мощность которого равна количеству слов, минус длина шингла, плюс один.

6) Сравниваемые тексты представляются в виде набора шинглов и вычисленных по ним контрольных сумм, рассчитанных через хеш-функцию (CRC32). Далее находим совпадающие контрольные суммы и включаем тексты в тематическую рубрику.

Чтобы сохранить высокую точность определения тематической близости, свойственную алгоритму шинглов, и при этом сократить длительность его работы без сокращения числа сравнений, как в модификации Фетерли, мы распараллеливаем сравнение значений хеш-функций и нахождение тематически близких текстов.

4. Параллельная реализация алгоритма шинглов

Рассмотрим результаты тестирования разработанных нами и программно-реализованных алгоритмов шинглов, использующих разные технологии распараллеливания.

Первый способ распараллеливания – реализация алгоритма шинглов с использованием технологий CUDA и Open CL. Результаты экспериментальной оценки работы алгоритма четырьмя разными реализациями с оценкой времени (в секундах) для каждой реализации показаны на Рис. 1. Число итераций можно рассматривать как число анализируемых новостных текстов.

Тестирование проводилось на устройстве с CPU «Intel core i5 3.0 GHz», видеокартой (Cuda) NVIDIA GeForce GTX 650Ti 2GB и OpenCL AMD Radeon 7870 2GB.

По сравнению с последовательным алгоритмом, для параллельной реализации алгоритма шинглов с использованием технологии Cuda (вторая кривая снизу) среднее ускорение составило 5.10, а с применением технологии OpenCL (первая кривая снизу) – 10.12. При этом параллельно выполнялся только подсчет и сравнение хешей, а нормализация текста проводилась последовательно. OpenCL превосходит CUDA в 1.93 раза, что в данном случае объясняется тем, что тестируемая с OpenCL видеокарта незначительно превосходит своего конкурента по вычислительной мощности.

Второй способ распараллеливания – реализация алгоритма шинглов с использованием технологий Google App Engine. Эта технология используется для обеспечения услуг по предоставлению ресурсов при размещении информации на сайтах и web-приложениях (хостинг) на серверах Google с бесплатным именем <имя_сайта>.appspot.com, либо с собственным именем, задействованным с помощью служб Google. Приложения, разворачиваемые на базе

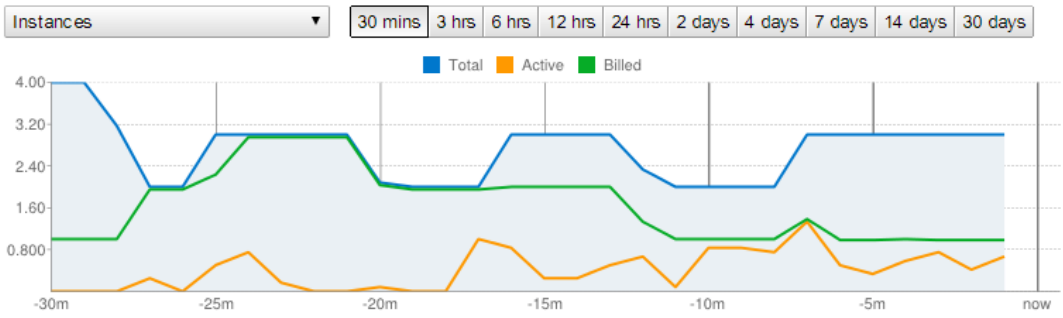

Рис. 2. График использования вычислительных единиц

Табл. 1. Время и ускорение с использованием Google App Engine

Итерации	5	10	15	20	25	30
Время с App Engine	9.558	18.454	27.792	36.450	45.416	54.436
Время локально	10.212	20.414	30.530	40.518	50.888	63.166
Ускорение	1,068	1,106	1,099	1,112	1,120	1,160

Рис. 3. Графики времени и ускорения обработки текстов

(по x – итерации, по y - ускорение)

App Engine, должны быть написаны на Python, Java, PHP. Для проведения тестирования, алгоритм был переписан на языке Python.

График использования вычислительных единиц приведен на Рис. 2. Тестирование работы программы проводилось на 30 текстах размером от 46 до 50 Кб. В тестовой выборке содержалось по 10 текстов на 3 новостные темы. Сначала запускалась программа при помощи Google App Engine, затем локально: вычислительный узел – одно ядро Core i5 3.3 GHz. Таким образом, за счет введения распараллеливания процессов обработки можно существенно увеличить скорость объединения новостных текстов в тематические кластеры. В Табл. 1 и на Рис. 3 показано время разделения текстов по трем тематикам.

Google App Engine позволяет достичь ускорения порядка 1,2 раза по сравнению с локальным запуском приложения с последовательным выполнением процессов. При этом необходимо учесть, что для выполнения операций с Google App Engine выделялось число вычислительных узлов не более четырех. При выделении большего количества вычислительных мощностей для работы с текстами больших размеров можно получить значительное ускорение по сравнению с использованием локальной машины.

На следующем этапе для каждого такого кластера выделяются ключевые фразы и слова, которые используются для построения визуального представления новости.

5. Сети терминов и представление в виде интеллект-карты новости (mind map)

Наиболее подходящим для визуального представления новости является использование графового метода [5, 8, 9, 13]. Среди таких методов создания сетей терминов были рассмотрены TextRank (приложение алгоритма PageRank к задачам обработки естественного языка), метод построения графа горизонтальной видимости (Horizontal Visibility Graph – HVG), метод Марковских случайных полей (Conditional Random Fields – CRF). В результате анализа их работы мы выбрали метод TextRank для установления связей между ключевыми словами и прорисовки графической схемы Mind Map. Подробное описание визуализации Mind Map показано в [10].

Для проверки алгоритмов была разработана программа, осуществляющая анализ новостных статей и их визуализацию. На вход программы для обработки подается новостной текст (в случае анализа одной новости) или ссылки на анализируемые новости (в случае анализа нескольких статей). Выходом является аннотация и визуализация статей. На Рис. 4-Рис. 6 представлены экранные формы окон разработанной программы, осуществляющей комплексный анализ одной новостной статьи и ее визуализацию.

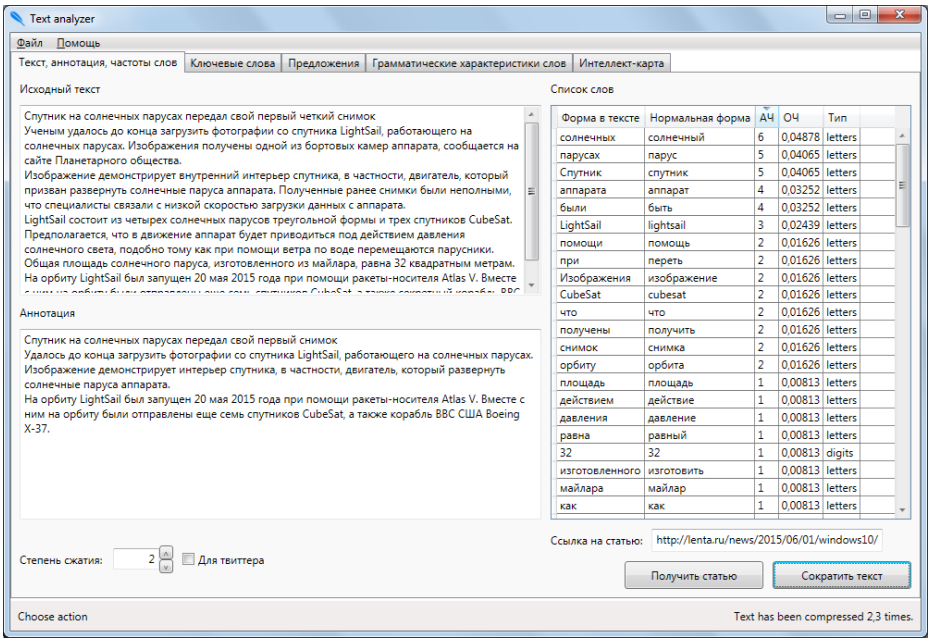

Рис. 4. Экран программы для анализа новостных статей. Аннотация новости

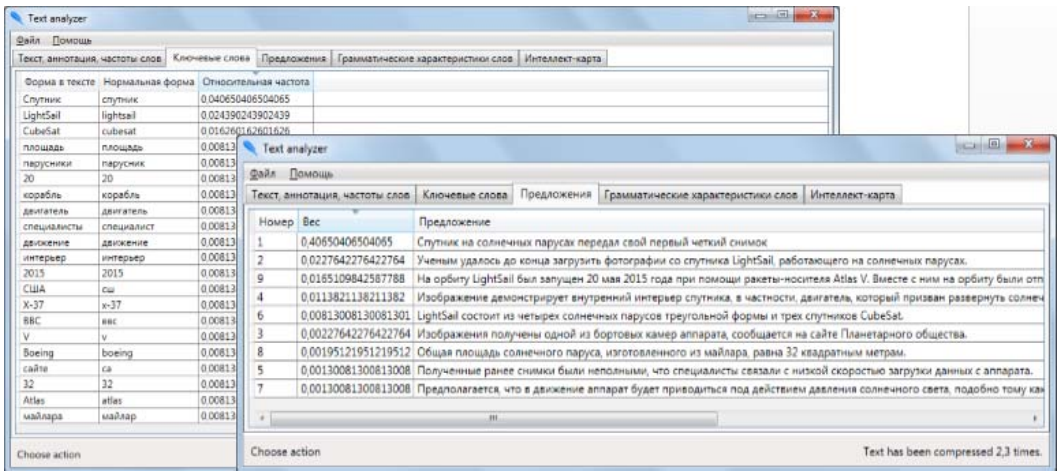

Рис. 5. Экран программы для анализа новостных статей. Ключевые слова и предложения

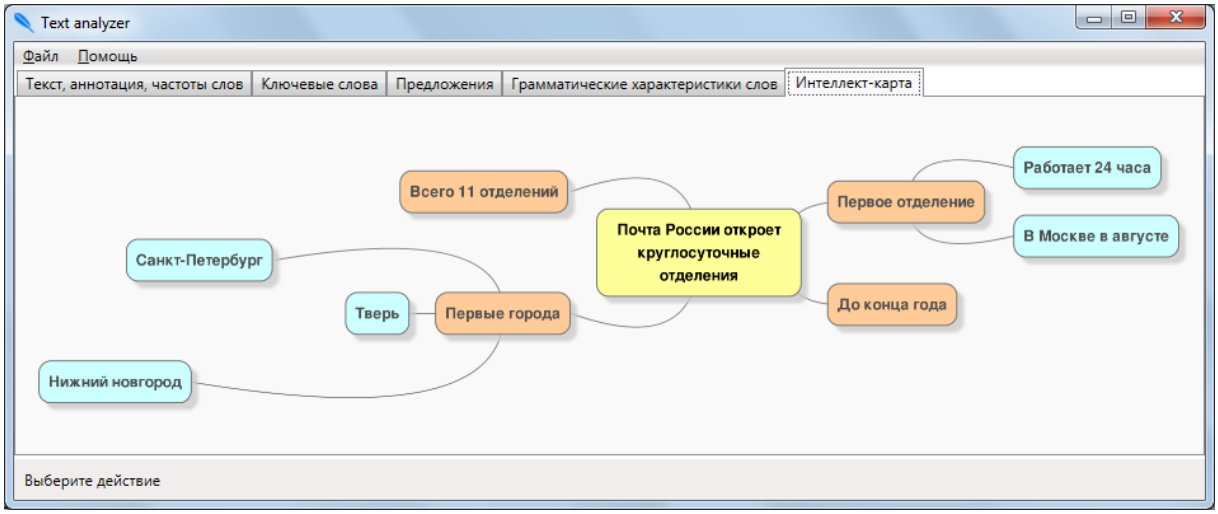

Рис. 6. Экран программы для анализа новостных статей. Схема Mind Map

На Рис. 6 показан пример визулизации статьи для новости со следующим текстом: «До конца года Почта России откроет круглосуточные отделения. Всего планируется открыть одиннадцать отделений. Первое отделение, работающее 24 часа, появится в Москве уже в августе. Первыми городами для апробирования работы круглосуточных центров получения и отправки корреспонденции выбраны Санкт-Петербург, Тверь и Нижний Новгород».

Заключение

На основании результатов сравнения скорости разделения текстов по тематикам, можно сделать вывод, что в зависимости от вида решаемой задачи и используемых аппаратных технологий, можно применять различные параллельные реализации алгоритма шинглов. При этом на наших тестовых выборках технология OpenCL значительно превосходила другие варианты.

Использование автоматизированной методики извлечения ключевых слов позволило повысить качество обработки новостных Интернет-статей. Отметим, что для параметра времени при автоматизированной обработке учитывается не только непосредственно время анализа системой, но и время, необходимое для окончательной корректировки текстов. Качество результата же оценивалось по таким критериям: сохранение ключевых фактов, связность ключевых сущностей, сохранение синтаксической структуры текста после удаления незначащих частей. Каждый из названных критериев оценивался экспертами по шкале от 0 до 10 баллов. Затем для оценки качества (адекватности) извлеченных ключевых сущностей аннотации находилось среднее арифметическое для перечисленных трех показателей по каждому тексту. В итоге по сравнению с ручным способом, время определения ключевых сущностей уменьшилось как минимум в 2 раза, а качество обработки новостей осталось на том же уровне, как и при анализе текста человеком.

Как одно из возможных приложений, рассмотренные методы визуализации новостной информации можно использовать для графической адаптации текстовой информации новостного потока образовательной организации для людей с ограниченными возможностями здоровья по зрению.

Литература

1. Алгоритм шинглов для веб-документов, поиск нечетких дубликатов текстов, сравнение текстов на похожесть [Электронный ресурс]. – Режим доступа: http://www.codeisart.ru/part-1-shingles-algorithm-for-web-documents.
2. Автоматизированный подход к определению авторства текста / А.В. Муха, В.Л. Розалиев, Ю.А. Орлова, А.В. Заболеева-Зотова // Известия ВолгГТУ. Серия "Актуальные проблемы управления, вычислительной техники и информатики в технических системах". Вып. 17: межвуз. сб. науч. тр. / ВолгГТУ. - Волгоград, 2013. - № 14 (117). - С. 51-54.
3. Заболеева-Зотова А.В. Автоматизация семантического анализа текста технического задания: монография / А.В. Заболеева-Зотова, Ю.А. Орлова. – Волгоград: ИУНЛ, 2010. – 155 с

4. Зеленков Ю.Г. Сравнительный анализ методов определения нечетких дубликатов для Web-документов / Ю.Г. Зеленков, И.В. Сегалович // Труды IX Всероссийской научной конференции «Электронные библиотеки: перспективные методы и технологии, электронные коллекции» – RCDL'2007. – 9 с.
5. Ландэ Д.В. Использование графов горизонтальной видимости для выявления слов, определяющих информационную структуру текста / Д.В. Ландэ, А.А. Снарский, Е.В. Ягунова // Труды XV Всероссийской научной конференции «Электронные библиотеки: перспективные методы и технологии, электронные коллекции» – RCDL'2013, Ярославль, 14-17 октября 2013 г. – 7 с.
6. Солошенко А.Н. Thematic Clustering Methods Applied to News Texts Analysis / А.Н. Солошенко, Ю.А. Орлова, В.Л. Розалиев, А.В. Заболеева-Зотова // Knowledge-Based Software Engineering: Proceedings of 11th Joint Conference, JCKBSE 2014 (Volgograd, Russia, September 17-20, 2014) / ed. by A. Kravets, M. Shcherbakov, M. Kultsova, Tadashi Iijima ; Volgograd State Technical University [etc.]. – [Б/м]: Springer International Publishing, 2014. – P. 294-310. – (Series: Communications in Computer and Information Science ; Vol. 466)
7. Солошенко А.Н. Establishing Semantic Similarity of the Cluster Documents and Extracting Key Entities in the Problem of the Semantic Analysis of News Texts / А.Н. Солошенко, Ю.А. Орлова, В.Л. Розалиев, А.В. Заболеева-Зотова // Modern Applied Science. - 2015. - Vol. 9, No. 5. – С. 246-268.
8. Орлова Ю.А. Processing of Spatial and Temporal Information in the Text / А.С. Дмитриев, А.В. Заболеева-Зотова, Ю.А. Орлова, В.Л. Розалиев // World Applied Sciences Journal (WASJ). – 2013. – Vol. 24, Spec. Issue 24: Information Technologies in Modern Industry, Education & Society. – С. 133-137.
9. Усталов Д.А. Извлечение терминов из русскоязычных текстов при помощи графовых моделей // Теория графов и приложения = Graphs theory and applications: материалы конференции. – 2012. – С. 62–69.
10. Automated Mind Map Generation from News Texts Based on Link Grammar / А.Н. Солошенко, Ю.А. Орлова, В.Л. Розалиев, А.В. Заболеева-Зотова // Creativity in Intelligent Technologies and Data Science. CIT&DS 2015: First Conference (Volgograd, Russia, September 15-17, 2015): Proceedings / ed. by A. Kravets, M. Shcherbakov, M. Kultsova, O. Shabalina. – [Switzerland]: Springer International Publishing, 2015. – P. 637-654. – (Ser. Communications in Computer and Information Science. Vol. 535)
11. Anisimov A.V. A method for the computation of the semantic similarity and relatedness between natural language words / A.V. Anisimov, O.O. Marchenko, V.K. Kysenko // Cybernetics and Systems Analysis, July 2011. – Volume 47, Issue 4. – Pp. 515-522
12. Broder A. On the resemblance and containment of documents. Compression and Complexity of Sequences / A. Broder // SEQUENCES'97, IEEE Computer Society, 1998, Pp. 21-29
13. Furu Wei. A document-sensitive graph model for multi-document summarization / Furu Wei, Wenjie Li, Qin Lu, Yanxiang He // Knowledge and Information Systems, February 2010. Volume 22, Issue 2. – Pp. 245-259.
14. Sheng-Tun Li. Constructing tree-based knowledge structures from text corpus / Sheng-Tun Li, Fu-Ching Tsai // Applied Intelligence, August 2010. – Volume 33, Issue 1. – Pp. 67-78.

Заболеева-Зотова Алла Викторовна. Начальник Управления региональных и межгосударственных программ Российского фонда фундаментальных исследований. Окончила Волгоградский государственный педагогический институт им. А.С. Серафимовича в 1980 году. Доктор технических наук, профессор. Автор более 250 печатных работ, включая 6 монографий. Область научных интересов: системный анализ, искусственный интеллект, нечеткая математика, компьютерная лингвистика, логико-лингвистическое моделирование, интеллектуальный анализ информации.
E-mail: zabzot@gmail.com

Орлова Юлия Александровна. Доцент кафедры «Системы автоматизированного проектирования и поискового конструирования» Волгоградского государственного технического университета. Окончила Волгоградский государственный технический университет в 2007 году. Кандидат технических наук, кандидат педагогических наук. Автор более 225 печатных работ, включая 8 монографий. Область научных интересов: анализ текстовой информации, искусственный интеллект, распознавание образов и анализ изображений, нечеткие системы и модели, распознавание и анализ движений человека. E-mail: yulia.orlova@gmail.com

Розалиев Владимир Леонидович. Доцент кафедры «Системы автоматизированного проектирования и поискового конструирования» Волгоградского государственного технического университета. Окончила Волгоградский государственный технический университет в 2007 году. Кандидат технических наук. Автор более 125 печатных работ. Область научных интересов: искусственный интеллект, распознавание образов и анализ изображений, анализ текстовой информации, нечеткие системы и модели, распознавание и анализ движений человека, определение и моделирование эмоциональных реакций человека. E-mail: vladimir.rozaliev@gmail.com