

Таким образом, при переписке может быть использовано вплоть до трёх уровней защиты информации, и для того, чтобы удостовериться в том, что сообщение не было перехвачено и изменено, может быть прикреплена цифровая подпись. Подобный подход обеспечивает необходимый уровень защиты информации, передаваемой посредством сети интернет.

Список использованных источников:

1. Шифрование [Электронный ресурс]. – Режим доступа: https://professorweb.ru/my/csharp/base_net/level2/2_3.php. – Дата доступа: 08.04.2018.
2. Криптографические методы и средства защиты информации [Электронный ресурс]. – Режим доступа: <http://itsphera.ru/it/cryptographic-methods-and-tools-for-information-protection.html>. – Дата доступа: 08.04.2018.

ПРИМЕНЕНИЕ РЕКУРСИВНЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ АНАЛИЗА ТОНАЛЬНОСТИ ТЕКСТА

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Витковский А.В.

Жвакина А. В. – к. т. н, доцент

Современные порталы в сети Интернет позволяют пользователям высказывать свое мнение о различных предметах, событиях, явлениях. Эти мнения могут быть полезными для различных исследований, необходимых аналитикам, SMM-специалистам, бренд-менеджерам, PR-агентам и иным специалистам, нуждающимся в получении агрегированной информации. Таким образом существует необходимость в инструментах для анализа отзывов пользователей. Пользователи оставляют свои комментарии в интернете на естественных языках, таких как английский, русский и др., что является проблемой для обработки программными средствами. В работе рассматривается такой метод обработки естественного языка, как анализ тональности текста.

Обработка естественного языка (Natural Language Processing, NLP) – общее направление искусственного интеллекта и математической лингвистики. Оно изучает проблемы компьютерного анализа и синтеза естественных языков. Одной из задач, решаемых в рамках обработки языка, является анализ тональности текста. Анализ тональности текстов – это класс методов анализа содержания, предназначенный для классификации автоматического распознавания в тексте лексики с эмоциональной окраской, а также мнений (эмоциональных оценок) автора об объектах, которые упоминаются в тексте.

Для решения задачи анализа тональности текста применяют нейронные сети. Искусственные нейронные сети (НС) — совокупность моделей биологических нейронных сетей. Представляют собой сеть элементов — искусственных нейронов — связанных между собой синаптическими соединениями. Сеть обрабатывает входную информацию и в процессе изменения своего состояния во времени формирует совокупность выходных сигналов. Работа сети состоит в преобразовании входных сигналов во времени, в результате чего меняется внутреннее состояние сети и формируются выходные воздействия. Обычно НС оперирует цифровыми, а не символьными величинами. Искусственные нейронные сети — набор математических и алгоритмических методов для решения широкого круга задач. Алгоритмы на основе машинного обучения показывают свою эффективность в задачах обработке естественных языков.

Большинство систем прогнозирования настроений работают по простому алгоритму, рассматривая слова в изоляции, давая положительные баллы для положительных слов и отрицательные баллы для отрицательных слов, а затем суммирует баллы. Таким образом, порядок слов игнорируется и теряется важная информация. Однако, модели на основе нейронных сетей фактически создают представления целых предложений, основанные на структуре предложения. Они вычисляют тональность, основанную на том, как слова влияют значение более длинных фраз. Таким образом, модель на основе нейронных сетей не так легко обмануть, как обычные алгоритмы.

Наиболее часто используемыми в исследованиях методами являются методы на основе машинного обучения с учителем. Сутью таких методов состоит в том, что сначала нейронная сеть получает коллекцию из данных и уже готовых точных решений, и в процессе машинного обучения сеть настраивается, и далее уже на других данных может выдавать нужные результаты.

Для решения задачи анализа тональности могут использоваться рекурсивные нейронные сети. Рекурсивные нейронные сети (англ. Recursive neural network; RvNN) – вид нейронных сетей, работающих с данными переменной длины. Модели рекурсивных сетей используют иерархические структуры образцов при обучении. Например, изображения, составленные из сцен, объединяющих подсцены, включающие много объектов. Выявление структуры сцены и её деконструкция – нетривиальная задача. При этом необходимо как идентифицировать отдельные объекты, так и всю структуру сцены. В рекурсивных сетях нейроны с одинаковыми весами активируются рекурсивно в соответствии со структурой сети. В процессе работы рекурсивной сети вырабатывается модель для предсказания для структур переменной размерности, так и скалярных структур через активацию структуры в соответствии с топологией. Рекурсивные нейронные сети успешно применяются при обучении последовательных структур и деревьев в задачах обработки естественного языка, при этом фразы и предложения моделируются через векторное представление слов. Рекурсивные сети первоначально появились для распределённого представления структур, используя

предикаты математической логики. Разработки рекурсивных сетей и первые модели появились в середине 1990-х.

Рекурсивные нейронные сети работают с векторными представлениями слов. Векторное представление слов – класс методов и подходов для обработки естественного языка, суть которых состоит в том, что слову словаря ставится в соответствие n-мерный вектор. Для получения векторных представлений используются нейронные сети, которые после обучения могут автоматически выполнять указанную задачу. Векторные представления позволяют программе работать со значениями слов, т.к. смысл векторных представлений состоит в том, чтобы для слов со схожими значениями получались близкие вектора, а для далеких понятий расстояние должно быть большое. На практике используются достаточно большие размерности, от 200 до 500. Таким образом для огромного количества слов из словаря можно получить векторы, которые могут обрабатываться на ЭВМ. С увеличением размерности результат анализа становится точнее, а скорость работы и обучения замедляется.

Список использованных источников:

1. Заенцев, И. В. Нейронные сети: основные модели / И. В. Заенцев. – Воронеж, 1999. – 76 с.
2. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank – Stanford University, 2013.
3. Википедия:Рекурсивные нейронные сети. [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Рекурсивные_нейронные_сети. – Дата доступа: 25.03.2018.
4. Википедия: Анализ тональности текста. [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Анализ_тональности_текста. – Дата доступа: 25.03.2017.

SQRT-ДЕРЕВО КАК НЕЗАСЛУЖЕННО ЗАБЫТАЯ СТРУКТУРА ДАННЫХ

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Вишневецкий В.А.

Гербик А.И. – ассистент

В рамках данной работы будет освещена структура данных под названием sqrt-дерево и её преимущество перед другими открытыми структурами данных.

Для объяснения принципа работы sqrt-дерева будет проще рассмотреть следующую теоретическую задачу: “Дан набор из N чисел. На некоторых его отрезках требуется посчитать произведение чисел по заранее заданному модулю M”.

Описание sqrt-дерева лучше начать с идеи разбиения массива на блоки длины \sqrt{n} (здесь и далее под \sqrt{n} имеется ввиду $\lfloor \sqrt{n} \rfloor$), за исключением последнего блока, который может иметь размер меньше, чем \sqrt{n} . Принцип разбиения указан на рисунке 1:

$$\{a_1, \dots, a_{\sqrt{N}}\}, \{a_{\sqrt{N}+1}, \dots, a_{2\sqrt{N}}\}, \dots, \left\{ a_{\left\lfloor \frac{N}{\sqrt{N}} \right\rfloor \cdot \sqrt{N} + 1}, \dots, a_N \right\}$$

Рис. 1 – принцип разбиения на блоки

Предпросчитаем для каждого блока: произведение всех чисел, принадлежащих ему, произведение чисел на каждом префиксе и суффиксе этого блока. А также для каждого отрезка блоков посчитаем произведение чисел. В итоге суммарное время работы предпросчета будет $O(n)$. Используя эти данные, мы уже можем решать вышеприведенную задачу за $O(\sqrt{n})$ времени на запрос. Заметим, что мы можем отвечать за $O(1)$ на все запросы, кроме тех, которые затрагивают элементы ровно одного блока. Чтобы эффективно отвечать на такой тип запросов, мы можем построить в этом блоке такую же структуру данных, как была описана выше. Будем выполнять это действие рекурсивно для каждого блока, который имеет размер больше единицы. Таким образом мы имеем древообразную структуру, изображенную на рисунке 2:

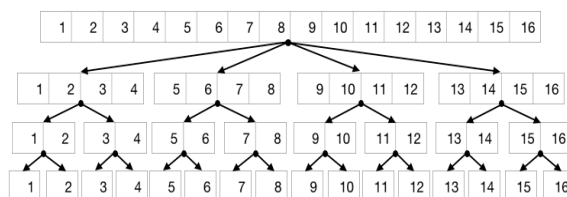


Рис. 2 – sqrt-дерево

На каждом слое находится ровно N вершин, поэтому для оценки сложности работы нам достаточно найти количество слоёв. Давайте заметим, что новый слой создается только когда предыдущий слой имеет