ПРОТОТИП СИСТЕМЫ КЛАССИФИКАЦИИ ВЕБ-СТРАНИЦ НА ОСНОВЕ КОНТЕНТА С ИСПОЛЬЗОВАНИЕМ ГЛУБОКИХ НЕЙРОННЫХ СЕТЕЙ

С.С. Мельниченко, *магистрант*
Московский технический университет связи и информатики
(Россия, г. Москва)

DOI:10.24412/2500-1000-2023-2-2-32-35

Аннотация. *Качество процесса классификации веб-страниц оказывает огромное влияние на системы поиска информации. В данной статье предложено решение, объединяющее результаты классификаторов текстовых и графических данных, чтобы получить точное представление веб-страниц. Процесс классификации графических и текстовых данных был реализован с помощью моделей глубокого обучения. Система классификации может быть использована как для рекомендации контента, так и для фильтрации нежелательной информации.*

Ключевые слова: *классификация, LSTM, CNN, глубокое обучение, агрегация данных, нейронная сеть.*

Информационно-поисковые системы играют важную роль в современном обществе [1]. Целью информационно-поисковой системы является сбор, хранение и предоставление эффективного механизма поиска для клиента. Качество процесса индексации и классификации играет решающую роль в процессе поиска информации.

Наиболее распространенные методы классификации веб-страниц основаны на анализе текста и графических данных [2]. Такой подход объясняется тем фактом, что классификация остальных встроенных мультимедийных данных, таких как изображения, аудио- и видеоданные, является трудоемким и дорогостоящим с точки зрения вычислений процессом. В данной статье представлен алгоритм классификации веб-страниц на основе текста и изображений с помощью глубокого обучения.

Предлагаемая архитектура системы

Архитектура классификатора включает в себя несколько блоков: парсер, классификатор изображений, классификатор текста и объединитель. Каждый из этих блоков отвечает за выполнение соответствующих задач. Парсер включает в себя веб-сканер, который собирает текстовые и графические данные из интернета, он также оценивает веса текстовых и графических данных, а затем сохраняет их в отдельных репозиториях. Классификатор изображений создает текст, связанный с изображениями, затем классификатор текста классифицирует текстовые данные из парсера и классификатора изображений. Последний блок – это объединитель, который объединяет результаты обоих классификаторов.

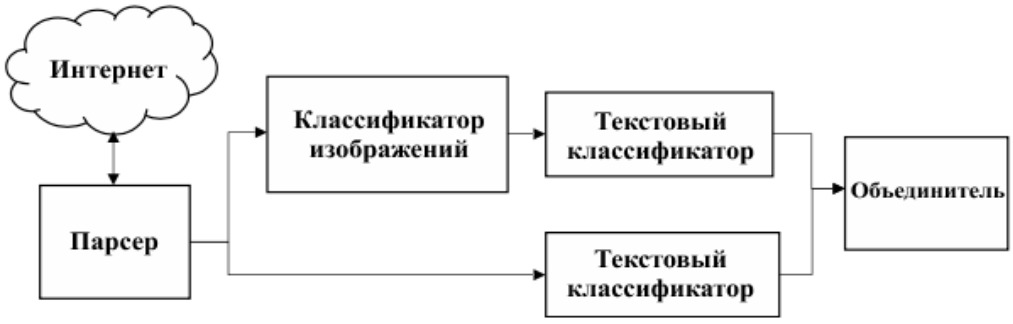

Рис. 1. Архитектура системы

Парсер

Парсер – это компонент, который включает в себя веб-сканер и систему хранения для этих структур данных. Слабо связанная архитектура системы позволяет использовать и другие подходы для интеллектуального анализа данных. В то время как поисковый робот перемещается по веб-страницам, он сохраняет данные в соответствии с принципами ключ-значение. Для каждой собранной веб-страницы ключ представляет собой хэш-код адреса веб-страницы, значение представляет ссылки на три компонента данных: текст, изображения и мета-теги с веб-страницы, которая содержала ключевые слова метаданных. Эти ссылки расположены в отдельных структурах данных для хранения текстовых и двоичных данных. Каждый абзац веб-страницы и изображение хранятся в отдельном блоке с соответствующим весом. Веса представляют приоритет каждого компонента данных, который позже используется в сводке вычисления категории. Изначально веса, относящиеся к каждому текстовому абзацу на веб-странице, равны единице, гибкость слабосвязанной архитектуры системы позволяет вычислять веса для каждого компонента данных отдельно на основе различных алгоритмов. Алгоритм вычисления весов может основываться на следующих свойствах:

1. Внешний вид текста: стили шрифта, цвета и размер текстовых данных для каждого параметра.
2. Расположение абзацев и изображений. Этот метод включает в себя: анализ иерархии тегов.
3. Численная статистика, где используются такие алгоритмы, как TF-IDF, Okapi BM25 [3].
4. Комбинирование методов, при котором один или несколько методов могут быть использованы для вычисления весов для каждого компонента данных.

Классификатор изображений

Классификатор изображений включает в себя глубокую нейронную сеть для генерации подписи к изображению. Он получает компоненты данных изображения от парсера и генерирует функцию с помощью алгоритма YOLO. Классификатор состоит из двух нейронных сетей: сверточная нейронная сеть (Convolutional neural network, CNN) на основе YOLO для извлечения признаков и нейронной сети долгой-краткосрочной памяти (Long short-term memory, LSTM) для генерации текстовой последовательности, поскольку она сохраняет релевантные данные во время процесса обучения и исключает нерелевантную информацию с помощью слоя фильтра забывания (Forget gate).

Текстовый классификатор

В проведенном исследовании для системы реального времени, которая собирает веб-страницы и непрерывно работает в фоновом режиме, достаточно простого и эффективного вычисления класса с точностью более 95%. Более сложные расширения, требующие большего количества вычислительных ресурсов, могут быть достигнуты с помощью платформ высокопроизводительных вычислений (High performance computing, HPC) и методов непрерывного развертывания DevOps [4].

Объединитель

В предлагаемой архитектуре целью объединителя является объединение компонентов текстов и изображений. Структура данных содержит заголовок и часть данных. Заголовок содержит хэш-код веб-страницы и два глобальных веса: $\boldsymbol{W_i}$ представляет глобальный вес изображения и $\boldsymbol{W_T}$ глобальный вес текста. Часть данных включает в себя набор компонентов данных с тремя параметрами: тег (*I, T*), который показывает, к какому типу данных принадлежит компонент, локальный вес и данные упорядоченного списка с числовым представлением меток классов. Этих данных достаточно для того, чтобы объединитель суммировал набор компонентов данных для каждой веб-страницы в отдельности. Объединитель генерирует теги результатов T путем агрегирования результатов отдельных изображений $\boldsymbol{C^{image}}$ и текстов $\boldsymbol{C^{text}}$. Где $\boldsymbol{C^{image}} = \{c^{\boldsymbol{image}} : c^{\boldsymbol{image}} \in R^n\}$, а функция агрегации может быть определена как

$$A: O(\{\boldsymbol{C^{text}} \rightarrow \boldsymbol{w^{text}}, \boldsymbol{W^{text}}, \boldsymbol{C^{image}} \rightarrow \boldsymbol{w^{image}}, \boldsymbol{W^{image}}\}) \rightarrow \boldsymbol{T}, \quad (1)$$

что включает в себя сопоставленные компоненты изображения и текста относительно их локальных весов:

$$c^{image} \rightarrow w^{image}: \{\forall_i: c_i^{image} \rightarrow w_i^{image}\} \quad (2)$$

$$c^{text} \rightarrow w^{text}: \{\forall_i: c_i^{text} \rightarrow w_i^{text}\} \quad (3)$$

Два глобальных параметра W^{text} и W^{image} регулируют приоритет классификации для каждого типа компонента. В данном случае существует только два типа компонентов (текст и изображение). Выходная функция *O*, может отличаться в зависимости от целей, представленных ниже:

Чтобы получить только один класс из агрегатора, *O* должно быть принято в качестве функции аргумента максимизации (Argmax).

Для получения постоянного количества категорий функция *O* должна выбрать первое n наибольшее количество категорий (если n не превышает общее количество классов).

Чтобы обнаружить новые n классов с веб-страницы, функция *O* должна обработать данные как псевдокод, представленный на рисунке 2.

```
function FIND_N_NEW_CLASSES(exited_tags, n, classes = {class: value})
    result ← set()
    inserted_classes ← 0
    sorted_classes ← sort(classes.values, descending)
    for class in sorted_classes do
        if len(inserted_classes) < n and class not in existed_tags then
            result.insert(class)
            n ← n + 1
        end if
    end for
return result
end function
```

Рис. 2. Псевдокод алгоритма обнаружения n новых классов

Заключение. В процессе фильтрации нежелательного контента в браузере, а также для систем рекомендации контента крайне важна классификация материалов, содержащихся в интернете. В данной статье представлен прототип метода классификации веб-страниц на основе текста и изображений при помощи технологии нейронных сетей глубокого обучения. Предложенное решение представляется возможным доработать и расширить для последующего увеличения точности за счет простоты его реализации.

Библиографический список

1. Jochen Hartmann, Juliana Huppertz, Christina Schamp, Mark Heitmann Comparing automated text classification methods, International Journal of Research in Marketing. – 2019. – Vol. 36, № 1. – Pp. 20-38.

2. Oliver Schulte, Kurt Routley, Aggregating Predictions vs. Aggregating Features for, in IEEE Symposium on Computational Intelligence and Data Mining (CIDM), 2014.

3. Yoon Kim, Convolutional Neural Networks for Sentence Classification, in Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 2014.

4. Alastair R. Rae, Daniel Le, Jongwoo Kim, George R. Thoma, Main Content Detection in HTML Journal Articles, in Conference: the ACM Symposium, 2018.

5. Peter Rousseeuw, Mia Hubert, Anomaly detection by robust statistics, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery. – 2018. – Vol. 8, № 2.

6. Linxuan Yu, Yeli Li, Qingtao Zeng, Yanxiong Sun, Yuning Bian, Wei He, Summary of web crawler technology research // Journal of Physics: Conference Series. – 2020. – Vol. 1449, № 1.

7. Alex Sherstinsky, Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network, Physica D: Nonlinear Phenomena. – 2020. – Vol. 404.

8. Linxuan Yu, Yeli Li, Qingtao Zeng, Yanxiong Sun, Yuning Bian, Wei He, Summary of web crawler technology research // Journal of Physics: Conference Series. – 2020. – Vol. 1449, №1.

PROTOTYPE OF A CONTENT-BASED WEB PAGE CLASSIFICATION SYSTEM USING DEEP NEURAL NETWORKS

S.S. Melnichenko, *Graduate Student*
Moscow Technical University of Communications and Informatics
(Russia, Moscow)

Abstract. *The quality of the web page classification process has a huge impact on information retrieval systems. In this article, a solution is proposed that combines the results of classifiers of text and graphic data in order to obtain an accurate representation of web pages. The process of classifying graphical and textual data was implemented using deep learning models. The classification system can be used both to recommend content and to filter unwanted information.*

Keywords: *classification, LSTM, CNN, deep learning, data aggregation, neural network.*