

## **ОЦЕНКА ЭФФЕКТИВНОСТИ PYTHON-БИБЛИОТЕК ДЛЯ СЕНТИМЕНТ-АНАЛИЗА (на материале англоязычных и русскоязычных тревел-блогов)**

В работе проводится оценка эффективности Python-библиотек для sentiment-анализа текстов тревел-блогов. Методика исследования основана на анализе корректности классификации отдельных предложений и определения общей тональности текста.

*Ключевые слова:* sentiment-анализ; Python-библиотеки; тревел-блоги; обработка естественного языка.

## **EVALUATION OF THE EFFECTIVENESS OF PYTHON LIBRARIES FOR SENTIMENT ANALYSIS (Based on English and Russian Travel Blogs)**

This study evaluates the effectiveness of Python libraries for sentiment analysis of travel blog texts. The research methodology is based on assessing the accuracy of classifying individual sentences and determining the overall sentiment of the text.

*Key words:* sentiment analysis; Python libraries; travel blogs; natural language processing.

В эпоху цифровых технологий пользовательский контент в социальных сетях играет ключевую роль в формировании общественного мнения, трендов и популярных тенденций. Тревел-блоги представляют собой ценный источник информации, содержащий субъективные оценки путешественников. Изучение контента тревел-блогов позволяет выявлять основные эмоции и настроения авторов, что важно для туристической индустрии, маркетинга и социологических исследований. С учетом значительного объема таких данных возникает необходимость в их систематическом и системном анализе. Sentiment-анализ текстов тревел-блогов предоставляет информацию о предпочтениях пользователей, уровне их удовлетворенности и общих настроениях в отношении тех или иных туристических направлений.

Объемные тексты тревел-блогов могут содержать оценки сразу нескольких аспектов путешествия – транспорта, проживания, достопримечательностей, сервиса. Выполняя задачу автоматизированного анализа такого текста, модель должна не просто интерпретировать отдельные его фрагменты, а учитывать многоаспектность текста и выявлять его общий тон, считывая контекст и соотношение положительных и отрицательных оценок.

В зависимости от подхода к обработке текстовых данных различают несколько основных методов sentiment-анализа: лексиконные, методы машинного обучения и нейросетевые модели. Лексиконные методы основаны на использовании заранее составленных словарей, в которых словам и фразам присваиваются соответствующие тональные оценки: положительные или отрицательные, иногда – нейтральные, в зависимости от модели. Такие методы эффективны при анализе коротких сообщений, но они слабо учитывают контекст, наличие сарказма, сложные языковые конструкции и многозначность слов. Методы машинного обучения используют размеченные корпуса текстов для обучения классификаторов, которые на основе статистических закономерностей определяют тональность новых текстов. Эти методы могут учитывать контекст, но требуют больших объемов данных для эффективного обучения и последующего функционирования. Современные подходы, ос-

нованные на нейронных сетях, особенно трансформерных архитектурах, таких как BERT и XLM-RoBERTa, способны анализировать тональность на более глубоком уровне, моделируя зависимости в тексте.

Одна из часто используемых Python-библиотек для анализа текстов на английском языке – VADER, лексиконно-ориентированный инструмент, который позволяет определять тональность текстов, анализируя не только лексику, но и неформальные элементы, такие как эмодзи, заглавные буквы и знаки препинания. Эта библиотека подходит для быстрого анализа коротких текстов, например, твитов. Другая библиотека, TextBlob, использует более общий подход на основе машинного обучения и предоставляет простые интерфейсы для обработки текста, таких как sentiment-анализ, распознавание сущностей и перевод текста. Для анализа русскоязычных текстов используется DeepPavlov, который включает в себя предобученные модели для различных задач анализа текстов непосредственно на русском языке, в том числе для sentiment-анализа. Например, RuSentiment представляет собой набор моделей, обученных на огромных корпусах данных на русском языке, и сочетает в себе методы машинного обучения и нейронных сетей.

Для оценки эффективности различных Python-библиотек для sentiment-анализа используются такие метрики, как *Precision* ‘точность’, *Recall* ‘полнота’ и *F1-Score*, которые позволяют количественно измерить качество анализа, выполненного моделью. *Precision* рассчитывается как доля предложений, корректно классифицированных как положительные или отрицательные, среди всех предложений, которые модель отнесла к этой категории. *Recall* показывает, сколько из всех действительно положительных или отрицательных предложений модель смогла правильно определить. *F1-Score* представляет собой гармоническое среднее между *Precision* и *Recall*.

Для проведения оценки эффективности Python-библиотек, используемых для sentiment-анализа, выбраны по одному тестовому тексту на русском (URL: <https://clck.ru/3KRP63>) и английском (URL: <https://clck.ru/3KRP2Y>) языках объёмом по 1000 слов, в которых нами предварительно вручную размечены предложения с положительной и отрицательной эмоциональной окраской. Для трёх Python-библиотек для русского языка (Hugging Face Transformers RuBERT и XLM-RoBERTa, DeepPavlov RuSentiment) и четырёх для английского (Hugging Face Transformers RuBERT и RoBERTa, VADER, TextBlob) разработаны алгоритмы, которые анализируют текст и выводят на экран выявленные моделями положительно и отрицательно окрашенные предложения, а также определяет общую тональность текста. Затем полученные результаты сопоставляются с предварительной разметкой.

Результаты оценки эффективности Python-библиотек для sentiment-анализа тестового текста на английском языке представлены в таблице 1.

Табл.1 Оценка эффективности Python-библиотек для sentiment-анализа тестового текста на английском языке

Библиотека Python	Hugging Face Transformers (BERT)	Hugging Face Transformers (RoBERTa)	VADER	TextBlob
Precision	47%	<b>83%</b>	62%	42%
Recall	94%	<b>78%</b>	87%	84%
F1-Score	63%	<b>80%</b>	72%	56%
Корректность определения итоговой тональности тестового текста	Нет	<b>Да</b>	Да	Да

Модель на базе Hugging Face Transformers RoBERTa демонстрирует максимальные показатели эффективности среди представленных библиотек, высокую точность и сбалансированную работу, на что указывает высокий F1-Score. Библиотеки VADER и TextBlob показывают хороший Recall, но выдают большее число ложных срабатываний, в то время как модель BERT, напротив, чаще выдает ложноположительные результаты.

Результаты оценки эффективности Python-библиотек для sentiment-анализа тестового текста на русском языке представлены в таблице 2.

*Табл.2* Оценка эффективности Python-библиотек для sentiment-анализа тестового текста на русском языке

Библиотека Python	Hugging Face Transformers (RuBERT)	DeepPavlov (RuSentiment)	Hugging Face Transformers (XLM-RoBERTa)
Precision	58%	86%	<b>72%</b>
Recall	84%	67%	<b>90%</b>
F1-Score	59%	75%	<b>80%</b>
Корректность определения итоговой тональности тестового текста	Нет	Нет	<b>Да</b>

В случае sentiment-анализа текстов на русском языке модель XLM-RoBERTa из библиотеки Hugging Face Transformers демонстрирует максимальные показатели эффективности, о чём говорит высокий F1-Score, а также единственная из исследуемых библиотек правильно определяет общую тональность тестового текста. Модели RuBERT и DeepPavlov демонстрируют значительное число ложных срабатываний и ошибочно определяют общую тональность тестового текста.

Результаты исследования показали, что выбор оптимальной Python-библиотеки для sentiment-анализа текстов тревел-блогов существенно зависит от особенностей анализируемого языка, объёма и сложности исходного материала. Модели на базе Hugging Face Transformers RoBERTa и XLM-RoBERTa показали лучшие результаты по точности и полноте, при этом модель XLM-RoBERTa продемонстрировала максимальную эффективность в анализе русскоязычных текстов, правильно определяя итоговую тональность текста.