Обзор литературы

Проанализируем существующую литературу, посвященную задаче определения фейковых отзывов.

В работе [1] текст отзыва обрабатывается (удаление стоп-слов + стемминг), затем отзывы векторизуются с помощью bag-of-words, удаляются те фичи, которые соответствуют словам, используемым менее, чем в 5 отзывах, затем обучаются модели Random Forest и наивный байесовский классификатор. Случайный лес по итогам экспериментов продемонстрировал лучшее качество по основным метрикам (accuracy, precision, recall, F1).

В публикации [2] представлен схожий подход решения задачи детекции фейковых отзывов. Сначала выполняется предобработка текстов, состоящая из стемминга, удаления пунктуации и стоп-слов, после чего для векторизации текстов используется bag-of-words. Далее авторы обучают несколько различных ML-моделей: SVM, решающее дерево, случайный лес и градиентный бустинг. (Основной акцент в этой работе был сделан на анализе влияния различных этапов предобработки текста на итоговое качество)

В статье [3] авторы описывают следующий пайплайн решения. Во-первых, выполняется препроцессинг текстовых данных, взятых из датасета Yelp: токенизация, удаление стоп-слов и лемматизация. Далее для получения эмбеддингов текстов отзывов используется TF-IDF на биграммах и триграммах. Помимо полученных с помощью TF-IDF признаков, авторы также используют «поведенческие признаки» (behavioural features) пользователей, например, caps-count (количество заглавных букв в отзыве), punct-count (количество знаков пунктуации) и emojis-count (количество смайликов). В качестве моделей для задачи классификации реальных/фейковых новостей были выбраны логистическая регрессия, наивный байесовский классификатор, KNN, SVM и Random Forest. Лучшее качество по F-мере в этом исследовании продемонстрировал KNN для триграмм и с использованием «поведенческих признаков».

В работе [4] предложена следующая имплементация: эмбеддинги для текстов получаются с помощью предобученной модели BERT, после чего на полученных векторных представлениях обучаются классификаторы (SVM, Random Forest, Bagging classifier, K-NN, наивный байесовский классификатор).

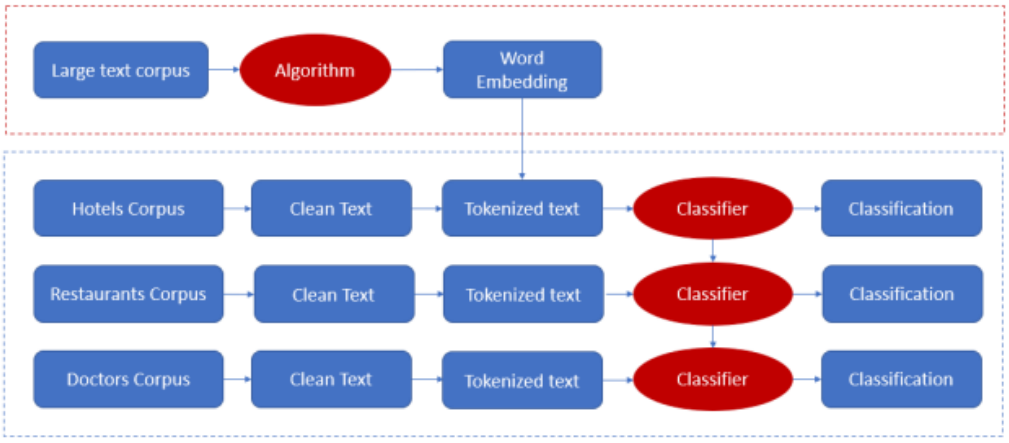
В публикации [5] используются датасеты OpSpam и Yelp. Авторы используют следующие признаки:

- «структурные признаки» для текстов (длина отзыва, средняя длина слова и предложения, доля заглавных букв и чисел);
- фичи, характеризующие пользователя, который оставил отзыв (максимальное число отзывов за день, средняя длина отзыва, стандартное отклонение оценок и доля положительных/отрицательных отзывов);
- POS-tags as percentages (видимо, доля тегов каждой части речи в отзыве);
- Доля положительно и отрицательно окрашенных слов.

Авторы установили, что часть признаков (доли POS-тегов и «сентиментные» доли) не являются значимыми (с помощью логистической регрессии).

Для получения эмбеддингов текстов отзывов авторы используют в разных экспериментах bag-of-words, TF-IDF и предобученный word2vec. Для классификации используются два типа моделей: нейросетевые и модели из классического ML. В качестве моделей классического ML авторы использовали SVM и логистическую регрессию, при этом в обоих случаях тексты были векторизованы с помощью TF-IDF. При применении нейросетевых моделей тексты векторизуются с использованием W2V и BOW (остальные фичи просто конкатенируются к эмбеддингам). В качестве нейросетевых моделей были использованы FFNN, CNN и LSTM. Последним описанным подходом решения задачи в данной статье является fine-tuning предобученной модели BERT, данный подход продемонстрировал в исследовании лучшее качество.

Статья [6] придерживается в целом той же методологии и фокусируется на исследовании влияния способа векторизации текстов отзывов на итоговый результат. Общая схема выглядит следующим образом:



После удаления дубликатов в датасете для подготовки текстовых данных используются более-менее стандартные шаги: авторы исправляют ошибки в словах (с помощью инструмента TextBlob), удаляются HTML-теги, числа и пунктуация, производится приведение к нижнему регистру и лемматизация, далее удаляются стоп-слова и слова длины менее 3-х символов. После этого для получения эмбеддингов текстов используются такие методы как BOW, TF-IDF, GloVe, W2V и BERT. После получения векторных представлений текстов были обучены модели машинного обучения (логистическая регрессия, SVM, KNN, Decision Tree, Random Forest, градиентный бустинг (XGBoost)). Кроме того, для каждой модели производился подбор гиперпараметров с помощью grid search. Наилучший результат в данной работе был продемонстрирован при использовании BOW с логистической регрессией.

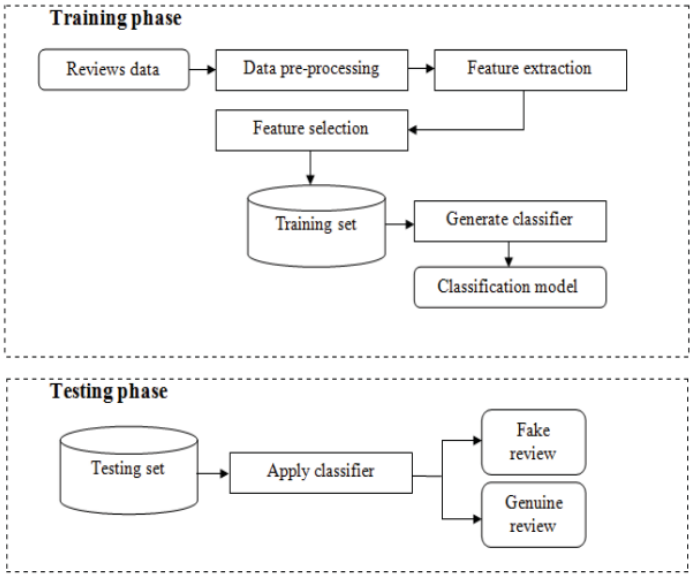
Работа [7] представляет собой общий обзор проблематики детекции фейковых отзывов. В работе утверждается, что глобально существует 3 подхода к решению задачи:

- Базирующийся на отзывах. Этот подход предполагает использование фичей, извлекаемых из текста отзыва;
- Базирующийся на пользователях, оставляющих отзывы. Данный метод основан на анализе поведения пользователей, здесь используются признаки, связанные непосредственно с пользователями (возраст, общее количество написанных пользователем отзывов, средний рейтинг отзыва и т.п.);
- Базирующийся на продукте. Для этого метода важны признаки, характеризующие продукт (цена продукта, рейтинг и т.п.).

Также в данной работе описывается общий пайплайн решения задачи определения фейковых новостей с использованием методов машинного обучения, упомянуты следующие этапы:

- Сбор данных;
- Предобработка данных;
- Извлечение и отбор признаков;
- Обучение модели классификации (здесь в качестве основных моделей авторы упоминают решающие деревья, случайный лес, SVM, KNN и логистическую регрессию).

Схематично это можно представить следующим образом (схема взята из текста работы):



В следующей статье (судя по всему, это просто статья на сайте, а не опубликованная работа, поэтому я ее обособил) авторы используют GPT, чтобы сгенерировать фейковые отзывы, а реальные (написанные людьми) отзывы парсят из интернета. Обработка текстов (стемминг, приведение к нижнему регистру и т.п.) не выполняется, производится только удаление дубликатов и нереалистичных отзывов. Далее обучают ML модели (SVM, XGBoost, Random Forest и ряд других), а также производится fine-tuning BERT. Лучший результат достигнут благодаря дообучению BERT.

Выводы и дальнейшие планы

Как можно понять из проанализированной научной литературы, подход к решению задачи определения фейковых отзывов в большинстве задач почти идентичен, в основном различия есть только в используемых признаках (что, конечно, в том числе зависит от того, какие данные удалось собрать или какие данные были изначально в датасете, если используется готовый набор данных), в способе получения эмбеддингов из текстов, а также в используемых моделях для классификации реальных и фейковых отзывов.

На основании проведенного анализа релевантной литературы, я принял решение, что пайплайн моего решения будет устроен следующим образом:

- Feature Engineering: проведение EDA, удаление дубликатов, генерация новых фичей препроцессинг текстов;
- Векторизация текстов: здесь, как это следует из обзора литературы, довольно много вариантов (BOW, TF-IDF, W2V, трансформеры)
- Обучение модели для классификации: опять же, как следует из обзора литературы, есть несколько вариантов выбора классификатора.

Возможно, имеет смысл попробовать в качестве слабого бейзлайна BOW и обучить, например, на полученных из BOW данных для текстов отзывов и имеющихся у нас числовых признаках (f1-f8) что-то несложное (например, логистическую регрессию). Далее в качестве бейзлайна посильнее (конечно, если качество окажется выше, чем у слабого бейзлайна) для получения эмбеддингов текстов можно использовать W2V (предобученный или обучить самостоятельно, но кажется, что для обучения W2V с нуля корпус не очень объемный) и взять здесь в качестве классификатора что-то посильнее (например, бустинг (CatBoost или LightGBM), хотя кажется, что для бустинга выборка из 3000 объектов может быть недостаточно), а далее пытаться улучшить качество: смотреть, какие признаки каким образом влияют на качество и насколько признаки значимы с точки зрения обученной модели (feature importance), пробовать взвешенный W2V с весами TF-IDF или предобученные трансформерные модели (BERT или Sentence-BERT) для получения эмбеддингов текстов.

Список литературы

1. Chowdhary N. S., Pandit A. A. Fake review detection using classification //Int. J. Comput. Appl. – 2018. – Т. 180. – №. 50. – С. 16-21.
   URL: https://typeset.io/pdf/fake-review-detection-using-classification-ee1qy1ry2j.pdf
2. Etaiwi W., Naymat G. The impact of applying different preprocessing steps on review spam detection //Procedia computer science. – 2017. – Т. 113. – С. 273-279.
   URL: https://www.sciencedirect.com/science/article/pii/S1877050917317787
3. Elmogy A. M. et al. Fake reviews detection using supervised machine learning //International Journal of Advanced Computer Science and Applications. – 2021. – Т. 12. – №. 1.
   URL:https://www.researchgate.net/publication/348962950_Fake_Reviews_Detection_using_Supervised_Machine_Learning
4. Mir A. Q., Khan F. Y., Chishti M. A. Online Fake Review Detection Using Supervised Machine Learning And BERT Model //arXiv preprint arXiv:2301.03225. – 2023.
   URL: https://arxiv.org/ftp/arxiv/papers/2301/2301.03225.pdf
5. Kennedy S. et al. Fact or factitious? Contextualized opinion spam detection //arXiv preprint arXiv:2010.15296. – 2020.
   URL: https://aclanthology.org/P19-2048.pdf
6. Macean D. Predictive model for detecting fake reviews: Exploring the possible enhancements of using word embeddings: дис. – 2023.
   URL: https://run.unl.pt/bitstream/10362/152177/1/TCDMAA2936.pdf

7. Patel N. A., Patel R. A survey on fake review detection using machine learning techniques //2018 4th International Conference on Computing Communication and Automation (ICCCA). – IEEE, 2018. – С. 1-6.
URL: https://www.sci-hub.ru/10.1109/CCAA.2018.8777594