Технічні науки

UDC 004.051

Sereda Daria

Student of the

Kharkiv National University of Radio Electronics

Golian Nataliia

Scientific Director, Associate Professor of the

Department of Software Engineering

Kharkiv National University of Radio Electronics

INVESTIGATION OF THE NATURAL LANGUAGE PROCESSING MODELS FOR TEXT-BASED RECOMMENDATION SYSTEMS

Summary. *This study investigates the application of natural language processing (NLP) methods in recommendation systems based on textual data. It identifies models with the highest utility coefficients for specific tasks based on criteria crucial for recommendation provision. The study underscores the importance and effectiveness of employing NLP methods in recommendation systems, utilizing the findings to develop a recommendation system.*

Key words: *recommendation systems, NLP models, natural language processing, text, BERT, fastText, movies, sentiment analysis.*

In today's digital landscape, users are inundated with a vast array of online content and platforms, making effective navigation essential. Recommender systems have become indispensable tools, guiding users through this abundance by offering personalized experiences. Leveraging advanced algorithms and machine learning, these systems analyze user behavior and preferences to deliver tailored recommendations that anticipate evolving needs.

There are two main types of recommender systems – content-based filtering and collaborative filtering. In content-based filtering systems, objects are determined based on their features. Based on the available attributes in the objects that the user rated, the interest profile of the new user is studied, in other words, this is a keyword-based recommendation system. Thus, content-based recommendation systems use algorithms that offer users similar items they liked in the past or are currently studying [1].

The integration of Natural Language Processing (NLP) techniques marks a significant shift in recommendation systems. NLP equips these systems to understand and extract insights from textual data, such as user reviews and product descriptions. By harnessing NLP, recommender systems can better comprehend user preferences and deliver more contextually relevant recommendations.

Transformer models, such as BERT and GPT, have demonstrated remarkable performance in natural language understanding tasks, making them well-suited for tasks like sentiment analysis and semantic understanding in recommender systems. Similarly, word embedding models, like fastText, offer efficient representations of words in vector space, facilitating meaningful comparisons and associations between words. By leveraging these sophisticated techniques, recommender systems can better comprehend user preferences and deliver more accurate and personalized recommendations. Therefore, exploring these advanced models is essential for enhancing the effectiveness and adaptability of recommendation algorithms in diverse application domains so this investigation is focused on relatively new transformer models and word embedding model.

The following metrics were chosen to evaluate the models.

Recommendation precision is a measure of how often the model correctly classifies positive predictions. It increases as the ratio of the number of correct positive predictions to the total number of positive predictions. Mathematically, it is defined as (1).

$$P = \frac{Tp}{Tp + Fp}, \tag{1}$$

where P is precision,

Tp is the number of correctly classified instances as positive,

Fp is the number of incorrectly classified instances as positive.

Recall quantifies is the ability of the model to capture all relevant instances of the positive prediction. It is calculated using the formula (2).

$$R = \frac{Tp}{Tp + Fn}, \tag{2}$$

where R is recall,

Tp is the number of correct positive forecasts,

Fn is the number of false negative predictions

The F-score is a combined measure of precision P and a Recall R. It is calculated using the harmonic mean between P and R (3).

$$Fscore = \frac{2 \cdot P \cdot R}{P + R}, \tag{3}$$

where P – precision value,

R – recall value.

To facilitate the investigation effectively, two main goals have been defined:

1. Sentiment analysis. Sentiment analysis enables a deeper understanding of the emotional context embedded within user-generated text, facilitating the creation of recommendations that resonate on a more personal level. By discerning the user's emotional state, recommendations can be customized to better suit their current needs, mood, and preferences.

2. Formulating recommendations based on users' preferences enhances the effectiveness of recommendation systems across various domains. By analyzing user-generated content or interactions and providing suggestions tailored to individual tastes, these systems offer more accurate and satisfying recommendations, thereby improving overall user satisfaction and engagement with the platform or service.

Within the scope of this study, the tasks above are formulated as follows:

1. The sentiment analysis within user-generated text;
2. Providing movie genre recommendations based on user-preferred movie attributes.

For the first task, a dataset containing users' tweets and labels for emotional states was utilized: sadness (0), joy (1), love (2), anger (3), fear (4), surprise (5) [2]. The training set comprises 15,969 unique values. 4666 tweets represent sadness, 5362 tweets – joy, 1304 – love, 2159 – anger, 1937 – fear and 572 represent surprise.

For the second task, the dataset was created with the use of The Movie Database (TMDb) API [3]. This dataset contains information about movies fetched from endpoints The resulting dataset consists of 20,000 rows and includes the following columns: title – the title of the movie; description – a brief overview or synopsis of the movie; genres – a list of genres associated with the movie: [genre 1, genre 2, … genre n]. There are a total of 18 genres listed, with varying counts of movies for each genre.

Experiments were conducted on GPT-2, BERT, XLNet and FastText models. The process includes data pre-processing, model training, model testing and evaluation. The Python programming language was used to conduct experiments and evaluate models.

To draw conclusions regarding the use of models for text-related tasks in recommendation systems, two tables comparing the results were created. As metrics for comparison, F-score and training time were chosen. Table 1 represents results for sentiment analysis problem and table 2 – genre recommendations. As metrics for comparison, F-score and training time were chosen.

Table 1

Comparison for sentiment analysis

Model	F-Score value	Execution time (h.)
BERT	0.858	12
GPT	0.738	17
XLNet	0.82	13,5
FastText	0,85	4,5

While BERT offers competitive performance in sentiment analysis, FastText outperforms models in terms of both accuracy and efficiency. Its ability to achieve a high F-Score with minimal computational resources makes FastText an attractive choice for sentiment analysis tasks, particularly in scenarios where timely processing and resource constraints are paramount.

Table 2

Comparison for genre recommendations

Model	F-Score value	Execution time (h.)
BERT	0.82	11,5
GPT	0.81	18
XLNet	0.849	16
FastText	0,66	6

In order to find the model that best fits, we will use a linear additive convolution with weighting coefficients to calculate the utility coefficient, as certain criteria have a greater impact on effectiveness and model selection. The convolution formula is given in (4).

$$Z^* = \max_{i=1,m} \sum_{j=1}^{n} \alpha_j \beta_j \alpha_{i_j} \tag{4}$$

where α_j – normalization factors,

β_j – weighting coefficients.

Some criteria contribute more and are more important than others, so it is necessary to introduce weighting coefficients. Using the ranking method, we will evaluate the importance of the criteria in the following order: the most important criterion is F-Score, its rank we set up as 5. Execution time on the other hand is also important, so its rank is 2. Rank sum = 8. The utility calculation of models using linear additive convolution is illustrated in figure 1.

	F-score	Time saving (h)	K
BERT	0,82	6,5	**0,275337522**
GPT	0,81	0	0,161277477
XLNet	0,849	2	0,203525447
FastText	0,66	6	0,234859553
βj	0,625	0,25	
αj	0,318572794	0,068965517	

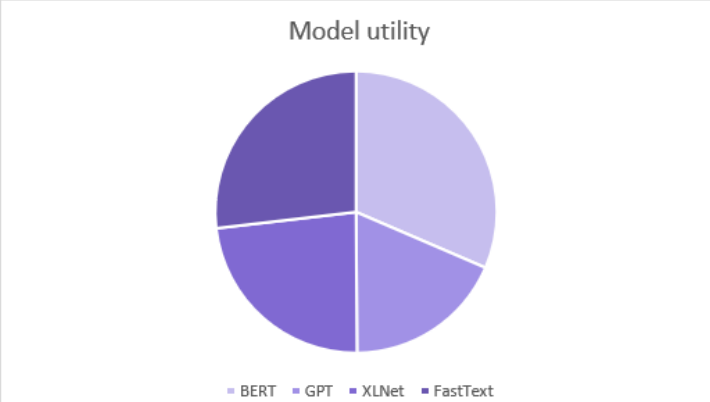

Fig. 1. Model's utility for genre recommendations

Based on the utility scores calculated using the linear additive convolution, the BERT model appears to be the most suitable choice for recommendation solutions among the models evaluated.

In conclusion, the integration of BERT and FastText offers a comprehensive approach to recommendation systems, enabling them to deliver personalized and emotionally resonant suggestions to users, thereby enhancing the overall user experience and satisfaction. Therefore, the results of the conducted research can be used to develop recommendations systems, specifically a recommendation system for providing movie recommendations based on the user's post, taking into account emotions and post's context.

References

1. Aggarwal C. C. Recommender Systems: The Textbook. 2016. 519 p. (1st Edition).
2. Riloff E., Tsujii J., Chiang D., Hockenmaier J. CARER: Contextualized Affect Representations for Emotion Recognition. Association for Computational Linguistics. 2018. URL: https://aclanthology.org/D18-1404 (date of access: 05.03.2024).
3. TMDb API. URL: https://developer.themoviedb.org/docs/getting-started (date of access: 06.03.2024).