**Multi-label Extraction of Movie Genres through Synopses**

**Joe** BU (2019312570)

Department of History,

SungKyunKwan University

**Shin GC** (2019310558)

Department of Classical Chinese Literature,

SungKyunKwan University

**ABSTRACT**

In this paper, we introduce an approach for classifying movie genres based on movie synopses. Unlike previous studies that utilized open-source data platforms like Kaggle, we extracted and used movie genres and synopses in Korean directly. Our model goes beyond the traditional binary classification of a single genre, instead adopting a multi-class classification method. After the data crawling process, we identified a total of 20 unique genres. Notably, our model accounted for all 20 genres without any omission. As a result, our research outperformed other movie genre classification models, achieving an impressive F1-score of 0.90.

**INTRODUCTION**

In this research paper, we delve into the development and implementation of an advanced multi-label classifier. This classifier, designed to take a movie synopsis as input and predict the movie's genres as output, represents a significant evolution from previous attempts at genre categorization based on data from IMDB and TMDB.

Past research has certainly made strides in this area, yet it has also revealed distinct limitations. The most significant of these is the oversimplification of classifying a movie under a single genre, failing to capture the multifaceted nature of many films. Moreover, these models have often suffered from a lack of accuracy, rendering them impractical for widespread adaptation and utility.

The film industry currently faces a significant challenge due to the manual, inefficient, and potentially inaccurate process of classifying movie genres based on their synopses. This traditional method typically involves experts who view the entire film and then subjectively determine its genre categorization based on various elements such as the script, characters, and plot. Despite being a standard practice, it often leads to inconsistencies and potential inaccuracies in genre classification due to individual interpretational variances.

Existing scholarly research predominantly focuses on single-label classification, overlooking the common industry practice of categorizing a single movie into multiple genres to reflect its multifaceted nature. Recognizing this gap, our research proposes

a more objective, efficient, and nuanced approach, a multi-label classification model. This model aims to streamline the genre classification process while more accurately reflecting the multi-dimensional nature of movie genres, thus marking a considerable advancement in the field of movie genre classification.

텍스트, 스크린샷, 폰트, 디자인이(가) 표시된 사진

자동 생성된 설명

**Fig 1. Schematic diagram of multilabel movie genre classification model.**

Our proposed approach entails several stages. Initially, we extract data that includes movie titles and genres from a database file on the KOBIS website. Concurrently, we also scrape synopses data from NAVER web-site through web crawling. Following the data collection, we vectorize the data using the AutoTokenizer from the BERT model of Klue/bert-base. Subsequently, we import the pre-trained BERT model and conduct fine-tuning by adjusting the final output layer to align with our project's objectives. Ultimately, we evaluate the model's precision and accuracy utilizing the F1 score and classification report. This comprehensive method ensures that our multi-label classification model is both efficient and reliable.

**RELATED WORKS**

[1] *Hasan, Md. Mehedi, Dip, Sadia Tamim, Rahman, Tasmiah (2021)*

In this study, they vectorized text data using BOW and TF-IDF methods and selected features through chi-square tests. In the modeling part, it combined SVM and Decision Tree, both of which construct a confusion matrix in the training set. When testing, if the y-values given by the two models differ, they adopted a method of choosing the y with a lower misclassification probability (the probability of incorrect classification) from SVM and DT. While this study achieved high performance, it only used seven genres, which is a limitation compared to our project that classifies 20 genres.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Algorithm | Feature Extraction | Precision | Recall | F1-score |
| BOW | KNN | 0.86 | 0.83 | 0.83 |
| SVM | 0.84 | 0.80 | 0.83 |
| DT | 0.80 | 0.79 | 0.81 |
| POHC | 0.92 | 0.90 | 0.91 |
| TF-IDF | KNN | 0.90 | 0.87 | 0.89 |
| SVM | 0.88 | 0.83 | 0.87 |
| DT | 0.85 | 0.83 | 0.84 |
| POHC | 0.97 | 0.94 | 0.96 |

**Tabel 1. Performance comparison using BOW and TF-IDF feature vector.**

[2]

This study employed various vectorization methods, among which the TF-IDF (Term Frequency-Inverse Document Frequency) approach yielded the highest performance, with an F1-score of 54.8. The methodology combined multiple models, including the Decision Tree and Random Forest, with different vectorization techniques. The results indicated significant variations depending on the chosen model and vectorization method. This led to the conclusion that it is crucial to experiment with diverse combinations in our team project.

텍스트, 폰트, 번호, 문서이(가) 표시된 사진

자동 생성된 설명

**Figure 2. Average F1-score results for each experiment.**

**DATA**

KOBIS, also known as the Korea Box-office Information System, is a robust database that provides detailed information on nearly 98,000 movies. This extensive database offers a wealth of information, including the movie title, the year of production, the genre, and the type. However, it falls short in one crucial area - it does not provide any synopsis data.

Recognizing this gap, we turned to another source for this essential piece of information. We decided to extract the synopsis directly from the movie service operated by the popular South Korean online platform, NAVER. While our initial intention was to extract data for all 98,000 movies, we quickly realized that this would not be feasible due to time constraints.

Therefore, we had to establish and apply some conditions to make the data extraction process more manageable. By doing so, we were able to successfully extract synopsis data for approximately 32,000 movies. The conditions we set for this extraction process are as follows:

**1. Removal of adult genres**: Excluded erotic films, not just films rated for adults only.

**2. Year setting**: Initially, we only extracted from 2010 to 2023. However, upon checking the genre labels frequency, we found that the number of movies in genres such as historical dramas and musicals was less than 100. Therefore, we additionally crawled synopsis data for movies from 2000-2009.

**3. Type of movie**: There are feature films, shorts, omnibus, etc. We decided to extract synopses only from feature films that are commonly seen in regular theaters.

스크린샷, 텍스트, 라인, 도표이(가) 표시된 사진

자동 생성된 설명

**Figure 3. Genre labels frequency in our dataset.**

In our research, we made a conscious decision to not merge any genres. Our goal was to preserve the uniqueness of the original data as much as possible. However, we encountered a setback since some movies were not registered on the NAVER movie service. This resulted in a significant number of movies from which we could not extract synopsis data. To ensure the integrity and accuracy of our data, we took several steps. First, we deleted movies for which we were unable to obtain synopsis data. We then proceeded to remove duplicate entries from our dataset. In addition to these steps, we also made the decision to delete all data rows that were characterized by the 'other' genre.

By adhering to these stringent criteria, we were able to secure a robust and reliable dataset. The final dataset, after all the filtering and pruning, comprised approximately 20,000 entries. This dataset forms the foundation of our research and plays a crucial role in the development and testing of our multi-label classification model.

**PREPROCESSING**

The BERT model, a critical component of our research, is renowned for its unique approach to understanding the holistic context of a sentence. Its advanced design allows it to consider all words within a sentence simultaneously. This concurrent processing facilitates a comprehensive understanding of the sentence, even allowing stop words and grammatical information to contribute significantly to the overall meaning comprehension.

This unique feature of the BERT model negates the need for conventional text processing techniques such as part-of-speech (POS) tagging or stop-word removal. These techniques, although valuable in certain contexts, are not necessary when using the BERT model due to its inherent capacity to utilize original raw text effectively. Considering these unique characteristics, we tailored our text processing techniques to align with the BERT model's capabilities. We limited our text processing to padding and truncation, techniques that are essential to ensuring efficient processing for our tasks.

This minimalist approach to text processing allows us to preserve the integrity of the original text as much as possible. It's crucial to note that these steps help maintain the original sentence structure, thereby providing a more accurate context for the BERT model to understand and analyze.

In essence, by aligning our text processing techniques with the unique capabilities of the BERT model, we aim to maximize the accuracy and efficiency of our multi-label classification model while ensuring the preservation of the original text's integrity.

***BERT for Text Classification -* [3]**

BERT-base model contains an encoder with 12 Transformer blocks, 12 self-attention heads, and the hidden size of 768. BERT takes an input of a sequence of no more than 512 tokens and outputs the representation of the sequence. The sequence has one or two segments that the first token of the sequence is always [CLS] which contains the special classification embedding and another special token [SEP] is used for separating segments. For text classification tasks, BERT takes the final hidden state h of the first token [CLS] as the representation of the whole sequence. A simple softmax classifier is added to the top of BERT to predict the probability of label c:

p(c | **h**) = softmax(W**h**),

where W is the task-specific parameter matrix. We fine-tune all the parameters from BERT as well as W jointly by maximizing the log-probability of the correct label.

텍스트, 스크린샷, 번호, 도표이(가) 표시된 사진

자동 생성된 설명

**Figure 4. Example of trained BERT for text classification – [4]**

**VECTORIZING**

In traditional machine learning applications, different data vectorization methods such as Word2Vec and TF-IDF have been employed. Among these, TF-IDF has demonstrated superior performance, with a single-label accuracy of 60 compared to 40. However, when conducting multi-label classification based on logistic regression, Word2Vec displayed significantly lower results. This outcome is speculated to be due to the model being trained on English data. The need for a pre-trained model utilizing deep learning was recognized, but to leverage such a model, the tokenizer used in that model should also be employed. Therefore, a model trained on Korean was chosen and its tokenizer was utilized. In the Transformer-based BERT model, the tokenizer decomposes the text into subunits and then uses a pre-trained vocabulary to convert each subword into vector representations. The AutoTokenizer used in the Klue/bert-base's BERT model was adopted for this process.

**MODEL SELECTION**

In our research, we strategically chose to utilize a pre-trained model. This decision not only saved us significant time but also provided us with a robust foundation. However, to ensure the model's accuracy, we knew we had to make some critical adjustments. One such adjustment was made to the hidden layer of the final output layer of the pre-trained model. We carefully modified its structure to correspond to the number of labels in our dataset, which was 20. This meticulous modification was vital to ensure the model's output would align accurately with our multi-label classification task.

When it came to selecting a loss function, we chose BCEWithLogitLoss. This choice was not made lightly. BCEWithLogitLoss, also known as Binary Cross-Entropy with Logits Loss, is a loss function that is particularly well-suited for multi-label classification tasks. It combines a Sigmoid layer and the BCELoss in one single class. This means that this loss function is useful when training a classification problem with C classes. If provided, the optional argument weight should be a 1D Tensor assigning weight to each of the classes. This is particularly useful when you have an unbalanced training set. By employing BCEWithLogitLoss, we aimed to enhance the effectiveness and reliability of our model in accurately classifying movie genres. This loss function is widely recognized for its exceptional performance in tasks involving multi-label classification and was thus an ideal choice for our model. In conclusion, through careful adjustments to the pre-trained model and the strategic selection of the loss function, we aimed to create a robust and reliable multi-label classification model that could accurately reflect the multifaceted nature of movie genres. In the context of the BERT model, the elements of ids, mask, and token type ids play a crucial role in forming the model's input. The 'ids' are responsible for the conversion of tokenized data into numeric form, the 'mask' is used to disregard padding in the text that doesn't meet a fixed length, and 'token type ids' help distinguish the sentence to which each token belongs. These components ensure that the model processes the input text data accurately and efficiently, thereby contributing to the overall performance of the model in multi-label classification tasks.

**TUNING**

We undertook a systematic process of hyperparameter optimization, specifically focusing on the learning rate, the number of epochs, and the batch size. This was achieved through the implementation of a grid search methodology, a well-established technique for tuning hyperparameters. By systematically working through multiple combinations of these parameters, we were able to identify the optimal configuration that led to the most effective learning and best overall performance of our model. This rigorous and thorough approach to hyperparameter optimization ensured the robustness of our model and contributed significantly to our ability to accurately classify movie genres.

**텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명**

**Figure 5. The Best combination of parameters**

**EVALUATION**

The results of the multi-label classification are presented as a single vector, which is a combination of 20 binary-classified labels. To assess the performance, we use accuracy to determine whether the predicted vector matches the actual vector. To identify which labels have lower prediction accuracy, and to simultaneously evaluate sensitivity and precision, we also employ the F1 score and a classification report.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Genre | Precision | Recall | F1-score | support |
| SF | 0.93 | 0.94 | 0.93 | 97 |
| Family | 0.93 | 0.89 | 0.93 | 55 |
| Show | 0.91 | 0.95 | 0.93 | 22 |
| Horror | 0.92 | 0.98 | 0.95 | 191 |
| Thriller | 0.98 | 0.97 | 0.97 | 330 |
| Drama | 0.92 | 0.94 | 0.93 | 683 |
| Melo | 0.92 | 0.92 | 0.92 | 207 |
| Documentary | 0.92 | 0.92 | 0.92 | 207 |
| Musical | 0.92 | 0.79 | 0.85 | 14 |
| Mystery | 0.80 | 0.88 | 0.84 | 90 |
| Animation | 0.94 | 0.81 | 0.87 | 145 |
| history | 1.00 | 0.79 | 0.88 | 14 |
| Western | 1.00 | 1.00 | 1.00 | 6 |
| Comedy | 0.90 | 0.93 | 0.92 | 338 |
| Crime | 0.92 | 0.93 | 0.93 | 169 |
| Action | 0.94 | 0.94 | 0.94 | 425 |
| Adventure | 0.87 | 0.87 | 0.87 | 128 |
| War | 0.89 | 0.93 | 0.91 | 42 |
| Romance | 0.92 | 0.88 | 0.90 | 232 |
| Fantasy | 0.92 | 0.85 | 0.88 | 104 |

**Table 2. Averaged classification results for the overall best experiment.**

**CONCLUSION**

In the process of selecting our final model, we conducted numerous trials and analyses. We managed to enhance the performance of a logistic regression-based classifier to a single classification criterion of 0.60. We also experimented with various vectorization methods, such as the Tf-idf method and Word2Vec. Despite the preprocessing steps we took, such as stop-word removal and POS(part-of-speech) tagging to eliminate postpositions and endings, these models still fell short in performance compared to our BERT model. Our model's accuracy criterion is stringent, only deeming a prediction correct if all 20 genres are accurately identified. Hence, an accuracy of 0.85 can be considered highly accurate in this context, further solidifying the effectiveness of our chosen model.

**REFERENCES**

[1] Hasan, Md. Mehedi, Dip, Sadia Tamim, Rahman, Tasmiah (2021). Multilabel Movie Genre Classification from Movie Subtitle: Parameter Optimized Hybrid Classifier

[2] Giuseppe Portolese, Valeria Delisandra Feltrim (2018). On the Use of Synopsis-based Features for Film Genre Classification

[3] Chi Sun, Xipeng Qiu(B), Yige Xu, and Xuanjing Huang (2019). How to Fine-Tune BERT for Text Classification?

[4] Usuga-Cadavid, Juan Pablo1,2 (2022). Using deep learning to value free-form text data for predictive maintenance