Case Classification System Based on Taiwanese Civil Summary Court Cases

Ming-Yi Chen[1](✉), Jia-Wei Chang[1], Hsiao-Chin Lo[2], and Ying-Hung Pu[3]

[1] Department of Computer Science and Information Engineering, National Taichung University of Science and Technology, Taichung, Taiwan 404348
{s1811132015,jwchang}@nutc.edu.tw

[2] Department of Japanese Studies, National Taichung University of Science and Technology, Taichung, Taiwan 404348
hclo@nutc.edu.tw

[3] Department of College of Languages, National Taichung University of Science and Technology, Taichung, Taiwan 404348
yhpu@nutc.edu.tw

Abstract. This experiment classifies cases based on the top 20 most frequently used categories of case reasons found in civil summary court judgments provided by the Judicial Yuan of Taiwan from 2012 to 2022. We built case classifiers using two methods: machine learning with TF-IDF+SVM and deep learning with BERT. We then compared the results of both classifiers. In the classification results using TF-IDF+SVM, an accuracy of 89.3% was achieved, while with BERT, an accuracy of 93.825% was achieved.

Keywords: Natural Language Processing (NLP) · Text Analysis · Machine Learning · Deep Learning · Case Classification

1 Introduction

Natural Language Processing (NLP) is a machine learning technique that allows computers to interpret and understand human language by translating and comprehending concepts similar to humans, using binary code (0 s and 1 s) as a medium. The difference between Chinese and English lies in the fact that Chinese sentences usually do not have any punctuation marks to separate words [1]. When machines process and understand texts, they often start by segmenting long strings of sentences into individual words or tokens, a process known as word segmentation or tokenization. This allows for the extraction of textual features through the process of word segmentation. In many Chinese NLP applications, such as machine translation, text summarization, and others. Chinese word segmentation is often a necessary preprocessing step. This process involves dividing Chinese sentences into individual words or tokens to facilitate further analysis and processing in various NLP tasks. The two major challenges in Chinese word segmentation are ambiguity and unknown words. The issue of ambiguity arises when the same Chinese character sequence may have different word segmentation results in different

J. C. Hung et al. (Eds.): FC 2023, LNEE 1134, pp. 66–76, 2024.
https://doi.org/10.1007/978-981-99-9342-0_8

texts or contexts. Unknown words refer to words that are not included in the Chinese word segmentation dictionary, including names of people, places, organizations, legal terms, and their abbreviations [2]. Legal Artificial Intelligence (LegalAI) refers to the application of artificial intelligence methods to legal tasks, which helps improve the efficiency of legal professionals and provides assistance to individuals who may not have a strong knowledge of the law [3]. In the past few years, deep learning-based methods have achieved significant advancements in text classification tasks. BERT (Bidirectional Encoder Representations from Transformers) [4] is a revolutionary language model that obtains text representations by pre-training on large-scale unlabeled data. It has achieved remarkable performance on various NLP tasks. The introduction of the BERT model has brought new breakthroughs to text classification tasks. Its capabilities surpass traditional feature engineering-based methods, enabling the model to automatically learn key features from raw text. In this experiment, we used publicly available civil summary court judgments from the Taiwan Judicial Yuan for the years 2012 to 2022. We selected the top 20 most frequently occurring case categories as our data and compared the classification results between machine learning and deep learning methods.

2 Related Works

2.1 Legal Artificial Intelligence

Indeed, even before the widespread adoption of artificial intelligence technologies, there were studies that employed statistical methods to analyze legal cases [5, 6]. With advancements in technology, there has been significant research in recent years on applying artificial intelligence to the field of law. Some examples include studies on legal judgment prediction [7–9], reading comprehension [10], and case retrieval [11]. These efforts aim to leverage AI to enhance various aspects of the legal domain. Luo et al. [7] utilized three methods, FastText, TFIDF+SVM, and CNN, to train and test on over 2.6 million criminal cases published by the Supreme People’s Court of China, and comparing the results of the three approaches. Xiao et al. [8] proposed a neural network approach based on attention mechanism. They developed a unified framework for jointly modeling the tasks of determining appropriate charges and extracting relevant legal articles for a given criminal case. Zhong et al. [9] mentioned that legal judgments consist of multiple sub-tasks, including decisions of applicable law articles, charges, fines, and the term of penalty. These sub-tasks are considered as a directed acyclic graph (DAG) with dependencies among them. They proposed a spectrum-based multi-task learning framework called TOPJUDGE, which integrates multi-task learning and DAG dependencies into judgment prediction. Duan et al. [10] introduced a dataset for Chinese legal reading comprehension, comprising approximately 10,000 court documents and 50,000 expert-annotated questions with answers. They built two powerful baseline models based on BERT and BiDAF for this task. Shao et al. [11] utilized BERT to capture paragraph-level semantic relations and inferred the relevance between two cases by aggregating paragraph-level interactions.

2.2 Application of NLP in Taiwanese Court Judgments

The current research on the application of NLP in Taiwanese court judgments can be broadly categorized into three types: “Judgment Retrieval Systems”, “Case Classification or Clustering”, “Judgment Factor Analysis and Prediction of Judgment Outcomes”.

“Judgment Retrieval Systems” refer to the development or improvement of systems used for retrieving court judgments, aiming to enhance retrieval efficiency and the accuracy of search results. Hsieh [12] proposed using vocabulary combinations from factual paragraphs in judgment documents to improve retrieval results. In this experiment, they presented methods for extracting Chinese vocabulary from judgment documents, extracting important word phrases from factual paragraphs, and searching for similar cases based on these word phrases. Lin [13] established a factor table for civil judgments related to copyright law and proposed a method to extract relevant factors from judgment documents using regular expressions. This allowed for the analysis of the relationships among various factors.

“Case Classification or Clustering” refers to the process of categorizing and grouping court judgments based on different case types or legal issues using various approaches. The goal is to organize and classify the judgments into meaningful groups, allowing for easier retrieval, analysis, and understanding of the legal content within the judgments. Lia [14] developed a case-based inference system based on gambling and theft cases, combining the system with rules established by domain experts to improve classification performance. Additionally, they proposed a method for automatically annotating semantic information in factual paragraphs of judgment documents to extract the abstract structure of case facts. Ho [15] employed a hierarchical clustering method for grouping civil judgment digests. The study proposed a similarity measurement approach for civil judgment digests and compared the clustering effectiveness of various hierarchical clustering methods in civil judgments. Furthermore, they utilized a method that incorporates weighted legal keywords to enhance the clustering performance.

Research in “Judicial Factor Analysis and Judgment Outcome Prediction” involves extracting judgment factors from court rulings and using them to predict judgment outcomes or analyze the relationship between various judgment factors and outcomes. Huang [16] proposes a method for extracting sentencing and penalty factors from guilty verdicts in criminal cases related to trademark law. The study utilizes regular expressions to extract the paragraphs containing the factors from the judgments and clusters the keywords. By manually labeling the clustering results according to the sentencing standards prescribed by criminal law, specific types of cases can be obtained with their corresponding prosecution and sentencing factors. Chen [17] examines the correlation between the textual consistency of written orders on applications for release from pretrial detention in criminal proceedings and various key influencing factors such as judgment time, court, and the alleged criminal charges. The study also analyzes the relationship between these factors and the judgment outcomes.

3 Experimental

3.1 Dataset

The dataset used in this experiment consists of all the judgments from the civil summary court in Taiwan provided by the Judicial Yuan from 2012 to 2022. The dataset comprises a total of 1,179,705 records, with each judgment stored in JSON format.

Each judgment includes eight label contents: "JID" (file name), "JYEAR" (year), "JCASE" (court of judgment), "JNO" (judgment number), "JDATE" (judgment date), "JTITLE" (judgment case), "JFULL" (full text of the judgment), and "JPDF" (PDF download link for the judgment) (Fig. 1).

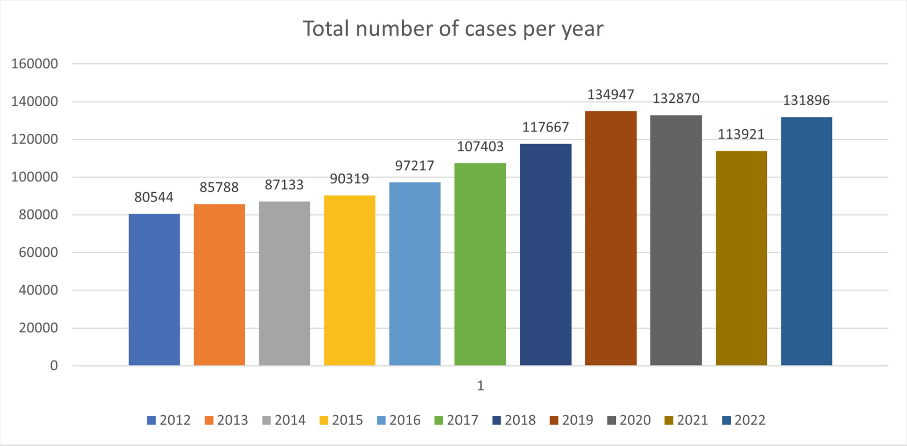


Fig. 1. Number of Cases from 2012 to 2022

First, the total number of judgment categories in the period from 2012 to 2022 was calculated. After the analysis, it was found that there were 9,613 different categories.

In this experiment, the top 20 categories with the highest number of cases were selected for case classification. The top 20 categories of cases are as follows: Damages compensation, Repayment of loans, Payment of credit card consumption expenses, Debt settlement, Damages compensation for tortious acts, Return of loans, Payment of bills, Return of credit card consumption expenses, Confirmation of non-existence of promissory note debt, Repayment of credit card consumption expenses, Transfer of property, Payment of credit card consumption expenses, etc., Payment of telecommunication fees, Payment of credit card debts, Repayment of credit card loans, Return of undue profits, Payment of management fees, Debt settlement for consumer purchases, Division of co-owned property, Lawsuit against debtor's objection. The judgment documents were filtered to include only the top 20 categories of cases, resulting in a total of 85,2168 documents. From each category, a random sample of 1000 documents was selected, resulting in a dataset of 20,000 documents in total (Fig. 2).

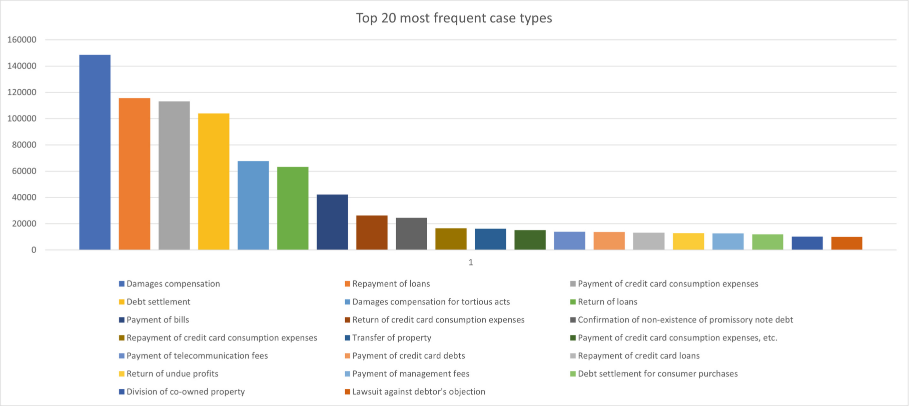


Fig. 2. Number of Top 20 Case Categories from 2012 to 2022

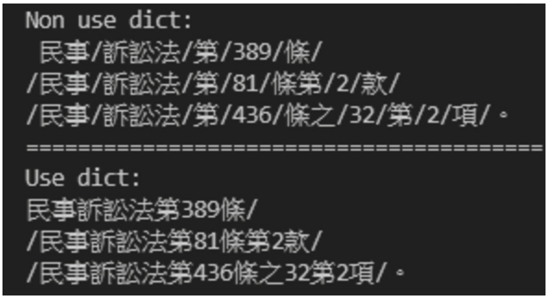
3.2 Data Preprocessing

The content of all the judgments was accessed through web scraping from the Taiwan Judicial Yuan's judgment query system. The system marks the words that appear in the Judicial Yuan's Legal Terminology Dictionary. Additionally, the system lists the legal provisions used in each judgment. In the experiment, web scraping was employed to retrieve the marked words and the referenced legal provisions. These words and legal provisions were then added to the dictionary to improve the accuracy of subsequent word segmentation experiments.

3.3 Word Segmentation

The word segmentation system used in this experiment is Jieba. It was chosen because it allows for the creation of custom word segmentation dictionaries, which helps to achieve more accurate word segmentation results according to our expectations. By adding customized word segmentation dictionaries, the accuracy and integrity of the word segmentation results can be improved. Below is a comparison of the word segmentation results without using a custom dictionary and the results after incorporating the custom dictionary (Fig. 3).

The comparison demonstrates that using a custom word segmentation dictionary improves the accuracy and alignment of the word segmentation results with the expected outcome.



```
Non use dict:
 民事/訴訟法/第/389/條/
/民事/訴訟法/第/81/條第/2/款/
/民事/訴訟法/第/436/條之/32/第/2/項/。
========================================
Use dict:
民事訴訟法第389條/
/民事訴訟法第81條第2款/
/民事訴訟法第436條之32第2項/。
```

Fig. 3. Comparison of Word Segmentation Results without Using a Custom Dictionary and with Using a Custom Dictionary

In this experiment, regular expressions [18] were used to extract paragraphs from the mention of plaintiffs and defendants in the judgment documents up to the end of the main text. Subsequently, Jieba was used for word segmentation.

3.4 Filtering Non-Chinese Words and Removing Stop Words

Stop words include the most frequently used words in daily life that have high occurrence but little meaningful content, such as “you”, “I”, “he”. In this experiment, not only common language terms but also frequently appearing words in judgment documents were added to the stop word list. These words do not have specific explanations in the Judicial Terminology Dictionary system.

After tokenization, the resulting tokens are further processed using regular expressions to remove non-Chinese characters (such as numbers and English words), followed by the removal of stop words. This preprocessing step helps to eliminate redundant words in the text, reduce computational resources, and improve the efficiency of training the model.

3.5 Term Frequency-Inverse Document Frequency

TF-IDF (Term Frequency-Inverse Document Frequency) is a statistical method for determining the importance of a word in a document. It utilizes two different parameters: term frequency (TF) and inverse document frequency (IDF).

Term frequency refers to the frequency of a term occurring in a document. If a term appears frequently within a document (high term frequency), it is assumed to be important for that particular document.

Inverse document frequency, on the other hand, measures the rarity of a term across the entire document collection. If a term is rare in other documents (high inverse document frequency), it suggests that the term is more significant for the document in question.

By combining term frequency and inverse document frequency, TF-IDF assigns a weight to each term, reflecting its importance within a specific document in the context of the entire document collection. This allows the identification of keywords that are indicative of the content and relevance of a particular document.

If a term appears frequently in a document but also appears frequently in other documents, it should not be considered a keyword for that document. The TF-IDF (Term Frequency-Inverse Document Frequency) algorithm uses two parameters: Term Frequency (TF) and Inverse Document Frequency (IDF), to calculate the importance of a term in a document. The calculation method of TF-IDF(k,f,A) is the multiplication of the term frequency and the inverse document frequency. The formula is shown as Eq. (1):

$$TF - IDF(k,f,A) = TF(k,f) * IDF(k,A) \tag{1}$$

The term frequency TF(k,f) represents the frequency of term k appearing in document f. Assuming the term "artificial intelligence" appears 20 times in a particular document, and the total number of words in that document is 320, the frequency of the term would be 20/320 = 0. 0625. The formula for TF(k,f) is given by Eq. (2):

$$TF(k,f) = \frac{F_f(i)}{max_{w \in f} F_f(w)} \tag{2}$$

The inverse document frequency IDF(k,A) represents the reciprocal of the proportion of documents in a collection that contain the term k. The IDF value decreases as the term appears more frequently across the documents, and vice versa. The formula for IDF(k,A) is given by Eq. (3):

$$IDF(k,A) = ln\left(\frac{|A|}{|\{a \in A : i \in k \in a\}|}\right) \tag{3}$$

Calculating term frequency TF(k,f) alone is insufficient to identify representative keywords for a document because the term may also appear frequently in other documents. Hence, the concept of inverse document frequency IDF(k,A) is introduced to provide a comprehensive evaluation. The combination of TF and IDF yields the final TF-IDF(k,f,A) result, which represents the weight and importance of the term within the document.

In this experiment, we compared the results using two classifiers: TF-IDF with Support Vector Machine (SVM) machine learning classifier and BERT deep learning classifier.

3.6 TF-IDF+SVM

By adding the processed text to the corpus and extracting the judgment categories from each judgment document as labels, we calculated the TF-IDF values for each word in the corpus. Next, the data was split into training and testing sets in a 7:3 ratio. The training set consisted of 14,000 instances, while the testing set had 6,000 instances. The distribution of judgment categories in the training and testing sets is shown in Figs. 4 and 5.

Next, a SVM classifier is constructed and trained on the training set. The trained model is then used to test the performance on the test set, resulting in an accuracy of 89. 3%.

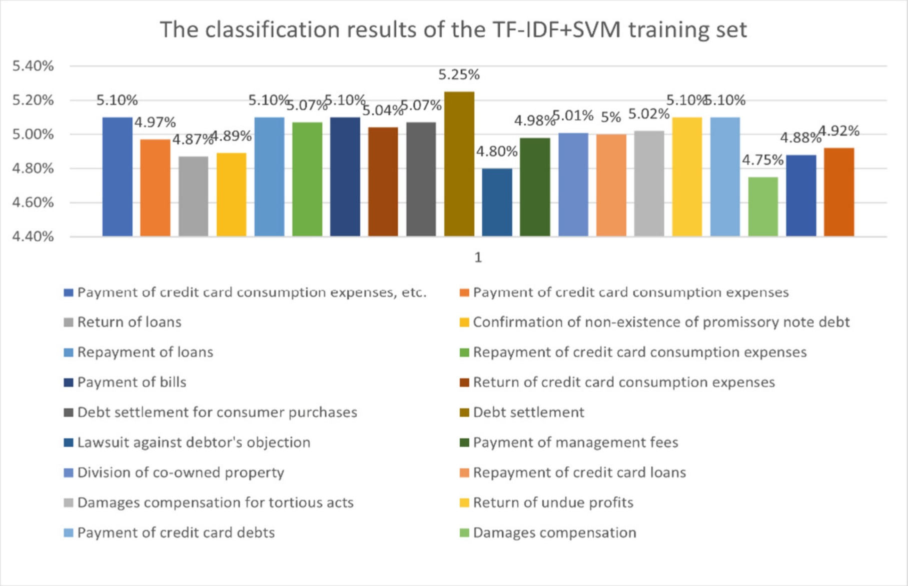


Fig. 4. Proportions of Each Case Category in the Training Set for TF-IDF+SVM

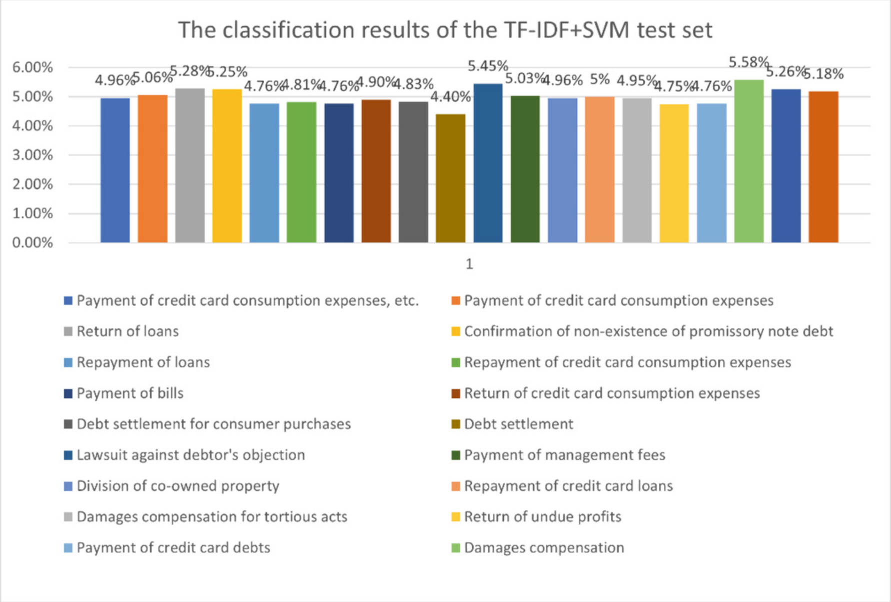


Fig. 5. Proportions of Each Case Category in the test Set for TF-IDF+SVM

3.7 BERT

In the BERT experiment, the pre-trained bert-base-chinese model was used. The dataset was split into training and test sets with a ratio of 8:2. The training set consisted of 16,000 samples, while the test set had 4,000 samples. The distribution of case categories in the training and test sets is shown in Figs. 6 and 7.

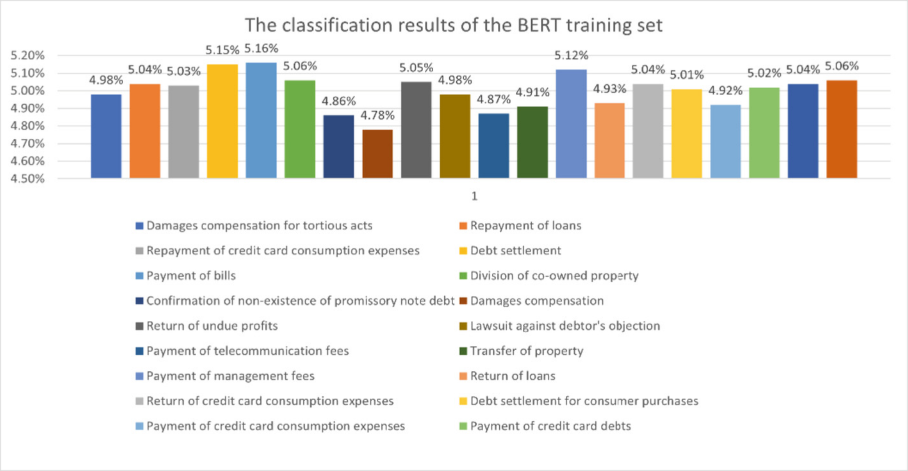


Fig. 6. Proportions of Each Case Category in the training Set for BERT

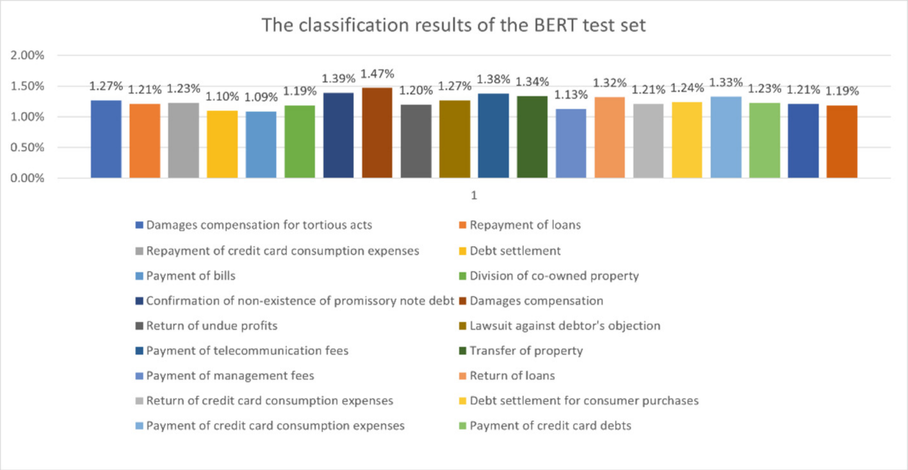


Fig. 7. Proportions of Each Case Category in the test Set for BERT

Next, the batch size was set to 32, and the maximum input length was set to 128. The training process was performed on an Intel(R) Core(TM) i7–10700 CPU@2.90 GHz. The AdamW optimizer was chosen with a learning rate of 1^{e-5}. After training, the model achieved a testing accuracy of 93.825% (Fig. 8).

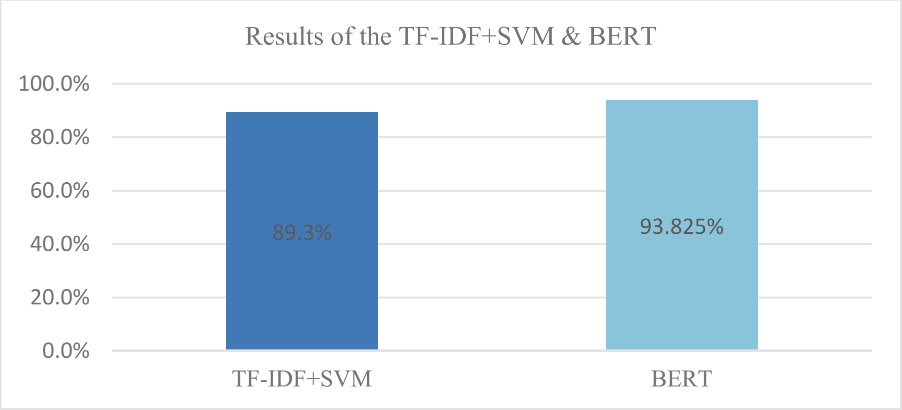


Fig. 8. Comparison of Results between BERT and TF-IDF+SVM

4 Conclusions and Future Directions

In this experiment, two approaches, machine learning and deep learning, were used for the classification of legal case categories based on court judgments. The results of both methods were compared to evaluate their performance.

Currently, only the top 20 most common case categories in civil summary courts are being used, with 1,000 judgment documents extracted for each category. In future experiments, the number of case categories will be expanded, and more judgment documents will be included to allow the model to learn the description patterns and relevant legal provisions associated with different case categories. This will enhance the accuracy of the classification results.

Acknowledgements. This work has received support from the funding provided by the National Science and Technology Council, Project No. 111-2410-H-025-018.

References

1. Ma, W.Y., Chen, K.J.: A bottom-up merging algorithm for Chinese unknown word extraction. In: Proceedings of the Second SIGHAN Workshop on Chinese Language Processing, pp. 31–38 (2003)
2. Lin, Q.X., et al.: A simple and effective closed test for Chinese word segmentation based on sequence labeling. Int. J. Comput. Linguist. Chinese Lang. Process. **15**(3–4) (2010)
3. Zhong, H., Xiao, C., Tu, C., Zhang, T., Liu, Z., Sun, M.: How does NLP benefit legal system: a summary of legal artificial intelligence. arXiv preprint arXiv:2004.12158 (2020). Zhang, D., Li, J., Shan, Z.: Implementation of Dlib deep learning face recognition technology. In: 2020 International Conference on Robots & Intelligent System (ICRIS), pp. 88–91. IEEE (2020)
4. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)

5. Kort, F.: Predicting Supreme Court decisions mathematically: a quantitative analysis of the "right to counsel" cases. Am. Polit. Sci. Rev. **51**(1), 1–12 (1957)
6. Segal, J.A.: Predicting Supreme Court cases probabilistically: the search and seizure cases, 1962–1981. Am. Polit. Sci. Rev. **78**(4), 891–900 (1984)
7. Luo, B., Feng, Y., Xu, J., Zhang, X., Zhao, D.: Learning to predict charges for criminal cases with legal basis. arXiv preprint arXiv:1707.09168 (2017)
8. Xiao, C., et al. Cail2018: a large-scale legal dataset for judgment prediction. arXiv preprint arXiv:1807.02478 (2018)
9. Zhong, H., Guo, Z., Tu, C., Xiao, C., Liu, Z., Sun, M.: Legal judgment prediction via topological learning. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 3540–3549 (2018)
10. Duan, X., et al.: CJRC: a reliable human-annotated benchmark dataset for Chinese judicial reading comprehension. In: Proceedings of the Chinese Computational Linguistics: 18th China National Conference (CCL 2019), Kunming, 18–20 October 2019, vol. 18, pp. 439–451. Springer, Cham (2019)
11. Shao, Y., et al.: BERT-PLI: modeling paragraph-level interactions for legal case retrieval. In: IJCAI, pp. 3501–3507 (2020)
12. Hsieh, C.D.: An exploration of indexing Chinese judicial documents with term (2005)
13. Lin, H.-L.: Implementation of text mining techniques in court decisions: focusing on compensation of copyright infringement (2013)
14. Lia, T.-M.: Classification and discourse analysis of gambling and larceny cases that infringe multiple criminal-law articles (2004)
15. Ho, J.H.: An application of hierarchical clustering of documents for civil judgments (2006)
16. Huang, Y.-T.: Study on the Prosecutorial Sentencing Factors by Text Mining – Focusing on the Intellectual Property Law in Taiwan (2012)
17. Chen, J.-Y.: A study of the consistency of judicial decisions based on text mining: using corpuses from rulings on applications for suspension of detention (2015)
18. Kleene, S.C.: Representation of events in nerve nets and finite automata. Autom. Stud. **34**, 3–41 (1956)