JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 41, XXX-XXX (2025)
DOI: 10.6688/JISE.202505_41(3).0004

Ensemble BERT Techniques for Financial Sentiment Analysis and Argument Understanding with Linguistic Features in Social Media Analytics

EUGENE SY, TZU-CHENG PENG, HENG-YU LIN, SHIH-HSUAN HUANG, YUNG-CHUN CHANG[+] AND CHIEN-PING CHUNG[†,1]
Graduate Institute of Data Science
Taipei Medical University
Taipei, 110 Taiwan
E-mail: {m946111012; m946111003; m946111008; m946110008; changyc}@tmu.edu.tw; thomas6311@mail.ntut.edu.tw
[1]*Department of Information and Finance Management*
National Taipei University of Technology
Taipei, 106 Taiwan

Financial argument mining provides a systematic approach to extract valuable insights from vast number of financial texts which can be used for tasks such as investment analysis, risk assessment and financial news summarization. Our study, which focused on argument unit identification and argument relation detection and classification provides a novel ensemble approach by combining traditional machine learning and fine-tuned language models such as BERT and its variations, with techniques to address data imbalance and a robust voting mechanism to form a consensus to enhance prediction accuracy. Through our experiments, we were able to achieve Macro-F_1 scores of 77.08% for argument unit identification. Macro-F_1 scores of 57.90% were achieved in our experiments for relation detection and classification. It demonstrates that ensemble models with voting mechanisms are a viable alternative that can outperform single model methods in this study. The developed method also sets a new benchmark in the field of argumentative mining in finance, demonstrating new ways to analyze financial discourse in a more advance and accurate way.

Keywords: financial NLP, argumentative mining, ensemble technique, data imbalance, log-likelihood ratio

1. INTRODUCTION

In the rapidly changing field of Natural Language Processing (NLP), the financial sector is increasingly using these technologies. Integrating NLP in finance is not only innovative but has become essential, driven by the need to manage and interpret the vast and complex data within the sector. This paper aims to explore this integration, focusing on the role of argumentation mining within the financial domain. The rise of NLP applications in various fields signals a new era where textual data serves as a rich resource

Received February 29, 2024; revised May 23 & August 13, 2024; accepted August 15, 2024.
Communicated by ——.
[+]Corresponding author: changyc@tmu.edu.tw
[†]Co-corresponding author: thomas6311@mail.ntut.edu.tw

for insights and predictions. In the financial sector, these applications are diverse and impactful, ranging from analyzing detailed financial documents to fraud detection, creating firm-specific narratives, assessing the readability of financial statements, and improving financial forecasting [1, 2]. The data sources are equally varied, including corporate disclosures, financial reports, professional journals, aggregated news, online forums, and social media [2]. The rise of deep learning in NLP has further changed this field. Techniques like BERT [3], a transformer-based pre-trained language model (PLM), have been adapted and fine-tuned for financial tasks, leading to the development of specialized models such as FinBERT [4], aiming for state-of-the-art (SOTA) performance in this domain.

While these advancements are impressive, there is still a significant area yet to be fully explored: the field of argument mining in finance [5]. Understanding the argumentative structure in financial texts is crucial, as it reveals not only the positions adopted by entities but also the reasons behind these stances [6]. This insight is valuable across various applications, from predicting financial market trends to planning public relations strategies. This paper addresses this emerging area by focusing on two critical tasks: argument unit identification (AUI) and argument relation detection and classification (ARDC). The former involves identifying whether a given text segment is a claim or a premise, while the latter involves determining the nature of the relationship between pairs of text segments, categorizing them as support, attack, or neutral [5]. Argument unit classification and relation detection are fundamental to understanding and automating the process of argumentation in financial texts.

Our paper aims to contribute to these two areas by developing a model capable of these tasks with SOTA performance, paving the way for more advanced and accurate analysis of financial discourse. We propose a unified ensemble architecture that integrates and combines multiple fine-tuned language models through a voting mechanism. This ensemble approach capitalizes on the strengths of various advanced deep learning models, harnessing their collective power to make more accurate and robust predictions. The unified architecture simplifies the model development workflow while offering adaptability to customize the ensemble with models specifically selected to offset each other's weaknesses. This strategy is particularly advantageous when dealing with challenges like data imbalance, where tailored models can be incorporated to address such issues.

The following study outlines our research in a structured manner across sections. Section 2 provides a comprehensive review of prior research on similar tasks, setting the stage for our investigation. Next, Section 3 explains the methodology behind our proposed approach and the innovative strategies we used. In Section 4, we present the outcomes of our experiments and analyze the results. This leads to the Discussion section, where we compare our findings with existing studies to highlight the performance and advancements achieved. Finally, in Section 5, we summarize our findings and draw conclusions from our research.

2. RELATED WORK

This section summarizes key research and developments in financial NLP, highlighting its role in market analysis, investor behavior, credit assessment, and sentiment analysis. It also covers the advancements in argumentative mining, focusing on the use of

pre-trained and large language models, their integration with other neural network architectures, and strategies to address data imbalance in argument mining.

2.1 Financial NLP

The evolution and current state of Financial NLP is a central topic in computational finance research [2]. Financial NLP began in the 1980s with basic text analysis techniques applied to financial reports and press releases. These early methods had limitations due to their inability to understand complex meanings. The rise of social media in the 2010s introduced real-time data into financial prediction models. More advance deep learning architectures like Convolutional Neural Networks (CNN), Restricted Boltzmann Machines (RBMs), and Long Short-Term Memory (LSTM) began to be used, sometimes alongside traditional models like Autoregressive Integrated Moving Average (ARIMA). This period saw improvements in sentiment analysis and semantic modeling using tools like SenticNet and SentiWordNet for interpreting market sentiment [2]. Building on this, Mishev et al. [7] highlight the critical role of sentiment analysis in finance, noting challenges with financial-specific language and the scarcity of large labeled datasets. They describe the evolution from lexicon-based methods to advanced deep learning techniques, including RNNs, CNNs, and PLMs like BERT and RoBERTa, which have significantly improved sentiment analysis accuracy in finance. Similarly, Fisher, Garnsey, and Hughes [1] provide an overview of NLP applications in accounting, auditing, and finance, showing NLP's capabilities in knowledge organization, fraud detection, and predictive analytics. Further expanding on these applications, Kumar and Ravi [8] explore the breadth of text mining applications in finance, from FOREX and stock market predictions to customer relationship management and cybersecurity. Their survey emphasizes the importance of preprocessing, feature selection, and various text mining techniques, indicating a shift towards more comprehensive applications in financial forecasting. This progression illustrates how Financial NLP has evolved from basic text analysis to sophisticated deep learning and transformer-based PLMs, enhancing the accuracy of financial forecasting and broadening the scope of NLP applications in finance.

Recent developments in Financial NLP have been driven by specialized workshops and shared tasks that address various aspects of financial text processing and analysis. Platforms like the Workshop on Financial Narrative Processing (FNP), Financial Technology and Natural Language Processing (FinNLP), FinNum Series, and the new FinArg Series have advanced research in this domain. Specifically, the FNP workshops [9–12] have focused on summarizing financial disclosures, extracting structures within financial documents, and detecting causal relationships within financial narratives. Likewise, the FinNLP workshops [13–17] have developed and benchmarked NLP tools specific to financial technology, including tasks like classifying financial terms into relevant hypernyms and sentence boundary detection in noisy financial texts. Simultaneously, the FinNum series [18–20], focuses on understanding numerals in financial contexts, including classifying numerals in financial tweets and analyzing the relationship between numerals and cashtags. Finally, the FinArg series [5], introduces tasks in financial argument mining, such as separating financial reports into premises and claims and analyzing social media discussion threads to determine argumentative links.

These workshops and shared tasks have significantly advanced automated financial text analysis and interpretation, promoting innovation and the creation of tools and

methodologies specifically designed for financial language processing. This collaborative approach enhances understanding and capabilities within the field, driving forward the state of the art in financial NLP.

2.2 Model Architectures in Argument Mining

Focusing on the FinArg series in argumentative mining, significant progress has been made in model architecture, especially with the use of pre-trained language models (PLMs) and large language models (LLMs) [5]. The use of transformer-based PLMs, particularly BERT [3] and its variants [4,21–23], has become prominent in financial argument mining. BERT's bidirectional understanding of textual context is a major improvement over older models that processed text in one direction. FinBERT [4], a version of BERT fine-tuned on financial texts, performs better in tasks like sentiment analysis and argument classification within financial texts. Building on the importance of context in argumentative relation mining, Nguyen and Litman [24] proposed a context-aware approach that utilizes features extracted from context windows surrounding argument components. Their work demonstrated the effectiveness of incorporating contextual information in improving the performance of argumentative relation classification tasks, such as identifying Support and Attack relationships. This research highlights the potential of leveraging broader contextual cues in argument mining, which aligns with our approach of using ensemble techniques and linguistic features for financial argument analysis.

Innovative research has explored hybrid models that combine PLMs with other neural network architectures. For instance, using BERT's embeddings as input to a CNN merges BERT's contextual understanding with CNN's pattern recognition capabilities [25]. This combination captures both the sequence of language and local textual features, improving the classification of complex argumentative structures. Another technique involves freezing the embedding layers of PLMs during training to prevent overfitting and focus on higher-level feature extraction [26]. These approaches highlight the adaptability and potential of transformer-based models in financial argument mining. The emergence of LLMs, like OpenAI's GPT models [27], has introduced new strategies in argumentative mining. Prompt-based learning [25, 28], such as using the T5 model [29] with prompts like "choose premise or claim," has shown promising results [25, 28]. ChatGPT has also been explored for prompt engineering, opening new possibilities in argumentative mining [30]. Techniques like zero-shot and few-shot learning with GPT-3.5 Turbo demonstrate the versatility of LLMs in this field [26].

A major challenge in argumentative mining, particularly in Argument Relation Detection and Classification, is data imbalance. This problem is worsened by the multi-class nature of the task, with some classes, like "Attack," being significantly underrepresented [31]. To address this imbalance, researchers have used various methods, including different sampling techniques (such as under-sampling or over-sampling specific classes) [28, 30, 32] and data augmentation strategies [8, 26]. Data augmentation involves creating additional data to support underrepresented classes. Cost-sensitive learning, which adds a specific cost in the machine learning process to improve classification accuracy, has also been used [28]. One method of data augmentation involves using the NLPAUG [33] library's Contextual Word Embedding Augmenter and Synonym Augmenter to paraphrase sentences [8]. Additionally, LLMs serve a dual role in this field: they are used as primary models for mining tasks and as tools for generating synthetic data to

address dataset imbalances [26, 30]. This dual use highlights the versatility of LLMs in solving problems and improving dataset quality.

2.3 Ensemble Techniques

Ensemble learning uses the idea of combining opinions from multiple experts to make better decisions [34]. This idea has been applied to automated decision-making, where ensemble methods usually perform better than single systems in many cases [35]. Techniques like bagging, boosting, and stacking combine predictions from multiple models. Algorithms such as Random Forest, AdaBoost, and Gradient Boosting Machines are popular examples [35, 36]. Ensemble voting is a common method that combines predictions from multiple classifiers through majority voting (hard voting) or weighted voting (soft voting). In hard voting, each classifier's prediction is equally weighted, while in soft voting, weights are based on the classifiers' performance or other factors. Many studies have investigated the effectiveness of these voting techniques in different applications.

In clustering ensembles, the Soft-Voting Clustering Ensemble (SVCE) method showed better performance compared to traditional voting methods on 15 UCI datasets. SVCE can handle both hard and soft clustering results, making it more flexible and general [37]. In bagging ensembles, a class-specific weighted soft voting method that adjusts weights based on test performance and intra-class variability showed slight improvements in accuracy, suggesting it is more reliable [38]. Additionally, a class-specific soft voting system for multiple extreme learning machines was proposed, which refines weights for each class. This enhances performance without increasing computational load, as demonstrated in various experiments [39].

In agriculture, a plant disease detection system using ensemble learning with both soft and hard voting classifiers was developed. Soft voting achieved 97.8% accuracy, while hard voting achieved 98.3%, showing the effectiveness of ensemble learning in improving diagnostic accuracy [40]. In medical diagnostics, the performance of hard and soft voting in breast tumor classification was evaluated. Hard voting achieved an accuracy of 99.42%, slightly outperforming soft voting, highlighting the potential of ensemble methods in improving medical diagnostic systems [41]. For predicting household food security status, soft voting outperformed hard voting with an accuracy of 99.79%, demonstrating its effectiveness in socio-economic applications [42].

Across multiple applications, both hard and soft voting improve prediction accuracy. Soft voting often offers better flexibility and accuracy by weighting predictions based on model performance, while hard voting remains a robust choice due to its simplicity. The consistent improvements in predictive accuracy across different fields underscore the value of ensemble learning methods, especially ensemble voting techniques, in complex classification tasks. Recent studies have further extended these concepts to BERT-based models, showcasing their potential in natural language processing tasks. For instance, BERT-based ensemble approaches have achieved state-of-the-art performance in biomedical literature classification [43], harmful news detection [44], sentiment analysis [45], idiom classification [46], and multi-aspect hate speech detection [47]. These studies highlight the versatility and effectiveness of combining BERT models with ensemble techniques, further advancing the field of text classification and demonstrating the power of integrating advanced language models with established ensemble methods.

3. MATERIALS AND METHOD

3.1 Dataset

The FinArg Dataset [31] is designed for annotating argumentation structures in financial earnings conference calls (ECCs). This dataset is important for research in computational argumentation, finance, and Financial NLP. It focuses on ECC transcripts and was annotated by four experts across 136 documents, resulting in a total of 804 documents. The creation of the dataset involved extensive data collection, strict annotation guidelines, and thorough checks for inter-annotator reliability. The dataset has a balanced distribution within the Argument Unit Identification Task. As shown in Table 1, it includes 5,078 'Premise' entries and 4,613 'Claim' entries, distributed across training, development, and testing phases with an 80-10-10 split. For our experiments, we combined the training and development subsets into one unified dataset. This unified dataset was then used for stratified 10-fold cross-validation during training. In contrast, the Argument Relation Detection and Classification section shows an imbalanced label distribution. This section uses a three-class model for sentence pairs, including 2,000 'No Detected Relation' instances, 4,823 'Support' instances, and only 78 'Attack' instances. This results in 28.98% 'No Detected Relation', 69.89% 'Support', and only 1.13% 'Attack'. The data partitioning for this section also follows an 80-10-10 split for training, development, and testing subsets. Both the training and development sets are used to train and internally evaluate the models, while the testing set is reserved exclusively for the final evaluation.

Table 1. Dataset Distribution for Argument Tasks

	Train (80%)	**Dev (10%)**	**Test (10%)**	**Whole (100%)**
Argument Unit Identification				
Premise (52.40%)	4,062	508	508	5,078
Claim (47.60%)	3,691	461	461	4,613
Total (100%)	7,753	969	969	9,691
Argument Relation Detection and Classification				
Support (69.89%)	3,859	482	482	4,823
Attack (1.13%)	62	8	8	78
No Detected Relation (28.98%)	1,600	200	200	2,000
Total (100%)	5,521	690	690	6,901

3.2 BERT-based Ensemble with Linguistic Features for Financial Sentiment Analysis and Argument Understanding

Our proposed methodology workflow, as shown in Figure 1, consists of two main phases: Ensemble Pre-Selection and Unified Ensemble, which includes methods for handling data imbalance. In the Ensemble Pre-Selection phase, we start by transforming the raw text data into formats suitable for training without any initial text cleaning. We then train selected models and rank them by performance. The best models undergo hyperparameter optimization to improve their performance further. In the Unified Ensemble

phase, we integrate the outputs of these optimized models using a voting mechanism. This combines model predictions to create a more accurate and resilient decision boundary. To address data imbalance, especially in tasks like argument relation detection and classification, we implement sampling methods during preprocessing. This ensures a balanced representation of all classes in the training data, which helps avoid biases and improves the model's generalization.

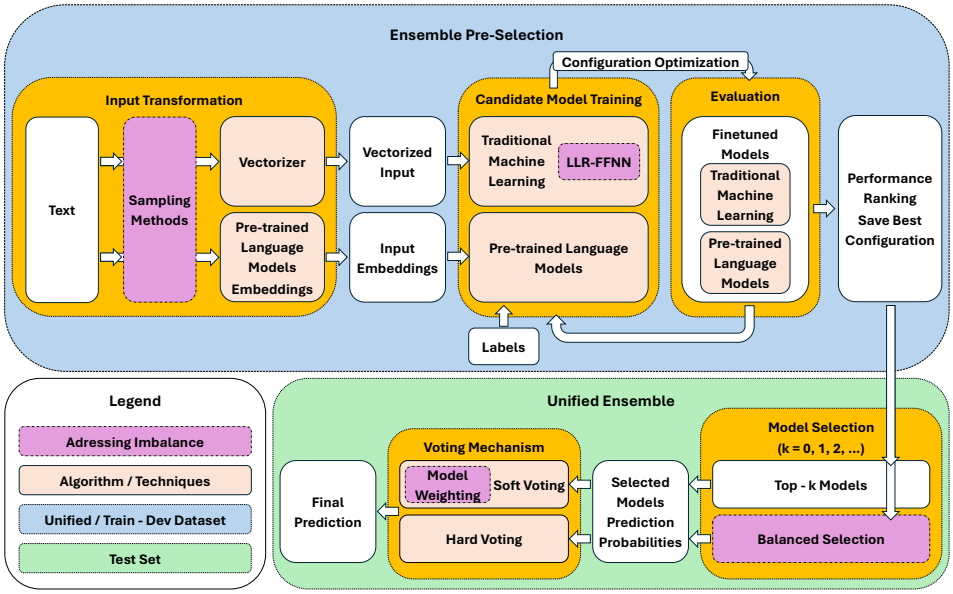

Fig. 1. Proposed Unified Ensemble Architecture Workflow

Our unified ensemble approach simplifies the model development workflow and aims to outperform existing methods by selecting and combining multiple fine-tuned language models through a voting mechanism. This method ensures the final prediction represents a consensus among the models, leveraging their combined strengths. A key advantage of this unified ensemble architecture is its adaptability compared to traditional single-model approaches, allowing for customized selection of models within the voting system to offset each other's weaknesses. This adaptability is particularly beneficial for tasks facing data imbalance issues. However, the challenge lies in determining the optimal combination of models given the vast array of available language models and their configurations, while also considering resource constraints.

To address this challenge, we propose an experimental setup in the Ensemble Pre-Selection phase. This setup evaluates trained models based on specific metrics, helping to narrow down the most effective ensemble combinations by focusing on the top-performing models in our rankings. By following this workflow, we aim to create a robust and adaptable model that can handle various challenges, particularly data imbalance, while providing accurate and reliable predictions.

3.2.1 Ensemble Pre-Selection

In this phase, our goal is to fine-tune individual models for their specific tasks before combining them into an ensemble. The process starts with transforming the raw text into inputs suitable for their respective methods. For Transformer-based Pre-trained Language Models (PLMs), we tokenize and transform input texts through their specific tokenizers. Text is padded or truncated to a consistent length of 512 tokens due to input size constraints. For traditional machine learning methods, we use the Term Frequency-Inverse Document Frequency (TF-IDF) vectorizer from scikit-learn to convert input text into vectors using the default configurations.

After preparing the inputs, we consider a range of models for this study. For traditional machine learning methods, we select Multinomial Naive Bayes (NB), Logistic Regression (LR), K-Nearest Neighbors (KNN), Decision Tree Classifier (DT), Support Vector Classifier (SVC), Random Forest Classifier (RF), XGBoost Classifier (XGB), and LightGBM Classifier (LGBM). For PLMs, we focus on BERT and its various versions to determine the most effective model for our tasks. We select BERT [3], FinBERT [4], ELECTRA [48], ROBERTA [21], ALBERT [49], and DISTILBERT [23] for Argument Unit Identification. For Argument Relation Detection and Classification, we choose BART [50] and DEBERTA [51].

Our objective is to find the most suitable model for our study's requirement of achieving a high Macro-F1 score. Each model's unique characteristics and strengths make them strong candidates for the tasks of Argument Unit Identification and Argument Relation Detection and Classification. In the initial training phase, we first train and evaluate the baseline performance of our selected models with their default configurations. Next, we select the top-performing models from each task and further optimize them through hyperparameter tuning.

3.2.2 Unified Ensemble

In this section, we implement a voting mechanism that integrates various model variations for enhanced decision-making. The ensemble technique for the final model selection incorporates a top-k approach. Here, the optimized models from the Ensemble Pre-Selection phase are selected based on their Macro-F_1 scores, with increasing values of k, specifically for the task of Argument Unit Identification. However, when handling a data imbalance issue like the one found in Argument Relation Detection and Classification, a more diverse selection of model is deemed more appropriate for this task rather than a straightforward ranking. Two types of voting mechanisms are employed. The first is Soft Voting, let $M = \{m_1, m_2, \ldots, m_n\}$ be the set of fine-tuned models in the ensemble, where each model m_i provides a prediction score for each class in a multiclass classification problem. The Soft Voting score, S_{soft}, for a given class c is calculated as:

$$S_{soft}(c) = \frac{1}{n} \sum_{i=1}^{n} p_{i,c} \tag{1}$$

$p_{i,c}$ is the prediction probability of model m_i for class c. The final prediction, C_{final}^{soft}, is the class with the highest Soft Voting score:

$$C_{final}^{soft} = \arg\max_{c}\{S_{soft}(c)\} \tag{2}$$

The second mechanism is Hard Voting, or Majority Voting. Each model m_i casts a binary vote $v_{i,c}$ for each class c, where $v_{i,c} = 1$ if m_i predicts class c and $v_{i,c} = 0$ otherwise. The Hard Voting score, S_{hard}, for class c is the sum of votes from all models:

$$S_{hard}(c) = \sum_{i=1}^{n} v_{i,c} \tag{3}$$

The final prediction, C_{final}^{hard}, is the class with the majority of votes:

$$C_{final}^{hard} = \arg\max_{c}\{S_{hard}(c)\} \tag{4}$$

3.2.3 Addressing Data Imbalance

To address the data imbalance inherent in the Argument Relation Detection and Classification dataset, we propose a strategy that encompasses sampling methods, model and class weighting, and the innovative use of the Log-Likelihood Ratio (LLR) as a feature within a Feed-Forward Neural Network (FFNN). To make the model more sensitive towards underrepresented classes, we explore two sampling methods: Random Sampling and SMOTE (Synthetic Minority Over-sampling Technique), along with Class Weighting Sampling to adjust class weights in the loss function, and Model Weighting to optimize ensemble model performance based on the F_1 score. Chang et al. [52] demonstrated a method for detecting topic-person interactions, utilizing LLR as an efficient feature selection mechanism. Recognizing the parallel between the requirement for measuring relationships in the Argument Relation Detection and Classification Task and the LLR method, we adapted Chang et al.'s LLR-based approach. This adaptation focuses on extracting and prioritizing word pairs across sentences, using LLR scores to find those with significant relevance to specific class labels.

Our approach starts with text preprocessing by removing noise (e.g., stopwords, punctuation) to prepare for LLR feature extraction. For each pair of sentences, we generate word pairs and calculate their LLR scores, identifying those with the most substantial association to the context of each sentence pair. Next, we select the top-n word pairs with the highest LLR scores for each class — top-50 for 'No Detected Relation', top-300 for 'Support', and top-2000 for 'Attack'. These word pairs are determined to be most effective through experimentation. To elaborate on the LLR calculation, let i and j denote specific word pairs within the corpus, where i refers to the first word in a pair and j to the second. For a given class label c, the LLR score for a word pair (w_i, w_j) is calculated as follows:

$$LLR(w_i, w_j|c) = 2\sum_{ij} O_{ij} \log\left(\frac{O_{ij}}{E_{ij}}\right) \tag{5}$$

where O_{ij} represents the observed frequency of the word pair (w_i, w_j) for class c, and E_{ij} represents the expected frequency of this pair, assuming independence between words.

Utilizing this methodology, we construct a new dataset for the FFNN, where each instance is represented by a feature vector comprising '1's and '0's, indicating the presence or absence of these top-ranked word pairs in the sentence pair. This binary representation captures the most salient relationships within the data, directly addressing class imbalance by focusing on features that are most indicative of each class. This dataset, enriched with LLR-based features, serves as the input to the FFNN model. Positioned within an ensemble framework, this model aims to improve the accuracy and consistency of predictions across different class categories in the dataset, thereby significantly advancing the Argument Relation Detection and Classification task. Our method, focusing on the strategic selection and use of top word pairs based on LLR scores, not only addresses data imbalance but also boosts the ensemble model's ability to generalize across varied class categories within the dataset.

3.3 Novelty of the Proposed Unified Ensemble Workflow

The main novelty of our proposed approach is the creation of a systematic and flexible workflow for building ensemble models. This workflow guides researchers and practitioners in using multiple models together, customized for specific tasks. In the Ensemble Pre-Selection phase, the workflow provides a structured method for evaluating potential models using various metrics. This helps in selecting the best models. Its flexible design allows easy integration of different types of models, from traditional machine learning to advanced deep learning, accommodating the fast-changing field of NLP. The Top-k method used in the Unified Ensemble phase highlights the workflow's adaptability. By changing the value of k, researchers can test different ensemble combinations, finding the best balance between performance and computational cost. This method goes beyond the limitations of single-model architectures, allowing the creation of ensembles suited to specific tasks. Additionally, the workflow's flexible design allows the use of specialized techniques to tackle task-specific issues. For example, in our study, we included sampling methods and the innovative Log-Likelihood Ratio feature extraction technique to address data imbalance, a common challenge in argument mining tasks. This flexibility allows researchers to develop and add custom solutions, enhancing the workflow's usefulness across various problems. By combining model selection, ensemble construction, and task-specific adjustments into one unified workflow, our approach offers a comprehensive guide for developing ensemble models. This structured method not only helps efficiently explore model combinations but also promotes reproducibility and transparency, supporting collaborative progress in the NLP community.

Our proposed workflow is a significant contribution to the field of ensemble learning for NLP tasks. It offers a principled and flexible approach to leveraging the strengths of multiple models while addressing specific challenges. This emphasis on systematic and adaptable methods underscores the novelty and utility of the workflow for constructing tailored ensemble models.

4. EXPERIMENT

4.1 Model Configuration and Evaluation Metrics

4.1.1 Finetuned Candidate Models

To optimize the selection of language models for our final voting ensemble, we conducted series of experiments to optimize the performance of top performing models from the initial training phrase. Results for the Argument Unit Identification task revealed that PLMs consistently outperformed traditional machine learning models. Table 2 demonstrates that ELECTRA emerged as the front-runner, boasting a Macro-F_1 score of 76.12%. It was closely followed by a hierarchy of PLMs, with RoBERTa, BERT-base-uncased, FinBERT, ALBERT, and DistilBERT. A similar trend was observed in the Argument Relation Detection and Classification task as shown in Table 3, where PLMs dominated in the Dev Set, and BART secured the top position with a Macro-F_1 score of 50.69%. Based on the results, PLMs were selected to form the basis of the ensemble model for both tasks.

Table 2. AUI Model Performance on Unified Set (10 Folds)

Methods	Unified Set	
	Macro-F_1 (%)	Micro-F_1 (%)
MLs		
SVC	**72.34**	**72.54**
LR	72.15	72.36
LGBM	71.14	71.28
RF	70.99	71.16
NB	70.51	70.58
XGB	69.83	70.02
KNN	63.59	64.22
DT	61.04	61.12
PLMs		
ELECTRA	**76.12**	**76.11**
RoBERTa	75.91	75.91
BERT	75.83	75.80
FinBERT	75.65	75.65
ALBERT	75.45	75.39
DistilBERT	75.28	75.24

Table 3. ARDC Model Performance on Dev Set

Methods	Dev Set	
	Macro-F_1 (%)	Micro-F_1 (%)
MLs		
KNN	**43.87**	72.23
LGBM	41.27	72.07
DT	39.55	64.52
XGB	38.99	71.13
RF	38.69	**73.28**
LR	36.55	70.31
SVC	36.36	65.21
LLR-FFNN	34.58	68.99
NB	28.89	69.99
PLMs		
BART	**50.69**	**80.87**
BERT	47.90	77.65
FinBERT	47.23	76.83
DEBERTA	44.98	72.03

For the task of Argument Unit Identification, our experiments with their hyperparameters provided us with distinct configurations that performed well. Each PLM were trained for 2 epochs, except for ALBERT which was trained for 3 epochs, using Mean Square Error loss function, optimized with AdamW, incorporating a dropout rate of 0.35 and a learning rate of 2e-05. In Argument Relation Detection and Classification, best performance was achieved when PLMs were trained for 30 epochs using Cross-Entropy loss function with AdamW optimizer. The dropout rate was kept at 0.3, and the learning rate was configured to 3e-07. The parameters and hyperparameters that were selected for

Table 4. PLM's Parameter Configuration

PLMs	Epochs	LF[1]	Optimizer	Dropout Rate	Learning Rate
Argument Unit Identification					
ALBERT [49]	3	MSE	AdamW	0.35	2e-05
BERT [3]	2	MSE	AdamW	0.35	2e-05
DistilBERT [23]	2	MSE	AdamW	0.35	2e-05
ELECTRA [48]	2	MSE	AdamW	0.35	2e-05
FinBERT [4]	2	MSE	AdamW	0.35	2e-05
RoBERTa [21]	2	MSE	AdamW	0.35	2e-05
Argument Relation Detection and Classification					
BART [50]	30	CE	AdamW	0.3	3e-07
BERT [3]	30	CE	AdamW	0.3	3e-07
DEBERTA [51]	30	CE	AdamW	0.3	3e-07
FINBERT [4]	30	CE	AdamW	0.3	3e-07

[1] LF: Loss Function, MSE: Mean Square Error, CE: Cross Entropy

the ensemble candidate PLMs in both tasks are summarized in Table 4.

For assessing model efficacy, we employ the Macro-F_1 score as the primary metric across both Argument Unit Identification and Argument Relation Detection and Classification tasks. This measure is particularly vital given the label imbalances present in the latter task, ensuring a balanced evaluation across diverse class distributions. Additionally, the Micro-F_1 score is recorded, serving as an essential counterpart to the Macro-F_1 in our analysis.

4.2 Results

4.2.1 Test Set Performance

In our analysis of the argument unit identification task, we evaluated various top-k ensemble configurations using both soft and hard voting methods on the test set. The results, as depicted in Table 5, indicate that the ensemble comprising the Top 4 PLMs from the unified dataset, employing soft voting, achieved the highest Macro-F_1 score of 77.083%. This score represents a significant improvement of 0.659% over the highest-performing individual PLM, BERT-base-uncased, which recorded a Macro-F_1 score of 76.424%.

Furthering our analysis, we observed that hard voting ensemble techniques generally achieved higher precision compared to their soft voting counterparts. This can be attributed to the hard voting approach requiring a majority of classifiers to have a prediction, thereby mitigating the occurrence of false positives. For instance, the Top 2 (H) configuration exhibited a precision of 76.941%, notably higher than the 74.268% precision attained by the Top 2 (S) configuration. However, this approach may adversely impact recall, as it can overlook true positives not agreed upon by the majority, as evidenced by the relatively lower recall of 73.102% in the Top 2 (H) compared to 77.007% in the Top 2 (S).

Conversely, soft voting technique tends to have more balance between precision and recall. This method considers the confidence levels of individual classifiers, potentially

Table 5. Ensemble Model and PLMs' Performance on AUI Test Set

Our Methods	Test Set			
	Precision (%)	**Recall (%)**	**Macro-F_1 (%)**	**Micro-F_1 (%)**
Hard Voting				
Top 2 (H)	**76.941**	73.102	76.658	76.780
Top 3 (H)	73.374	78.308	76.155	76.161
Top 4 (H)	74.477	77.223	76.551	76.574
Top 5 (H)	72.691	78.525	75.746	75.748
Top 6 (H)	73.795	76.356	75.826	75.851
Soft Voting				
Top 2 (S)	74.268	77.007	76.345	76.367
Top 3 (S)	74.167	77.223	76.348	76.367
Top 4 (S)	74.338	**79.176**	**77.083**	**77.090**
Top 5 (S)	74.530	77.440	76.656	76.677
Top 6 (S)	73.904	76.790	76.036	76.058
Finetuned PLMs				
ELECTRA	**75.162**	75.488	76.420	76.471
RoBERTa	71.727	81.996	76.049	76.058
BERT	75.054	75.705	**76.424**	**76.471**
FinBERT	70.849	**83.297**	75.718	75.748
ALBERT	72.893	69.414	73.032	73.168
DistilBERT	72.292	75.271	74.489	74.510

capturing a greater proportion of true positives without significantly increasing false positives. Our data justified this, with the Top 4 (S) configuration achieving the highest Macro-F_1 score of 77.083%, demonstrating a better balance between precision and recall compared to the Top 4 (H) configuration which had a Macro-F_1 score of 76.551%. Additionally, the Top 4 (S) showed a recall of 79.176%, significantly higher than the 77.223% recall observed in the Top 4 (H).

Overall, our results suggest that while hard voting may be preferable for applications prioritizing precision, as seen in the Top 2 (H) setup, soft voting provides a more balanced approach suitable for scenarios requiring a fair trade-off between precision and recall, as demonstrated by the Top 4 (S) configuration. This finding aligns with prior studies on ensemble learning methodologies, highlighting the trade-offs between precision and recall in different voting schemes [53, 54].

When juxtaposed with performances of other models in Table 6, it is evident that our ensemble model surpasses IDEA-1 which achieved the Macro-F_1-score of 76.46%, the next closest competitor, by a margin of 0.62%. Notably, other ensemble combinations from our experiments, including Top 2 Hard, Top 5 Soft, and Top 4 Hard, also demonstrated superior performance compared to IDEA-1 and the remaining models. Moreover, our approach shows significant robustness and effectiveness even when compared to newer LLMs such as T5 and GPT models. Specifically, while T5 prompting achieved a Macro-F_1 score of 76.36%, and the GPT-3.5-turbo prompting method reached only 56.82%, our method with BERT-based models maintained a higher performance level.

Table 6. AUI and ARDC Models' Performance Comparison

Methods	Architecture	Test Set	
		Macro-F_1 (%)	Micro-F_1 (%)
Argument Unit Identification			
Our Method	**Top-4 Soft Voting**	**77.08**	**77.09**
IDEA-1 [25]	BERT's final embeddings as CNN input	76.46	76.47
TUA1-1 [28]	T5 prompting	76.36	76.37
IMNTPU-2 [32]	RoBERTa-base	76.05	76.06
MONETECH-3 [26]	Finetune BERT (base)	75.53	75.54
MONETECH-1 [26]	Finetune BERT (base) w/ GPT augmented dataset	75.13	75.13
WUST-1 [55]	Finetune BERT (base)	74.41	74.41
LIPI-3 [8]	Finetune BERT-SEC	73.86	73.89
SCUNLP-1-2 [30]	Finetune DistilBERT w/ GPT augmented dataset	71.07	71.10
IMNTPU-3 [32]	GPT 3.5-turbo prompting	56.82	56.97
Argument Relation Detection & Classification			
TUA1-1 [28]	**Finetune T5 w/ Financial Phrasebank**	**61.50**	**85.65**
LIPI-3 [8]	Finetune FinBERT	60.22	79.42
Our Method	BART (CW) & LLR-FFNN (SM)	57.90	82.03
SCUNLP-1-3 [30]	Finetune DistilBERT w/ GPT augmented dataset	54.06	72.17
WUST-1 [55]	Finetune BERT	53.97	78.70
IMNTPU-2 [32]	Finetune FinBERT	52.97	82.61
IDEA-3 [25]	Finetune BERT	51.85	81.74

This demonstrates that our method's ensemble strategy and architectural optimizations leverage the strengths of BERT models effectively, yielding superior results.

For the argument relation detection and classification task, our experimentation focused on two main ensemble approaches. The first ensemble applied soft voting with model weighting between BART with Class Weighting (CW) and LLR-FFNN with SMOTE (SM). The second ensemble was similar to the first but included the addition of DEBERTA with Class Weighting, alongside a baseline of just BART with Class Weighting. As demonstrated in Table 7, the first ensemble, combining BART with Class Weighting and LLR-FFNN with SMOTE through soft voting, achieved the most notable Macro-F_1 score of 57.90%. This score represents a significant improvement of 1.18% over using BART with Class Weighting alone. Furthermore, compared to other approaches done as shown in Table 6, our most effective model for argument relation detection and classification ranked third overall. The second ensemble, which included DEBERTA, closely followed in fourth place, demonstrating the potential and effectiveness of these ensemble configurations.

To further elaborate on the effectiveness of our approach, particularly regarding the LLR + FFNN method: While this method showed lower performance on the development set, as shown in Table 3, its true value emerges when integrated into the ensemble model and evaluated on the test set. As evidenced in Table 7, the ensemble models R002 and R003, which incorporate LLR-FFNN with SMOTE, demonstrate improved performance over the baseline BART model (R001). Specifically, R002 (BART with Class Weighting

Table 7. Our Methods on ARDC Test Set

Our Methods[2]	Test Set (%)										
	Class 0: No Relation			Class 1: Support			Class 2: Attack			Macro F_1	Micro F_1
	Support: 200			Support: 482			Support: 8				
	P	R	F_1	P	R	F_1	P	R	F_1		
R001	**73.45**	65.00	68.97	**85.77**	**90.04**	**87.85**	14.29	12.50	13.33	56.72	81.88
R002	72.93	**66.00**	**69.29**	85.74	89.83	87.74	**25.00**	**12.50**	**16.67**	**57.90**	**82.03**
R003	72.78	65.60	68.95	85.74	89.83	87.74	20.00	12.50	15.38	57.36	81.88

[2] R001: BART (CW), R002: BART (CW) & LLR-FFNN (SM), R003: BART(CW) & LLR-FFNN (SM) & DEBERTA (CW)

& LLR-FFNN with SMOTE) achieved the highest Macro-F_1 score of 57.90% on the test set, surpassing the baseline by 1.18%. This improvement underscores the significance of the LLR + FFNN method when combined with other techniques in addressing the challenges of argument relation detection and classification, particularly in handling class imbalance.

It's important to note the difference between the Macro-F_1 score and Micro-F_1 score reported in Table 3 and Table 6 for both the dev set and test set evaluations. The significant differences between Micro and Macro-F_1 scores for all models, including benchmark models, can be attributed to the dataset's characteristics. The dataset for Relation Detection & Classification is heavily imbalanced, consisting of three classes where one class is notably underrepresented. As detailed in Table 1 and Table 7, the 'Attack' class constitutes only 1.13% of the data, leading to substantial imbalance. This imbalance affects the Macro-F_1 score, which gives equal weight to each class, thereby highlighting the performance on the underrepresented class. In contrast, the Micro-F_1 score aggregates the contributions of all classes, thus reflecting the model's overall performance across the dataset more evenly. This difference in Macro and Micro-F_1 scores can also be seen when compared to the other models in Table 6.

Our findings reveal the nuanced advantages of our proposed unified architecture. While our method did not surpass existing approaches in Argument Relation Detection and Classification, its significance lies in its ability to achieve state-of-the-art performance in Argument Unit Identification while maintaining comparable scores in relation tasks. This balanced performance across different aspects of argument mining demonstrates the potential of our unified approach. In the Argument Unit Identification task, our method's superior performance suggests an enhanced understanding of language semantics and argument structure. The comparable scores in Argument Relation Detection and Classification, though not surpassing current state-of-the-art methods, indicate that our unified architecture can effectively handle multiple aspects of argument mining without sacrificing performance in individual tasks. These results affirm the potential of our approach in addressing complex linguistic tasks holistically. While our findings align with current research on the effectiveness of advanced models in natural language processing, they offer a fresh perspective on leveraging a unified architecture to balance performance across different aspects of argument mining. This approach presents a more nuanced interpretation of model dynamics, potentially paving the way for more integrated solutions in the field.

5. CONCLUSION

In our research, we addressed argumentative mining tasks by developing a unified ensemble workflow that combines multiple fine-tuned language models. This new method of selecting and combining models outperformed traditional single-model approaches, showing significant improvements in both argument unit identification and argument relation detection and classification. Our ensemble included various models, from classic machine learning to advanced deep learning models. We tackled data imbalance with innovative strategies like the Log-Likelihood Ratio, which improved the model's sensitivity to minority classes—an important factor in argumentative analysis. A key feature of our study is a strong voting mechanism for model selection, ensuring that the final en-semble takes the best aspects of each model. This approach handled complex, imbalanced datasets effectively and was crucial in our experiments. The results showed that our en-semble models, especially those using soft voting, are effective in both tasks. This success validates our approach and sets a new benchmark in the field. Our study represents a sig-nificant advancement in argumentative mining. By combining innovative methods with thorough experimentation, we have established a new standard in the field, opening new possibilities for efficient and accurate argument analysis.

ACKNOWLEDGMENT

This study was supported by the National Science and Technology Council of Taiwan under grants NSTC 112-2410-H-038-007, and Joint Research Program Funding Sponsorship by University System of Taipei under Grant Number USTP-NTUT-TMU-112-06.

REFERENCES

1. I. E. Fisher, M. R. Garnsey, and M. E. Hughes, "Natural language processing in accounting, auditing and finance: A synthesis of the literature with a roadmap for future research," *Intelligent Systems in Accounting, Finance and Management*, Vol. 23, 2016, pp. 157-214.
2. F. Z. Xing, E. Cambria, and R. E. Welsch, "Natural language based financial forecasting: a survey," *Artificial Intelligence Review*, Vol. 50, 2018, pp. 49-73.
3. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidir- ectional transformers for language understanding," *arXiv Preprint*, 2018, arXiv:1810. 04805.
4. Z. Liu, D. Huang, K. Huang, Z. Li, and J. Zhao, "FinBERT: A pre-trained financial language representation model for financial text mining," in *Proceedings of the 29th International Conference on International Joint Conferences on Artificial Intelligence*, 2021, pp. 4513-4519.
5. C.-C. Chen, C.-Y. Lin, C.-J. Chiu, H.-H. Huang, A. Alhamzeh, Y.-L. Huang, H. Takamura, H.-H. Chen, and J. Zhao, "Overview of the ntcir-17 finarg-1 task: Fine-grained argument understanding in financial analysis," in *Proceedings of the 17th NTCIR Conference on Evaluation of Information Access Technologies*, 2023, pp. —-.

6. J. Lawrence and C. Reed, "Argument mining: A survey," *Computational Linguistics*, Vol. 45, 2020, pp. 765-818.
7. K. Mishev, A. Gjorgjevikj, I. Vodenska, L. T. Chitkushev, and D. Trajanov, "Evaluation of sentiment analysis in finance: from lexicons to transformers," *IEEE Access*, Vol. 8, 2020, pp. 131662-131682.
8. S. Chakraborty, A. Sarkar, D. Suman, S. Ghosh, and S. K. Naskar, "Lipi at the ntcir-17 finarg-1 task: Using pre-trained language models for comprehending financial arguments," in *Proceedings of the 17th NTCIR conference on evaluation of information access technologies*, 2023, pp. —-.
9. M. El-Haj, V. Athanasakou, S. Ferradans, C. Salzedo, A. Elhag, H. Bouamor, M. Litvak, P. Rayson, G. Giannakopoulos, and N. Pittaras, in *Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation*, 2020.
10. M. El-Haj, P. Rayson, S. Young, H. Bouamor, and S. Ferradans, in *Proceedings of the Second Financial Narrative Processing Workshop*, 2019.
11. M. El-Haj, P. Rayson, and N. Zmandar, in *Proceedings of the 3rd Financial Narrative Processing Workshop*, 2021.
12. M. El-Haj, P. Rayson, and N. Zmandar, in *Proceedings of the 4th Financial Narrative Processing Workshop@ LREC2022*, 2022.
13. C.-C. Chen, H.-H. Huang, H. Takamura, and H.-H. Chen, in *Proceedings of the 1st Workshop on Financial Technology and Natural Language Processing*, 2019.
14. C.-C. Chen, H.-H. Huang, H. Takamura, and H.-H. Chen, in *Proceedings of the 2nd Workshop on Financial Technology and Natural Language Processing*, 2020.
15. C.-C. Chen, H.-H. Huang, H. Takamura, and H.-H. Chen, in *Proceedings of the 3rd Workshop on Financial Technology and Natural Language Processing*, 2021.
16. C.-C. Chen, H.-H. Huang, H. Takamura, and H.-H. Chen, in *Proceedings of the 4th Workshop on Financial Technology and Natural Language Processing*, 2022.
17. C.-C. Chen, H. Takamura, P. Mathur, R. Sawhney, H.-H. Huang, and H.-H. Chen, in *Proceedings of the 5th Workshop on Financial Technology and Natural Language Processing and the 2nd Multimodal AI For Financial Forecasting*, 2023.
18. C.-C. Chen, H.-H. Huang, Y.-L. Huang, H. Takamura, and H.-H. Chen, "Overview of the ntcir-16 finnum-3 task: investor's and manager's fine-grained claim detection," in *Proceedings of the 16th NTCIR Conference on Evaluation of Information Access Technologies*, 2022, pp. —-.
19. C.-C. Chen, H.-H. Huang, H. Takamura, and H.-H. Chen, "Overview of the ntcir-14 finnum task: Fine-grained numeral understanding in financial social media data," in *Proceedings of the 14th NTCIR Conference on Evaluation of Information Access Technologies*, 2019, pp. 19-27.
20. C.-C. Chen, H.-H. Huang, H. Takamura, and H.-H. Chen, "Overview of the ntcir-15 finnum-2 task: Numeral attachment in financial tweets," *Development*, Vol. 850, 2020, No. 1-044.
21. Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv Preprint*, 2019, arXiv:1907.11692.

22. L. Loukas, M. Fergadiotis, I. Chalkidis, E. Spyropoulou, P. Malakasiotis, I. Androutsopoulos, and G. Paliouras, "Finer: Financial numeric entity recognition for XBRL tagging," *arXiv Preprint*, 2022, arXiv:2203.06482.
23. V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of bert: smaller, faster, cheaper and lighter," *arXiv Preprint*, 2019, arXiv:1910.01108.
24. H. Nguyen and D. Litman, "Context-aware argumentative relation mining," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, Vol. 1, 2016, pp. 1127-1137.
25. S. Tang and L. Li, "Idea at the ntcir-17 finarg-1 task: Argument-based sentiment analysis," in *Proceedings of the 17th NTCIR Conference on Evaluation of Information Access Technologies*, 2023.
26. S. Jiarakul, T. Tokunaga, and H. Yamada, "MONETECH at the NTCIR-17 FinArg-1 task: Layer freezing, data augmentation, and data filtering for argument unit identification," in *Proceedings of the 17th NTCIR Conference on Evaluation of Information Access Technologies*, 2023, p. 82-88.
27. A. Radford and K. Narasimhan, "Improving language understanding by generative pre-training," 2018. [Online]. Available: https://api.semanticscholar.org/CorpusID:49313245
28. D. Yamane, F. Ding, and X. Kang, "Tua1 at ntcir-17 finarg-1 task," in *Proceedings of the 17th NTCIR conference on evaluation of information access technologies. https://doi. org/10.20736/0002001288*, 2023.
29. C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *The Journal of Machine Learning Research*, Vol. 21, 2020, pp. 5485-5551.
30. Y.-M. Cheng and J.-L. Wu, "Scunlp-1 at the ntcir-17 finarg-1 task: Enhancing classification prediction through feature generation based on chatgpt," in *Proceedings of the 17th NTCIR Conference on Evaluation of Information Access Technologies*, 2023.
31. A. Alhamzeh, R. Fonck, E. Versmée, E. Egyed-Zsigmond, H. Kosch, and L. Brunie, "It's time to reason: Annotating argumentation structures in financial earnings calls: The finarg dataset," in *Proceedings of the 4th Workshop on Financial Technology and Natural Language Processing*, 2022, pp. 163-169.
32. C.-T. Tsai, W.-H. Liao, H.-C. Liu, V. Nataraj, T.-Y. Liu, M. T.-J. Jiang, and M.-Y. Day, "Imntpu at the ntcir-17 finarg-1 argument-based sentiment analysis and identifying attack and support argumentative relations in social media discussion threads," 2023.
33. E. Ma, "Nlp augmentation," 2019. [Online]. Available: https://github.com/makcedward/nlpaug
34. R. Polikar, "Ensemble based systems in decision making," *IEEE Circuits and Systems Magazine*, Vol. 6, 2006, pp. 21-45.
35. D. Mienye and Y. Sun, "A survey of ensemble learning: Concepts, algorithms, applications, and prospects," *IEEE Access*, 2022, p. 1.
36. T. R. N and R. K. Gupta, *Ensemble Learning Techniques and its Efficiency in Machine Learning: A Survey*, 2020, pp. 1-6. [Online]. Available: https://api.semanticscholar.org/CorpusID:221280210

37. H. Wang, Y. Yang, H. Wang, and D. Chen, *Soft-Voting Clustering Ensemble*, Z.-H. Zhou, F. Roli, and J. Kittler, (eds.). Berlin, Heidelberg: Springer Berlin Heidelberg, 2013.
38. L. N. Eeti and K. M. Buddhiraju, "A modified class-specific weighted soft voting for bagging ensemble," in *Proceedings of IEEE International Geoscience and Remote Sensing Symposium*, 2016, pp. 2622-2625.
39. J. Cao, S. Kwong, R. Wang, X. Li, K. Li, and X. Kong, "Class-specific soft voting based multiple extreme learning machines ensemble," *Neurocomputing*, Vol. 149, 2015, pp. 275-284.
40. H. K. Kondaveeti, K. G. Ujini, B. V. V. Pavankumar, B. S. Tarun, and S. C. Gopi, "Plant disease detection using ensemble learning," in *Proceedings of the 2nd International Conference on Computational Systems and Communication*, 2023, pp. 1-6.
41. A. S. Assiri, S. Nazir, and S. A. Velastin, "Breast tumor classification using an ensemble machine learning method," *Journal of Imaging*, Vol. 6, 2020, No. 39, 2313-433x Assiri, Adel S Orcid: 0000-0002-3678-0956 Nazir, Saima Orcid: 0000-0001-6577-8188 Velastin, Sergio A Orcid: 0000-0001-6775-7137 Journal Article Switzerland 2020/05/29 J Imaging. 2020 May 29;6(6):39. doi: 10.3390/jimaging6060039.
42. M. Nigus and H. Shashirekh, "Prediction of household food security status using ensemble learning models," *International Journal of Sensors, Wireless Communications and Control*, 2022.
43. S. J. Lin, W. C. Yeh, Y. W. Chiu, Y. C. Chang, M. H. Hsu, Y. S. Chen, and W. L. Hsu, "A bert-based ensemble learning approach for the biocreative vii challenges: full-text chemical identification and multi-label classification in pubmed articles," *Database*, Vol. 2022, 2022.
44. S.-Y. Lin, Y.-C. Kung, and F.-Y. Leu, "Predictive intelligence in harmful news identification by bert-based ensemble learning model with text sentiment analysis," *Information Processing I& Management*, Vol. 59, 2022, p. 102872. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306457322000073
45. H. Batra, N. S. Punn, S. K. Sonbhadra, and S. Agarwal, "Bert-based sentiment analysis: A software engineering perspective," in *Database and Expert Systems Applications*, C. Strauss, G. Kotsis, A. M. Tjoa, and I. Khalil, (eds.). Springer International Publishing, 2021, pp. 138-148.
46. J. Briskilal and C. N. Subalalitha, "An ensemble model for classifying idioms and literal texts using bert and roberta," *Information Processing I& Management*, Vol. 59, 2022, p. 102756. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306457321002375
47. A. C. Mazari, N. Boudoukhani, and A. Djeffal, "Bert-based ensemble learning for multi-aspect hate speech detection," *Cluster Computing*, Vol. 27, 2024, pp. 325-339. [Online]. Available: https://doi.org/10.1007/s10586-022-03956-x
48. K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning, "Electra: Pre-training text encoders as discriminators rather than generators," *arXiv Preprint*, 2020, arXiv:2003.10555.
49. Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," *arXiv Preprint*, 2019, arXiv:1909.11942.

50. M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," *arXiv Preprint*, 2019, arXiv:1910.13461.
51. P. He, X. Liu, J. Gao, and W. Chen, "Deberta: Decoding-enhanced bert with disentangled attention," *arXiv Preprint*, 2020, arXiv:2006.03654.
52. Y.-C. Chang, C. C. Chen, and W.-L. Hsu, "Spirit: A tree kernel-based method for topic person interaction detection," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 28, 2016, pp. 2494-2507.
53. L. Derczynski, "Complementarity, F-score, and NLP evaluation," in *Proceedings of the 10th International Conference on Language Resources and Evaluation*, N. Calzolari, K. Choukri, T. Declerck, S. Goggi, M. Grobelnik, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, and S. Piperidis, (eds.). Portorož, Slovenia: European Language Resources Association (ELRA), May 2016, pp. 261-266. [Online]. Available: https://aclanthology.org/L16-1040
54. L. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms: Second Edition*. John Wiley I& Sons, Ltd, 2014, Vol. 47.
55. M. Wu, M. Liu, and T. Zhang, "Wust at the ntcir-17 finarg-1 task," in *Proceedings of the 17th NTCIR Conference on Evaluation of Information Access Technologies*, 2023, pp. —-.

Eugene Sy received his Bachelor's degree in Electronics Engineering from Far Eastern University - Institute of Technology, Manila, in 2020. He completed his Master's degree in Data Science at Taipei Medical University. Currently, he is pursuing a PhD in the Artificial Intelligence of Things (AIoT) program at Academia Sinica in collaboration with National Taiwan University. His research interests include applying advanced Natural Language Processing techniques to financial and clinical applications, as well as exploring the integration of AI with IoT to enable data-driven decision-making.

Tzu-Cheng Peng received his Bachelor's degree in Business Administration from National Taipei University of Business, in 2020. He completed his Master's degree in Data Science at Taipei Medical University. Currently, he is pursuing a PhD in the department of Information Management, National Taiwan University. His research interests include artificial intelligence, data science, and the natural language processing.

Heng Yu Lin received her Bachelor's degree in Health Care Administration from Taipei Medical University in 2022. She completed her Master's degree in Data Science at Taipei Medical University. Her research interests include artificial intelligence, data science, and natural language processing.

Shih-Hsuan Huang received his Bachelor's degree in Medical Informatics of Computer Science from Fu Jen Catholic University, Taiwan, in 2021. He completed a Master's degree in Data Science at Taipei Medical University. Currently, he is employed at the National Center for Research on Earthquake Engineering. His research interests include artificial intelligence, data science, natural language processing, and software engineering.

Yung-Chun Chang received his PhD degree in information management from National Taiwan University, Taiwan, in 2016. He is currently a professor in the Graduate Institute of Data Science at Taipei Medical University and serves as the deputy chief data officer in the Office of Data Science. His significant scholarly contributions have been published in leading journals and conferences, including IEEE Transactions on Knowledge and Data Engineering, Information I& Management, International Journal of Information Management, International Journal of Nursing Studies, ACL, HICSS, and ICDE. His research primarily focuses on natural language processing, knowledge discovery, and the application of AI in electronic health records. His innovative work in these fields has made a considerable impact on both academic and applied aspects of information science.

Chien-Ping Chun received his Ph.D. in Economics from National Chengchi University, Taiwan, in 2008. He is an Associate Professor in the Department of Information and Finance Management at National Taipei University of Technology. His academic background is in the fields of Financial Technology, AI Applications, and International Finance. His current research focuses on AI development, integrating AI with Financial Technology, and Technology Innovation and Entrepreneurship.