ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

SELF-LEARNING METHODS IN EDUCATIONAL CONTEXTS

SEMESTER PROJECT ML4ED

Maxime Lelièvre

5th January 2024

ABSTRACT

In recent years, we have seen a dramatic increase in the rich data collected from technologically-enhanced learning environments. Machine learning (ML) can effectively analyze these data and inform the design of data-driven mechanisms to support learners, teachers, and other stakeholders. However, the process most often requires large labeled datasets, and labeling data is especially challenging in education. It usually requires experts' involvement and demands high pedagogical knowledge and expertise. Semi-supervised learning, particularly self-learning, is one of the ML approaches to overcome this challenge by leveraging the power of small amounts of labeled data. The basic idea behind self-training is to use the model trained on the labeled part of the data to infer predictions on unlabeled data. So, one can treat all or a portion of the predictions as labels for subsequent training. Such a process can be repeated iteratively while the entire dataset is labeled.

This paper explores the feasibility and potential applications of semi-supervised methodologies within the realm of multi-class text classification, specifically focusing on analyzing reflective writings.

Contents

Chapter 1

Introduction

1.1 Motivation

In recent years, the landscape of machine learning (ML), notably with the emergence of transformer-based models in Natural Language Processing (NLP), has witnessed a significant evolution. These transformer models, trained for general language comprehension, have redefined the capabilities of machines in educational tasks like automated essay scoring, automated scoring of responses to open-ended questions, and evaluating the quality of reflective writing. They've elevated the standards in areas such as adaptive learning systems, intelligent tutoring, and linguistic analysis of student writing to remarkable heights, making previously intricate endeavors attainable without exhaustive expert oversight.

However, such ML and NLP-based technologies are mainly based on supervised learning techniques and face a significant challenge: the need for large, labeled educational datasets. Creating these datasets requires considerable time and effort from human experts, who must manually annotate significant data volumes. This requirement becomes a major obstacle in effectively scaling such systems, especially in educational settings where quality labeling demands human raters with advanced pedagogical skills (Ullmann 2019), and pedagogical judgment is often subjective and can differ among experts. Consequently, this dependence on labeled data limits the practicality of providing pedagogical feedback driven by ML and NLP in educational contexts.

One of the promising ways to mitigate this challenge is data augmentation, which involves using algorithms to create additional, synthetic examples that mimic real data or applying semi-supervised learning methods (e.g., active learning, self-learning, etc.) that can utilize both labeled and unlabeled data. In other words, such semi-supervised approaches aimed to utilize large unlabeled datasets by incorporating and effectively leveraging small labeled datasets. In this research project, we investigate if and how semi-supervised approaches can be used in reflective practice teaching and learning.

Reflective practice, a critical element for professional development in education, provides individuals with deep insights from their personal experiences. This practice involves reflective writing, a key tool that enables educators and learners to express, analyze, and learn from their experiences. Reflective writing, traditionally done through journals and portfolios, is now being enhanced by digital platforms. These modern mediums, from online blogs to learning networks, are increasingly incorporating AI-powered technology to facilitate more structured and interactive forms of self-reflection. This evolution in reflective practice is about changing the mode of expression and enriching the quality of reflection. Through tools like AI-driven text analysis and interactive digital portfolios, technology allows for a more nuanced dissection and understanding of experiences.

The aforementioned semi-supervised approaches are particularly relevant in the context of reflective writing analysis, as the nuanced and subjective nature of the data makes the acquisition of large labeled datasets very challenging.

In this research project, we aim to answer the following research question:

If and how can the semi-supervised approaches assist in reducing human effort in labeling reflective writings?

To this end, we first establish several baselines on the existing labeled reflection dataset using BERT models and several ML approaches such as multi-class classification, binary classification, and downsampling for imbalanced datasets. Second, we apply the learning curves approach to investigate how the size of the dataset influences the resulting model's performance and the model's confidence in its predictions. Finally, we discuss how these insights can shed light on the effectiveness and usefulness of applying data augmentation and semi-supervised learning techniques such as self-learning.

1.2 Related Work

1.2.1 Reflection and reflective writing

Reflection on one's performance through reflective writing offers substantial educational benefits. As Krol 1996 suggests, it fosters dialogue between students and teachers, serving as an effective medium for self-assessment to evaluate personal epistemology. Reflective writing also facilitates peer discussion/assessment, providing insights into students' experience structuring (Holdan 2009; Maloney and Campbell-Evans 2002; Wallin and Adawi 2018). This practice aids student teachers in developing their identities and understanding their teaching practices.

However, effective reflective writing poses significant challenges both for students to master and for teachers to teach and assess (Buckingham Shum et al. 2016). For students, reflective writing requires a deep understanding of the subject matter and the ability to introspect and critically analyze their own thoughts and experiences. This introspection demands self-awareness and cognitive skills that can be challenging to develop, particularly for those new to the practice. Timely and frequent formative feedback can assist students in revising and improving their reflective writing skills. However, from the perspective of educators, teaching and assessing reflective writing (e.g., guiding students in writing mechanics and the more abstract aspects of reflection) is equally complex (Sumsion and Fleet 1996).

ML and NLP offer a potential solution to streamline feedback provision on students' reflective writing (Ullmann, Wild and Scott 2012). Traditional assessment methods primarily rely on qualitative or quantitative content analysis, demanding extensive manual effort. Automated evaluation techniques have emerged, initially employing dictionary-based or rule-based approaches, which have shown promise but need more sophistication (Ullmann 2019, 2015).

1.2.2 BERT fine-tuned models

BERT (Bidirectional Encoder Representations from Transformers) represents a breakthrough in Natural Language Processing (NLP), renowned for its prowess in understanding context and nuances within text data. Launched by Google AI Devlin et al. 2018, BERT revolutionized language understanding by pretraining a Transformer-based neural network on a vast corpus of text.

- Unlike earlier models, BERT grasps context from both left and right directions in a sentence, enabling a deeper comprehension of word relationships and meanings.

- Built on the Transformer architecture, BERT employs attention mechanisms to capture long-range dependencies within text, enhancing its ability to contextualize information.

- BERT is pre-trained on massive text corpora, learning rich representations of words, phrases, and sentences. These pre-trained models can be fine-tuned for specific downstream tasks (see Figure 1.1), adapting their learned representations for tasks like text classification, named entity recognition, and more.

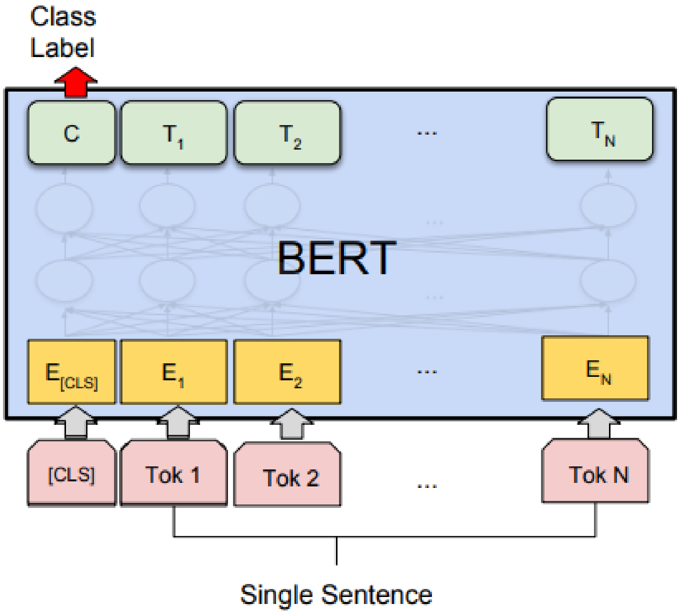


FIGURE 1.1
Illustration of Fine-tuning BERT on Sentence Classification Task

1.2.3 SEMI-SUPERVISED LEARNING AND SELF-LEARNING

In ML, various learning paradigms exist (see Figure 1.2), the two main ones being supervised learning and unsupervised learning. Supervised learning involves training models on labeled data, enabling them to learn patterns between input features and corresponding output labels. This method excels in delivering accurate predictions and structured learning. However, its reliance on labeled datasets poses challenges in data collection and might lead to overfitting or underfitting when facing unseen data. Thus, supervised learning is usually time-consuming and expensive for the labelization and requires domain expertise for specialized tasks (labeling medical images or reflective writings, for instance).

In contrast, unsupervised learning focuses on unlabeled data to identify inherent patterns or structures within datasets. While advantageous in utilizing vast amounts of unlabeled data and revealing hidden relationships, it faces challenges in evaluating models without clear metrics for performance assessment. The absence of explicit guidance in unsupervised learning might also lead to ambiguous or less interpretable representations, limiting the model's learning capabilities.

Both paradigms, supervised and unsupervised learning, have strengths and limitations, often influencing the choice based on data availability, task complexity, and the need for interpretability or generalization.

Semi-supervised learning and, in particular, self-supervised learning (or self-learning) represents a promising paradigm that bridges the gap between supervised and unsupervised methods and has already shown great results for various tasks, from images classification (Xie et al. 2020) to text classification (Liu et al. 2022). It operates under the premise that vast quantities of unlabeled data contain latent information

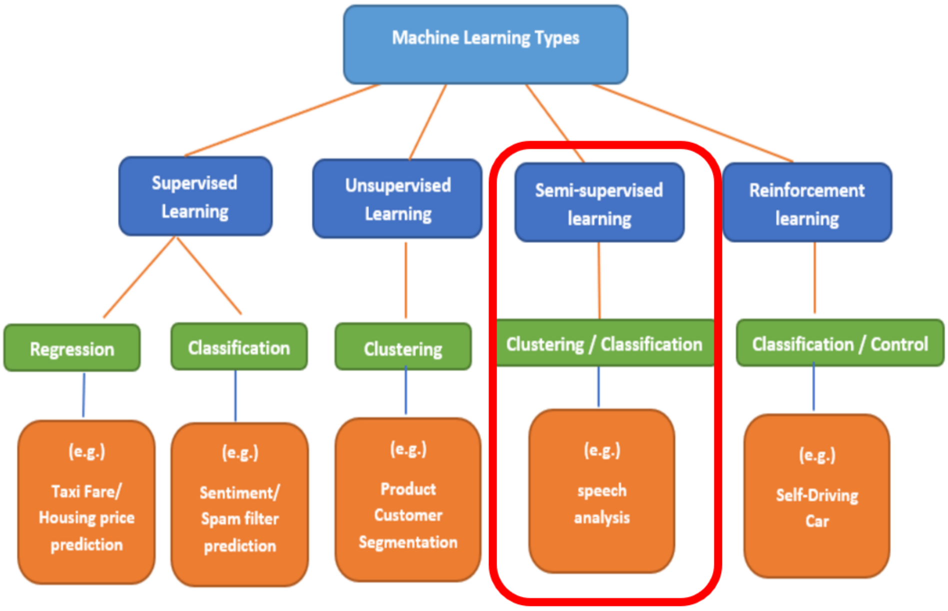


FIGURE 1.2
The different types of learning, Self-learning circled in red

that can be exploited without human annotation. This approach diverges from traditional supervised learning by generating labels directly from the input data itself, without requiring external annotations. Instead of relying on explicit labels, self-supervised learning designs pretext tasks that leverage inherent structures or relationships within the data. These tasks guide the learning process by defining auxiliary objectives, allowing the model to learn meaningful representations by predicting certain aspects of the data without explicit supervision.

Self-supervised learning offers different advantages: it can learn general-purpose representations, like supervised learning, reduce the financial and labor-intensive burdens of hand-labeling large datasets and capitalize on the abundance of unlabeled data.

Below is a detailed step by step explanation of pseudo-labeling, a self-supervised learning technique (see also Figure 1.3.

Teacher Model Training: Initially, a teacher model is trained solely using a labeled dataset. Different sizes of labeled samples are randomly chosen from the training set to train this initial teacher model.

Pseudo-label Generation: After training the teacher model, pseudo-labels are generated for an unlabeled dataset. These pseudo-labels are predictions made by the teacher model on the unlabeled data.

Selecting Pseudo-labeled Subset: Different strategies are explored for selecting subsets of pseudo-labeled data to train the student model:

- Full Selection: Using the entire pseudo-labeled set alongside the labeled set to train the student model.
- Random Selection: Randomly choosing a subset from the pseudo-labeled set and combining it with the labeled set for training.
- Top-k% Selection: Retaining prediction scores from the teacher model and selecting only the samples with the highest prediction scores (k% highest scores).

- Selection of samples with rare labels: Choosing a subset of samples predicted to have rare labels from the pseudo-labeled set.

Iterative Pseudo-labeling: The process involves iterations where the selected pseudo-labeled subset is used for training the student model and the remaining pseudo-labeled data serves as the unlabeled set for the next iteration.

This iterative process continues, enhancing the model's performance as it learns from its own predictions, see Figure 1.3.

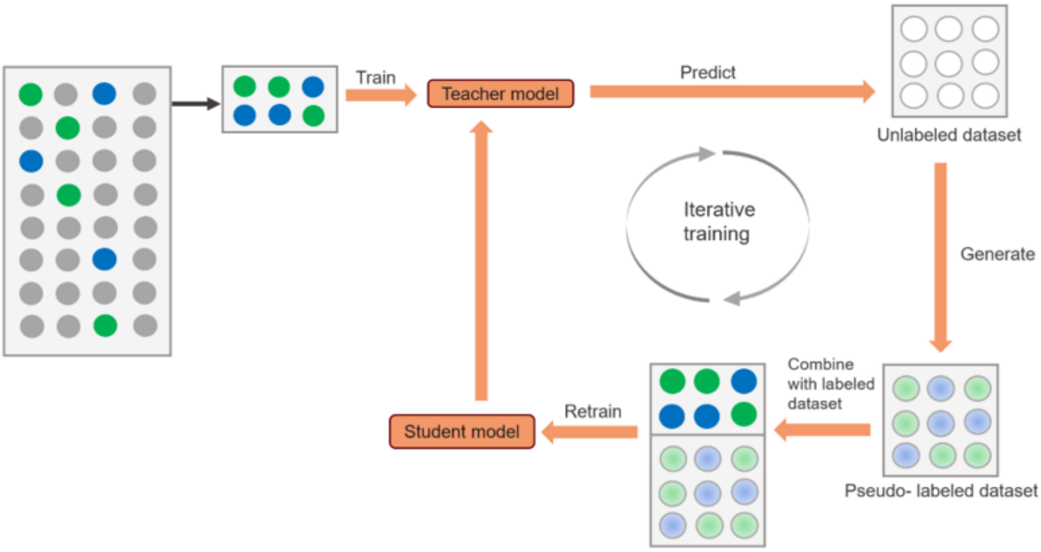


FIGURE 1.3
Illustration of Self-training workflow

The self-learning approach has been shown to be effective for image recognition (Xie et al. 2020) and for peer assessment in an educational context (Liu et al. 2022).

Chapter 2

Methods

2.1 Dataset

2.1.1 Original Data Collection and Annotation

Data collection process and composition

The Czech-English Reflective Dataset (CEReD) (Nehyba and Štefánik 2023) comprises 1,070 reflective journals (segmented in 33,859 sentences) from Czech student teachers in their fourth year of a five-year master's degree program. Students were tasked with creating reflective journals five times over a year, offering free-form text about their teaching practice experiences.

The distribution of reflective journals per subject exhibited a varied spread among different teaching subjects such as Czech, English, Geography, Physics, and chemistry education, among others.

Adoption of Annotation Meta-Scheme

This dataset follows the annotation meta-scheme established by Ullmann 2019, consisting of eight categories related to reflection: Reflection, Description of an Experience, Feelings, Personal Belief, Awareness of Difficulties, Perspective, Outcome – Lessons Learned, and Outcome – Future Intention (see Table 2.1). The selection of these categories stemmed from an analysis of 24 models of reflective writing by the authors, ensuring diversity and ease of comprehension for annotators.

Originally, the study of Ullmann 2019 dived into models used for reflective writing, revealing two significant aspects: depth and breadth.

In exploring these categories, the depth side focuses on distinguishing between reflective and non-reflective writing. On the other hand, the breadth categories encompass seven common components found in many reflective writing models. These include describing experiences, expressing emotions and personal beliefs, acknowledging difficulties, considering different perspectives, and discussing outcomes and future plans (see Ullmann 2015).

In (Nehyba and Štefánik 2023), the above two dimensions were combined into a single model identifying the eight categories mentioned.

Challenges and refinement in annotation

During the annotation process (see Figure 2.1), annotators were asked to categorize arbitrary units of text based on their connection to one of these predefined reflective categories. Despite meticulous annotation efforts, annotators faced challenges resulting in inconsistencies affecting classification quality. To address

Category	Indicator
Reflection	The sentence is reflective
Experience	The writer describes an experience he or she had in the past
Feeling	The writer describes his or her feelings
Belief	The writer describes his or her beliefs
Difficulty	The writer recognizes difficulties/problems
Perspective	The writer takes into account another perspective
Learning	The writer has learned something
Intention	The writer intends to do something

TABLE 2.1
Reflective Categories from Ullmann 2019 and their Indicators

this, a refined methodology was employed. Sentences, where the majority of annotators agreed on the true category, were selected to create a more decisive and separable dataset.

Subsequently, to address inconsistencies among annotators, a re-collection of annotations was conducted, taking into account only the annotations from the two best-performing annotators based on the classification performance of a random forest model trained on their personal annotations. This rigorous strategy aimed to mitigate inner inconsistencies among annotators, ensuring the dataset's reliability and enhancing its value for research in reflective writing within educational settings. This second round of annotations resulted in 7,128 labeled sentences.

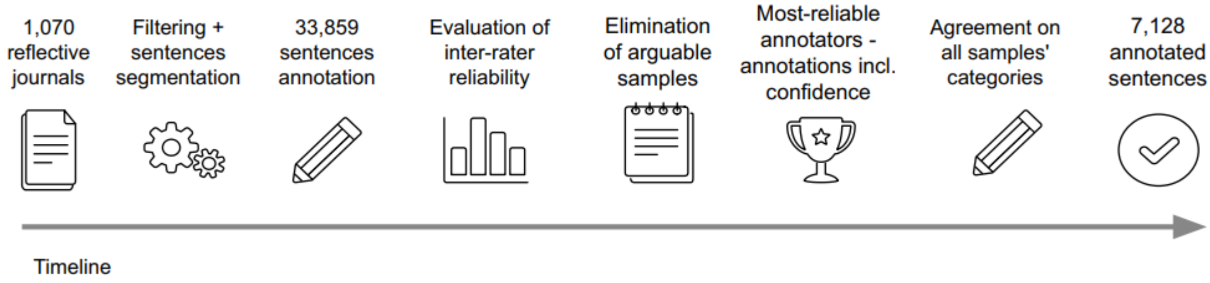

1,070 reflective journals
Filtering + sentences segmentation
33,859 sentences annotation
Evaluation of inter-rater reliability
Elimination of arguable samples
Most-reliable annotators - annotations incl. confidence
Agreement on all samples' categories
7,128 annotated sentences
Timeline

FIGURE 2.1
Flowchart of the data collection and annotation process outlines the two iterations of annotation collection that were performed to reach an inter-annotator agreement sufficient for training machine learning algorithm, from Nehyba and Štefánik 2023

For this project, we focused exclusively on the English version of the CEReD dataset. It's important to note that the original dataset was collected in Czech. However, the authors of the dataset translated it into English during their research, which is the version we utilized for our investigation.

2.1.2 PREPROCESSING AND CLEANING

During the initial stages, we encountered challenges with the distribution across the three individual datasets, observing an imbalance that would adversely affect the test dataset's adequacy for evaluation (see Figure 2.2). As a remedial measure, we opted to merge the initial datasets into one comprehensive dataset. The original 90-5-5% distribution did not align well with our evaluation needs, especially considering the minimal representation in the test dataset, which could compromise result reliability.

Post-merging, we conducted a meticulous resplitting of the combined dataset, allocating 85% to the training set and 15% to the test set. This division was performed while ensuring uniform distributions of classes across both datasets, enhancing the robustness of our subsequent analyses and evaluations.

Additionally, as part of the cleaning process, we identified certain discrepancies within the dataset's annotations. As explained in the previous section, the original Ullmann's taxonomy presented was 2-dimensional (Ullmann 2019), with one dimension measuring the depth of the reflection and the second dimension classifying the type (breadth) of the reflection. However, in Nehyba and Štefánik 2023, the authors flattened these two dimensions into one by introducing the *Reflection* category. In this research, we decided to stick with the original Ullmann's annotation. This decision prompted the removal of the *Reflection* category from the dataset. This step aimed to preserve the fundamental structure of the 2-dimensional taxonomy proposed by Ullmann, focusing on both the depth and breadth of reflection. By retaining this taxonomy without the *Reflection* class, we aim to classify various types of reflections present within the dataset, aligning with our overarching objective of classifying reflective writings.

Moreover, we coherently adjusted the original test dataset, renaming the category *Difficulties* to *Difficulty* to maintain consistency across the dataset attributes (see Figure 2.3).

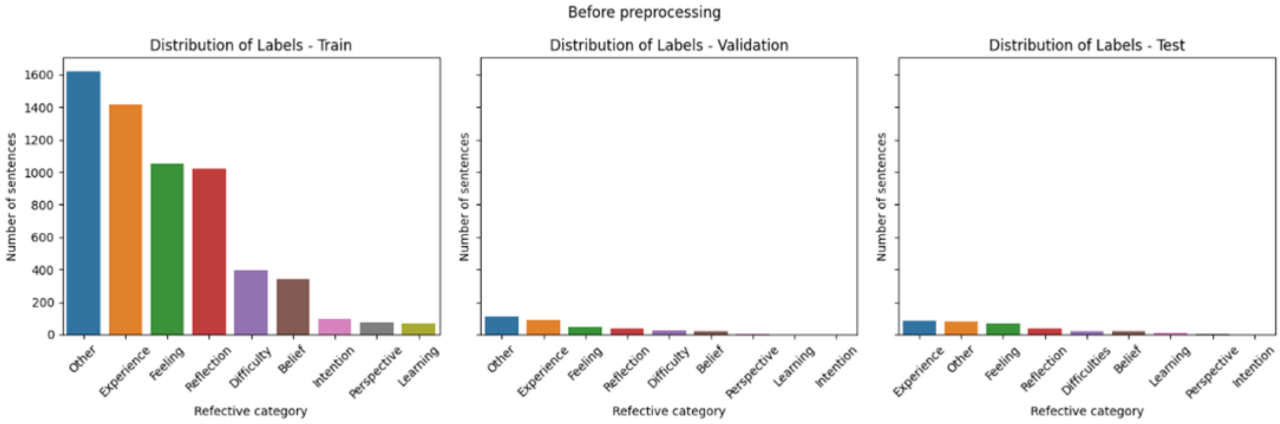


FIGURE 2.2
The 3 original datasets

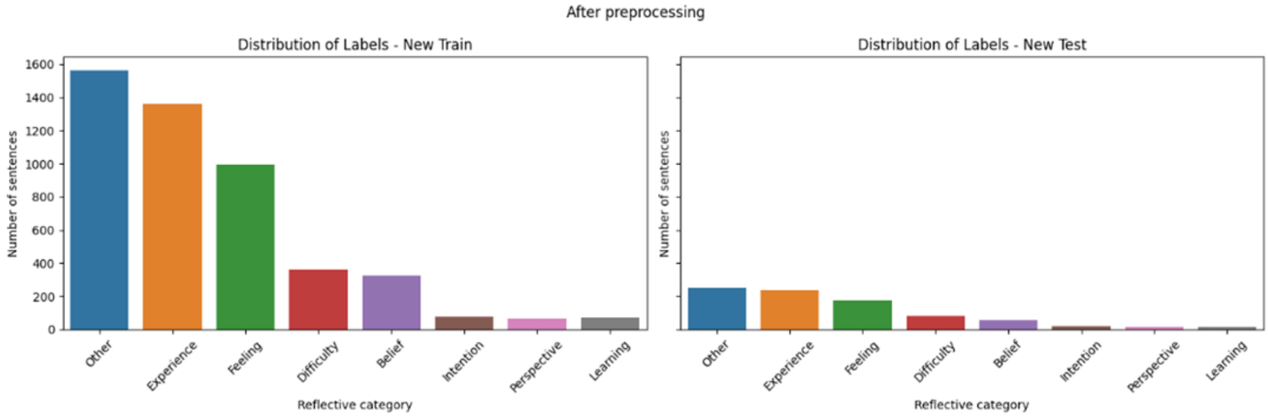


FIGURE 2.3
Train and Test datasets after merging without *Reflection* category

2.2 MULTI-CLASS CLASSIFICATION AND MULTIPLE BINARY CLASSIFICATIONS

Supervised learning's classification tasks involve predicting variables sorted into finite classes. When class counts exceed two, these tasks shift to multi-class classification. Real-world applications like recognizing handwritten digits or identifying faces fit into this classification domain.

To delve into classifications (see Figure 2.4), binary classification entails predicting between two classes—an instance belongs to one or the other. For instance, categorizing an animal image as a cat or dog falls under this category. Multiclass classification assigns each instance to a single class, while multi-label classification associates instances with multiple classes. In our context, we initially tackle multiclass text classification across various reflective categories.

Multi-class classification is generally more intricate compared to binary-class scenarios because the decision boundary tends to be more intricate. Some classification algorithms aren't equipped to handle multi-class problems directly, and other studies showcased the superiority of binarization strategies (Galar et al. 2011).

Binarization strategies, commonly using 'one-against-one' and 'one-against-rest' approaches, empower binary classifiers to effectively address multi-class problems. Even algorithms specialized in multi-class tasks benefit from these approaches, like the BERT models, as they simplify complex problems into more manageable binary sub-problems. The effectiveness stems partly from the ensemble effect created by multiple classifiers. In this project, we will leverage the performance of the 'one-against-rest' approach, which distinguishes between each class and all other classes individually. For instance, in a scenario with four classes (A, B, C, D), instead of predicting A, B, C, or D, the model makes separate predictions like "A vs. not A," "B vs. not B," and so on. Additionally, in some binarization methods, a cascaded classifier is employed, leveraging a hierarchical arrangement of the binary classifiers. Initially, they distinguish between one class and the rest or between pairs of classes. Misclassified samples proceed to subsequent levels for finer classification.

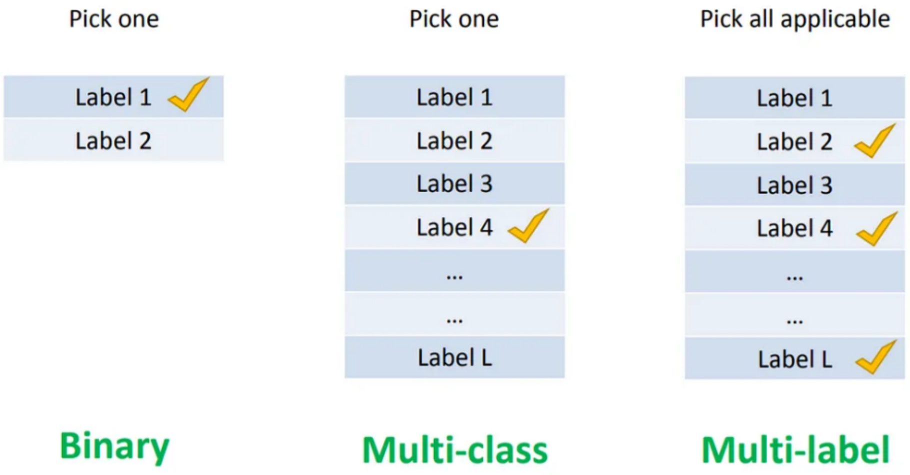


FIGURE 2.4
Differences between Binary, Multiclass and Multi-label Classification

2.3 MULTI-CLASS CLASSIFICATION ON IMBALANCED DATASETS AND ADAPTED EVALUATION METRICS

In numerous real-world scenarios, datasets constructed for multiclass classification frequently demonstrate imbalanced distributions across classes. This imbalance emerges when there's a significant discrepancy in the frequency of instances among various classes. This disproportionality can introduce hurdles during the training and assessment phases of multiclass classification models.

Imbalanced datasets often exhibit a scenario where one or multiple classes contain notably fewer instances compared to others. This disproportion can influence the model, causing it to favor the majority class and affecting its accuracy in predicting instances from minority classes.

There exists a wide range of metrics to handle imbalanced datasets. Below is a list of a few of them.

Confusion Matrix

The Confusion Matrix is a foundational tool for evaluating classification models. It provides insight into model performance by displaying True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN). These elements are crucial for computing various metrics like precision, recall, specificity, accuracy, and the AUC of a classification model.

Balanced Accuracy

Balanced Accuracy steps in as a more fitting metric in scenarios where datasets are imbalanced. Unlike standard accuracy, which assumes an equal distribution among classes, Balanced Accuracy considers the uneven distribution by computing the average recall across all classes.

In imbalanced datasets, accuracy might paint an inaccurate picture. For instance, a model achieving high accuracy might perform poorly on minority classes, overshadowed by strong performance on the majority class. Balanced Accuracy offers a more nuanced evaluation by accounting for the varying class sizes, providing a better understanding of overall classification performance in such situations.

F1 Score

The F1 Score is a composite metric that combines Precision and Recall into a single value, providing a balanced assessment of a model's performance, especially in scenarios where class distribution is imbalanced. It's calculated as the harmonic mean of Precision and Recall, offering a more holistic view than accuracy, especially when class distribution is unequal.

Precision focuses on the accuracy of positive predictions among all instances predicted as positive, while Recall measures the accuracy of positive predictions among all actual positive instances. The harmonic mean balances both Precision and Recall, giving an overall assessment of a model's ability to make precise and comprehensive predictions across all classes.

Area Under the Curve (AUC)

The Area Under the Curve, often represented as AUC-ROC (Receiver Operating Characteristic) or AUC-PR (Precision-Recall), assesses the overall performance of a model across various classification thresholds. AUC-ROC evaluates the True Positive Rate (TPR) against the False Positive Rate (FPR), showcasing the model's ability to discriminate between classes across different thresholds. A higher AUC-ROC indicates better model performance.

2.4 DOWNSAMPLING

As discussed in the previous section, the dataset imbalance can hinder the accuracy and reliability of machine learning models, leading to biased predictions skewed toward the majority classes.

Generally, rectifying the skewness among class labels involves augmenting the number of samples in underrepresented categories or reducing the instances in overrepresented ones. This process aims to mitigate the disparity between different classes, ensuring a more equitable representation and improving the performance of machine learning models in handling imbalanced datasets.

In particular, downsampling is a technique to reduce the number of instances in the overrepresented class or classes to align with the minority class. The process involves randomly removing instances from the majority class or classes until a more balanced distribution among classes is achieved.

The downsampling strategy has been implemented to be able to adjust the balance among the different classes by employing diverse downsampling thresholds, which are equal to the maximum number of samples from one class. The Figure 2.5 illustrates the outcomes across different scenarios resulting from these adjustments for our dataset.

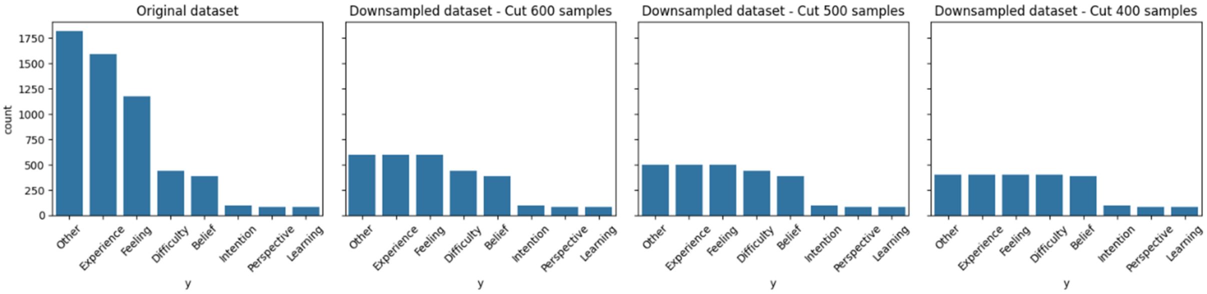


FIGURE 2.5
Distributions of downsampled datasets with different thresholds (600, 500, 400)

2.5 HYPERPARAMETER SEARCH

Hyperparameters in machine learning, especially in deep learning and models like transformers used for understanding language, are like settings or options that control how these models learn. They affect things like the structure of the model and how fast it learns. Finding the best settings for these hyperparameters is important because it can make the model work better. The learning rate, the number of training epochs, and the batch size are some examples of common hyperparameters.

There are various methods for discovering these optimal settings. One method involves **cross-validation**, where different configurations of the model are tested using sections of the data to select the most suitable one. **Grid search** involves experimenting with a predetermined set of various settings to determine the most effective one. **Random search**, akin to grid search, tests settings randomly. **Bayesian optimization** is more intricate—it leverages information from testing different settings to guide the search toward improved configurations. Each approach has its advantages and drawbacks, aiding in the selection of the most fitting settings for models applied in language and other machine learning domains.

In this research project, we will use N-fold cross-validation to look for the optimal hyperparameters.

2.6 Confidence Scores of BERT Predictions in the Context of Self-learning

In the context of text classification using BERT models, the confidence scores are essentially the probabilities assigned by the model to each potential class label for a given input text. These scores reflect the model's certainty or confidence in its predictions. BERT, like many neural network-based models, generates these scores as part of its prediction process, offering a probability distribution over the possible classes, indicating the likelihood of each class being the correct one for a given input.

In the self-learning framework discussed earlier, confidence scores play a crucial role. As the model generates pseudo-labels for unlabeled data, these confidence scores act as a measure of the model's certainty in its predictions. Leveraging these scores helps in selecting the most reliable predictions for inclusion in the labeled dataset for subsequent model training iterations. By considering and potentially filtering based on these scores, the self-learning loop aims to enhance the quality of the training data fed to the student model, thereby potentially improving the overall performance of the model over iterations.

The relevance of confidence scores lies in their potential correlation with prediction accuracy. Higher confidence scores typically indicate a higher certainty of the model in its predictions. Therefore, exploring the relationship between these scores and the actual accuracy of the predictions is critical. Understanding whether higher confidence scores align with more accurate predictions is crucial for evaluating the reliability of the model's confidence estimation and determining its utility in the self-learning process. If a strong correlation exists between high confidence scores and accurate predictions, it strengthens the justification for utilizing confidence scores as a criterion for pseudo-label selection in self-learning frameworks.

2.7 Learning Curves

Learning curves visualize the model's performance as a function of training data size. They showcase how the model's performance evolves concerning the amount of data it learns from, highlighting potential overfitting or underfitting and aiding in determining the sufficiency of data for optimal performance. It thus offers a visual representation (see Figure 2.6) of two critical aspects: the model's learning process from training data and its ability to generalize to new, unseen data.

In the context of self-learning, where models iteratively learn from labeled and pseudo-labeled data, learning curves provide valuable insights. They can assess the potential effectiveness of self-learning for various reflective categories. By observing the evolution of learning curves for each category, these curves can reveal whether self-learning, with its iterative approach and incorporation of pseudo-labels, could adequately leverage the unlabeled data to enhance model performance across diverse reflective categories.

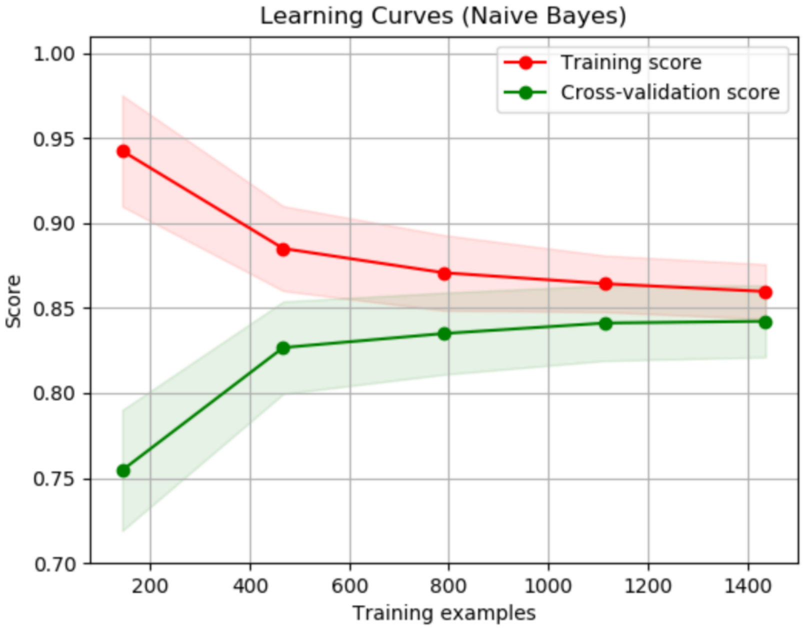


FIGURE 2.6
Learning curves showing training score and cross-validation score

CHAPTER 3

RESULTS AND DISCUSSION

3.1 BASELINE MULTI-CLASS CLASSIFICATION

In this paper, we used a BERT base model uncased (110M parameters) for computational reason, instead of the XLM-RoBERTa large model used by Nehyba and Štefánik 2023 (560M parameters). The section below unfolds the results obtained, which serve as a first reproducibility experiment of the work of Nehyba and Štefánik 2023

A data exploration analysis has been done before implementing the classification models. A short overview, which can be seen in Figure 3.1, highlights the imbalance of the reflective categories and shows that the sentences that will be classified have, on average, 20 words.

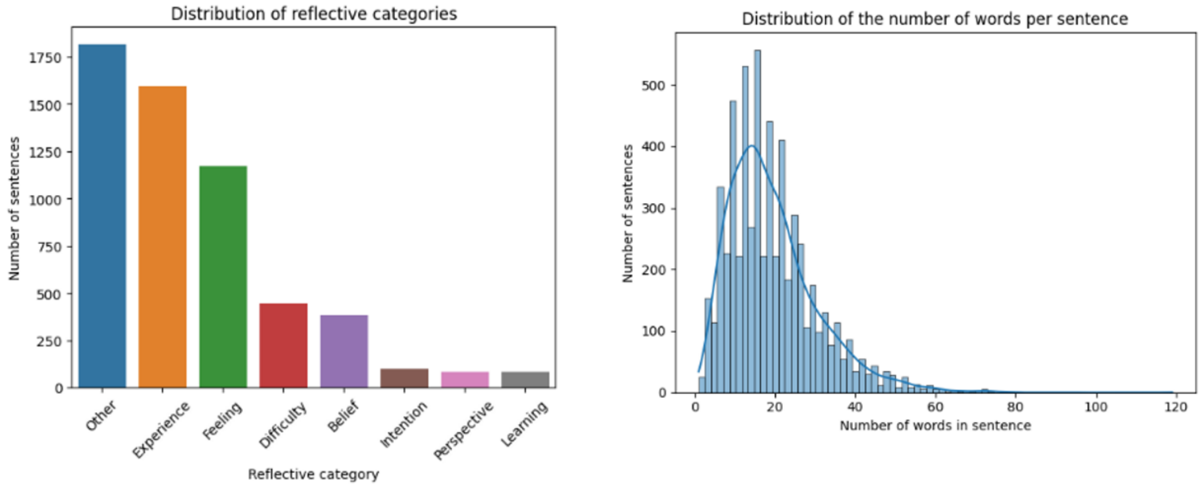


FIGURE 3.1
Left: Distribution of reflective categories in the merged dataset.
Right: Distribution of the number of words per sentence in the merged dataset.

We first performed a hyperparameter search (for the batch size and the number of epochs) with cross-validation for a batch size of 4, 8, and 16 and training on 1 to 10 epochs.

Looking at Figure 3.2, we opted to train the model for 3 epochs while maintaining batches of size 8 for subsequent training sessions, as this configuration demonstrated superior performance on this dataset. Additionally, using batches of size 8 enhances computational speed. A batch size of 8 could be beneficial as it balances computational efficiency and model stability. Moreover, selecting 3 epochs stemmed from observing that the training and validation loss plateaued after this period, indicating saturation in learning.

Looking at the evaluation with 3 epochs (Figure 3.3), we can see that the best accuracy is reached for a batch size of 8 (0.80), which is comparable to the results obtained by Nehyba and Štefánik 2023 (accuracy between 0.75 and 0.82 for thresholds of mean confidence of annotators of 3 and 4). Interestingly, we can see that for a batch size of 8, most of the predictions the model was confident about were actually correct. We can also see, as expected, that the predicted labels are heavily imbalanced, with mostly the categories *Experience*, *Feeling* and *Other* represented.

3.2 Baseline Multiple Binary Classifications

As explained in the *Methods* section, binarization strategies can be very promising compared to multi-class classification.

This section unfolds the results obtained by training 7 different binary classifiers (for each reflective category except *Other*) with the BERT base uncased model again and a batch size of 8 (resulting from the previous section). In each case, we preprocessed the original dataset by setting to *Other* all the labels that were not of the given category.

We first performed a hyperparameter search for the number of epochs, training each classifier on 1 to 10 epochs, see Figure 3.4. Looking at the train and validation losses, we can see that each classifier seems to learn in the first 3 epochs and overfits quite quickly (loss of 0 after 4-5 epochs). The losses for the reflective categories *Intention*, *Learning* and *Perspective* are very flat, which is also confirmed by the accuracy plots. Considering that the dataset is imbalanced, it makes more sense to look at the balanced accuracy for the under-represented classes, namely *Intention*, *Learning* and *Perspective*. For these latest, there's a clear indication that the models are acquiring knowledge, which contrasts with the earlier observations based on the loss and accuracy plots. Additionally, it's noteworthy that 4 epochs suffice for substantial learning to occur, which is a bit more than for the adequately represented classes.

As for the evaluation, we can see that only few classes (*Experience*, *Feeling* and *Difficulty*), which are adequately represented in the dataset, are well identified by the model, as shown by their confusion matrix as well as their F1-score (Figure 3.5). However, it is clear that the binary classifiers struggle to identify the positive labels of the under-represented classes.

3.2.1 Comparison Multi-class Classification - Multiple binary Classifications

Looking at Figure 3.6, one can note that the accuracy of the binary classifiers (0.95) is much better than the accuracy of the multi-class classifier (0.80). However, this metric is not representative of the situation considering the imbalance of the dataset. In fact, when we compare the balanced accuracies of the two methods, we can see that the multi-class classifier (0.79) actually performs better compared to the binary classifiers combined (0.71). More specifically, we can see that the binarization strategy works well for some classes (*Experience*, *Feeling* and *Difficulty*) which have a F1-score above 70%. This is probably due to the fact that those categories are adequately represented in the dataset. In fact, the other classes, even though they have a balanced accuracy of around 50%, have an F1-score very low and even null for the categories *Belief*, *Intention*, and *Perspective*. Acknowledging the potential of using a cascaded classifier to compare multi-class and binarization approaches, we aimed to combine binary classifiers hierarchically to mimic a multi-class classifier. However, the practicality of this approach was hindered by the performance of specific binary classifiers, particularly those dealing with under-represented categories.

In this case, it is hard to compare the multi-class classifier with the combination of binary classifiers. We thus chose to downsample the dataset and retrain binary classifiers.

3.3 MULTIPLE BINARY CLASSIFICATIONS WITH DOWNSAMPLING

We opted to downsample the original dataset to strike a balance between achieving a more balanced dataset and maximizing the number of samples available. We decided to downsample the original dataset with 500 samples maximum per category. The distribution of such decision can be seen in Figure 2.5.

With this configuration, we are looking for a more rigorous analysis of the results of binary classifiers on the under-represented reflective categories. This time, we trained 7 binary classifiers on 1 to 5 epochs to speed up the training because we saw previously that the loss and accuracy curves reached a plateau after 3-4 epochs for all categories.

Looking at the balanced accuracy curves of Figure 3.7, we can see that the binary classifiers manage to learn in 5 epochs. Specifically, 1-2 epochs are enough for the adequately represented categories (*Experience*, *Feeling*, *Difficulty* and *Belief*), whereas the under-represented categories (*Intention*, *Learning* and *Perspective*) need more epochs (3 or 4). The plots of Figure 3.7 clearly show that providing less examples could allow the model to learn better than with more examples, especially when the dataset is imbalanced.

The following section unfolds the results on the evaluation set and compare them with the results without downsampling.

3.3.1 COMPARISON MULTIPLE BINARY CLASSIFICATIONS WITH AND WITHOUT DOWNSAMPLING

Looking at Figure 3.8, we can see that the accuracy is pretty similar with and without downsampling (0,941 and 0,951 respectively). However, looking at the other evaluation metrics, namely the balanced accuracy, the F1-score, and the AUC, provides a totally different conclusion. In fact, with a downsampling at 500 samples maximum per class, we observe an increase of respectively the balanced accuracy, the F1-score and the AUC of **16%**, **31%** and **16%** respectively.

Specifically, as shown by the green arrows in Figure 3.8, the categories *Belief*, *Intention* and *Perspective* have seen a huge increase of their F1-score (+ 60-70%). It's important to highlight that the binary classifiers with downsampling notably surpass the multi-class classifier (balanced accuracy of 0.87 compared to 0.79). This showcases the significant potential of binarization strategies for the classification task of reflective writings within this dataset.

From these outcomes, it becomes clear that models can achieve better learning outcomes even with fewer samples, surpassing models trained on larger datasets, particularly in cases of imbalanced datasets. This leads us to the inquiry: How many training examples are truly necessary to acquire labels, and to what extent can self-learning contribute to labeling this dataset? The subsequent section aims to address this query using the learning curves approach.

3.4 LEARNING CURVES

For each reflective category (except *Other*), we trained the same BERT model as before by first splitting between a train set and a test set and then trained one model for each class, with an increasing number of training examples (500, 1000, 1500, 2000). At each step, the model reuses the exact same training examples of the previous step (if there are any) and adds 500 new training examples. This aims to simulate the self-learning training loop explained in the *Methods* section. At each step the model is evaluated on the test set. This whole process is repeated for N=10 shuffles to have robust results.

Looking at the results in Figure 3.9, we can see that the sufficient number of training examples, namely the number from which the loss and balanced accuracy reach a plateau, is different depending on the reflective category. For *Experience*, *Feeling*, *Difficulty*, *Belief* and *Intention*, the loss and balanced accuracy curves reach a plateau at 1500-2000 training examples, whereas no plateau is reached for the categories *Perspective* and *Learning*.

In the context of self-learning, it means that labeling 1500 examples for the first 5 categories mentioned might be enough, as the model won't learn more, even with additional examples. However, the categories *Perspective* and *Learning* will require to label more examples for the model to learn their features.

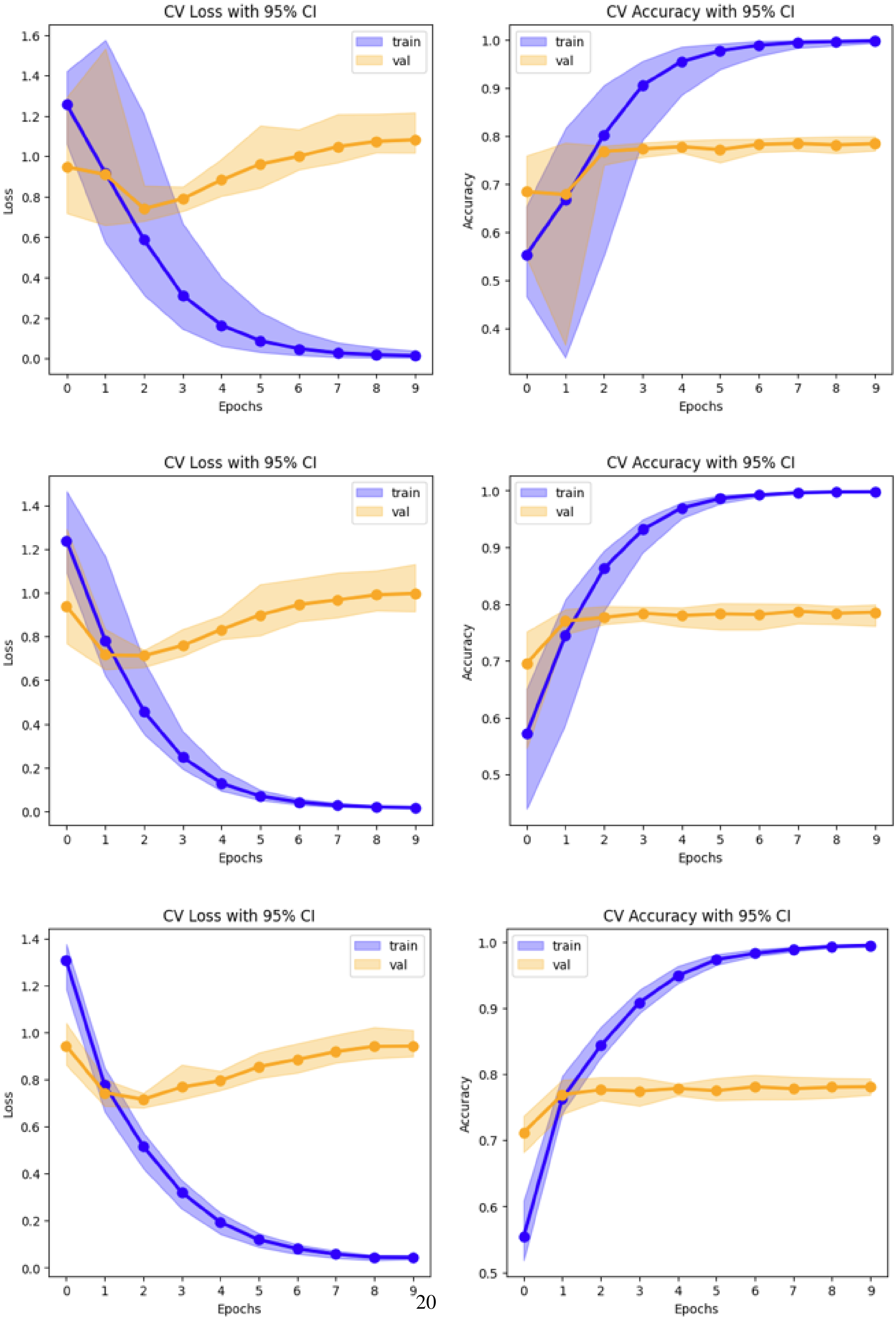
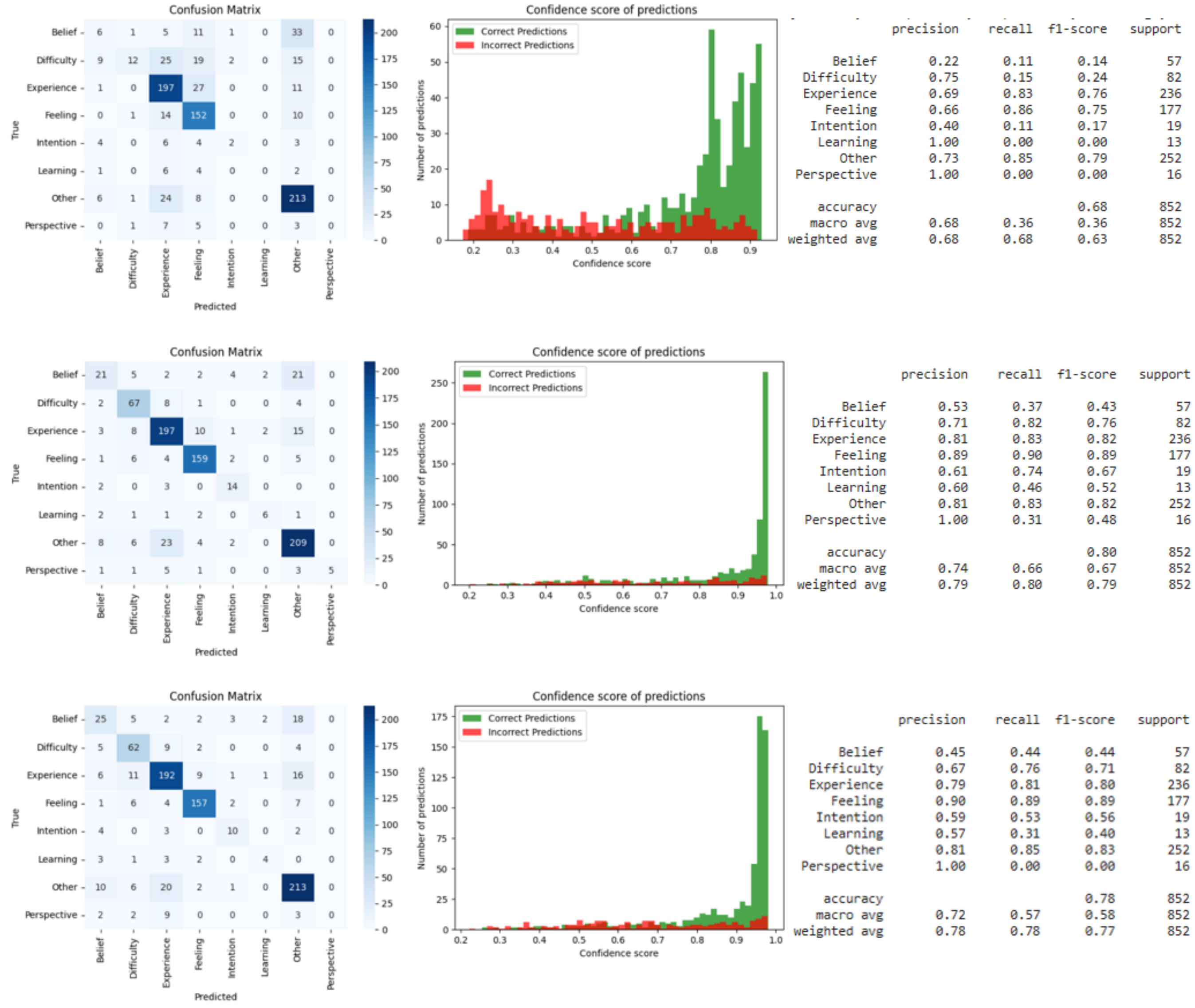


FIGURE 3.2
Multi-class classification with 5-fold cross-validation, 10 epochs, batch size of 4,8,16 (from top to bottom)



True \ Predicted	Belief	Difficulty	Experience	Feeling	Intention	Learning	Other	Perspective
Belief	6	1	5	11	1	0	33	0
Difficulty	9	12	25	19	2	0	15	0
Experience	1	0	197	27	0	0	11	0
Feeling	0	1	14	152	0	0	10	0
Intention	4	0	6	4	2	0	3	0
Learning	1	0	6	4	0	0	2	0
Other	6	1	24	8	0	0	213	0
Perspective	0	1	7	5	0	0	3	0

```
              precision    recall  f1-score   support

      Belief       0.22      0.11      0.14        57
  Difficulty       0.75      0.15      0.24        82
  Experience       0.69      0.83      0.76       236
     Feeling       0.66      0.86      0.75       177
   Intention       0.40      0.11      0.17        19
    Learning       1.00      0.00      0.00        13
       Other       0.73      0.85      0.79       252
 Perspective       1.00      0.00      0.00        16

    accuracy                           0.68       852
   macro avg       0.68      0.36      0.36       852
weighted avg       0.68      0.68      0.63       852
```

True \ Predicted	Belief	Difficulty	Experience	Feeling	Intention	Learning	Other	Perspective
Belief	21	5	2	2	4	2	21	0
Difficulty	2	67	8	1	0	0	4	0
Experience	3	8	197	10	1	2	15	0
Feeling	1	6	4	159	2	0	5	0
Intention	2	0	3	0	14	0	0	0
Learning	2	1	1	2	0	6	1	0
Other	8	6	23	4	2	0	209	0
Perspective	1	1	5	1	0	0	3	5

```
              precision    recall  f1-score   support

      Belief       0.53      0.37      0.43        57
  Difficulty       0.71      0.82      0.76        82
  Experience       0.81      0.83      0.82       236
     Feeling       0.89      0.90      0.89       177
   Intention       0.61      0.74      0.67        19
    Learning       0.60      0.46      0.52        13
       Other       0.81      0.83      0.82       252
 Perspective       1.00      0.31      0.48        16

    accuracy                           0.80       852
   macro avg       0.74      0.66      0.67       852
weighted avg       0.79      0.80      0.79       852
```

True \ Predicted	Belief	Difficulty	Experience	Feeling	Intention	Learning	Other	Perspective
Belief	25	5	2	2	3	2	18	0
Difficulty	5	62	9	2	0	0	4	0
Experience	6	11	192	9	1	1	16	0
Feeling	1	6	4	157	2	0	7	0
Intention	4	0	3	0	10	0	2	0
Learning	3	1	3	2	0	4	0	0
Other	10	6	20	2	1	0	213	0
Perspective	2	2	9	0	0	0	3	0

```
              precision    recall  f1-score   support

      Belief       0.45      0.44      0.44        57
  Difficulty       0.67      0.76      0.71        82
  Experience       0.79      0.81      0.80       236
     Feeling       0.90      0.89      0.89       177
   Intention       0.59      0.53      0.56        19
    Learning       0.57      0.31      0.40        13
       Other       0.81      0.85      0.83       252
 Perspective       1.00      0.00      0.00        16

    accuracy                           0.78       852
   macro avg       0.72      0.57      0.58       852
weighted avg       0.78      0.78      0.77       852
```

FIGURE 3.3
Multi-class classification evaluated on test dataset with 3 epochs, batch size of 4,8,16 (from top to bottom)

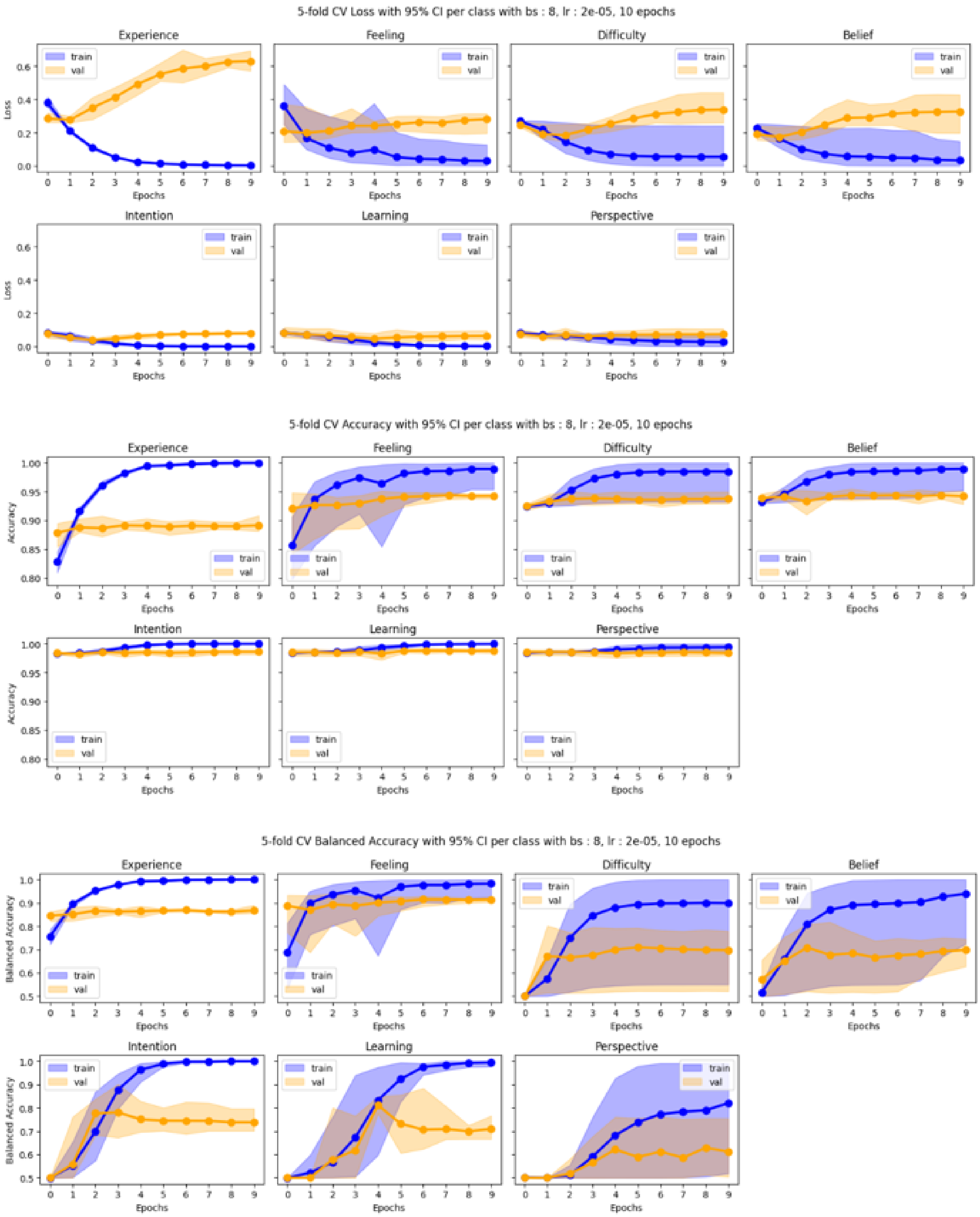


FIGURE 3.4
Multiple binary classifications with 5-fold cross-validation, 10 epochs, batch size 8

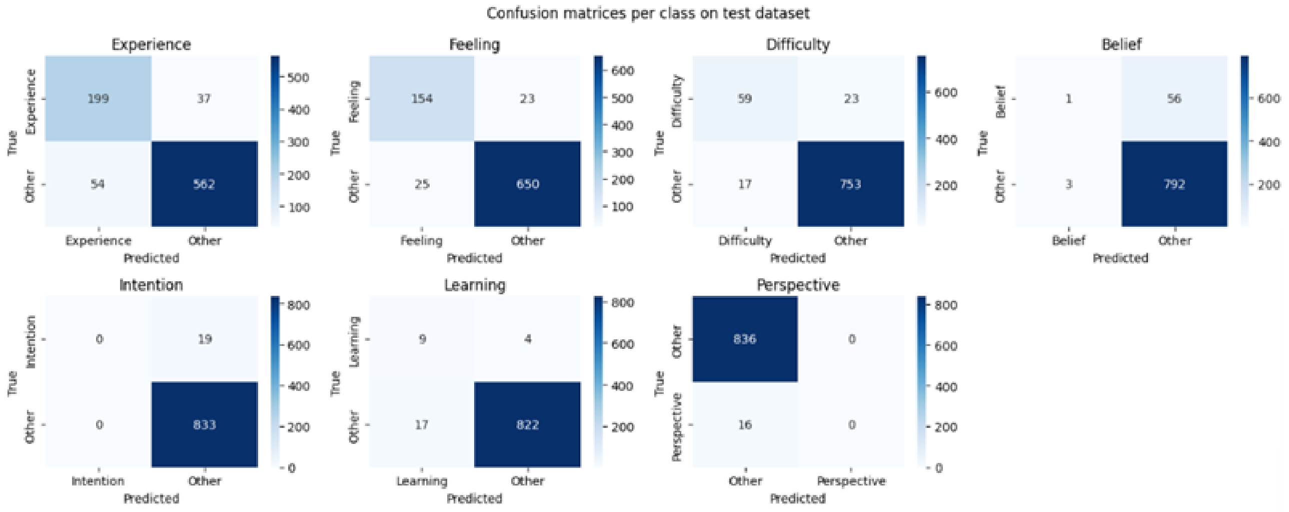
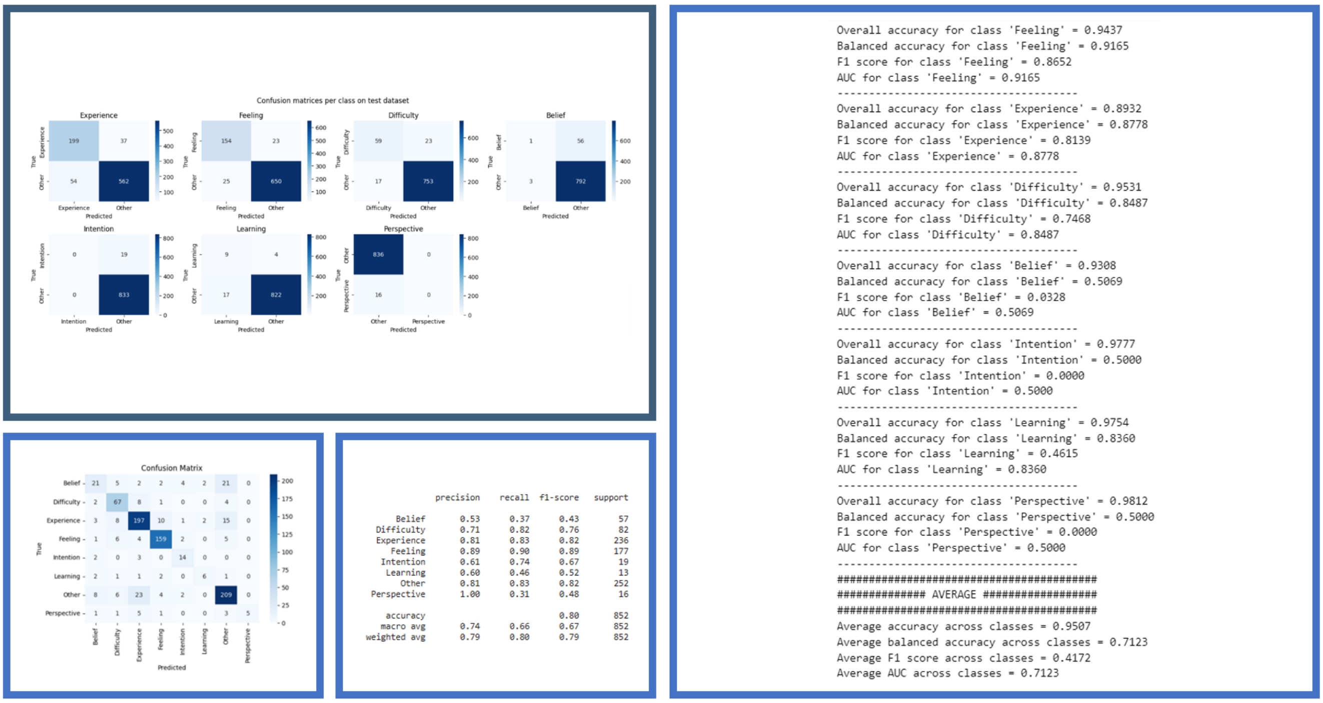


FIGURE 3.5
Multiple binary classifications evaluated on test dataset with 3 epochs



Confusion Matrix

True \ Predicted	Belief	Difficulty	Experience	Feeling	Intention	Learning	Other	Perspective
Belief	21	5	2	2	4	2	21	0
Difficulty	2	67	8	1	0	0	4	0
Experience	3	8	197	10	1	2	15	0
Feeling	1	6	4	159	2	0	5	0
Intention	2	0	3	0	14	0	0	0
Learning	2	1	1	2	0	6	1	0
Other	8	6	23	4	2	0	209	0
Perspective	1	1	5	1	0	0	3	5

```
              precision    recall  f1-score   support

      Belief       0.53      0.37      0.43        57
  Difficulty       0.71      0.82      0.76        82
  Experience       0.81      0.83      0.82       236
     Feeling       0.89      0.90      0.89       177
   Intention       0.61      0.74      0.67        19
    Learning       0.60      0.46      0.52        13
       Other       0.81      0.83      0.82       252
 Perspective       1.00      0.31      0.48        16

    accuracy                           0.80       852
   macro avg       0.74      0.66      0.67       852
weighted avg       0.79      0.80      0.79       852
```

```
Overall accuracy for class 'Feeling' = 0.9437
Balanced accuracy for class 'Feeling' = 0.9165
F1 score for class 'Feeling' = 0.8652
AUC for class 'Feeling' = 0.9165
----------------------------------------
Overall accuracy for class 'Experience' = 0.8932
Balanced accuracy for class 'Experience' = 0.8778
F1 score for class 'Experience' = 0.8139
AUC for class 'Experience' = 0.8778
----------------------------------------
Overall accuracy for class 'Difficulty' = 0.9531
Balanced accuracy for class 'Difficulty' = 0.8487
F1 score for class 'Difficulty' = 0.7468
AUC for class 'Difficulty' = 0.8487
----------------------------------------
Overall accuracy for class 'Belief' = 0.9308
Balanced accuracy for class 'Belief' = 0.5069
F1 score for class 'Belief' = 0.0328
AUC for class 'Belief' = 0.5069
----------------------------------------
Overall accuracy for class 'Intention' = 0.9777
Balanced accuracy for class 'Intention' = 0.5000
F1 score for class 'Intention' = 0.0000
AUC for class 'Intention' = 0.5000
----------------------------------------
Overall accuracy for class 'Learning' = 0.9754
Balanced accuracy for class 'Learning' = 0.8360
F1 score for class 'Learning' = 0.4615
AUC for class 'Learning' = 0.8360
----------------------------------------
Overall accuracy for class 'Perspective' = 0.9812
Balanced accuracy for class 'Perspective' = 0.5000
F1 score for class 'Perspective' = 0.0000
AUC for class 'Perspective' = 0.5000
----------------------------------------
##########################################
############## AVERAGE ###################
##########################################
Average accuracy across classes = 0.9507
Average balanced accuracy across classes = 0.7123
Average F1 score across classes = 0.4172
Average AUC across classes = 0.7123
```

FIGURE 3.6
Comparison between Multi-class classification and Multiple binary classifications both evaluated on test dataset with 3 epochs

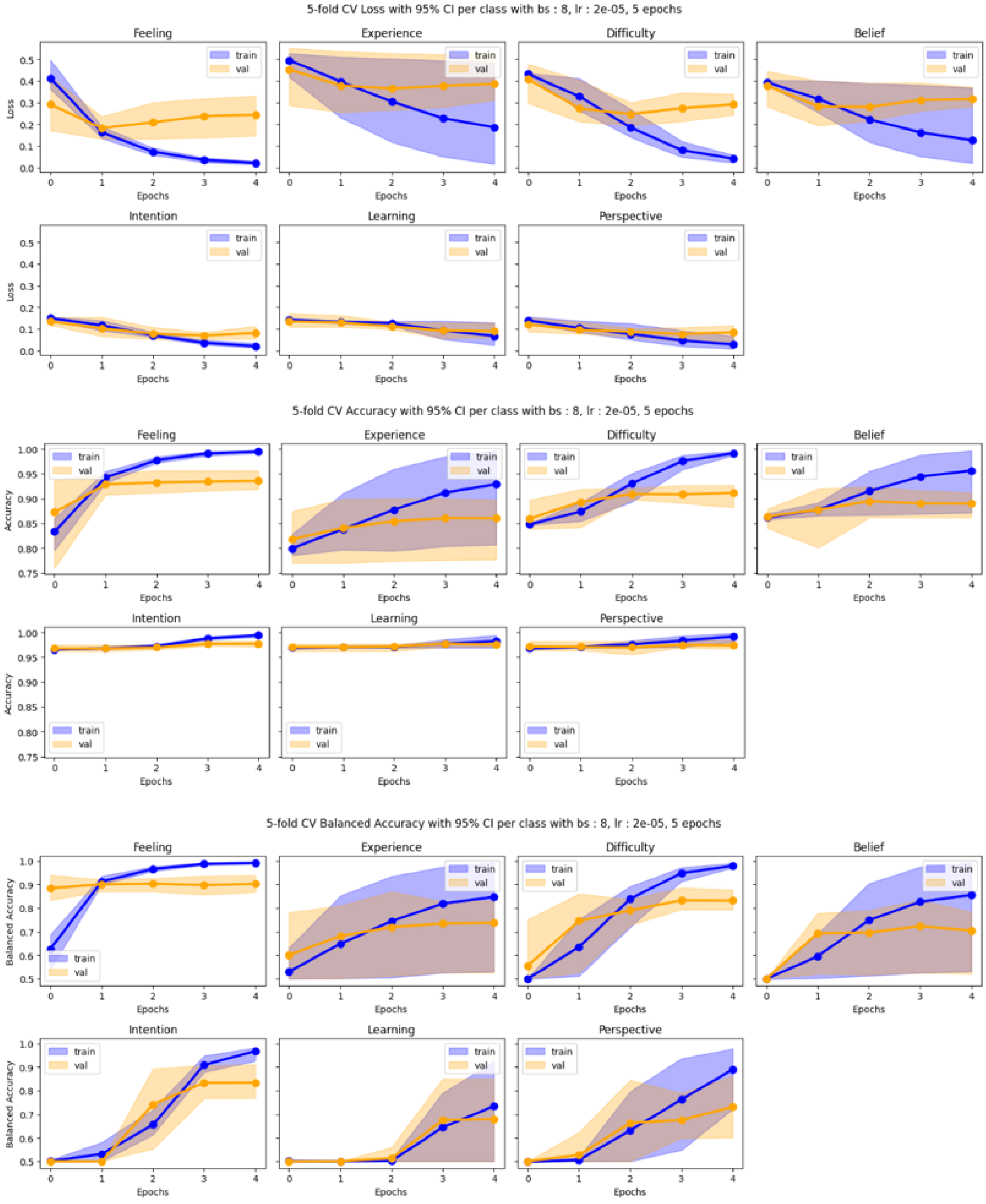
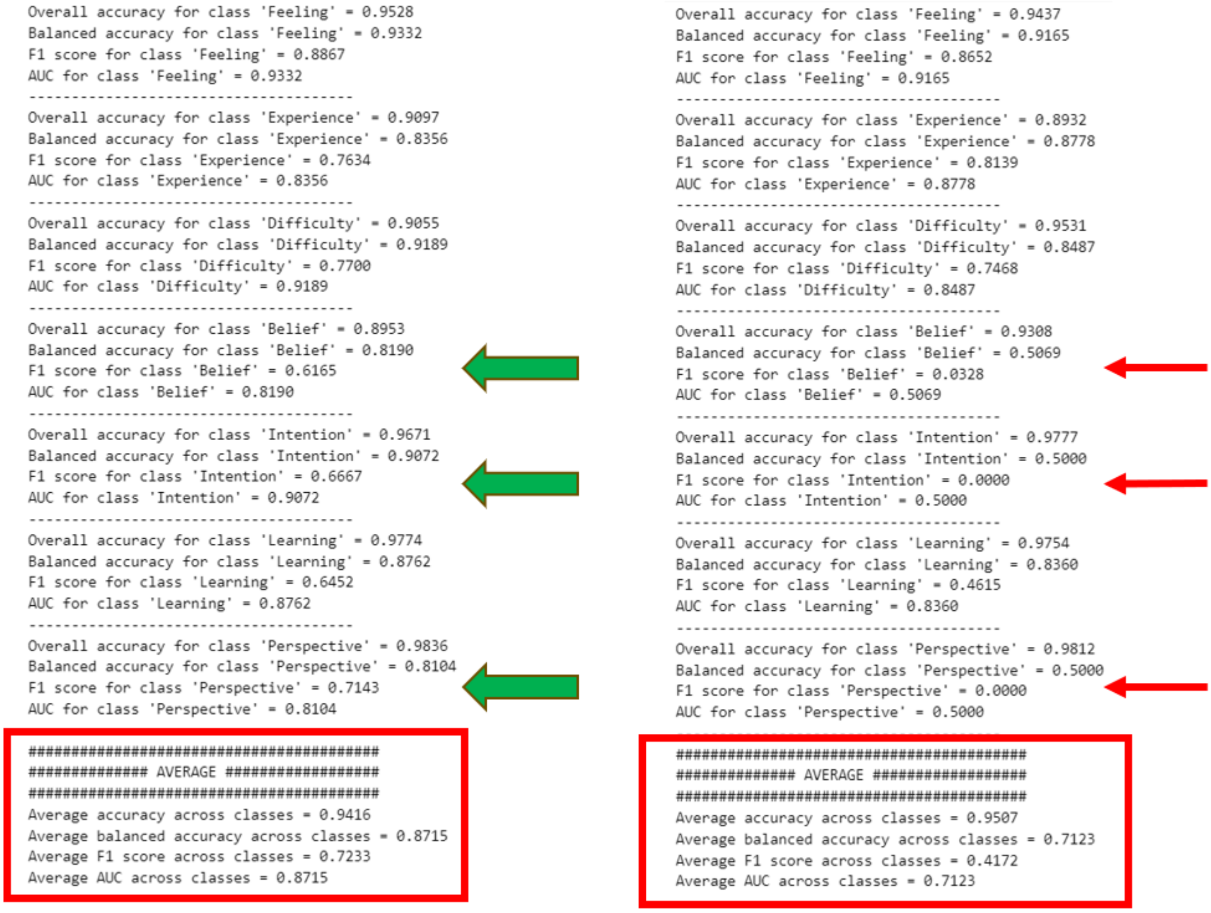


FIGURE 3.7

Multiple binary classifications on downsampled dataset with 5-fold cross-validation, 5 epochs, batch size 8



```
Overall accuracy for class 'Feeling' = 0.9528
Balanced accuracy for class 'Feeling' = 0.9332
F1 score for class 'Feeling' = 0.8867
AUC for class 'Feeling' = 0.9332
--------------------------------------
Overall accuracy for class 'Experience' = 0.9097
Balanced accuracy for class 'Experience' = 0.8356
F1 score for class 'Experience' = 0.7634
AUC for class 'Experience' = 0.8356
--------------------------------------
Overall accuracy for class 'Difficulty' = 0.9055
Balanced accuracy for class 'Difficulty' = 0.9189
F1 score for class 'Difficulty' = 0.7700
AUC for class 'Difficulty' = 0.9189
--------------------------------------
Overall accuracy for class 'Belief' = 0.8953
Balanced accuracy for class 'Belief' = 0.8190
F1 score for class 'Belief' = 0.6165
AUC for class 'Belief' = 0.8190
--------------------------------------
Overall accuracy for class 'Intention' = 0.9671
Balanced accuracy for class 'Intention' = 0.9072
F1 score for class 'Intention' = 0.6667
AUC for class 'Intention' = 0.9072
--------------------------------------
Overall accuracy for class 'Learning' = 0.9774
Balanced accuracy for class 'Learning' = 0.8762
F1 score for class 'Learning' = 0.6452
AUC for class 'Learning' = 0.8762
--------------------------------------
Overall accuracy for class 'Perspective' = 0.9836
Balanced accuracy for class 'Perspective' = 0.8104
F1 score for class 'Perspective' = 0.7143
AUC for class 'Perspective' = 0.8104
##########################################
############# AVERAGE #################
##########################################
Average accuracy across classes = 0.9416
Average balanced accuracy across classes = 0.8715
Average F1 score across classes = 0.7233
Average AUC across classes = 0.8715
```

```
Overall accuracy for class 'Feeling' = 0.9437
Balanced accuracy for class 'Feeling' = 0.9165
F1 score for class 'Feeling' = 0.8652
AUC for class 'Feeling' = 0.9165
--------------------------------------
Overall accuracy for class 'Experience' = 0.8932
Balanced accuracy for class 'Experience' = 0.8778
F1 score for class 'Experience' = 0.8139
AUC for class 'Experience' = 0.8778
--------------------------------------
Overall accuracy for class 'Difficulty' = 0.9531
Balanced accuracy for class 'Difficulty' = 0.8487
F1 score for class 'Difficulty' = 0.7468
AUC for class 'Difficulty' = 0.8487
--------------------------------------
Overall accuracy for class 'Belief' = 0.9308
Balanced accuracy for class 'Belief' = 0.5069
F1 score for class 'Belief' = 0.0328
AUC for class 'Belief' = 0.5069
--------------------------------------
Overall accuracy for class 'Intention' = 0.9777
Balanced accuracy for class 'Intention' = 0.5000
F1 score for class 'Intention' = 0.0000
AUC for class 'Intention' = 0.5000
--------------------------------------
Overall accuracy for class 'Learning' = 0.9754
Balanced accuracy for class 'Learning' = 0.8360
F1 score for class 'Learning' = 0.4615
AUC for class 'Learning' = 0.8360
--------------------------------------
Overall accuracy for class 'Perspective' = 0.9812
Balanced accuracy for class 'Perspective' = 0.5000
F1 score for class 'Perspective' = 0.0000
AUC for class 'Perspective' = 0.5000
##########################################
############# AVERAGE #################
##########################################
Average accuracy across classes = 0.9507
Average balanced accuracy across classes = 0.7123
Average F1 score across classes = 0.4172
Average AUC across classes = 0.7123
```

FIGURE 3.8

Comparison Multiple Binary Classifications With (left) - Without (right) Downsampling on test dataset with 3 epochs, 8 batches

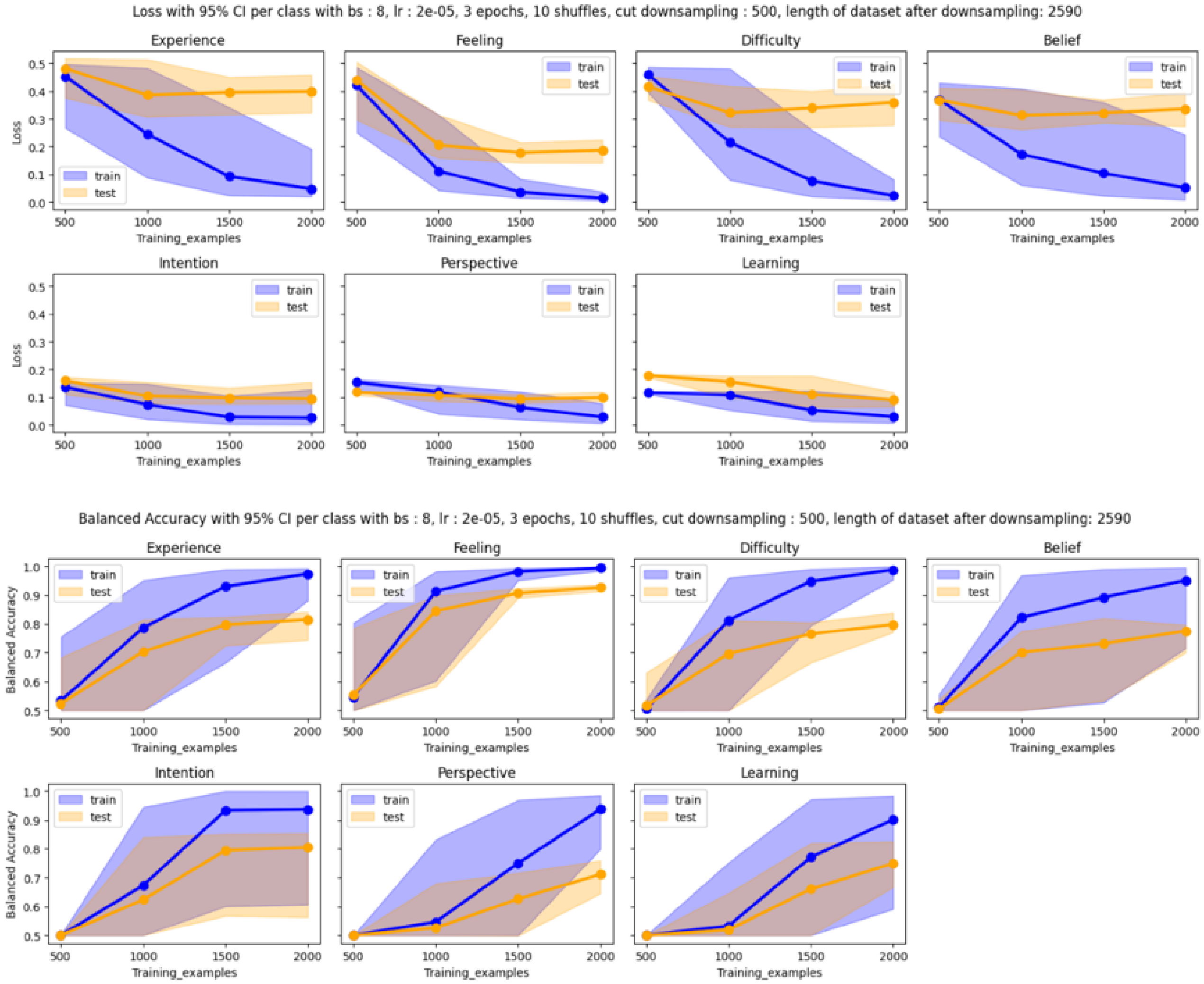


FIGURE 3.9
Learning curves for each reflective category with 10 shuffles
Top: Train and Test losses
Bottom: Train and Test Balanced Accuracies

Chapter 4

Conclusion

This study sets out to explore how models could learn from limited labeled dataset in the context of reflective writings through self-learning techniques. Leveraging a BERT base model, the experiment aimed to replicate previous work by Nehyba and Štefánik 2023, affirming similar outcomes under different computational conditions. Notably, training the model for three epochs with a batch size of eight demonstrated optimal performance, balancing computational efficiency with stability. The assessment revealed a satisfying accuracy (80%), especially in predicting over-represented categories.

Comparing multi-class and binary classifiers revealed favorable results for the latter in specific categories, mainly due to their higher representation in the dataset. However, less represented classes faced difficulties, despite improved performance in other categories by the binary classifiers. Adjusting the distribution of the classes in the dataset through downsampling notably increased the binary classifiers' performance, particularly in enhancing the F1-score for categories like Belief, Intention, and Perspective. This emphasizes the potential of binary strategies in effectively classifying reflective writings, surpassing the performance of multi-class classifiers.

Furthermore, exploring learning curves showcased varying optimal training examples across different reflective categories. Some categories like Experience, Feeling, Difficulty, Belief, and Intention reached a learning plateau of around 1500-2000 examples, while Perspective and Learning demanded additional labeled instances.

Future endeavors might involve tailoring the approach for each category. This could include determining the optimal number of epochs, refining the split of training examples, and fine-tuning the learning rate to suit the nuances of each category. Augmenting the dataset, especially for underrepresented categories, emerges as a critical aspect for enhancing model performance. Exploring diverse data augmentation strategies to generate additional labels could be pivotal. Techniques like synthetic data generation, transfer learning from related domains, or using samples with pseudo-labels obtained from a teacher model trained on small labeled datasets may significantly improve the model's understanding of these less-represented categories.

In conclusion, this study highlights the potential of models learning from limited samples, especially beneficial in datasets with imbalanced representations. It prompts further investigation into the minimum required training examples and the potential of self-learning for data annotation in educational contexts.

Bibliography

Ullmann, Thomas Daniel (2019). ‘Automated analysis of reflection in writing: Validating machine learning approaches’. In: *International Journal of Artificial Intelligence in Education* 29.2, pp. 217–257.

Krol, Christine A (1996). ‘Preservice Teacher Education Students’ Dialogue Journals: What Characterizes Students’ Reflective Writing and a Teacher’s Comments.’ In.

Holdan, E Gregory (2009). ‘Using On-Line Discussion to Encourage Reflective Thinking’. In.

Maloney, Carmel and Glenda Campbell-Evans (2002). ‘Using interactive journal writing as a strategy for professional growth’. In: *Asia-Pacific Journal of Teacher Education* 30.1, pp. 39–50.

Wallin, Patric and Tom Adawi (2018). ‘The reflective diary as a method for the formative assessment of self-regulated learning’. In: *European Journal of Engineering Education* 43.4, pp. 507–521.

Buckingham Shum, Simon et al. (2016). ‘Reflecting on reflective writing analytics: Assessment challenges and iterative evaluation of a prototype tool’. In: *Proceedings of the sixth international conference on learning analytics & knowledge*, pp. 213–222.

Sumsion, Jennifer and Alma Fleet (1996). ‘Reflection: can we assess it? Should we assess it?’ In: *Assessment & Evaluation in Higher Education* 21.2, pp. 121–130.

Ullmann, Thomas Daniel, Fridolin Wild and Peter Scott (2012). ‘Comparing automatically detected reflective texts with human judgements’. In: *CEUR Workshop Proceedings*.

Ullmann, Thomas Daniel (2015). *Automated detection of reflection in texts: A machine learning based approach*. Open University (United Kingdom).

Devlin, Jacob et al. (2018). ‘Bert: Pre-training of deep bidirectional transformers for language understanding’. In: *arXiv preprint arXiv:1810.04805*.

Xie, Qizhe et al. (2020). ‘Self-training with noisy student improves imagenet classification’. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10687–10698.

Liu, Chengyuan et al. (2022). ‘Improving Problem Detection in Peer Assessment through Pseudo-Labeling Using Semi-Supervised Learning.’ In: *International Educational Data Mining Society*.

Nehyba, Jan and Michal Štefánik (2023). ‘Applications of deep language models for reflective writings’. In: *Education and Information Technologies* 28.3, pp. 2961–2999.

Galar, Mikel et al. (2011). ‘An overview of ensemble methods for binary classifiers in multi-class problems: Experimental study on one-vs-one and one-vs-all schemes’. In: *Pattern Recognition* 44.8, pp. 1761–1776.