Article

Automated Assessment of Parent-Child Interaction Quality from Dyadic Dialogue

Chaohao Lin [1,‡,*], Ou Bai, PhD [1,‡] and Jennifer Piscitello, PhD [2,‡], Emily L. Robertson, PhD[2,‡], Kellina Lupas, PhD[2,†], Williame E. Pelham Jr., PhD[2]

[1] Department of Electrical & Computer Engineering, Florida International University, Miami, USA; eceinfo@fiu.edu
[2] Center for Children and Families, Florida International University, Miami, USA; ccf@fiu.edu
* Correspondence: clin@fiu.edu
† Current address: Division of Behavioral Medicine and Clinical Psychology, Cincinnati Children's Hospital, Winslow, USA.
‡ These authors contributed equally to this work.

Abstract: The quality of parent-child interaction is critical for child cognitive development. The Dyadic Parent-Child Interaction Coding System (DPICS) is commonly used to assess parent and child behaviors. However, manual annotation of DPICS codes by parent-child interaction therapists is a time-consuming task. To assist therapists in the coding task, researchers have begun to explore the use of artificial intelligence in natural language processing to classify DPICS codes automatically. In this study, we utilized datasets from the DPICS book manual, five families, and an open-source PCIT dataset. To train DPICS code classifiers, we employed the pre-trained fine-tune model Roberta as our learning algorithm. Our study shows that fine-tuning the pre-trained RoBERTa model achieves the highest results compared to other methods in sentence-based DPICS code classification assignments.For the DPICS manual dataset, the overall accuracy was 72.3% (72.2% macro-precision, 70.5% macro-recall, and 69.6% macro F-score). Meanwhile, for the PCIT dataset, the overall accuracy was 79.8% (80.4% macro-precision, 79.7% macro-recall, and 79.8% macro F-score), surpassing the previous highest results of 78.3% accuracy (79% precision, 77% recall) averaged over the eight DPICS classes. These results presented that fine-tuning the pre-trained RoBERTa model could provide valuable assistance to experts in the labeling process.

Keywords: Parent-Child Interaction; DPICS; Text Classification; Natural Language Processing(NLP); Transformers; Artificial Intelligence

1. Introduction

The quality of parent-child interaction (PCI) has a critical influence on child cognitive and socio-emotional development [24,33]. Parent-Child Interaction Therapy(PCIT) is a treatment designed to help parents of children with early behavior problems to improve their relationship with their child and to manage their child's behavior more effectively [15]. PCIT is associated with positive benefits for children and families, including reduced child behavior problems and family stress [37,47]. The Dyadic Parent–Child Interaction Coding System was developed for purposes of treatment monitoring and has been widely used for the assessment of PCI and is utilized in PCIT to assess treatment progress, which is typically coded manually by a trained therapist or research staff [42]. This can be problematic, as time spent training to code to fidelity is costly. Additionally, if large amounts of data are being collected, time spent coding can delay the research process significantly.

Artificial intelligence is a new trend, driven by the rapid development of machine learning and deep learning. The goal of artificial intelligence is to create intelligent agents that are capable of completing tasks in a manner similar to humans. State-of-the-art results and superhuman achievements have been attained in many fields, including AlphaGo in Go

Citation: Lastname, F.; Lastname, F.; Lastname, F. Title. *Journal Not Specified* **2023**, *1*, 0. https://doi.org/

Received:
Revised:
Accepted:
Published:

game, Atlas of Boston Dynamics in whole-body robots, and recent conversational dialogue agent ChatGPT. In the field of natural language processing, pre-trained autoregressive deep learning language models such as BERT and GPT have become increasingly popular [5,12]. Giving computers the ability to understand human language has long been a goal of artificial intelligence in natural language processing, and pre-trained models are fed massive raw documents in the hopes of identifying relationships among words or sentences.

Labeling DPICS codes is a tedious and time-consuming task for experts and therapists. For assisting PCIT Therapists, Huber et al. proposed the SpecialTime system for providing parents with feedback during their at-home practice of PCIT skills [23]. The developed SpecialTime system can automatically classify child-directed dialogue acts into the eight DPICS classes. Based on the SpecialTime system, we developed and implemented sentence-based classifiers to improve the DPICS code classification results and extend eight DPICS code classes to ten DPICS code classes which include Unlabeled Praise(UP), Labeled Praise(LP), Reflection(RF), Behavior Description(BD), Information Question(IQ), Descriptive Question(DQ), Indirect Commands(IC), Direct Commands(DC), Negative Talk(NTA) and Neutral Talk(TA) following the instruments of the Dyadic Parent-Child Interaction Coding System (DPICS) Comprehensive Manual for Research and Training, Fourth Edition(DPICS-IV)[42]. To do that, we first collected three section datasets include 1753 instances provided by DPICS-IV manual, a total of 1952 utterances from five families and a PCIT dataset containing 6021 utterances provided by [23]. For comparison, we also deploy typical and popular text feature extraction methods like Word of Bag, Term Frequency-Inverse Document Frequency, and Global Vectors for Word Representation. Additionally, machine learning approaches such as logistic regression, support vector machine, and XGBoost tree were compared as downstream classifiers. In terms of pre-trained deep learning models, we utilized and fine-tuned two variants of BERT: DistilBERT and RoBERTa.

After comparing different text representations and machine learning methods, we found that Roberta outperformed other methods in our datasets. Especially, Roberta's performance surpassed the previous best results in the public PCIT dataset.

In summary, we make the following contributions:

- Introduce the state-of-the-art pre-train language models, DistilBERT and Roberta, as deep learning approaches for automatically classifying DPICS codes, which have not been deployed in generating DPICS code classifiers before;
- The results of our study demonstrate that the use of pre-trained language models, such as DistilBERT and Roberta, can significantly improve the accuracy of DPICS code classification compared to traditional text feature extraction methods and machine learning approaches. In particular, fine-tuning the Roberta model achieved the highest accuracy and outperformed other machine learning models. An advantage of pre-trained language models is that they can handle raw data without the need for extensive feature engineering. Additionally, we extended the classification to ten DPICS codes, in contrast to previous studies that used only eight classes;
- Pre-trained language models offer a powerful tool for transfer learning. By using models trained on larger unrelated datasets, reasonable results can be achieved when transferring learning from one task to another. In our work, different families' communication was evaluated, and an overall accuracy of 71.0% was achieved across five families. These results demonstrate the potential of pre-trained language models for improving our understanding of communication patterns and behavior;
- Boosting the performance of pre-trained language models can be achieved by training on a wider range of data, such as subject-independent data, leading to more robust and accurate sentence-based classifiers. While sentence-based classifiers have achieved acceptable performance, context-based classifiers should also be considered in future work to enhance performance. Specifically, context-based classifiers could help to capture the nuances and complexities of natural language use.

2. Related Work

2.1. Text Feature Extraction

2.1.1. Text Representation

When working with text in machine learning models, we need to convert the text into numerical vectors so that the models can process it. Two common methods for achieving this are one-hot encoding and integer encoding. One-hot encoding creates a vector with a length equal to the size of the vocabulary and places a "1" in the index that corresponds to the word. This approach is inefficient because most values in the resulting vector are zero. In contrast, integer encoding assigns a unique integer value to each word. While this approach creates a dense vector that can be more efficient for machine learning models, it doesn't capture any relationships between the words, meaning that there is no inherent similarity between the encoded values of two words. For example, the integer values assigned to "he" and "she" have no relationship to each other, despite their semantic similarity. This limitation can be problematic for certain natural language processing tasks, where understanding the relationships between words is critical.

Apart from one-hot encoding and unique numbers, previous techniques such as Bag of Words (BoW) and Term Frequency-Inverse Document Frequency (TF-IDF) have been used for converting the text to numerical vectors [21,28]. BoW and TF-IDF are both statistical measurement methods. There are also several variants, such as n-gram models and smoothed variants of TF-IDF.

Bag of Words

Bag of words (BoW) is a widely used text representation technique in natural language processing (NLP). It involves converting a piece of text into a collection of individual words or terms, along with their respective frequencies[21]. To create a BoW model, the text is pre-processed to remove stopwords and punctuation. Each word in the preprocessed text is then tokenized and counted, resulting in a dictionary of unique words and their respective frequencies. Finally, the text is represented as a vector with a length equal to the size of the dictionary. Despite its widespread use, BoW has several limitations. First, BoW disregards the order and context of the words in the text, which can result in the loss of important information about the meaning and context of the words. Second, the vocabulary size can be very large, resulting in a high-dimensional vector space that can be computationally expensive and require too much memory. Third, stopwords, which are common words like "the" and "a", can dominate the frequency count and mislead the model. Finally, most documents only contain a small subset of the words in the vocabulary, resulting in sparse vectors that can make it difficult to compare documents or compute similarity measures.

Although BoW has some weaknesses, BoW is a widespread and effective technique for tasks such as text classification or sentiment analysis especially when combined with other techniques like feature selection and dimensionality reduction. [14,19,22,49].

Term Frequency-Inverse Document Frequency

Term Frequency-Inverse Document Frequency (TF-IDF) is a statistical measure used to find the relevance of words in a text document or corpus. TF-IDF is often used as a weighting factor in searches for information retrieval, text mining, and user modeling [3].

TF-IDF is composed of two metrics: term frequency (TF) and inverse document frequency (IDF). The TF score measures how often words appear in a particular document. In simple words, TF counts the occurrences of words in a document. The weight of a term is proportional to its frequency in the document, meaning that words that appear more frequently in a document are assigned a higher weight [28]. In contrast, IDF measures the rarity of words in the text. It gives more importance to rarely used words in the corpus that may hold significant information. By incorporating IDF, TF-IDF diminishes the weight of frequently occurring terms and increases the weight of rarely occurring terms [43].

TF–IDF has been one of the most widely used methods in natural language processing and machine learning for tasks like document classification, text summarization, sentiment

classification, and spam message detection. For example, identifying the most relevant words is commonly used in a document and then apply these words as features in a classification model. A survey conducted in 2015 of text-based recommender systems in digital libraries found that 83% of them used TF-IDF [4]. Furthermore, many previous studies have demonstrated the effectiveness of TF-IDF for tasks like automated text classification and sentiment analysis [8,17,20,41]. However, TF-IDF has limitations. TF-IDF does not efficiently capture the semantic meaning of words in a sequence or consider the order in which terms appear. Additionally, TF-IDF can be biased towards longer documents, meaning that longer documents will generally have higher scores than shorter ones.

2.1.2. Word Embedding

Word embeddings are a type of representation learning used in natural language processing that enables computers to understand the relationship between words. Humans have always excelled at understanding the relationship between words such as man and woman, cat and dog, etc. Word embedding has been developed to represent these relationships as numeric vectors in an n-dimensional space. In this space, words with similar meanings have similar representations, meaning that two similar words are represented by almost similar vectors that are closely placed in the vector space. This technique has been used effectively in various natural language processing tasks, such as sentiment analysis and machine translation. However, creating effective word embeddings is a significant and premier issue in natural language processing because the quality of word embeddings can impact the performance of downstream tasks. Moreover, ingenious word representations in a lower dimensional space can be more beneficial and faster to train a model, making the creation of effective word embeddings a critical research area.

Word2Vec

Word2Vec is a popular technique for learning word embeddings using shallow neural networks, developed by [29]. Word2Vec comprises two distinct models: Continuous Bag of Words (CBOW) and Continuous Skip-gram. The CBOW model predicts the middle word based on surrounding context words, while Skip-gram predicts the surrounding words given a target word. The context consists of a few words before and after the middle word in CBOW [30].

Global Vectors for Word Representation

Global Vectors for Word Representation (GloVe) is an algorithm that generates word embeddings by using matrix factorization techniques on a word-context matrix. To create the word-context matrix, a large corpus is scanned for each term, and context terms within a window defined by a window size before and after the term are counted. The resulting matrix contains co-occurrence information for each word (the rows) and its context words (the columns). To account for the decreasing importance of words as their distance from the target word increases, a weighting function is used to assign lower weights to more distant words. [35]

2.1.3. Transformer

In a study by Vaswani et al. (2017), an attention-based algorithm called Transformers was introduced [48]. Transformers are a unique type of sequence transduction model that rely solely on attention rather than recurrence. This approach allows for the consideration of more global relationships in longer input and output sequences. As a result, Transformers have recently been utilized in natural language processing to address various challenges.

Bidirectional encoder representations from transformers

BERT is a self-supervised learning model for learning language representations that was released by Google AI in 2018 [12]. BERT introduces a masked bidirectional language modeling objective that leverages context learned from both directions to predict randomly

masked tokens, allowing it to better capture contextualized word associations. BERT belongs to a class of models known as transformers, and comes in two variants: BERT-Base, which includes 110 million parameters, and BERT-Large, which has 340 million parameters. BERT relies on an attention mechanism to generate high-quality, contextualized word embeddings [48]. The attention mechanism captures the word associations based on the words to the left and right of each word as it passes through each BERT layer during training. Compared to traditional techniques like BoW and TF-IDF, BERT is a revolutionary technique for creating better word embeddings, thanks to its pre-training on Wikipedia data sets and massive word corpus. BERT has been successfully applied to many natural language processing tasks, including language translation [16,26,31].

DistilBERT

DistilBERT is a highly efficient and cost-effective variant of the BERT model that was developed by distilling BERT-base. With 40% fewer parameters than bert-base-uncased, DistilBERT is both small and lightweight. Additionally, it runs 60% faster than BERT while maintaining an impressive 97% performance on the GLUE language understanding benchmark [38].

RoBERTa

Yinhan Liu et al. proposed a robust approach called the Robustly Optimized BERT-Pretraining Approach (RoBERTa) in 2019, which aims to improve upon the original BERT model for pretraining natural language processing (NLP) systems [27]. RoBERTa shares the same architecture as BERT, but incorporates modifications to the key hyperparameters and minor embedding tweaks to increase robustness. Unlike BERT, RoBERTa does not use the next-sentence pretraining objective, and instead trains the model with much larger mini-batches and learning rates. Additionally, RoBERTa is trained using full sentences, dynamic masking, and a larger byte-level Byte-pair encoding (BPE) technique. RoBERTa has been widely adopted in downstream NLP tasks and has achieved outstanding results compared to other models [1,9,45].

2.2. Text Classification

Text classification is also known as text tagging or text categorization. The aim is to categorize and classify text into organized groups. Text classifiers can automatically analyze given text and assign a set of pre-defined tags or categories based on its content.

While human experts are still considered the most reliable method for text classification, manual classification can be a complex, tedious, and costly task. With the advancement of Natural Language Processing (NLP), text classification has become increasingly important, particularly in areas such as sentiment analysis, topic detection, and language detection. Various machine learning and deep learning methods have been employed for sentiment analysis, with Twitter being a popular data source[13,18,25,46]. Supervised methods, including Decision Trees, Random Forests, Logistic Regression, Support Vector Machines (SVM), and Naive Bayes, have been used to train classifiers [2,40]. However, supervised approaches require labeled data, which can be expensive. To address this, unsupervised learning methods, such as the proposed by Pandarachalil et al., have been suggested [34]. Additionally, Qaisar and Saeed Mian utilized a Long Short-Term Memory (LSTM) Classifier for sentiment analysis of movie reviews [36]. Similar to our task, a sentence is assigned to a label.

2.3. Dyadic Parent–Child Interaction Coding System and Parent-Child Interaction Therapy

The Dyadic Parent–Child Interaction Coding System, fourth edition (DPICS-IV), is a structured behavioral observation tool that assesses essential parent and child behaviors in standardized situations. The DPICS-IV has proven to be a valuable adjunct to Parent-Child Interaction Therapy (PCIT) and has been used extensively to evaluate other parenting interventions and research objectives as well [42]. Over the years, the DPICS has been

utilized in various studies addressing a wide range of clinical and research questions. Nelson et al. highlight the development of the DPICS and discuss its current usage as a treatment process or outcome variable. The authors also summarize the ways in which the DPICS has been adapted and the process by which it is designed to be adapted [32].

The DPICS-IV scoring system is based on the frequency counts of ten main categories, including Neutral talk, labeled Praise, unlabeled Praise, Behavior Description, Reflection, Information Question, Descriptive Question, Direct commands, Indirect commands, and Negative talk. However, in previous work, eight categories were commonly used, where Information Question and Descriptive Question were combined into a single category called Question, and Indirect Commands and Direct Commands were combined as Commands. [7,10,23]. Both Cañas et al. and Huber et al. have suggested that not all DPICS codes are equally important for therapy outcomes and have placed more emphasis on Negative talk. In addition, Cañas et al. found that the DPICS Negative talk factor had high discriminant capacity (AUC = 0.90) between samples, and a cut-off score of 8 enabled mother-child dyads to be classified with 82% sensitivity and 89% specificity [7].

The process of labeling DPICS codes manually for each sentence in a conversation is a time-consuming and labor-intensive task that requires trained experts. Confirmatory factor analysis is then used to verify the factor structure of the observed variables [6]. However, Huber et al. have developed SpecialTime, an automated system that can classify transcript segments into one of eight DPICS classes. The system uses a linear support vector machine trained on text feature representations obtained using TF-IDF and part-of-speech tags. The system achieves an overall accuracy of 78%, as evaluated by the authors using an expert-labeled corpus [23].

Parent-Child Interaction Therapy(PCIT) helps parents improve interaction quality with children with behavior problems. The therapy trains parents to use effective dialogue when interacting with their children [11].

3. Methodology

The proposed approach contains the four components given below:

1. Feature Extraction phase: This phase aims to convert utterances into numerical vector inputs for the next phase. We deploy various typical algorithms to generate different vectors.
2. Training and fine-tuning: The generated vectors from the feature extraction phase are then used as input for machine learning algorithms to train the classifiers. In certain instances, for vectors from pre-train models, both algorithm structures (DistillBERT and Roberta) are fine-tuned and then trained on the training set. We fine-tune the resulting network and append the pre-trained model by two dense layers of neural networks to do the pre-classification. Then the results are produced either from concatenated machine learning methods or directly classified.
3. Automatic DPICS classification: After the classifiers are developed and trained, they can be deployed to test expert-annotated data to compare results. In this stage, parent utterances from real-life parent-child interactions are tested by the classifiers to determine which DPICS class the sentences belong to. The DPICS classification results can then be provided to parent-child interaction therapists to assist in improving the quality of parent-child interaction. It is important to note that the labels used in this work are based on the DPICS-IV. [42].

To determine the most effective methods, we developed classifiers using seven widely used methods, including Bow, TF-IDF, Word2Vec, Glove, DistilBert, and Roberta. We then evaluated their effectiveness using classical machine learning methods such as logistic regression, support vector machines (SVMs), and XGBoosting and compared our results to those defined by experts. To summarize, the present study includes the following three main points:

1. Transitional text representation vs. Word embedding vs. Transformers
2. Transfer Learning

3. Boosting Improvement

4. Experiments

4.1. Dataset

Our dataset consists of three sections. The first section includes 1753 instances of Parent Verbalizations, which were obtained from the Dyadic Parent-Child Interaction Coding System (DPICS) Comprehensive Manual for Research and Training, Fourth Edition(DPICS-IV) [42]. Experts provided guided examples and rules to help human coders distinguish labels for the 10 DPICS classes in Parent Verbalizations, including Unlabeled Praise(UP), Labeled Praise(LP), Reflection(RF), Behavior Description(BD), Information Question(IQ), Descriptive Question(DQ), Indirect Commands(IC), Direct Commands(DC), Negative Talk(NTA), and Neutral Talk(TA).

In the second section, we recorded five families engaged in daily conversations, with each family providing at least 30 minutes of audio recordings. These recordings were transcribed into text and labeled by trained Research Assistants using the 10-class DPICS coding system for Parent Verbalizations. A local IRB committee has approved the recruiting and consenting procedure.

The third section data are provided by [23]. Bernd Huber et al. created an expert-annotated 6,021 utterance samples dataset for parent-child interaction therapy. But in this PCIT dataset, the utterances are classified into 8 classes that Information Question(IQ) and Descriptive Question(DQ) are combined into Question(QU), and Indirect Commands(IC) and Direct Commands(DC) are put together as commands (CMD).

A summary of our dataset shown classes and total numbers is given in Table 1.

Table 1. Dataset Summary.

DPICS Classes	DPICS mamual	5-Family	PCIT
Unlabeled Praise(UP)	143	39	213
Labeled Praise(LP)	122	22	723
Reflection(RF)	61	87	693
Behavior Description(BD)	141	56	748
Information Question(IQ)	197	259	782[a]
Descriptive Question(DQ)	102	292	
Indirect Commands(IC)	192	115	924[b]
Direct Commands(DC)	168	395	
Negative Talk(NTA)	206	126	634
Neutral Talk(TA)	421	561	1304
Total	1753	1952	6021

[a] Question(QU).
[b] Commands (CMD).

4.2. Evaluation Metric

Our dataset is imbalanced, as shown in Table 1. However, accuracy is not an appropriate performance measure for imbalanced classification problems, as models that always predict the majority class will achieve high accuracy scores even if they fail to identify samples from the minority class. For example, if a dataset contains 95% samples from the majority class and 5% samples from the minority class, a model that always predicts the majority class will achieve an accuracy score of 95%, even if it fails to correctly identify any samples from the minority class.

To evaluate the performance of our classifiers, we used precision, recall, and F-Measure metrics. Precision quantifies the number of positive class predictions that actually belong to the positive class, while recall quantifies the number of positive class predictions made out of all positive examples in the dataset. F-Measure provides a way to balance the tradeoff between precision and recall by combining both metrics into a single score. These metrics are useful for evaluating the performance of classifiers in scenarios where one class is more important than the other.

In our multi-class classification problem, we calculated both micro-average and macro-average precision, recall, and F-Measure scores. Micro-average metrics give equal weight to each example in the dataset, while macro-average metrics give equal weight to each class in the dataset. The macro-average precision and recall scores are calculated as the arithmetic mean of individual classes' precision and recall scores, while the macro-average F1-score is calculated as the arithmetic mean of individual classes' F1-score.

We consider the 10 classes equally important, while some research experts suggested that not all DPICS codes are equally influential for therapy outcomes.

Table 2. Measures of Classification Performance.

Measure	Formula
Accuracy	$\frac{C}{A}$
Macro Precision	$\frac{\sum_{i=1}^{n} precision_i}{n}$
Macro Recall	$\frac{\sum_{i=1}^{n} i}{n}$
Macro F	$\frac{\sum_{i=1}^{n} i}{n}$

Table 2 gives formulas of all metrics used in our experimental tests where:

- C: Number of correct predictions;
- A: Total number of all samples;
- $precision_i$: precision for each class; where

$$precision = \frac{TruePostive}{TruePostive + FalsePositive}, \tag{1}$$

- $recall_i$: recall for each class; where

$$recall = \frac{TruePostive}{TruePostive + FalseNegative}, \tag{2}$$

- F_i: recall for each class; where

$$F_i = 2 \times \frac{precision_i \cdot recall_i}{precision_i + recall_i}, \tag{3}$$

5. Results

5.1. Text Representation vs. Word Embedding vs. Transformers

In this section, we used two expert-labeled datasets, DIPCS-IV and PCIT. DIPCS-IV and PCIT were used to train and test our models. To evaluate model performance, we employed five-fold cross-validation, using four folds for training and one fold for testing. Additionally, we utilized grid search in hyperparameter spaces to identify parameter combinations that maximized performance on the validation fold. Using two different datasets for evaluation can increase the robustness of the results by demonstrating the efficiency and accuracy of the classification across different datasets. This approach helps ensure that the results are not just specific to one particular dataset.

5.1.1. Performance on DPICS Manual

Table 3. 10 Classes on DPICS Manual results.

		Accuracy	Macro Precision	Macro Recall	Macro F
BoW	LR	52.312 ± 3.801	52.619 ± 4.237	46.632 ± 2.676	47.476 ± 3.632
	SVM	48.715 ± 1.986	55.980 ± 4.960	39.626 ± 3.020	41.701 ± 3.613
	XGBoost	53.388 ± 1.067	52.455 ± 2.327	47.847 ± 1.296	48.574 ± 1.072
TF-IDF	LR	47.975 ± 2.379	53.529 ± 4.517	38.669 ± 0.765	40.430 ± 1.292
	SVM	47.976 ± 2.258	56.663 ± 5.859	37.642 ± 2.014	40.103 ± 2.747
	XGBoost	50.942 ± 1.253	49.655 ± 2.977	45.314 ± 1.448	46.088 ± 1.479
Glove	LR	47.294 ± 3.745	46.203 ± 4.628	44.000 ± 3.973	44.171 ± 4.016
	SVM	42.727 ± 0.318	54.213 ± 4.074	32.556 ± 0.876	35.368 ± 1.259
	XGBoost	52.201 ± 3.495	50.315 ± 5.158	46.711 ± 3.005	47.026 ± 3.946
DistilBERT	Fine-tune	69.426 ± 5.657	69.346 ± 4.219	66.248 ± 3.619	65.543 ± 4.571
Roberta	Fine-tune	72.337 ± 3.559	72.284 ± 3.929	70.502 ± 2.642	69.617 ± 3.769

BoW: Bag of Word;
TF-IDF: Term Frequency-Inverse Document Frequency;
GloVe: Global Vectors for Word Representation;
LR: Logistic Regression;
SVM: Support Vector Machine;
XGBoost: eXtreme Gradient Boosting.

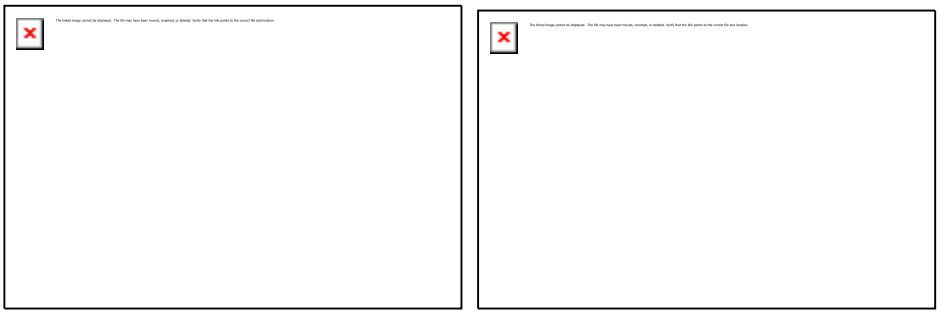

Figure 1. Confusion matrix of utterance classifier in the DPICS scheme with predictions in rows and actual labels in columns. The left is the confusion matrix of DPICS-IV, and the right is the confusion matrix of PCIT dataset. The overall accuracy of the DPICS-IV classifier is 72.3%, and the PCIT classifier is 79.8%.

The performance results are presented in Table 3, showing the average $\pm$ standard deviation for five-fold cross-validation. This format is a useful way to show the performance as it provides both the average performance and the degree of variability across the folds. The classifier developed by Roberta achieves the best results of 72.3% accuracy (72.3% macro-precision, 70.5% macro-recall, and 69.6% macro F-score). The confusion matrix is presented at figure 1(a).

5.1.2. Performance on PCIT

Using the same method as in the DPICS Manual, we evaluated the performance of DistilBert and Roberta. Table 4 demonstrated that both models achieved better performance compared to other methods, with Roberta surpassing the previous best results of 78.3% accuracy (79% precision, 77% recall) averaged over the eight DPICS classes [23]. Roberta achieved an overall accuracy of 79.8% (80.4% macro-precision, 79.7% macro-recall, and 79.8% macro F-score). While Huber et al. found that TF-IDF with SVM can achieve similar results with additional feature engineering such as part-of-speech tagging, our approach directly feeds raw data without any extra feature engineering. Confusion matrix is listed in figure 1(b).

Table 4. 8 Classes on PCIT results.

		Accuracy	Macro Precision	Macro Recall	Macro F
BoW	LR	63.494 ± 1.409	64.596 ± 1.781	61.772 ± 1.226	62.570 ± 1.435
	SVM	64.158 ± 0.700	68.075 ± 2.175	61.128 ± 0.723	62.505 ± 1.130
	XGBoost	63.743 ± 0.908	64.352 ± 1.504	60.763 ± 1.475	61.389 ± 1.506
TF-IDF	LR	61.783 ± 0.977	63.606 ± 1.465	57.416 ± 0.912	58.484 ± 1.026
	SVM	64.474 ± 1.189	67.553 ± 2.333	61.296 ± 1.380	62.657 ± 1.716
	XGBoost	63.627 ± 0.778	65.832 ± 1.286	62.047 ± 1.334	62.944 ± 1.372
Glove	LR	62.248 ± 1.699	62.639 ± 2.567	62.795 ± 1.969	62.619 ± 2.276
	SVM	66.816 ± 0.942	69.346 ± 1.456	64.407 ± 1.606	65.981 ± 1.571
	XGBoost	72.280 ± 1.475	74.118 ± 1.527	71.493 ± 1.437	72.527 ± 1.414
DistilBERT	Fine-tune	77.495 ± 1.328	77.449 ± 1.797	78.049 ± 1.441	77.584 ± 1.583
Roberta	Fine-tune	79.854 ± 0.557	80.443 ± 1.241	79.762 ± 0.867	79.824 ± 0.748

BoW: Bag of Word;
TF-IDF: Term Frequency-Inverse Document Frequency;
GloVe: Global Vectors for Word Representation;
LR: Logistic Regression;
SVM: Support Vector Machine;
XGBoost: eXtreme Gradient Boosting.

5.2. Transfer Learning

In this section, we applied our trained DPICS models to classify PCI in real-life scenarios. This step is important as an application in real-life data allows us to see how well our models perform outside of the controlled environment of the training and testing datasets. The results of our real-life classification are presented in Table 5, where we report overall accuracy as the evaluation metric. This metric is intuitive and straightforward, as overall accuracy simply measures the proportion of correct classifications out of all classifications made.

Table 5. Five Families Experimental Results.

Family		BoW	TF-IDF	Glove	DistilBERT	Roberta
001	LR	40.240	46.024	40.723		
	SVM	45.783	43.614	43.373	61.928	66.024
	XGBoost	42.891	42.891	44.096		
002	LR	43.762	44.950	41.188		
	SVM	43.762	42.772	39.603	63.762	71.881
	XGBoost	43.762	47.524	46.732		
003	LR	51.685	53.089	36.235		
	SVM	53.370	52.247	43.258	74.157	75.843
	XGBoost	53.089	50.280	40.168		
004	LR	48.611	43.229	39.583		
	SVM	38.368	39.930	43.229	63.020	72.222
	XGBoost	46.527	41.667	49.826		
005	LR	44.231	44.231	38.461		
	SVM	36.538	44.231	42.307	59.615	69.231
	XGBoost	42.307	44.231	57.692		

BoW: Bag of Word;
TF-IDF: Term Frequency-Inverse Document Frequency;
GloVe: Global Vectors for Word Representation;
LR: Logistic Regression;
SVM: Support Vector Machine;
XGBoost: eXtreme Gradient Boosting.

According to the results presented in Table 5, Roberta achieved the highest accuracy, with an average accuracy of 71.0% when applied to the five family datasets, while

DistilBERT also demonstrated successful transfer learning. However, both Bag of Words (BoW) and Term Frequency-Inverse Document Frequency (TF-IDF) did not perform well in transfer learning. Despite being widely used and recommended in previous work [8,17,41], these methods require a complete and adequate corpus to produce accurate results. Our dataset was limited, and expert-annotated data is expensive, which likely contributed to their poor performance. The Glove method, which is word-based, performed better than BoW and TF-IDF but did not surpass the accuracy of DistilBERT and Roberta.

The results presented in our study demonstrate that transfer learning is applicable and can achieve acceptable results in DPCIS code classification. The primary goal of developing DPCIS classifiers is to assist and speed up the labeling process for experts. As such, it is important to consider the few-shot task in this development process. By leveraging the pre-trained models and fine-tuning them on small labeled datasets, we can efficiently classify parent-child interactions with a limited amount of annotated data.

5.3. Boosting Improvement

The DPICS-IV manual provides typical data, but the amount of data is insufficient according to [23]. Collecting data is a gradual process. In this section, we attempted to improve performance when newly labeled data is enrolled. We combined data from four families with the DPICS-IV manual as the training set and tested the model on the remaining family. This was done to determine if the performance could be improved compared to using the DPICS-IV manual as the training set alone.

Table 6. Five Families Boosting Improvement Experimental Results.

Family		Glove	DistilBERT	Roberta
001	LR	48.192		
	SVM	55.180	71.807	74.216
	XGBoost	58.313		
002	LR	40.792		
	SVM	48.712	71.683	73.267
	XGBoost	54.851		
003	LR	41.292		
	SVM	53.089	76.685	79.213
	XGBoost	52.528		
004	LR	45.138		
	SVM	51.388	73.784	75.694
	XGBoost	62.673		
005	LR	36.538		
	SVM	53.846	73.076	75.000
	XGBoost	57.692		

GloVe: Global Vectors for Word Representation;
LR: Logistic Regression;
SVM: Support Vector Machine;
XGBoost: eXtreme Gradient Boosting.

We conducted five-fold cross-validation using data from five families, testing each family with a combination of data from the other four families plus the DPICS-IV manual. As Table 6 presented, our results showed that the accuracy of each family improved with an average accuracy of 75.5%, indicating that the sentence-level classifiers can be enhanced with the addition of more data. However, we did not observe any significant improvement in Glove's performance, indicating that word-based methods are not well-suited for DPICS code classification.

6. Discussion

With the development of machine learning and artificial intelligence, there is potential to improve the summarization and synthesis of complex PCI data. For example, automatically classifying parent behavior and utterances into DPICS categories can facilitate streamlined data collection and analysis to accelerate the research process. From a clinical standpoint, this information can be utilized to provide personalized feedback to families who may be in treatment. This feedback can then be translated into just-in-time interventions that can be incorporated into application-based treatment. Rather than relying on manual coding of DPICS categories, which is time and labor intensive, leveraging artificial intelligence and machine learning can help automate this process and significantly advance behavioral treatment for families with children with elevated behavior problems (e.g., Attention-Deficit/Hyperactivity Disorder, Oppositional Defiant Disorder, Conduct Disorder). This would be a substantial advancement given that there is a shortage of providers specializing in childhood disruptive behavior disorder (American Psychological Association, 2022).

In our study, we compared several different language models for classifying DPCIS codes. The performance of four DPICS categories (RF, DQ, NTA, and TA) in the DPICS-IV dataset, as presented in Figure 1(Left), still needs improvement. Additionally, the accuracy of RF in the PCIT dataset, as presented in Figure 1(Right), is fair. After reviewing all the experimental results, we believe that the limited improvement in accuracy can be attributed to the fact that DPCIS code classification requires contextual information, especially context from children's responses and speech, as well as tone of speech and visual information. While Parent DPICS codes are currently labeled sentence by sentence, the DPICS-IV manual in Table 1 provides examples that illustrate how a single speech sentence can be labeled as different DPCIS codes depending on the situation. Given the context dependent nature of DPICS codes, future work could also investigate the possibility of labeling DPICS codes at the level of a conversation or an interaction, rather than at the sentence level, to better capture the context and nuances of the parent-child interaction.

7. Conclusions

In our study, we proposed various learning approaches to develop a sentence-based classifier for DPICS codes with 10 classes. Our results demonstrated that pre-trained language models, particularly Roberta, outperformed the other methods for DPICS classification. Specifically, Roberta surpassed previous results on an open-source PCIT dataset. Furthermore, our results suggested that word-based methods such as GloVe may not be suitable for DPICS code classification. Our findings suggest that pre-trained language models such as Roberta can be highly effective in accurately classifying DPICS codes. This could provide valuable assistance to experts in the labeling process, leading to more efficient and accurate labeling of children's speech and language.

Although sentence-based DPICS code classifiers can achieve acceptable results, it's important to consider the crucial role that context plays in determining DPICS codes. Future research should consider incorporating contextual information into DPICS classification tasks to improve performance. The DPICS-IV manual provides examples of how DPICS codes are related to context and are determined by different communication environments. However, previous work has primarily focused on sentence-based DPICS code classifiers. To address this limitation, future studies could explore the use of context-based classifiers either alone or in combination with sentence-based classifiers. By doing so, researchers could gain a more comprehensive understanding of how children's speech and language are related to different communication environments. In addition, the classifiers should be extended to more families when data is increasingly collected.

Funding: This work was partially supported by NSF-SCC-2125549.

References

1. Adoma, A. F., Henry, N. M., & Chen, W. (2020, December). Comparative analyses of bert, roberta, distilbert, and xlnet for text-based emotion recognition. In 2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP) (pp. 117-121). IEEE.
2. Ahammed, M. T., Gloria, A., Oion, M. S. R., Ghosh, S., Balaii, P., & Nisat, T. (2022, March). Sentiment Analysis using a Machine Learning Approach in Python. In 2022 International Conference on Communication, Computing and Internet of Things (IC3IoT) (pp. 1-6). IEEE.
3. Anand, Rajaraman & Jeffrey David, Ullman.(2011). Mining of massive datasets. Cambridge University Press.
4. Beel, J., Gipp, B., Langer, S., & Breitinger, C. (2016). Paper recommender systems: a literature survey. International Journal on Digital Libraries, 17, 305-338.
5. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. Advances in neural information processing systems, 33, 1877-1901.
6. Cañas Miguel, M., Ibabe Erostarbe, I., Arruabarrena Madariaga, M. I., & Paúl Ochotorena, J. D. (2021). Dyadic parent-child interaction coding system (Dpics): Factorial structure and concurrent validity. Psicothema.
7. Cañas, M., Ibabe, I., Arruabarrena, I., & De Paúl, J. (2022). The dyadic parent-child interaction coding system (DPICS): Negative talk as an indicator of dysfunctional mother-child interaction. Children and Youth Services Review, 143, 106679.
8. Christian, H., Agus, M. P., & Suhartono, D. (2016). Single document automatic text summarization using term frequency-inverse document frequency (TF-IDF). ComTech: Computer, Mathematics and Engineering Applications, 7(4), 285-294.
9. Cortiz, D. (2021). Exploring transformers in emotion recognition: a comparison of bert, distillbert, roberta, xlnet and electra. arXiv preprint arXiv:2104.02041.
10. Cotter, A. M. (2016). Psychometric properties of the dyadic parent-child interaction coding system (DPICS): Investigating updated versions across diagnostic subgroups (Doctoral dissertation, Auburn University).
11. Cotter, A. M., & Brestan-Knight, E. (2020). Convergence of parent report and child behavior using the Dyadic Parent-Child Interaction Coding System (DPICS). Journal of Child and Family Studies, 29, 3287-3301.
12. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
13. Diyasa, I. G. S. M., Mandenni, N. M. I. M., Fachrurrozi, M. I., Pradika, S. I., Manab, K. R. N., & Sasmita, N. R. (2021, May). Twitter sentiment analysis as an evaluation and service base on python textblob. In IOP Conference Series: Materials Science and Engineering (Vol. 1125, No. 1, p. 012034). IOP Publishing.
14. El-Din, D. M. (2016). Enhancement bag-of-words model for solving the challenges of sentiment analysis. International Journal of Advanced Computer Science and Applications, 7(1).
15. Eyberg, S. M., Boggs, S. R., & Algina, J. (1995). Parent-child interaction therapy: a psychosocial model for the treatment of young children with conduct problem behavior and their families. Psychopharmacology bulletin.
16. Gao, Z., Feng, A., Song, X., & Wu, X. (2019). Target-dependent sentiment classification with BERT. Ieee Access, 7, 154290-154299.
17. Ghag, K., & Shah, K. (2014). SentiTFIDF–Sentiment classification using relative term frequency inverse document frequency. International Journal of Advanced Computer Science and Applications, 5(2).
18. Gupta, B., Negi, M., Vishwakarma, K., Rawat, G., Badhani, P., & Tech, B. (2017). Study of Twitter sentiment analysis using machine learning algorithms on Python. International Journal of Computer Applications, 165(9), 29-34.
19. HaCohen-Kerner, Y., Miller, D., & Yigal, Y. (2020). The influence of preprocessing on text classification using a bag-of-words representation. PloS one, 15(5), e0232525.
20. Hakim, A. A., Erwin, A., Eng, K. I., Galinium, M., & Muliady, W. (2014, October). Automated document classification for news article in Bahasa Indonesia based on term frequency inverse document frequency (TF-IDF) approach. In 2014 6th international conference on information technology and electrical engineering (ICITEE) (pp. 1-4). IEEE.
21. Harris, Z. S. (1954). Distributional structure. Word, 10(2-3), 146-162.
22. Huang, C. R., & Lee, L. H. (2008, November). Contrastive approach towards text source classification based on top-bag-of-word similarity. In Proceedings of the 22nd pacific asia conference on language, information and computation (pp. 404-410).
23. Huber, B., Davis III, R. F., Cotter, A., Junkin, E., Yard, M., Shieber, S., ... & Gajos, K. Z. (2019, May). SpecialTime: Automatically detecting dialogue acts from speech to support parent-child interaction therapy. In Proceedings of the 13th EAI International Conference on Pervasive Computing Technologies for Healthcare (pp. 139-148).
24. Jeong, J., Franchett, E. E., Ramos de Oliveira, C. V., Rehmani, K., & Yousafzai, A. K. (2021). Parenting interventions to promote early child development in the first three years of life: A global systematic review and meta-analysis. PLoS medicine, 18(5), e1003602.
25. Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2016). Bag of tricks for efficient text classification. arXiv preprint arXiv:1607.01759.
26. Koroteev, M. V. (2021). BERT: a review of applications in natural language processing and understanding. arXiv preprint arXiv:2103.11943.
27. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.

28. Luhn, H. P. (1957). A statistical approach to mechanized encoding and searching of literary information. IBM Journal of research and development, 1(4), 309-317.
29. Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. Advances in neural information processing systems, 26.
30. Mikolov, T., Le, Q. V., & Sutskever, I. (2013). Exploiting similarities among languages for machine translation. arXiv preprint arXiv:1309.4168.
31. Müller, M., Salathé, M., & Kummervold, P. E. (2020). Covid-twitter-bert: A natural language processing model to analyse covid-19 content on twitter. arXiv preprint arXiv:2005.07503.
32. Nelson, M. M., & Olsen, B. (2018). Dyadic parent–child interaction coding system (DPICS): an adaptable measure of parent and child behavior during dyadic interactions. Handbook of parent-child interaction therapy: Innovations and applications for research and practice, 285-302.
33. Nilsen, F. M., Ruiz, J. D., & Tulve, N. S. (2020). A meta-analysis of stressors from the total environment associated with children's general cognitive ability. International Journal of Environmental Research and Public Health, 17(15), 5451.
34. Pandarachalil, R., Sendhilkumar, S., & Mahalakshmi, G. S. (2015). Twitter sentiment analysis for large-scale data: an unsupervised approach. Cognitive computation, 7(2), 254-262.
35. Pennington, J., Socher, R., & Manning, C. D. (2014, October). Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP) (pp. 1532-1543).
36. Qaisar, S. M. (2020, October). Sentiment analysis of IMDb movie reviews using long short-term memory. In 2020 2nd International Conference on Computer and Information Sciences (ICCIS) (pp. 1-4). IEEE.
37. Thomas, R., Abell, B., Webb, H. J., Avdagic, E., & Zimmer-Gembeck, M. J. (2017). Parent-child interaction therapy: A meta-analysis. Pediatrics, 140(3).
38. Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. arXiv preprint arXiv:1910.01108.
39. Schütze, H., Manning, C. D., & Raghavan, P. (2008). Introduction to information retrieval (Vol. 39, pp. 234-265). Cambridge: Cambridge University Press.
40. Singh, J., & Tripathi, P. (2021, June). Sentiment analysis of Twitter data by making use of SVM, Random Forest and Decision Tree algorithm. In 2021 10th IEEE International Conference on Communication Systems and Network Technologies (CSNT) (pp. 193-198). IEEE.
41. Sjarif, N. N. A., Azmi, N. F. M., Chuprat, S., Sarkan, H. M., Yahya, Y., & Sam, S. M. (2019). SMS spam message detection using term frequency-inverse document frequency and random forest algorithm. Procedia Computer Science, 161, 509-515.
42. SM Eyberg, MM Nelson, NC Ginn, N Bhuiyan, & SR Boggs. (2013). Dyadic Parent–Child Interaction Coding System (DPICS) comprehensive manual for research and training. 4th PCIT International. Gainesville, FL.
43. Sparck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. Journal of documentation, 28(1), 11-21.
44. Suhartono, D., Purwandari, K., Jeremy, N. H., Philip, S., Arisaputra, P., & Parmonangan, I. H. (2023). Deep neural networks and weighted word embeddings for sentiment analysis of drug product reviews. Procedia Computer Science, 216, 664-671.
45. Tarunesh, I., Aditya, S., & Choudhury, M. (2021). Trusting roberta over bert: Insights from checklisting the natural language inference task. arXiv preprint arXiv:2107.07229.
46. Wagh, R., & Punde, P. (2018, March). Survey on sentiment analysis using twitter dataset. In 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA) (pp. 208-211). IEEE.
47. Valero Aguayo, L., Rodríguez Bocanegra, M., Ferro García, R., & Ascanio Velasco, L. (2021). Meta-analysis of the efficacy and effectiveness of parent child interaction therapy (PCIT) for child behaviour problems. Psicothema.
48. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30.
49. Yan, D., Li, K., Gu, S., & Yang, L. (2020). Network-based bag-of-words model for text classification. IEEE Access, 8, 82641-82652.