Utterance classification using the Six Thinking Hats framework

Salvatore Puccio s.puccio1
Emiliano Marrale e.marrale
Federico Tamponi f.tamponi2
Mostafa Eid m.eid2

@studenti.unipi.it

Abstract

Communication is almost always driven by an underlying intention—whether consciously recognized or not. This intention might be to express emotion, critique logically, or convey factual information. However, such motives are often hidden from listeners and even from speakers themselves, leading to misunderstandings—for example, when emotional expression is met with a factual counterargument. This report aims to move beyond surface-level intent detection by uncovering the invisible "hat" behind each utterance—the deeper purpose guiding the communication. Drawing inspiration from Edward de Bono's Six Thinking Hats framework (1985), we propose a hat-based classification system to capture these hidden dimensions of intent. Despite its potential, identifying such latent intentions remains a significant challenge.

1 Introduction

Intent detection has been used in dialogue systems to recognize the users' needs for several years at the moment, because it forms a key component in human-machine interaction.
What makes intent detection crucial is that it is a translation of the user's wants. Intents are usually formed from a verb + noun, such as query weather, *book a taxi*, *find a hotel*. The intent is usually classified from a set of intent categories that were already defined, depending on the type of need the user has.

A step further than facilitating human-machine interaction, we aim to ease and make interaction between humans easier to navigate. We aim to do this by classifying – the naturally rich in emotion human dialogues and trying to understand the type of thinking behind them – into one of the hats presented by the Six Thinking Hats framework, providing a more cognitive-level classification of user intent that reflects different modes of reasoning.

Six Hats Here, we briefly explain each of the Six Thinking Hats. For a more detailed and comprehensive understanding, we recommend reading the original book.

- The **White Hat** focuses on facts. Whoever wears it discusses objective information without expressing personal opinions or interpretations.
- The **Red Hat** represents emotion, feelings, reactions, and intuitions. A person wearing the red hat may express their emotions or gut feelings freely, without needing to justify them.
- The **Black Hat** is used for critical judgment. It involves logical analysis that highlights potential problems, risks, or negative aspects.
- The **Yellow Hat** stands for optimism. It is the counterpart to the black hat, emphasizing the positive aspects of an idea.

HLT final project report (2025).

- The **Green Hat** symbolizes creativity. It encourages proposing new ideas, exploring alternatives, and thinking outside the box.
- The **Blue Hat** is responsible for managing the thinking process. Those who wear it organize and direct the use of the other hats. For instance, in a meeting, a blue hat may decide that it's time for green hat thinking, prompting everyone to focus on generating creative ideas.

For our purposes, the blue hat is not relevant, as we aim to detect the underlying intent of a speaker. Blue hat statements are typically declarative and presuppose awareness of the framework.

In this paper, we describe our approach, the challenges we encountered, and how we addressed them.

2 Background

2.1 Datasets

DailyDialog The primary dataset used in this project is DailyDialog [Li et al., 2017], which contains dialogues covering various aspects of daily life, such as *work*, *relationships*, *ordinary life*, and *tourism*.

Each dialogue is associated with a unique ID and a corresponding *topic*, which provides contextual information. Additional columns in the dataset describe individual utterances, including *emotion*, *act*, and *turn*.

A major limitation of this dataset—and a key challenge in our project—is the absence of a "hat" label, which is essential for training models to recognize the type of thinking reflected in an utterance according to the Six Thinking Hats framework.

Since DailyDialog is a relatively large dataset, we needed a way to automatically label utterances with hat categories. We explored different strategies for each hat which we are going to explain in more details in section 3.1.

Total Dialogues	13,118
Average Speaker Turns Per Dialogue	7.9
Average Tokens Per Dialogue	114.7
Average Tokens Per Utterance	14.6

Table 1: Basic Statistics of DailyDialog. [Li et al., 2017]

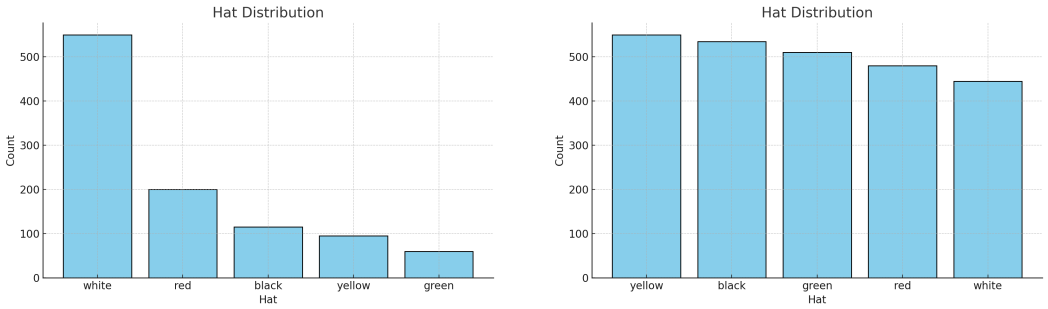
2.2 Hand-labeled Dataset

We selected a total of 200 dialogues from the DailyDialog dataset, comprising 50 from the *Work* topic, 50 from *Relationship*, and 100 from *Emotion & Attitude*. Before annotation, we established a consensus on the definitions of the hat categories, slightly adapting them from the original Six Thinking Hats framework as follows:

- **White:** Exchanging or providing plain informations. Things generally true, things that happened to someone.
- **Black:** Analysis of a situation. Negative analysis, explaining the weak points of something.
- **Red:** Emotions involved in the answer. Intuitions, feelings, gut reactions. No need for logical justification.
- **Yellow:** Statements that highlight the positive sides. Open up remote but highly desirable possibilities. Provide encouragement to take action. Express positive judgments.
- **Green:** Proposing new point of views. New Ideas. Solutions to problems. Imagining new scenarios, going beyond what is known.

Each team member manually labeled 50 dialogues according to these definitions, resulting in approximately 1,000 utterances overall. The resulting class distribution is illustrated in Figure 1a, where a noticeable imbalance in favor of the white hat label can be observed.

To address this imbalance, we employed the **Easy Data Augmentation** (EDA) technique Wei and Zou [2019]. The dataset was first split into training and test sets using an 80-20 ratio. EDA was then applied exclusively to the training set to prevent overfitting. The post-augmentation class distribution is shown in Figure 1b.



(a) Human-Labeled Dataset: Hat Distribution

(b) Human-Labeled Dataset: EDA Hat Distribution

Figure 1: Comparison of Hat Distributions Before and After EDA

2.3 BERT

Bidirectional Encoder Representations from Transformers (BERT) is a transformer-based architecture that utilizes only the encoder component of the original Transformer model. It is designed to be pre-trained on large corpora to capture language semantics in a deeply bidirectional manner. This pre-training enables the model to be fine-tuned with minimal task-specific modifications—typically by adding a single output layer—resulting in state-of-the-art performance across a wide range of Natural Language Processing (NLP) tasks Devlin et al. [2018].

The BERT family of models, which currently represents the state of the art in several NLP benchmarks, largely adheres to the architecture of the original BERT-base model. This architecture comprises six transformer layers, each consisting of two key sub-components:

1. A multi-head self-attention mechanism
2. A position-wise fully connected feed-forward network

Each sub-layer is wrapped with a residual connection, followed by layer normalization. Formally, the output of each sub-layer is computed as:

$$\text{LayerNorm}(x + \text{Sublayer}(x))$$

For the purposes of this report, we omit a detailed explanation of the transformer mechanism—available in Vaswani et al. [2017]—and instead focus on summarizing the specific BERT model variants employed in our project.

roBERTa (Robustly Optimized BERT Pretraining Approach) It is an enhanced variant of BERT that maintains the core masked language modeling (MLM) objective while introducing several key modifications that lead to substantial performance gains across various NLP tasks, resulting in better generalization and improved transfer capabilities to downstream NLP tasks. Additionally, by simplifying the pretraining process (focusing solely on MLM), RoBERTa achieves more robust performance. Empirically, RoBERTa outperforms BERT on nearly all major NLP benchmarks Liu et al. [2021].

DistilBERT is a 40% smaller version of BERT, which was created using a technique called knowledge distillation to create a smaller transformer that can achieve results close to the ones produced by the original model, and is 60% faster during inference. Sanh et al. [2019]

DistilRoBERTa With the same previous idea of knowledge distillation, the model DistilRoBERTa is the result of distilling RoBERTa model, which is an enhanced successor of BERT, resulting from training the same architecture on more data, longer sequences, and removing the next sentence prediction objective and other changes that worked towards getting the best performance from the already existing architecture. **?**

2.4 Fine-tuning

There are several strategies for fine-tuning a Large Language Model (LLM), including full fine-tuning, Low-Rank Adaptation (LoRA), Parameter-Efficient Fine-Tuning (PEFT), and layer freezing. In this work, we focus primarily on full fine-tuning and Low-Rank Adaptation (LoRA), as these are the two techniques we primarily employed.

Full Fine-Tuning works on updating all of the model parameters, but it requires a substantial amount of computational power to get the job done.
LoRA freezes the model weights and applies changes to a different set of weights that are added to the original parameters. LoRA transforms the model parameters to a dimension with a lower rank, making the training faster by cutting the number of trainable parameters. It is usually good when there are few clients with different applications that require fine-tuning. Ding et al. [2023]

3 Methodology

3.1 Automated Labeling

In this section, we present the approaches used to automatically label the different types of hats in our dataset. We selected a subset of dialogues from the DailyDialogue dataset, focusing on the topics *Work*, *Relationship*, and *Emotion & Attitude*. Various labeling strategies inspired by prior research were explored to automatically assign hat labels to our dataset.

Red Hat

For the *Red Hat*, we leveraged the labels provided in the DailyDialogue dataset. Specifically, we labeled as red hat all utterances annotated with an emotion label corresponding to `anger`, `sadness`, `fear`, or `disgust`.

Black & White Hats

To infer the *Black* and *White Hats*, we utilized a pre-trained model introduced in Wróblewska [2025].

The referenced study investigated the application of natural language processing (NLP) to assess the creativity of questions—a critical element in information acquisition, decision-making, and problem-solving. Questions can take many forms (e.g., open, closed, rhetorical, hypothetical) and serve functions beyond information retrieval, such as exploring perspectives or evaluating scenarios.

The creative process often begins with asking the right questions. For instance, Reiter-Palmon found that individuals skilled in problem solving tend to reframe problems as questions Mumford et al. [1997]Reiter-Palmon et al. [1998]. More recently, Raz showed that question complexity correlates positively with question's creativity Raz et al. [2023]. The study measured complexity using Bloom's Taxonomy Anderson et al. [2000], a widely recognized framework for categorizing cognitive complexity in education.

Bloom's Taxonomy includes the following six hierarchical levels:

- **Knowledge** – recalling facts (e.g., "What is a computer?")
- **Comprehension** – understanding meaning (e.g., "How does a computer work?")
- **Application** – applying knowledge (e.g., "How can a computer facilitate daily life?")
- **Analysis** – identifying relationships (e.g., "What are the effects of computer usage on productivity?")
- **Synthesis** – generating new ideas (e.g., "How can we make computers more efficient?")

- **Evaluation** – judging information (e.g., "How beneficial are computers for society?")

Using RoBERTa and CNN-based architectures, the authors trained a model on an Exam Question Dataset. The model achieved near-perfect accuracy on the training set and approximately 80% accuracy on the test set by the 10th epoch Wróblewska [2025].

We applied this model to our dataset and conducted a manual inspection of the resulting labels. We found that utterances labeled as `Analysis` were indicative of black hat thinking, while those labeled as `Knowledge` and `Evaluation` were best suited for white hat labeling. The distribution of the various labels over our dataset is as shown in Figure 2:

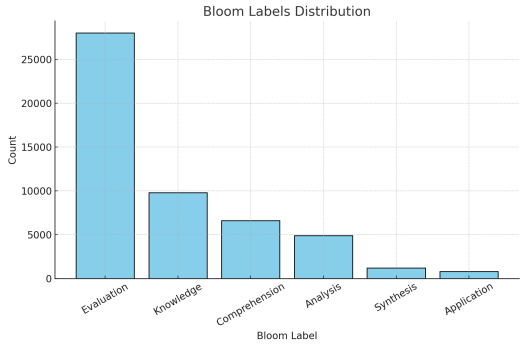


Figure 2: DailyDialogue Bloom Predictions

Yellow Hat

Optimism, the defining feature of the *Yellow Hat*, has been shown to correlate with various positive outcomes, including increased life expectancy and improved physical and mental health Helen Achat and Sparrow. [2000]. Further studies link optimism to personality traits such as extroversion, emotional stability, conscientiousness, and agreeableness J Patrick Sharpe and Roth. [2011]. As a relatively stable personality trait, optimism influences how individuals interpret and interact with the world. Conversely, recognizing pessimistic language may help identify individuals at risk for mental health challenges, such as depression.

We trained a BERT-based classifier on the tweet dataset introduced in the referenced study. The model yielded approximately 300 utterances labeled as optimistic with a confidence score exceeding 80%. However, the limited number of high-confidence predictions can be attributed to the substantial domain difference between the tweet dataset and our target utterances. This domain mismatch likely accounts for the suboptimal performance of the classifier in our context.

Green Hat

Defining creativity is a challenging task; for this reason, the green hat was the most difficult to label automatically.

The following approaches were considered:

- **Jordanous (2012) Approach**: We decided to use a simple approach to start, the one from Anna Jordanous, who in the work *"Defining Creativity: Finding Keywords for Creativity Using Corpus Linguistics Techniques"* Jordanous [2010] analyzes 30 academic papers in the field of creativity and with the help of a G-squared function, unveils the most frequent keywords in creativity-related academic papers. Our first approach was to take these keywords and make a simple lexicon to find out sentences in the dataset containing words from that list, or similar words, and so to label them as creative sentences.
- **Creativity Words with BART**: Extraction of creativity-related keywords using BART (Bidirectional and Auto-Regressive Transformer), which combines a bidirectional encoder (BERT-like) with an autoregressive decoder (GPT-like).

- **Zero-Shot Classification**: Conversion of labels into linguistic hypotheses (e.g., "This example is about creativity.") and assignment of the class whose entailment probability is highest (entailed, neutral, contradicted).

Application on DailyDialog: Utilization of the top 100 creativity-associated keywords to classify utterances; display of the eight utterances with the highest entailment scores, highlighting the scarcity of genuinely "creative" expressions in the corpus.

Table 2: Model predictions for the eight most creative utterances in DailyDialog

Label	Score	Text
variables	0.9790	What's the house rent? When is the rent due? And how much security deposit do you require?
positively	0.9731	This is encouraging news, good for you! What is your solution, then?
investigators	0.9702	It's a story about a policeman investigating a series of strange murders. I play the detective. He has to catch the killer, but there's very little evidence. It's a psychological thriller with some frightening scenes, but I hope people won't be too scared to go.
scores	0.9697	I'd like to talk to you about my grades.
positively	0.9697	Hooray! I can play football with daddy. Mom and sister can play badminton.
complexity	0.9692	This may be complicated. But I think he may need to build up his self-esteem.
unconscious	0.9688	She is still in a coma.
rated	0.9683	It's too raw. I wanted mine well done, but this one here is almost medium.

Zero-shot classification using Mistral-7B

After observing suboptimal results in previous experiments, we opted for a zero-shot classification approach using a large language model. Specifically, we used the **Mistral-7B-Instruct-v0.2** [AI, 2023] checkpoint, quantized in 4-bit to enable inference on consumer-grade GPUs.

The goal was to automatically annotate utterances as being expressed under a *green hat* or not, following Edward de Bono's Six Thinking Hats framework.

Prompt template used for classification:

> Definition: "A green hat is the hat of creativity.
> Under the green hat, you are permitted to put forward 'possibilities'.
> It is under the green hat that suggested courses of action are put forward: 'We could do this, or this, or this.'
> The green hat includes both 'the top of the head' creativity and 'deliberate' creativity.
> New ideas, new concepts, and new perceptions. The deliberate creation of new ideas.
> Alternatives and more alternatives. Change. New approaches to problems.
> Label the next utterance as a green hat or not.
> Answer only with "Y" for yes or "N" for No."
> Utterance: `<insert utterance here>`
> Answer:

Prediction function: a batch of utterances was classified using a simple loop, where each utterance was formatted into the prompt, encoded, and passed to the model. The model generated a single token ('Y' or 'N'), which was decoded and stored.

This approach allowed for rapid annotation of several thousand utterances with minimal supervision, using a deterministic prompt and reproducible pipeline.

3.2 Population-Based Training

For hyperparameter tuning, we employed two different approaches: a randomized search with cross-validation, which yielded suboptimal results, and Population-Based Training (PBT), which we briefly describe below.

Population-Based Training (PBT) Jaderberg [2017] maintains a small population of models trained concurrently. At regular intervals (every T steps), the models are evaluated on a validation set and ranked based on performance. The lower-performing models adopt the weights of higher-performing ones (*exploit*) and then perturb their hyperparameters (*explore*). This approach enables online hyperparameter optimization without the need to restart training from scratch.

Our Configuration

Base model: `distilroberta-base`, sequence classification with five labels.

Population: $|\mathcal{P}| = 2$; exploit interval $T = 1{,}000$ updates; bottom 20% replaced.

Search space: Learning rate: $\{1\times10^{-5},\ 5\times10^{-5},\ 1\times10^{-4},\ 2\times10^{-4}\}$;
Weight decay: $\{0,\ 0.01,\ 0.1\}$;
Warm-up ratio: $\{0.06,\ 0.10,\ 0.20\}$;
Dropout: $\{0.1,\ 0.2,\ 0.3\}$;
Continuous values are perturbed by $\times\{0.8,\ 1.2\}$; categorical ones are resampled.

Optimizer & schedule: `AdamW`; linear decay with warm-up.

Runtime: Batch size 8, 5 epochs, gradient clipping at 1.0.

4 Results

BERT

For both BERT and RoBERTa models, we performed hyperparameter tuning using randomized cross-validation. The search space included the following configuration options shown in Table 3. Despite this extensive search, the best configurations were manually selected after empirical evaluation.

Parameter	Values Explored	Final Value
epochs	10, 15, 20, 30	10
model_dropout	0.1, 0.2, 0.3	-
optimizer_lr	10^{-3}, 10^{-4}, 10^{-5}	2×10^{-5}
scheduler_warmup_steps	0, 100, 200	-
lora_r	4, 8, 16	-
lora_alpha	16, 32, 64	-
lora_dropout	0.05, 0.1, 0.2	-
tokenizer_max_length	32, 64, 128	128
dataloader_batch_size	8, 16, 32	8
clip_grad_norm	1.0, 2.0	1.0
early_stopping_patience	3, 5, 10	3
early_stopping_delta	0.01, 0.05, 0.1	0.1
scheduler_type	*linear*, *cosine*, *constant*	*linear*
optimizer_type	*AdamW*, *Adafactor*	*AdamW*
weight_decay	0.01, 0.001, 0.0001	-

Table 3: Hyperparameter search space for BERT and RoBERTa tuning

The evaluation results using these parameters across the different datasets are presented in Table 4. It is interesting to note that Bert trained over the automatically labeled dataset and tested over the hand-labeled dataset provided some results among the hats, white and red, and also some recall for the yellow hat. Thus, despite the poor performance, the bloom approach was approximately correct in labelling the white hat. Overall, the best performance was obtained in BERT trained over the easy augmented hand-labeled dataset and tested over the normal hand-labeled dataset. This yielded a minor improvement in the Weighted Avg F1 score and a boost in predicting all the hats, with the exception of the black hat, which lost some points.

The evaluation results using the selected hyperparameters across different datasets are presented in Table 4. Notably, BERT trained on the automatically labeled dataset and tested on the hand-labeled dataset achieved some minor improvement for the white and red hats, and to a lesser extent, for the yellow hat. Despite the generally low performance, this suggests that the Bloom-based labeling approach was approximately effective for identifying the white hat class.

Overall, the best performance was observed when BERT was trained on the EDA-augmented hand-labeled dataset and evaluated on the original hand-labeled test set. This configuration yielded a slight

improvement in the weighted average F1 score and enhanced prediction performance across most hat colors, except for the black hat, which experienced a slight decrease in all metrics.

Table 4: BERT Metrics Across Different Datasets

Trainint-Set	Test-Set	Class	Precision	Recall	F1-score
Hand-Labeled	Hand-Labeled	red	0.45	0.47	0.46
		white	0.68	0.84	0.75
		black	0.45	0.22	0.29
		yellow	0.67	0.21	0.32
		green	0.44	0.33	0.38
		Weighted Avg. F1	0.58 (204 samples)		
EDA Hand-Labeled	Hand-Labeled	red	0.63	0.47	0.54
		white	0.66	0.89	0.76
		black	0.29	0.09	0.13
		yellow	0.78	0.37	0.50
		green	0.56	0.42	0.48
		Weighted Avg. F1	0.60 (204 samples)		
Automatically Labeled	Automatically Labeled	red	0.72	0.75	0.73
		white	0.41	0.41	0.41
		black	0.52	0.60	0.56
		yellow	0.56	0.51	0.53
		green	0.63	0.57	0.60
		Weighted Avg. F1	0.57 (1000 samples)		
Automatically Labeled	Hand-Labeled	red	0.36	0.40	0.38
		white	0.63	0.47	0.54
		black	0.15	0.17	0.16
		yellow	0.24	0.47	0.32
		green	0.15	0.17	0.16
		Weighted Avg. F1	0.42 (204 samples)		

RoBERTa

RoBERTa yielded more or less the same overall results as BERT in all datasets using the same hyperparameter configurations. We present the metrics in the appendix in Table 8.

DistilBert

- **Model and tokenizer**: `DistilBertForSequenceClassification` with `DistilBertTokenizerFast`, max sequence length 128.
- **Optimizer & learning rate**: AdamW with initial LR 5×10^{-5}; default linear scheduler.
- **Early stopping**: monitor validation loss, patience = 3 steps, min delta = 10^{-4}.
- **Training setup**:
  - `hand_labelled_dataset.json`: 80%/10%/10% stratified split (train/val/test).
  - `ald_train_dataset.json`/`ald_test_dataset.json`: 80%/20% train/val; held-out test in `ald_test_dataset.json`.
  - Up to 100 epochs; batch size 16 (train), 32 (val).
- **Metrics**: precision, recall, F1-score per class; overall accuracy; macro & weighted averages.

Dataset 1 Analysis (`hand_labelled_dataset.json`, 30 samples) The model achieves 57% overall accuracy on this small, manually balanced set (6 samples per class). It performs decently on `red` (F1 = 0.77) and struggles most with `white` (F1 = 0.44).

Dataset 2 Analysis (`ald_test_dataset.json`, 1 000 samples) Here, accuracy is 55% across 1 000 examples (200 per class). `red` remains a decent performer (F1 = 0.71), but `white` suffers a dramatic drop in recall (0.17) compared to Dataset 1, yielding a low F1 of 0.25. Other classes show modest declines in F1, reflecting greater diversity and noise in this larger, automatically balanced set.

Table 5: Performance on `hand_labelled_dataset.json` (left) and `ald_test_dataset.json` (right)

Class	P	R	F1	Supp.
black	0.40	0.67	0.50	6
green	1.00	0.33	0.50	6
red	0.71	0.83	0.77	6
white	0.67	0.33	0.44	6
yellow	0.50	0.67	0.57	6
Accuracy		0.57		30
Macro avg	0.66	0.57	0.56	30
Weighted avg	0.66	0.57	0.56	30

Dataset 1: hand_labelled

Class	P	R	F1	Supp.
black	0.49	0.56	0.52	200
green	0.50	0.70	0.58	200
red	0.74	0.68	0.71	200
white	0.50	0.17	0.25	200
yellow	0.51	0.61	0.56	200
Accuracy		0.55		1000
Macro avg	0.55	0.55	0.53	1000
Weighted avg	0.55	0.55	0.53	1000

Dataset 2: ald_test

DistilRoBERTa

All models below were trained using the same architecture, **DistilRoBERTa** [HuggingFace, 2019]. In the Methodology section, we explained the hyperparameter space and specified the population of 2 for PBT. The only exception is **Hand-label (4)**, which refers to the model trained on the hand-labeled dataset but with a population of 4. EDA refers to the Easy Data Augmentation dataset, and ALD refers to the automatically labeled dataset.

Table 6: Macro average metrics and Cohen's Kappa for models trained and tested on Hand-label, ALD, and EDA datasets. Hand-label (4) indicates training with population = 4.

Training Set	Test Set	Training Time	Precision	Recall	F1	Cohen's Kappa
Hand-label (4)	Hand-label	1h 40min	0.480	0.430	0.450	0.329
Hand-label (4)	ALD	1h 40min	0.350	0.300	0.280	0.125
Hand-label	Hand-label	5min 47s	0.410	0.300	0.280	0.260
Hand-label	ALD	5min 47s	0.270	0.280	0.180	0.095
EDA	Hand-label	27min 32s	0.420	0.340	0.360	0.254
EDA	ALD	27min 32s	0.360	0.260	0.230	0.079
ALD	ALD	24min 34s	0.430	0.440	0.430	0.295
ALD	Hand-label	24min 34s	0.280	0.310	0.260	0.108

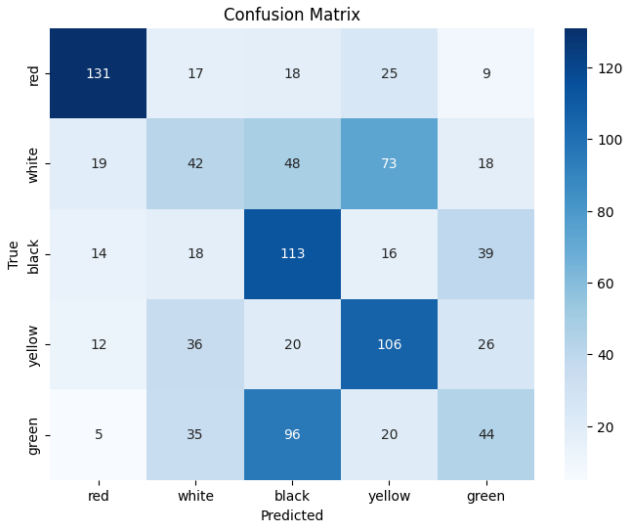


Figure 3: Confusion matrix on the ALD test set – model trained on ALD.

Observations

The model trained on `Hand-label` with a population size of 4 achieved the most consistent results across datasets. In-domain performance was highest for the ALD-trained model on ALD (accuracy = 0.436), but it exhibited poor generalization. Larger population size in PBT (as in `Hand-label (4)`) improved generalization at the cost of longer training time. As we can see in Table 7 below, comparing DistilRoBERTa with the other models which was not trained using the PBT technique, there is no such improvement.

Table 7: Comparison of BERT-family models trained and tested on ALD (automatically labeled dataset)

Class	**BERT**			**DistilBERT**			**RoBERTa**			**DistilRoBERTa**		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
black	0.51	0.59	0.55	0.49	0.56	0.52	0.45	0.58	0.51	0.38	0.56	0.46
green	0.65	0.37	0.47	0.50	0.70	0.58	0.31	0.40	0.35	0.22	0.26	0.24
red	0.75	0.77	0.76	0.74	0.68	0.71	0.72	0.68	0.70	0.72	0.66	0.69
white	0.42	0.39	0.41	0.50	0.17	0.25	0.50	0.17	0.25	0.28	0.21	0.24
yellow	0.50	0.67	0.57	0.51	0.61	0.56	0.53	0.22	0.31	0.53	0.22	0.31
Weighted avg	-	-	0.55	0.55	0.55	0.53	-	-	0.57	0.43	0.44	0.43

5 Conclusion and Future Work

This project presents a novel approach to intent classification by leveraging Edward de Bono's Six Thinking Hats framework—a direction, to the best of our knowledge, that has not been previously explored in the NLP field. Unlike traditional intent detection tasks, which focus on surface-level actions, our goal was to uncover the cognitive mode behind an utterance: whether it reflects facts, emotions, critique, optimism, or creativity.

One of the main challenges we encountered was the complete absence of any dataset labeled according to the Six Hats framework. As a result, we had to adapt an existing dataset (DailyDialog), and devise a multi-step pipeline to automatically annotate utterances based on various linguistic properties.

To achieve this, we combined techniques and models from several research papers, each focusing on a specific linguistic dimension, such as optimism, creativity, or critical reasoning. These techniques were integrated into a unified system that attempted to label DailyDialog utterances into the Six Hats categories. Naturally, this introduced significant noise and inconsistencies in the training data, which likely influenced the overall performance of the models. We believe that access to a carefully labeled dataset, specifically designed for this task, would have led to better generalization and deeper insights.

Looking ahead, one promising direction for future work is multimodal analysis. Language is not just textual—it is also vocal and visual. Important cues such as tone of voice, facial expressions, and gesture all contribute to how intent is perceived and should not be ignored. If annotated video and audio datasets were available, and sufficient computational resources could be allocated, we could significantly enrich the classification process and capture more nuanced forms of human communication.

In conclusion, this work represents a first step toward cognitive-level intent classification, pushing the boundaries of traditional dialogue understanding and opening new avenues for research in explainable and human-centric NLP.

References

Mistral AI. Mistral-7b. `https://mistral.ai/news/announcing-mistral-7b/`, 2023. Accessed: 2024-06.

Lorin W. Anderson, David R. Krathwohl, and Benjamin S. Bloom. *A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives*. Longman, New York, 2000. ISBN 978-0-8013-1903-7.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2018. URL https://arxiv.org/abs/1810.04805. Version Number: 2.

Qinxu Ding, Ding Ding, Yue Wang, Chong Guan, and Bosheng Ding. Unraveling the landscape of large language models: a systematic review and future perspectives. *Journal of Electronic Business & Digital Economics*, 2023. URL https://api.semanticscholar.org/CorpusID:266379549.

Avron Spiro Deborah A. DeMolles Helen Achat, Ichiro Kawachi and David Sparrow. Optimism and depression as predictors of physical and mental health functioning. 08 2000.

HuggingFace. Distilroberta base model, 2019. URL https://huggingface.co/distilbert/distilroberta-base#model-details. Available on HuggingFace Hub.

Nicholas R Martin J Patrick Sharpe and Kelly A Roth. Optimism and the big five factors of personality: Beyond neuroticism and extraversion. personality and individual differences. 08 2011.

Max et al. Jaderberg. Population based training of neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, 2017.

Anna Jordanous. Defining creativity: Finding keywords for creativity using corpus linguistics techniques. In *Proceedings of the International Conference on Computational Creativity*, Lisbon, Portugal, 2010. URL https://computationalcreativity.net/iccc2010/papers/jordanous-2.pdf.

Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. Dailydialog: A manually labelled multi-turn dialogue dataset. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 986–995. Asian Federation of Natural Language Processing, 2017. URL https://aclanthology.org/I17-1099.

Zhuang Liu, Wayne Lin, Ya Shi, and Jun Zhao. A robustly optimized bert pre-training approach with post-training. In *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, pages 1218–1227, Huhhot, China, 2021. Chinese Information Processing Society of China.

Michael Mumford, Denise L. Whetzel, and Roni Reiter-Palmon. Thinking creatively at work: Organization influences on creative problem solving. *The Journal of Creative Behavior*, 31(1): 7–17, 1997. doi: 10.1002/j.2162-6057.1997.tb00777.x. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/j.2162-6057.1997.tb00777.x.

Tomer Raz, Roni Reiter-Palmon, and Yoed N. Kenett. The role of asking more complex questions in creative thinking. *Psychology of Aesthetics, Creativity, and the Arts*, 2023. doi: 10.1037/aca0000658.

Roni Reiter-Palmon, Michael D. Mumford, and Kimberly V. Threlfall. Solving everyday problems creatively: The role of problem construction and personality type. *Creativity Research Journal*, 11 (3):187–197, 1998. doi: 10.1207/s15326934crj1103_3.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108, 2019. URL https://api.semanticscholar.org/CorpusID:203626972.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Jason Wei and Kai Zou. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6382–6388, Hong Kong, China, 2019. Association for Computational Linguistics. URL https://aclanthology.org/D19-1670.

Korbin M. Kenett Y. N. Dan D. Ganzha M. Paprzycki M. Wróblewska, A. Applying text mining to analyze human question asking in creativity research. 07 2025. URL https://arxiv.org/abs/2501.02090.

A Appendix / supplemental material

Supplemental material (complete proofs, additional experiments and plots).

Table 8: roBERTa Metrics Across Different Datasets

Training-Set	**Test-Set**	**Class**	**Precision**	**Recall**	**F1-score**
Hand-Labeled	Hand-Labeled	red	0.46	0.57	0.51
		white	0.81	0.65	0.72
		black	0.33	0.57	0.42
		yellow	0.57	0.42	0.48
		green	0.23	0.25	0.24
		Weighted Avg. F1	0.59 (204 samples)		
EDA Hand-Labeled	Hand-Labeled	red	0.49	0.47	0.48
		white	0.70	0.78	0.74
		black	0.50	0.26	0.34
		yellow	0.53	0.53	0.53
		green	0.25	0.25	0.25
		Weighted Avg. F1	0.60 (204 samples)		
automatically Labeled	automatically Labeled	red	0.75	0.77	0.76
		white	0.42	0.39	0.41
		black	0.51	0.59	0.55
		yellow	0.50	0.67	0.57
		green	0.65	0.37	0.47
		Weighted Avg. F1	0.55 (1000 samples)		
automatically Labeled	Hand-Labeled	red	0.32	0.30	0.31
		white	0.69	0.47	0.56
		black	0.19	0.22	0.20
		yellow	0.21	0.58	0.31
		green	0.25	0.25	0.25
		Weighted Avg. F1	0.43 (204 samples)		