Training-Free Text Emotion Tagging via LLM-Based Best-Worst Scaling

Lukas Christ[1], **Shahin Amiriparian**[1,2]
[1]Huawei, The Netherlands [2]TU Munich, Germany
lukas.christ1@huawei.com

Abstract

Large Language Models (LLMs) have been frequently used as automatic annotators for tasks such as Text Emotion Recognition (TER). We consider a scenario in which annotators assign at least one emotion label from a large set of options to a text snippet. For this *emotion tagging* task, we propose a novel zero-shot algorithm that leverages Best-Worst Scaling (BWS), prompting the LLM to choose the least and most suitable emotions for a given text from several label subsets. The LLM's choices can be represented by a graph linking labels via *worse-than* relations. Random walks on this graph yield the final score for each label. We compare our algorithm with naive prompting approaches as well as an established BWS-based method. Extensive experiments demonstrate the suitability of the method. It proves to compare favorably to the benchmarks in terms of both accuracy and calibration with respect to human annotations. Moreover, our algorithm's automatic annotations are shown to be suitable for finetuning lightweight emotion classification models. The proposed method consumes considerably fewer computational resources than the established BWS approach.

1 Introduction

Large Language Models (LLMs) are often utilized for automatically labeling textual data in a zero-shot manner, thus avoiding the tedious work of collecting labels from humans. To give but a few examples, zero-shot prompting-based prediction has been applied to stance detection, humor recognition, and Text Emotion Recognition (TER) (Ziems et al., 2024). Such tasks are typically approached in a straightforward manner by prompting an instruction-tuned LLM with a question such as *Which emotion does the given text express?*, usually in combination with further instructions on the task and the desired return format.

In this work, we propose a more sophisticated zero-shot approach to the TER problem intended to improve upon such simple approaches in terms of both accuracy and calibration. Specifically, we consider a variant of TER which we dub *emotion tagging*: given a text snippet t and an extensive set of emotion labels L, t should be assigned a subset of L describing the emotions potentially expressed in t. We assume that L goes considerably beyond the rather small emotion label sets often encountered in TER studies, which typically do not exceed 6-10 choices (Alm et al., 2005; Saravia et al., 2018; Öhman et al., 2020; Yu et al., 2024). Moreover, the texts in question may (potentially) exhibit a number of different emotions at different degrees, as is typical for short, potentially ambiguous sentences lacking context.

Given the size of L, naive approaches that simply "ask" the model to provide a score for each $l \in L$ confront the LLM with a complicated task, potentially impairing its performance. Hence, we propose a novel framework that breaks down the emotion tagging problem into smaller problems via Best-Worst Scaling (BWS). Introduced by (Louviere and Woodworth, 1991), BWS is devised for tasks that require ranking a set of choices L with respect to some criterion or item t. Instead of presenting annotators with the entire set L, small subsets (referred to as *tuples* henceforth) of size M are created and the annotator asked to select the most suitable (*best*) and least suitable (*worst*) choices from each such tuple, thus reducing the cognitive complexity of the task. From the responses, numerical scores can be computed in different ways (Louviere et al., 2015). Moreover, BWS does not rely on asking numeric values on a Likert scale from annotators. Likert scales come with several limitations that are mitigated by BWS forcing annotators to choose a best and worst option. For example, different annotators may interpret and use the items of the scale in different ways, making inter-annotator comparisons difficult (Baumgartner and

Findings of the Association for Computational Linguistics: EACL 2026, pages 6678–6694

Steenkamp, 2001). Moreover, *central tendency bias*, i.e., annotators avoiding the "extreme" options of the scale, is often observed in Likert scale ratings (Stevens, 1971; Douven, 2018). For a more thorough discussion of the advantages of BWS over Likert scales, see, e.g., (Heo et al., 2022).

We cast emotion tagging as a BWS task. BWS has been utilized for TER via LLM before (Bagdon et al., 2024; Zhang and Feng, 2025; Duong et al., 2025). In these works, tuples are constructed from texts and evaluated with respect to a given emotion, such that a large text corpus must be available. Our approach, in contrast, builds tuples from emotions and evaluates them regarding a given text, thus eliminating this prohibitive prerequisite. Furthermore, we improve upon previous work by proposing a novel graph-based scoring method that leads to well-calibrated predictions, a factor neglected by existing BWS-based TER methods.

We show that the proposed method outperforms naive prompting approaches and previous LLM-based BWS approaches both in terms of agreement with human annotations and calibration to them. Moreover, the scores obtained by our framework prove to be useful as pseudo-labels for automatic TER in a supervised finetuning setting. Besides, our method is shown to make efficient use of computational resources.

2 Related Work

A plethora of works employing LLMs as automatic annotators for text snippets exists. (Gilardi et al., 2023) investigate ChatGPT's performance on four problems including stance detection. (Rathje et al., 2024) prompt different GPT versions (Brown et al., 2020) on 15 psychology-related datasets, among them sentiment analysis and TER corpora. To the best of our knowledge, the most extensive study of this kind is conducted by (Ziems et al., 2024), who analyze the zero-shot performance of 13 different LLMs on 25 datasets related to (computational) social science, including tasks such as humor detection and TER. All of these works, however, are restricted to rather coarse-grained emotion taxonomies comprising up to 8 *basic* emotions following theories as proposed by Ekman (Ekman, 1992) or Plutchik (Plutchik, 1982). In this work, we employ BWS to equip LLMs with the capacity to efficiently predict more extensive sets of emotions. In general, BWS has enjoyed increasing popularity in several research fields recently (Schuster et al., 2024). It has also been applied to obtain human annotations for emotions in texts, namely to obtain valence, arousal, and dominance scores for words (Mohammad, 2018) and emotion intensity labels for tweets (Mohammad and Bravo-Marquez, 2017; Mohammad and Kiritchenko, 2018).

A few studies combine BWS with LLMs. Arguably most relevant to our work, (Bagdon et al., 2024) closely follow the BWS-based annotation process of the dataset introduced by (Mohammad and Kiritchenko, 2018), but replace human annotators with different LLMs. Their experiments demonstrate the feasibility of LLM-based BWS annotations for the problem of emotion intensities. Different from our setup, they generate tuples consisting of texts and rank them with respect to a given emotion label, where the label set only comprises *anger*, *fear*, *joy*, and *sadness*. When N is the data set size, $2N$ tuples are evaluated and the scores $s(t, l)$ (t text snippet, $l \in L$) are simply estimated via Equation (1), i.e., by counting how often a text was ranked best and worst, respectively.

$$s(t,l) = \frac{\#\text{best}_l(t) - \#\text{worst}_l(t)}{\#\text{overall}_l(t)} \quad (1)$$

This ignores the other *better-than* relations implied by best-worst choices: if A is the best and D the worst item from $\{A, B, C, D\}$, this also implies $A > \{B, C, D\}$ and $D < \{A, B, C\}$, but this information is discarded by Equation (1). Our graph-based method (cf. Section 3) seeks to leverage these relations as well. Other than our method, the approach by (Bagdon et al., 2024) requires a sufficiently large textual data set and assumes that the 4 emotions in question are present in the data at a sufficient degree. Our framework, in contrast, selects emotions per text example and thus does not rely on any assumptions about the dataset size or the distribution of emotion intensities or labels within the data. The method by (Bagdon et al., 2024) is applied by (Zhang and Feng, 2025) for TER of 9 emotions in student comments. Similarly, (Duong et al., 2025) perform LLM-based BWS for 6 emotions.

3 Methods

Our approach consists of three steps: First, BWS samples are automatically generated taking the structure of the label space L into account (Section 3.1). Next, an LLM conducts BWS on these samples (Section 3.2), which leads to a graph in

which nodes represent emotions from the label set L and edges represent the relations among them yielded by BWS. Finally, we compute fine-grained scores from this graph using random walks (Section 3.3). The entire pipeline is visualized in Figure 1. In the following, N denotes the size of the dataset and M is a parameter specifying the tuple size.

3.1 Pseudo-VAD Clustering

Discrete emotions such as *anger*, *disgust*, *joy*, etc. do not form a homogeneous space. Some emotions may often occur simultaneously (e.g., *anger* and *disgust*), while others are usually incompatible (e.g., *sadness* and *joy*). One way to account for such differences is mapping discrete emotions into the continuous Valence/Arousal (V/A) space (Russell and Mehrabian, 1977), as done in several previous emotion recognition studies (Gerczuk et al., 2021; Park et al., 2021; Christ et al., 2024; Amiriparian et al., 2024). *Valence* refers to pleasantness and *arousal* to the degree of excitement.

Such a two-dimensional representation of the labels L is leveraged to construct informative tuples for BWS. These tuples specifically group a) similar and b) dissimilar emotions, forcing the model to make both coarse-grained (e.g., *is the author rather angry or happy?*) and fine-grained (e.g., *is the author rather angry, annoyed, or disgusted?*) decisions. We generate the V/A mapping for the emotions in L by prompting an LLM. The tuples are constructed via clustering of the V/A representations. In the initial sample generation, the special label *neutral*, arguably not an emotion, is excluded. We generate four sets of $\lfloor \frac{|L|-1}{M} \rfloor$ tuples, each of them defining a non-overlapping partition of $L \setminus \{\textit{neutral}\}$: 1) emotions similar in terms of valence, 2) emotions diverse in terms of valence, 3) emotions similar in terms of arousal, and 4) emotions diverse in terms of arousal. The maximum tuple size is limited to $M + 2$. Thus, overall $4 * \lfloor \frac{|L|-1}{M} \rfloor$ tuples of up to $M + 2$ labels (including *neutral*) are generated. This process, its result and its validation with human labels are described in detail in Section A. The label *neutral* is added to each of the tuples in order to always provide the special choice of no emotion at all. Moreover, this ensures that each label is compared with *neutral*, thus preventing isolated components in the graph built in the next step.

3.2 LLM BWS

The tuples are presented to an instruction-finetuned LLM in the format shown in Figure 1. From all decisions, the implied *better-than* relations are inferred. The union of all relations forms a *worse-than* graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where each vertex $\mathcal{V}$ represents a label from L, and an edge (u, v) means that u has been evaluated as less suitable than v.

3.3 Random Walks

Given $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, one way of modeling the decision process of a single annotator is to assume that the annotator moves through $\mathcal{V}$ via the edges, whereas stopping this trajectory at a node $u \in \mathcal{V}$ means selecting the label corresponding to u. Emotion tagging scores $s : L \to [0, 1]$ can be obtained by simulating this annotation process n times, leading to n label sets $\{\hat{Y}_1 \subseteq L, ..., \hat{Y}_n \subseteq L\}$. From them, s can be computed in a simple manner as the relative frequencies, cf. Equation (2).

$$s(l) = \frac{|\{\hat{Y}_i \mid l \in \hat{Y}_i|\}|}{n} \tag{2}$$

We now describe the proposed annotation process model in detail. Let $in(u) := \{v \in \mathcal{V} \mid (v, u) \in \mathcal{E}\}$ and $out(u) := \{v \in \mathcal{V} \mid (u, v) \in \mathcal{E}\}$. Intuitively, $in_{\mathcal{G}}(u)$ and $out_{\mathcal{G}}(u)$ correspond to emotions evaluated as worse suitable and better suitable than the emotion represented by u, respectively. Moreover, let $rand()$ denote a function that uniformly samples a random number in $[0, 1]$. Assuming that there is an *node importance function* $\mathcal{I} : \mathcal{V} \to \mathbb{R}_0^+$, a simple random walk on the annotation graph is defined by Algorithm 1.

Algorithm 1 Single Random Walk algorithm.

```
1: function RandomWalk((V, E), I : V → R_0^+)
2:     u := sample random node from V uniformly
3:     while True do
4:         if out(u) == ∅ then
5:             return u
6:         p_stop = I(u) / (I(u) + Σ_{v∈out(u)} I(v))
7:         if rand() < p_stop then
8:             return u
9:         u := sample random node from out(u) using I
```

To summarize Algorithm 1, a random walk is conducted on the annotation graph with the stopping criterion depending on $\mathcal{I}$. The process always stops at sinks, i.e., emotion labels which were deemed best in every tuple they occur in (cf. line 4). For any other node u, the process stops with probability $\frac{I(u)}{I(u)+\sum_{v \in out(u)} I(v)}$, thus accounting for the

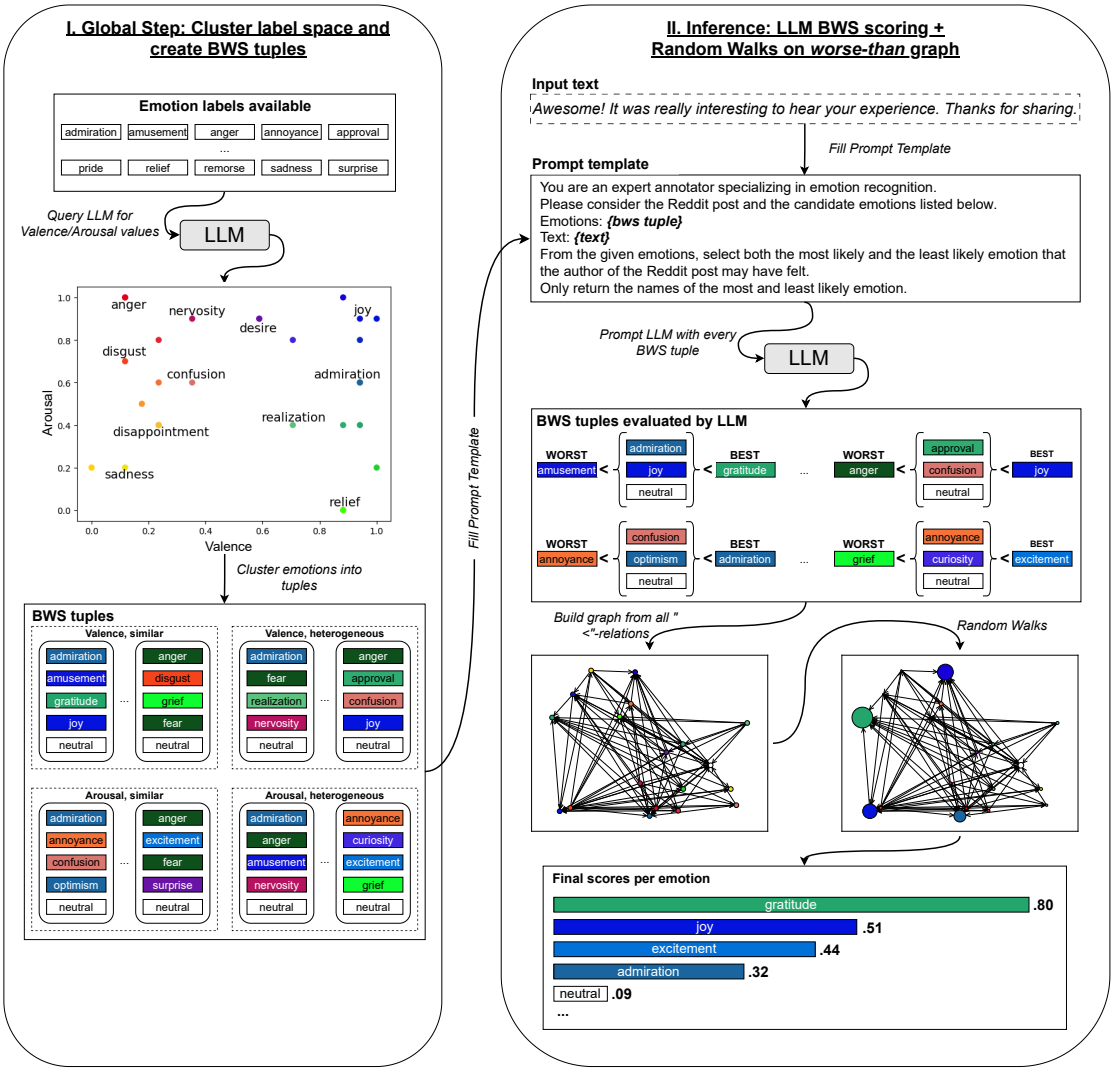

Figure 1: Overview of the proposed method. The graph is simplified with fewer vertices and edges for clarity.

importance of its successors. If the process does not stop, the next node is sampled from $out(u)$, taking the importance function $\mathcal{I}$ into account. Algorithm 1 only selects one node, i.e., emotion. In the emotion tagging task, however, multiple emotions can be assigned to a given text. To account for this, we propose Algorithm 2.

Essentially, SIMULATEANNOTATION executes Algorithm 1 multiple times (lines 14-24). After each execution, the selected node u is added to $\hat{Y}$ (l.16) and removed from the graph which is also extended with edges (v, w), if v is indirectly connected to w via u (lines 21-23). The stopping criterion for this multilabel selection process relies on a weight function $\mathcal{W} : \mathcal{V} \rightarrow [0, 1]$ with $\sum_{u \in \mathcal{V}} \mathcal{W}(u) = 1$. After each update of the set of selected nodes $\hat{Y}$, the entire process is halted with probability $\sum_{u \in \hat{Y}} \mathcal{W}(u)$ (cf. lines 17-20). The importance function $\mathcal{I}$ utilized for the random walks is initialized with a simple score depending on how many emotion labels are deemed better or worse, respectively, cf. line 2. Subsequently, $\mathcal{W}$ is initialized via the results of 100 simple random walks (lines 3-7).

4 Experimental Setup

4.1 Dataset

Most TER datasets feature only a comparatively small set of no more than 8 labels (Alm et al., 2005; Öhman et al., 2020; Duong et al., 2025), rendering them not suitable for the emotion tagging problem, which presupposes a wide range of fine-grained emotion labels. The GoEmotions (Demszky et al., 2020) dataset is arguably the only notable exception, featuring 27 emotion classes plus *neutral*.

Algorithm 2 Annotation Algorithm.

function INITANNOTATION($(\mathcal{V}, \mathcal{E})$)
  let $\mathcal{I}(u) := \frac{|in(u)|}{|in(u)|+|out(u)|}\ \forall u \in \mathcal{V}$
  let $\mathcal{W}(u) := 0\ \forall u \in \mathcal{V}$
  for 100 times **do**
    $u :=$ RANDOMWALK($(\mathcal{V}, \mathcal{E}), \mathcal{I}$)
    $\mathcal{W}(u) := \mathcal{W}(u) + \frac{1}{100}$
  return $\mathcal{W}, \mathcal{I}$

function SIMULATEANNOTATION($(\mathcal{V}, \mathcal{E})$)
  $\mathcal{W}, \mathcal{I} :=$ INITANNOTATION($(\mathcal{V}, \mathcal{E})$)
  $\hat{Y} := \emptyset$
  while $|\hat{Y}| < |\mathcal{V}|$ **do**
    $u :=$ RANDOMWALK($(\mathcal{V}, \mathcal{E}), \mathcal{I}$)
    $\hat{Y} := \hat{Y} \cup \{u\}$
    $p_{stop} := \sum_{v \in \hat{Y}} \mathcal{W}(v)$
    if $rand() < p_{stop}$ **then**
      return $\hat{Y}$
    $\mathcal{V} := \mathcal{V} \setminus \{u\}$
    $\mathcal{E} := \mathcal{E} \cup \{(v, w) \mid (v, u) \in \mathcal{E} \wedge (u, w) \in \mathcal{E}\}$
    $\mathcal{E} := \mathcal{E} \setminus \{(v, w) \mid (v == u) \vee (w == u)\}$
  return $\hat{Y}$

Each datapoint in GoEmotions is a Reddit post labeled by at least three annotators, who are allowed to select more than one label per text. Despite this multilabel setup, only around 17 % of the texts are assigned more than one emotion.

For our experiments, we construct a subset of 5000 Reddit posts from GoEmotions. This is motivated by two considerations. First, we conduct a large number of resource-intensive experiments, which would not be feasible with the full dataset. Second, the majority of data points in GoEmotions is only annotated with one label, and hence of limited relevance to the problem at hand. We build our GoEmotions subset along the following considerations: first, making sure to consider multilabel examples, we include all 561 texts with at least three different labels as well as those with two labels where one of them is *neutral* (1396). Second, to ensure sufficient representation of all labels, all texts annotated with underrepresented emotions, i.e., those that occur fewer than $\lceil \frac{5000}{|L|} \rceil = \lceil \frac{5000}{28} \rceil$ times in the whole dataset, are considered (503 data points). 2347 texts match at least one of these criteria. The remaining datapoints are randomly sampled from GoEmotions. Further information on the dataset can be found in Section B.

From the 5000 examples, a small validation set of size 200 is separated for the purpose of hyperparameter search. Arguably, in a real-world scenario, it is realistic to curate such a small-scale data set manually to optimize automatic labeling approaches.

A pressing issue in the evaluation of LLM-based methods is the problem of potential training data contamination (Sainz et al., 2023; Li and Flanigan, 2024). If an LLM is exposed to a benchmark dataset during pretraining, the thus acquired information invalidates zero-shot evaluation procedures of the model using this dataset. In order to check the LLMs for contamination with the GoEmotions dataset, we a) check the provided documentation regarding their pretraining data and b) apply a variant of Data Contamination Quiz (Golchin and Surdeanu, 2023). The details for the latter can be found in Section D. We find no indication that the models have memorized the GoEmotions data during their pretraining procedure.

4.2 Hyperparameters

Both the proposed method and Text-BWS introduce a hyperparameter M related to the size of the BWS tuples: in (Bagdon et al., 2024), M determines the size of each tuple, while in our tuple construction algorithm (cf. Section A), tuple sizes are M+2 at most. M is optimized via the validation set for both methods. Among candidates ranging from 2 to 9, $M = 4$ is selected for our approach and $M = 6$ for Text-BWS. To obtain the final scores via Equation (2), we run SIMULATEANNOTATION (cf. Algorithm 2) $n = 100$ times.

4.3 Benchmarks

Our method's performance is compared against the following prompt-based approaches. The exact prompts can be found in Section C.

Direct asking, all at once (D-A): for each text snippet t, the LLM is prompted to provide scores for all emotions in L at once. Hence, this method entails N prompts.

Direct asking, individual (D-I): alternatively, a prompt is presented for each emotion individually. Hence, $N * |L|$ prompts are required.

Text-BWS (Bagdon et al., 2024) employ BWS for emotion intensity estimation. Different from our method, their BWS tuples comprise sentences instead of emotions. Hence, we refer to this approach as *Text-BWS* in the following. Given an emotion $l \in L$, $2N$ such tuples are presented to the model. A score for text t and emotion l is calculated based on the number of times t is selected as the best and worst option, respectively, cf. Equation (1). These values are subsequently normalized to the range $[0, 1]$. Text-BWS is of considerable complexity, requiring $2 * N * |L|$ prompts. In con-

trast, our approach requires only $4 * \lfloor \frac{L-1}{M} \rfloor * N$ prompts, as there are 4 partitions of $|L-1|$ labels (excluding *neutral*)..

The prompts utilized for the baseline methods are provided in Section C.

For all three approaches, as well as ours, we additionally apply a **rescaling** approach to the predictions in order to increase the F1-scores. Specifically, for each $l \in L$, a threshold τ_l is selected that maximizes the respective label's F1 score on the validation set. Then, a prediction $\hat{y}$ is rescaled to $\hat{y}'$ as specified in Section 4.3.

$$\hat{y}' = \begin{cases} 0.5 * \frac{\hat{y}}{\tau_l} & \text{if } \hat{y} \leq \tau_l \\ 0.5 + 0.5 * \frac{\hat{y}-\tau_l}{1-\tau_l} & \text{else} \end{cases} \quad (3)$$

4.4 Models and Decoding

We conduct our experiments with different LLMs of the open-source OLMO-2 family (OLMo et al., 2024). We experiment with OLMO-2 Instruction-finetuned models of different sizes, namely 1B, 7B, 13B, and 32B parameters[1]. In every experiment, the model's response is obtained via greedy decoding.

All prompting-based methods are implemented via the `outlines` (Willard and Louf, 2023)[2] library. Its structured decoding options are utilized to ensure that the model outputs adhere to predefined JSON formats. In some cases, however, we still encounter invalid outputs. If the score for an emotion is missing, it is assumed to be 0. If the score for an emotion is given twice, the latter value is chosen. In case the best or worst guess returned by the model is not part of the tuple in question, it is treated as if it actually were.

4.5 Evaluation

The scores are evaluated against the human labels for the data. This, however, comes with caveats. First, the labels in GoEmotions are arguably sparse, i.e., the annotators seemingly preferred selecting rather few labels per instance. Second, the labels are binary, thus not accounting for the gradual nature of emotional expressions. Faced with these challenges, we employ different metrics to obtain a holistic impression of the different methods' performances: the **F1**-score indicates to what degree a method can directly be used as a classifier for the human scores. We consider all scores greater than or equal 0.5 as positive predictions for the respective class. To get a better understanding of how well a method's scores separate positive from negative instances per emotion, we additionally compute the **Area under the ROC Curve (AUC)** w.r.t. the human annotations. Lastly, a paramount demand for automatically generated continuous scores is that they are calibrated with respect to human annotations. We thus compute the approximated **Expected Calibration Error (ECE)** utilizing 10 bins of equal width.

Following (Bagdon et al., 2024), we also investigate the *usefulness* of automatic scores for training an emotion classifier. We finetune a 86M-parameter DeBERTa-V3 *base* model (He et al., 2023)[3] on the automatically generated scores as well as on the human labels from the dataset. The model's performance is evaluated on the complete test dataset of GoEmotions via the Macro-F1 score. Detailed hyperparameters of these experiments are described in Section F.

4.6 Ablation Studies

We investigate several variants of our methods. To assess the impact of V/A clustering, we replace the $4 * \lfloor \frac{|L|-1}{M} \rfloor$ Valence and Arousal-based clusters/tuples (cf. Section 3.1) with a) $2 * \lfloor \frac{|L|-1}{M} \rfloor$ Valence-only clusters ($\mathcal{C}_V^{sim}$ and $\mathcal{C}_V^{div}$), b) $2 * \lfloor \frac{|L|-1}{M} \rfloor$ Arousal-only clusters ($\mathcal{C}_A^{sim}$ and $\mathcal{C}_A^{div}$), and c) $4 * \lfloor \frac{|L|-1}{M} \rfloor$ random partitions of L.

Moreover, we validate the impact of the proposed random walk method. Instead of the proposed algorithm (cf. Algorithm 2), we experiment with a) computing scores via Equation (1) without any graph construction and b) modifying SIMULATEANNOTATION in Algorithm 2 such that only one vertex/label is selected, thus abandoning the multi-label selection approach.

In all ablation studies, the 32B variant of OLMO-2 is utilized.

4.7 Computational Setup

All experiments are carried out on three NVIDIA-RTX6000 cards. The models are always loaded in their 16-bit quantized versions. A rough estimation of the overall GPU hours spent for this work is 200 hours. The experiments are carried out with batch sizes between 2 and 8, depending on the

[1] huggingface checkpoints: allenai/OLMo-2-{0425-1B, 1124-7B, 1124-13B, 0325-32B}-Instruct

[2] https://dottxt-ai.github.io/outlines

[3] https://huggingface.co/microsoft/deberta-v3-base

model sizes and available GPU resources. To give a general estimation for the resource requirements, we perform 500 prompts for each experimental configuration with a batch size of 1 and extrapolate to estimate the required GPU time, input tokens, and output tokens. The result of these estimations are presented in Section 5.4.

5 Results

The following presents the results for the comparison with human labels (Section 5.1), the scores' usefulness in training TER classifiers (Section 5.2), and a closer analysis of resource requirements (Section 5.4).

5.1 Comparison with Human Labels

Table 1 lists the AUC, F1, and ECE values obtained for our method and the benchmarks. As the results for the 1B model only slightly surpass chance level for AUC (.5), we omit them in the following.

Model	Method	AUC [↑]	F1 [↑]	ECE [↓]
7B	D-A	.6831 (±.09)	.1742 (±.10)	.1507 (±.13)
	+*rescaling*	.6831 (±.09)	.1983 (±.12)	.2036 (±.12)
	D-I	.7555 (±.09)	.1565 (±.08)	.4250 (±.15)
	+*rescaling*	.7555 (±.09)	.2372 (±.10)	.2627 (±.10)
	Text-BWS	**.7733** (±.10)	.1369 (±.09)	.4448 (±.06)
	+*rescaling*	**.7733** (±.10)	**.2594** (±.12)	.2946 (±.05)
	ours	.7443 (±.09)	.2153 (±.14)	**.0885** (±.09)
	+*rescaling*	.7443 (±.09)	.2652 (±.13)	.1020 (±.07)
13B	D-A	.7439 (±.09)	.2517 (±.13)	.0884 (±.07)
	+*rescaling*	.7439 (±.09)	.2708 (±.13)	.1020 (±.06)
	D-I	**.8076** (±.08)	.3136 (±.09)	.1410 (±.09)
	+*rescaling*	**.8076** (±.08)	**.3327** (±.12)	.1362 (±.08)
	Text-BWS	.7842 (±.10)	.1374 (±.09)	.4455 (±.06)
	+*rescaling*	.7842 (±.10)	.2647 (±.12)	.2794 (±.05)
	ours	.7702 (±.08)	.2551 (±.12)	**.0755** (±.07)
	+*rescaling*	.7702 (±.08)	.2932 (±.12)	.0940 (±.07)
32B	D-A	.7363 (±.08)	.3226 (±.11)	.0915 (±.07)
	+*rescaling*	.7363 (±.08)	.3301 (±.13)	.0743 (±.05)
	D-I	.7801 (±.08)	.2439 (±.11)	.2116 (±.11)
	+*rescaling*	.7801 (±.08)	.2980 (±.11)	.1692 (±.11)
	Text-BWS	**.8123** (±.09)	.1446 (±.10)	.4428 (±.07)
	+*rescaling*	**.8123** (±.09)	.2919 (±.11)	.2836 (±.06)
	ours	.8119 (±.07)	**.3617** (±.11)	**.0604** (±.06)
	+*rescaling*	.8119 (±.07)	.3531 (±.13)	.0614 (±.04)

Table 1: Experimental results for the comparison of different methods' predictions with the gold standard labels. The *Model* column refers to the different variants of OLMo-2 (cf. Section 4.4). All values are means and standard deviations over the 28 emotion labels in the dataset. Each experiment is deterministic due to greedy decoding and hence only executed once. Text-BWS refers to the method introduced by (Bagdon et al., 2024)

All methods exhibit improvements w.r.t. F1 and AUC with growing model size. Our method is better calibrated than the benchmarks for 7B, 13B, and 32B. While for the 7B and 13B model, our method is clearly outperformed by D-I and Text-BWS in terms of AUC and F1, these two methods prove to yield considerably miscalibrated predictions. To give an example, for the 7B model, the ECE of D-I before rescaling is .4250, while our approach's ECE is .0885. For the 32B model, the proposed method outperforms D-I and D-A also w.r.t. AUC and F1. Text-BWS outperforms our method in terms of AUC for the 32B model, but only by a slight margin (.8123 vs. .8119 AUC), while yielding lower F1 scores, considerably worse calibration. Thus, our approach accounts for the best mean F1 results overall, namely .3617.

The rescaling approach almost always improves the F1 scores, with the largest absolute improvement being around 15 percentage points for Text-BWS on the 32B model. When the ECE value is larger than .1, rescaling tends to also improve the predictions' calibration, while for lower ECE values, it usually increases them. Consequently, the lowest ECE overall, namely .0604 for our method with the 32B model, is achieved without rescaling. Calibration curves for all 28 labels can be found in Section G. Section H describes an example in detail.

5.2 Text Emotion Recognition Training

The results for training DeBERTa models on the different methods' predictions are given in Table 2.

F1 [↑]	7B	13B	32B
D-A	.0198 (± .03)	.0508 (± .06)	.2015 (± .08)
+*rescaling*	.0849 (± .01)	.1504 (± .00)	.1251 (± .10)
D-I	.0995 (± .02)	.1970 (± .10)	.2069 (± .01)
+*rescaling*	**.1709** (± .00)	.1421 (± .11)	.1170 (± .08)
Text-BWS	.1082 (± .00)	.1114 (± .00)	.1118 (± .00)
+*rescaling*	.1417 (± .09)	**.2141** (± .00)	.1715 (± .09)
ours	.0037 (± .06)	.1745 (± .09)	**.3346** (± .01)
+*rescaling*	.1339 (± .10)	.1619 (± .12)	.3207 (± .02)
human labels		**.4384** (±.01)	

Table 2: Results for training classifiers on the automatically predicted emotion annotations. The result for training on the human labels is given in the last row for convenience. All classifiers are evaluated on the human-labeled GoEmotions test set. Here, the means and standard deviations of Macro-F1 scores over 5 fixed random seeds are reported.

Our approach proves to be superior over the benchmarks when using the predictions by the 32B model. It accounts for the highest mean macro-F1

score of .3346. Notably, this result is still around 10 percentage points worse than training on the human-assigned labels, i.e., .4384. None of the other three methods exceed a mean macro-F1 score of .2141 as observed for Text-BWS with the 13B model and without rescaling. The results for the 7B and 13B model mirror the comparison with human labels (cf. Table 1), where our method proves to be less accurate than D-I and Text-BWS. A comparison between initial and rescaled predictions does not show a clear pattern for any method. To give an example, rescaling improves the D-I results for the 7B predictions by around 7 percentage points, but worsens them for the 13B and 32B predictions by around 5 and 8 percentage points, respectively. We assume that our method's superior performance can be attributed to the combination of relatively high agreement with human labels on the one hand and good calibration to them on the other.

5.3 Ablation Study Results

The results of the ablation studies with the 32B model are presented in Table 3.

	AUC [↑]	F1 [↑]	ECE [↑]
Reference			
ours	.8119 (±.07)	**.3617** (±.11)	.0604 (±.06)
BWS tuple variations			
Valence only	.7950 (±.08)	.3391 (±.11)	.0634 (±.05)
Arousal only	.7921 (±.08)	.3370 (±.10)	.0682 (±.06)
Random	.8111 (±.07)	.3597 (±.11)	.0597 (±.06)
Scoring variations			
Equation (1) scores	**.8189** (±.08)	.1594 (±.11)	.4414 (±.14)
one walk only	.7968 (±.08)	.1395 (±.13)	**.0346** (±.07)

Table 3: Ablation study results analogously to Table 1. All results are obtained with the 32B model.

Using only $2 * \lfloor \frac{|L|-1}{M} \rfloor$ negatively impacts the results w.r.t. all three metrics. Replacing the $2 * \lfloor \frac{|L|-1}{M} \rfloor$ tuples as constructed according to Section 3.1, however, proves to yield results that are only slightly outperformed by the proposed method in terms of AUC (.8119 vs. .8111) and F1 (.3617 vs. .3597). Regarding ECE, random tuples even lead to a slightly lower value, namely .0597 as compared to .0604 for our proposed method. Hence, the main advantage of the tuple construction method introduced in Section 3.1 is its systematic approach that yields informative comparisons, e.g., among all very positive emotions, very negative emotions, or very calm emotions etc. These may be of further interest in many practical scenarios, while random tuples lead to comparable results.

When substituting the proposed random walk approach (Algorithm 1) with the naive voting given by Equation (1), the AUC results slightly increase from .8119 to .8189. This, however, comes at the cost of considerably worsened F1 (.1594 vs. .3617) and ECE (.4414 vs. .0604) results. It can be concluded that this naive scoring method indeed yields predictions that capture differences between positive and negative samples, but these predictions are not directly usable as a substitute for human votes. They only (roughly) indicate that an emotion in question is a better choice than a certain percentage of other emotion labels, but this does not directly lead to an estimation of how likely a human annotator would assign this emotion.

The simplified version of Algorithm 1 is particularly detrimental to the F1 scores. This is mainly due to its negative impact on the recalls, as fewer labels are selected during the simulated annotation process.

5.4 Resource Requirement Estimations

Table 4 shows the estimations for each method's complexity in terms of the number of input and output tokens and runtime. Note that, in practice, the runtime is faster as all estimations given here are based on batch size 1.

	D-A	D-I	Text-BWS	ours
Input/Output complexity				
# prompts	5K	140K	280K	120K
# input tokens (M)	**1.0**	17.1	63.1	13.4
# output tokens (M)	2.1	**1.1**	3.4	1.7
GPU runtime estimations [m]				
7B	1,087	**729**	2,830	921
13B	1,720	**1,158**	3,601	1,614
32B	3,836	**2,568**	16,894	3,633

Table 4: Comparison of input/output complexity and estimated GPU runtime. All values except the number of prompts are estimations. The lowest numbers of input/output tokens and GPU minutes per model are boldfaced, and the second lowest values are underlined.

Text-BWS is the most resource-intensive method both in terms of the number of tokens and the runtime. This is due to the large number of tuples and lengthy input prompts comprising M texts (instead of short emotion label terms). Our method, in contrast, is more efficient than D-A and Text-BWS w.r.t. both token counts and runtime. It must, however, be noted that the random walks in our method (cf. Section 3.3) are executed on CPUs and are thus

not considered in the estimations discussed here. D-I turns out to be the most efficient method, arguably mainly due to a comparatively small output complexity.

6 Discussion

The proposed combination of LLM-based BWS and subsequent scoring via the thus induced graph proves to be effective w.r.t. all evaluation metrics. It outperforms the benchmarks in terms of AUC and is competitive with regards to F1-score, at least on the largest model considered. Regardless of the model size, our method displays superior calibration. In combination with the ablation studies, these results show that calibration is to be attributed to the novel random walk algorithm, which leverages the full information provided by the model's BWS decisions instead of simply counting "wins" and "losses". As a consequence, our algorithm's predictions are also most suitable for pseudo-labeling aimed at training TER classifiers. Furthermore, it comes with appealing runtime and input/output complexity, in particular when compared to previous BWS-based methods. The construction of tuples of emotions rather than texts leads to a reduction of both the number of prompts and input and output tokens. Moreover, Text-BWS assumes a relatively large text corpus to construct the tuples, while our method also works on isolated text snippets. To summarize, our algorithm offers a promising combination of competitive predictive accuracy, well-calibrated predictions, and resource efficiency. Its main disadvantage is its conceptual complexity, especially when compared to the straightforward D-I and D-A baselines. In terms of AUC and F1, Text-BWS (Bagdon et al., 2024) is arguably superior, but at the cost of substantially higher resource requirements in terms of computational power and data.

7 Conclusion

We introduced a novel BWS-based method for zero-shot emotion tagging via LLMs. Our experiments demonstrated that it outperforms a set of benchmarks in terms of accuracy and calibration when compared to human annotations, making it an advantageous choice for automatic text emotion tagging. Moreover, it compares favorably against the benchmarks in terms of computational efficiency. Future work may include more sophisticated BWS tuple construction, the evaluation of further models and more refined decoding strategies than the vanilla greedy approach employed here. We hope that our work contributes to fostering research into this often neglected direction. The code to reproduce our experiments is made available[4].

Limitations

The experiments presented in this paper are limited to one particular dataset and model family, respectively. Unfortunately, there is a regrettable shortage of TER datasets featuring annotation schemes that go beyond standard basic emotion categories. A more thorough future investigation of the emotion tagging problem would greatly benefit from more suitable data resources. In particular, the dataset employed is rather specific, as it exclusively features English Reddit posts.

LLMs are known to be highly sensitive to the wording of their input prompts. We did not perform any experiments on prompt sensitivity or prompt optimization in this work.

Moreover, our method's results may vary depending on the specific emotion set chosen for the emotion tagging problem. The proposed method is only suitable for a reasonably large label set L, as it arguably heavily relies on a large number of comparisons induced by BWS decisions on several different label tuples.

The proposed methodology for tuple creation is rather simplistic as it maps emotion labels to static valence/arousal values and utilizes these as representations for clustering. In practice, better results may be obtained by handcrafting the tuples or devising methods that generate individual tuples for each input text based on its characteristics.

Further, we want to emphasize that the proposed random walk method (Section 3.3) does not make any reference to or claims about cognitive processes underlying actual annotation tasks. Its design is solely motivated by the pragmatic search for a well-calibrated emotion tagging method.

While the limitations mentioned above pertain to our specific method, there are also general caveats when employing LLMs as annotators, in particular for highly subjective tasks such as TER. In these tasks, one should aim to acquire several annotations per data point in order to capture diverse subjective perspectives (Mohammad, 2022). It is yet unclear if and how such a process can be simulated with LLMs. (Lee et al., 2023) demonstrate

[4] https://github.com/lc0197/bws_emo_tagging

that LLMs struggle with capturing human disagreement. A plethora of works reveal biases towards certain perspectives and opinions in LLMs (Gallegos et al., 2024). Some studies suggest that *socio-demographic prompting* may let LLMs respond from diverse perspectives (Hayati et al., 2024; Schäfer et al., 2024), while others have called this approach into question (Mukherjee et al., 2024). Therefore, we argue that LLM-based annotations should only be considered in combination with human labeling or when human annotations can not be easily obtained.

When conducting TER, regardless of the chosen technical approach, a few general limitations should be kept in mind. A detailed discussion of ethical considerations on TER is beyond the scope of this work, but can be found, e.g., in (Mohammad, 2022). Longstanding debates among psychologists centers around the existence and the degree of universality of discrete *basic* emotions, see e.g., (Ekman, 1992; Barrett, 2017; Keltner et al., 2019). Several empirical studies show that emotional expressions are, at least to a degree, culturally relative (Matsumoto, 2013), a facet typically neglected in existing TER datasets. Another critical aspect to consider is that, naturally, the interpretation of emotional expressions is a subjective task, which is reflected by often moderate inter-annotator agreement scores in TER studies, e.g. (Demszky et al., 2020; Troiano et al., 2021; Christ et al., 2024). An extensive recent study on the shortcomings of (third-party) annotations in TER is presented by (Li et al., 2025), who also highlight the role of demographic similarity between annotators and authors of annotated texts.

Acknowledgments

Shahin Amiriparian is also with MDSI – Munich Data Science Institute as well as MCML – Munich Center of Machine Learning.

References

Cecilia Ovesdotter Alm, Dan Roth, and Richard Sproat. 2005. Emotions from text: Machine learning for text-based emotion prediction. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 579–586, Vancouver, British Columbia, Canada. Association for Computational Linguistics.

Shahin Amiriparian, Filip Packań, Maurice Gerczuk, and Björn W Schuller. 2024. Exhubert: Enhancing hubert through block extension and fine-tuning on 37 emotion datasets. In *Proc. Interspeech 2024*, pages 2635–2639.

Christopher Bagdon, Prathamesh Karmalkar, Harsha Gurulingappa, and Roman Klinger. 2024. "you are an expert annotator": Automatic best–worst-scaling annotations for emotion intensity modeling. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7924–7936, Mexico City, Mexico. Association for Computational Linguistics.

Lisa Feldman Barrett. 2017. The theory of constructed emotion: an active inference account of interoception and categorization. *Social cognitive and affective neuroscience*, 12(1):1–23.

Hans Baumgartner and Jan-Benedict EM Steenkamp. 2001. Response styles in marketing research: A cross-national investigation. *Journal of marketing research*, 38(2):143–156.

Paul S Bradley, Kristin P Bennett, and Ayhan Demiriz. 2000. Constrained k-means clustering. *Microsoft Research, Redmond*, 20(0):0.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Lukas Christ, Shahin Amiriparian, Manuel Milling, Ilhan Aslan, and Björn Schuller. 2024. Modeling emotional trajectories in written stories utilizing transformers and weakly-supervised learning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 7144–7159, Bangkok, Thailand. Association for Computational Linguistics.

Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. 2020. GoEmotions: A dataset of fine-grained emotions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4040–4054, Online. Association for Computational Linguistics.

Igor Douven. 2018. A bayesian perspective on likert scales and central tendency. *Psychonomic bulletin & review*, 25:1203–1211.

Phan Anh Duong, Cat Luong, Divyesh Bommana, and Tianyu Jiang. 2025. CHEER-Ekman: Fine-grained embodied emotion classification. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1118–1131, Vienna, Austria. Association for Computational Linguistics.

Paul Ekman. 1992. An argument for basic emotions. *Cognition & emotion*, 6(3-4):169–200.

Isabel O. Gallegos, Ryan A. Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K. Ahmed. 2024. Bias and fairness in large language models: A survey. *Computational Linguistics*, 50(3):1097–1179.

Maurice Gerczuk, Shahin Amiriparian, Sandra Ottl, and Björn W Schuller. 2021. Emonet: A transfer learning framework for multi-corpus speech emotion recognition. *IEEE Transactions on Affective Computing*, 14(2):1472–1487.

Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. 2023. Chatgpt outperforms crowd workers for text-annotation tasks. *Proceedings of the National Academy of Sciences*, 120(30):e2305016120.

Shahriar Golchin and Mihai Surdeanu. 2023. Data contamination quiz: A tool to detect and estimate contamination in large language models. *CoRR*.

Shirley Anugrah Hayati, Minhwa Lee, Dheeraj Rajagopal, and Dongyeop Kang. 2024. How far can we extract diverse perspectives from large language models? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5336–5366, Miami, Florida, USA. Association for Computational Linguistics.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. DeBERTav3: Improving deBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations*.

Cindy Yoonjoung Heo, Bona Kim, Kwangsoo Park, and Robin M Back. 2022. A comparison of best-worst scaling and likert scale methods on peer-to-peer accommodation attributes. *Journal of business research*, 148:368–377.

Dacher Keltner, Disa Sauter, Jessica Tracy, and Alan Cowen. 2019. Emotional expression: Advances in basic emotion theory. *Journal of nonverbal behavior*, 43(2):133–160.

Noah Lee, Na Min An, and James Thorne. 2023. Can large language models capture dissenting human voices? In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4569–4585, Singapore. Association for Computational Linguistics.

Changmao Li and Jeffrey Flanigan. 2024. Task contamination: Language models may not be few-shot anymore. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18471–18480.

Jiayi Li, Yingfan Zhou, Pranav Narayanan Venkit, Halima Binte Islam, Sneha Arya, Shomir Wilson, and Sarah Rajtmajer. 2025. Can third parties read our emotions? In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 21478–21499, Vienna, Austria. Association for Computational Linguistics.

Jordan J Louviere, Terry N Flynn, and Anthony Alfred John Marley. 2015. *Best-worst scaling: Theory, methods and applications*. Cambridge University Press.

Jordan J Louviere and George G Woodworth. 1991. Best-worst scaling: A model for the largest difference judgments. Technical report, Working paper.

David Matsumoto. 2013. Culture and emotional expression. *Understanding Culture*, pages 271–287.

Saif Mohammad. 2018. Obtaining reliable human ratings of valence, arousal, and dominance for 20,000 english words. In *Proceedings of the 56th annual meeting of the association for computational linguistics (volume 1: Long papers)*, pages 174–184.

Saif Mohammad and Felipe Bravo-Marquez. 2017. WASSA-2017 shared task on emotion intensity. In *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 34–49, Copenhagen, Denmark. Association for Computational Linguistics.

Saif Mohammad and Svetlana Kiritchenko. 2018. Understanding emotions: A dataset of tweets to study interactions between affect categories. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Saif M. Mohammad. 2022. Ethics sheet for automatic emotion recognition and sentiment analysis. *Computational Linguistics*, 48(2):239–278.

Sagnik Mukherjee, Muhammad Farid Adilazuarda, Sunayana Sitaram, Kalika Bali, Alham Fikri Aji, and Monojit Choudhury. 2024. Cultural conditioning or placebo? on the effectiveness of socio-demographic prompting. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 15811–15837, Miami, Florida, USA. Association for Computational Linguistics.

Emily Öhman, Marc Pàmies, Kaisla Kajava, and Jörg Tiedemann. 2020. XED: A multilingual dataset for sentiment analysis and emotion detection. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6542–6552, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, and 1 others. 2024. 2 olmo 2 furious. *arXiv preprint arXiv:2501.00656*.

Sungjoon Park, Jiseon Kim, Seonghyeon Ye, Jaeyeol Jeon, Hee Young Park, and Alice Oh. 2021. Dimensional emotion detection from categorical emotion.

In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4367–4380, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Robert Plutchik. 1982. A psychoevolutionary theory of emotions.

Steve Rathje, Dan-Mircea Mirea, Ilia Sucholutsky, Raja Marjieh, Claire E Robertson, and Jay J Van Bavel. 2024. Gpt is an effective tool for multilingual psychological text analysis. *Proceedings of the National Academy of Sciences*, 121(34):e2308950121.

James A Russell and Albert Mehrabian. 1977. Evidence for a three-factor theory of emotions. *Journal of research in Personality*, 11(3):273–294.

Oscar Sainz, Jon Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. 2023. Nlp evaluation in trouble: On the need to measure llm data contamination for each benchmark. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10776–10787.

Elvis Saravia, Hsien-Chi Toby Liu, Yen-Hao Huang, Junlin Wu, and Yi-Shin Chen. 2018. CARER: Contextualized affect representations for emotion recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3687–3697, Brussels, Belgium. Association for Computational Linguistics.

Johannes Schäfer, Aidan Combs, Christopher Bagdon, Jiahui Li, Nadine Probol, Lynn Greschner, Sean Papay, Yarik Menchaca Resendiz, Aswathy Velutharambath, Amelie Wührl, and 1 others. 2024. Which demographics do llms default to during annotation? *arXiv preprint arXiv:2410.08820*.

Anne LR Schuster, Norah L Crossnohere, Nicola B Campoamor, Ilene L Hollin, and John FP Bridges. 2024. The rise of best-worst scaling for prioritization: a transdisciplinary literature review. *Journal of choice modelling*, 50:100466.

Stanley S Stevens. 1971. Issues in psychophysical measurement. *Psychological review*, 78(5):426.

Enrica Troiano, Sebastian Padó, and Roman Klinger. 2021. Emotion ratings: How intensity, annotation confidence and agreements are entangled. In *Proceedings of the Eleventh Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 40–49, Online. Association for Computational Linguistics.

Brandon T Willard and Rémi Louf. 2023. Efficient guided generation for large language models. *arXiv preprint arXiv:2307.09702*.

Fangxu Yu, Junjie Guo, Zhen Wu, and Xinyu Dai. 2024. Emotion-anchored contrastive learning framework for emotion recognition in conversation. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 4521–4534, Mexico City, Mexico. Association for Computational Linguistics.

Mengchen Zhang and Xiang Feng. 2025. Automated annotation of academic emotion intensity in online learning comment texts: A bws method based on llms. In *Proceedings of the 2024 16th International Conference on Education Technology and Computers*, ICETC '24, page 317–323, New York, NY, USA. Association for Computing Machinery.

Caleb Ziems, William Held, Omar Shaikh, Jiaao Chen, Zhehao Zhang, and Diyi Yang. 2024. Can large language models transform computational social science? *Computational Linguistics*, 50(1):237–291.

A V/A Clustering

The V/A values are generated by prompting the 32B instruction-tuned version of OLMo-2 as described in Table 5.

System Prompt	You are an expert annotator specializing in emotion recognition.
User Prompt	Please consider the emotions listed below. Emotions: admiration,...,surprise. For each of the given emotions, give estimates for its corresponding representation in the valence/arousal/dominance space. Return three numeric values between 0 and 1, corresponding to the emotion's degree of valence, arousal, and dominance. A value of zero denotes very low valence/arousal/dominance, while a value of 1 denotes very high valence/arousal/dominance Round to the format of two decimal places. Only reply with the triple of numbers for each emotion.
Example Response	[{"emotion": "admiration", "valence": .90, "arousal": .70, "dominance": .80} ... {"emotion": "surprise", "valence": .60, "arousal": .85, "dominance": .65}]

Table 5: Prompt for LLM-based Valence/Arousal/Dominance predictions.

We find the predictions for the *dominance* dimension to be of little use and thus discard them for clustering. Table 6 shows the resulting predictions.

We validate these ratings by correlating them with the human-annotated valence/arousal labels for the respective emotion terms given in the crowdsourced NRC-VAD dictionary (Mohammad, 2018). Here, we observe Pearson's correlation coefficients of .946 and .856 for human-labelled valence and arousal, respectively, both of which are statistically significant with $p < 0.01$.

Algorithm 3 describes the clustering process. The tuples grouping emotions similar in terms of valence/arousal are obtained via the constrained KMeans algorithm (Bradley et al., 2000)[5], allowing to specify maximum cluster/tuple sizes (lines 5 and 6). From a set of such clusters comprising similar emotions, the diverse clusters are created in a round-robin manner (lines 7- 15).

With $M = 4$ and a maximum cluster size of 6, the approach results in the clusters listed by Table 7.

[5] https://github.com/joshlk/k-means-constrained

Label	Valence	Arousal
admiration	.9412	.6000
amusement	.9412	.9000
anger	.1176	1.0000
annoyance	.2353	.6000
approval	.8824	.4000
caring	.8824	.0000
confusion	.3529	.6000
curiosity	.7059	.8000
desire	.5882	.9000
disappointment	.2353	.4000
disapproval	.1765	.5000
disgust	.1176	.7000
embarrassment	.2353	.8000
excitement	.8824	1.0000
fear	.1176	1.0000
gratitude	.9412	.4000
grief	.0000	.2000
joy	1.0000	.9000
love	1.0000	.2000
nervousness	.3592	.9000
optimism	.9412	.6000
pride	.9412	.8000
realization	.7059	.4000
relief	.8824	.0000
remorse	.2353	.4000
sadness	.1176	.2000
surprise	.5882	.9000

Table 6: Valence and arousal predictions for discrete emotion labels as given by OLMo-2 32B-Instruct, normalized to $[0, 1]$.

Algorithm 3 Clustering algorithm. The operator $\circ$ denotes (list) concatenations.

1: **function** CREATECLUSTERS(L, $valence : L \to [0,1]$, $arousal : L \to [0,1]$, $M \in \mathbb{N}^+$, $tol \in \mathbb{N}^+$)
2: max_size := $M + tol$
3: num_clusters := $\lfloor \frac{|L|}{M} \rfloor$
4: ▷ All clusters are lists of lists
5: $\mathcal{C}_V^{sim}$:= CONSTRAINEDKMEANS(L, $valence$, num_clusters, max_size)
6: $\mathcal{C}_A^{sim}$:= CONSTRAINEDKMEANS(L, $arousal$, num_clusters, max_size)
7: $\mathcal{C}_V^{div}$:= list of num_clusters empty lists
8: $\mathcal{C}_A^{div}$:= list of num_clusters empty lists
9: flat_V = $\mathcal{C}_V^{sim}[0] \circ \cdots \circ \mathcal{C}_V^{sim}$[num_clusters]
10: flat_A = $\mathcal{C}_A^{sim}[0] \circ \cdots \circ \mathcal{C}_A^{sim}$[num_clusters]
11: **for** i in $[1, |L|]$ **do**
12: div_idx = i mod num_clusters
13: $\mathcal{C}_V^{div}$[div_idx] = $\mathcal{C}_V^{div}$[div_idx] $\circ$ [flat_V[i]]
14: $\mathcal{C}_A^{div}$[div_idx] = $\mathcal{C}_A^{div}$[div_idx] $\circ$ [flat_A[i]]
15: **return** $\mathcal{C}_V^{sim}, \mathcal{C}_A^{sim}, \mathcal{C}_V^{div}, \mathcal{C}_A^{div}$

B Dataset Label Distributions

Figure 2 shows the frequencies of the emotional labels excluding *neutral*. The *neutral* label occurs in 2396 of the 5000 texts, i.e., 47.92 %.

C Prompts

The prompts utilized for the benchmarks (D-I, D-A, (Bagdon et al., 2024)) and our method are given

$\mathcal{C}_V^{sim}$	amusement gratitude joy love optimism pride	admiration approval caring excitement relief	anger disgust fear grief sadness	annoyance disappointment disapproval embarrassment remorse	curiosity desire realization surprise	confusion nervousness
$\mathcal{C}_V^{div}$	confusion relief curiosity disapproval love	nervousness anger desire embarrassment optimism	admiration disgust realization remorse pride	approval fear surprise amusement	caring grief annoyance gratitude	excitement sadness disappointment joy
$\mathcal{C}_A^{sim}$	approval disappointment disapproval gratitude realization remorse	curiosity embarrassment joy nervousness pride surprise	admiration annoyance confusion disgust optimism	amusement anger desire excitement fear	grief love sadness	caring relief
$\mathcal{C}_A^{div}$	approval curiosity admiration relief desire	disappointment embarrassment annoyance grief excitement	disapproval joy confusion love fear	gratitude nervousness disgust sadness	realization pride optimism amusement	remorse surprise caring anger

Table 7: Clustering results for the 27 emotion labels in the GoEmotions dataset. Every cell corresponds to one of the clusters as specified by the first column.

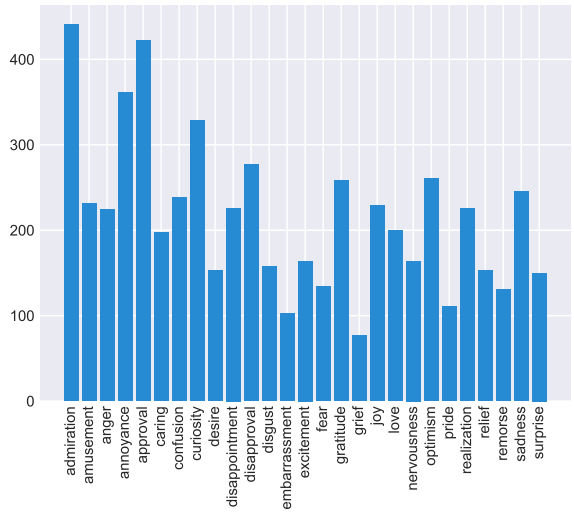

Figure 2: Label distribution in our subset of GoEmotions (5000 samples), excluding *neutral*.

in Table 8. Note that they are heavily inspired by the prompts used by (Bagdon et al., 2024). Correct response formats are ensured by the structured generation options offered by the `outlines` library.

D Data Contamination Investigation with DCQ

We briefly review the intuition behind DCQ in the following; for details see (Golchin and Surdeanu, 2023). DCQ aims at assessing causal LLMs for contamination with a dataset D by providing it with prefixes from the texts in D as inputs. Then, the model is requested to predict the continuation of these prefixes in the form of a multiple-choice quiz. If the LLM systematically predicts the actual continuations, data contamination during pretraining is likely. The quiz questions are corrected for potential positional biases the model may exhibit (e.g., an LLM may tend to always predict the first option when presented with a multiple-choice question).

The GoEmotions labels are only provided via integer codes, i.e., texts are paired with numbers ranging from 0 to 28. Hence, the original CSV files from GoEmotions are already unlikely to leak the actual emotion labels to a model during pretraining. Nevertheless, it could be that the model memorizes the emotion labels in the form of their numerical codes.

We thus adapt DCQ to the problem at hand by posing requests as exemplified in Table 9.

Specifically, we conduct DCQ with two different system prompts. Each model is confronted with 100 randomly selected examples from our subset of GoEmotions. The observed κ values range from $-.2040$ and .1176, giving no indication of systematic data contamination.

E Hyperparameter *M* Considerations

The hyperparameter M determining the tuple size in both our approach and Text-BWS is optimized via the development set. AUC serves as the deci-

	D-A	D-I	Text-BWS (Bagdon et al., 2024)	ours
System Prompt	You are an expert annotator specializing in emotion recognition.			
User Prompt	Please consider the Reddit post and the candidate emotions listed below. Emotions: admiration,..., surprise. Text: Example Text. For each of the given emotions, estimate the probability that the author of the Reddit post felt this emotion. Reply with a float between 0 and 1 where 0 means that the author did not feel the emotion at all and 1 means that the author definitely felt the emotion. Round to the format of two decimal places. Return the float estimate for each emotion.	Please consider the Reddit post below. Text: Example text. Estimate the probability that the author of the Reddit post felt the emotion admiration. Reply with a float between 0 and 1, where 0 means that the author did not feel **blue**admiration at all, and 1 means that the author definitely felt admiration. Round to the format of two decimal places.	Please consider the emotion admiration and the Reddit posts given below. Which of the six texts' authors is likely to feel admiration the MOST and which of the six authors is likely to feel admiration the LEAST? Texts: A) Example Text 1 B) Example Text 2 C) Example Text 3 D) Example Text 4 E) Example Text 5 F) Example Text 6 Which of the six texts' authors is likely to feel admiration the MOST and which of the six authors is likely to feel admiration the LEAST? Only return the identifier (A, B, C, D, E, or F) of the texts that express admiration the most and least, respectively.	Please consider the Reddit post and the candidate emotions listed below. Emotions: admiration,..., relief. Text: Example Text From the given emotions, select both the most likely and the least likely emotion that the author of the Reddit post may have felt. Only return the names of the most and least likely emotion.
Example Response	[{"emotion": "admiration", "score": 0.73} ... {"emotion": "surprise", "score": 0.05}]	{"score": 0.73}	{"most": "C", "least": "E"}	{"most": "admiration", "least": "neutral"}

Table 8: Prompts used for the benchmark and our proposed approach.

System Prompt 1	You are a language model that has seen and memorized open-source datasets.
System Prompt 2	You are an expert annotator specializing in emotion recognition.
User Prompt	You are provided with a five-choice quiz. Your task is to correctly select the option that exactly corresponds to the label of the given text in the *GoEmotions* dataset. Only generate the single letter corresponding to your answer. Text: *This is an example text.* Labels: A) 12, 23 B) 7 C) 7, 9 D) 19 E) 25
Example Response	{"choice": "B"}

Table 9: Example for a DCQ prompt. The five options correspond to different possible labels for the given text. The two system prompts are mutually exclusive and experiments are conducted with both of them. The response is in JSON format for more convenient parsing.

sive criterion for choosing M, as arguably accuracy of the algorithm is more important than calibration (as measured by ECE). Moreover, other than F1, AUC is not sensitive to a prediction threshold. Table 10 displays both the validation AUC results and the number of prompts required in the full experimental setup for M values between 2 and 9.

Based on the AUC results, M is set to 4 and 6 in the experiments with our approach and Text-BWS, respectively (cf. Section 4.2). Due to the design of the clustering approach, the clusters and resulting AUC values of our approach are identical for $M \in \{7, 8, 9\}$, as $\lfloor \frac{27}{7} \rfloor = \lfloor \frac{27}{8} \rfloor = \lfloor \frac{27}{9} \rfloor = 3$. The validation results for our approach suggest that $M = 4$ provides a reasonable middle ground between conducting numerous fine-grained BWS comparisons and evaluating a small number of rather extensive BWS tuples. On the one hand, too fine-grained tuples, e.g. pairwise comparisons for $M = 2$, easily lead to irrelevant comparisons that cause too high scores for emotion labels that are not actually supported by the input, but still "win" some evaluations against other irrelevant labels. On the other hand, too large tuples lead to a loss of information, as they entail fewer decisions to be made by the model. In this case, the majority of labels will typically never be selected as the most or least suitable label, such that only a few labels receive meaningful scores.

As for Text-BWS, the tuples are made up of texts instead of labels. The results in Table 10 suggest that values of $M < 4$ are not suitable here, arguably due to lack of support for several labels in the dataset, leading to too many text tuples in which no text is relevant to the label in question. This observation highlights again Text-BWS's de-

	ours		Text-BWS	
M	dev. AUC	# prompts	dev. AUC	# prompts
2	.7626	260k	.6189	280k
3	.7588	180k	.7547	280k
4	**.7781**	120k	.7950	280k
5	.7444	100k	.7828	280k
6	.7584	80k	**.7999**	280k
7	.7593	60k	.7912	280k
8	.7593	60k	.7936	280k
9	.7593	60k	.7898	280k

Table 10: Hyperparameter search results and number of prompts across different M values for both our approach and Text-BWS. Note that the validation AUC results are based on the development set comprising 200 data points. The number of prompts refers to the full experimental setup with N=5k.

pendence on large-scale corpora, which must be considered a major disadvantage. Our approach, in contrast, does not rely on other texts to annotate a given input.

F DeBERTa Finetuning Details

For all methods' predictions and the human gold standard labels, we finetune the pretrained DeBERTA-v3 *base* model[6] (12 layers, 86M parameters). Training is conducted for a maximum of 10 epochs, with an early stopping patience of 3 epochs. In all experiments, a development set consisting of 1250 predictions (25 %) is sampled from the 5000 predictions. Following initial results on the gold standard values, the learning rate is set to $5e-5$. Binary cross-entropy is employed as the loss function.

G Calibration Curves

Figure 3 provides label-wise calibration curves for all methods. Consistent with the mean ECE results (cf. Table 1), our model's curve is usually the closest to the optimal curve, i.e., the main diagonal. This is particularly true for predictions between 0 and 0.5 on many labels, cf., e.g., the curves for *admiration*, *desire*, *fear*, *love*, and *remorse*. In this range, the curves for D-I and D-A are frequently close to $y = 0$, arguably because these methods tend to predict values that are either close to 0 or close to 1, cf. Section H. Furthermore, the curve for *neutral* reveals how all methods are struggling to correctly assess this special class.

[6] https://huggingface.co/microsoft/deberta-v3-base

H Example

We consider the post *This just made me feel so much better about myself :)* (ID in GoEmotions: ed71bw9) as an example in the following. The gold standard label for this text is solely *pride*. Table 11 shows all BWS decisions yielded by the 32B model for the tuples listed in Table 7.

Tuple group	Worst	in between	Best
$\mathcal{C}_{\mathcal{V}}^{sim}$	neutral admiration disgust disapproval realization nervousness	{amusement,...,pride} {approval,...,neutral} {anger,...,sadness} {annoyance,...,remorse} {curiosity,...,neutral} {confusion}	joy relief joy neutral desire neutral
$\mathcal{C}_{\mathcal{V}}^{div}$	confusion nervousness realization fear grief sadness	{curiosity,...,neutral} {anger,...,neutral} {admiration,...,remorse} {approval,...,surprise} {annoyance,...,neutral} {disappointment,...,neutral}	relief optimism pride amusement gratitude joy
$\mathcal{C}_{\mathcal{A}}^{sim}$	realization nervousness disgust fear grief caring	{approval,...,remorse} {curiosity,...,pride} {admiration,...,neutral} {anger,...,neutral} {neutral, sadness} {neutral}	gratitude joy optimism amusement love relief
$\mathcal{C}_{\mathcal{A}}^{div}$	curiosity grief disapproval disgust realization remorse	{approval,...,neutral} {annoyance,...,neutral} {confusion,...,neutral} {nervousness,...,sadness} {amusement,...,pride} {anger,...,surprise}	relief excitement joy gratitude optimism caring

Table 11: Best-worst evaluation for the text *This just made me feel so much better about myself :)* by the 32B model. Note that each line corresponds to a cell in Table 7, with the additional option *neutral*. We omit most of the labels not selected as the best or worst option for brevity. Predictions in red mark hallucinations, i.e., a label is returned that was not part of the given tuple.

In line with the clearly positive text, the model selects positive emotions as the most suitable options throughout. In one case, it even hallucinates a positive label (*joy*) when presented only with negative options such as *anger* and *sadness* (line 3). In lines 4 and 6, the model deals with such cases in the desired manner, i.e., it selects the special label *neutral* as the best option compared to the rather negative alternatives.

In Table 12, the four different methods' predictions for the text are given. The D-A method falls short of capturing nuanced emotionality, even predicting a score above 0 for only one emotion (joy). Both D-I and Text-BWS arguably over-generate labels, with 11 and 14 predictions above .5, respectively. Such overgeneration also explains why Text-BWS and D-I exhibit higher ECEs than our approach. Our method arguably yields more rea-

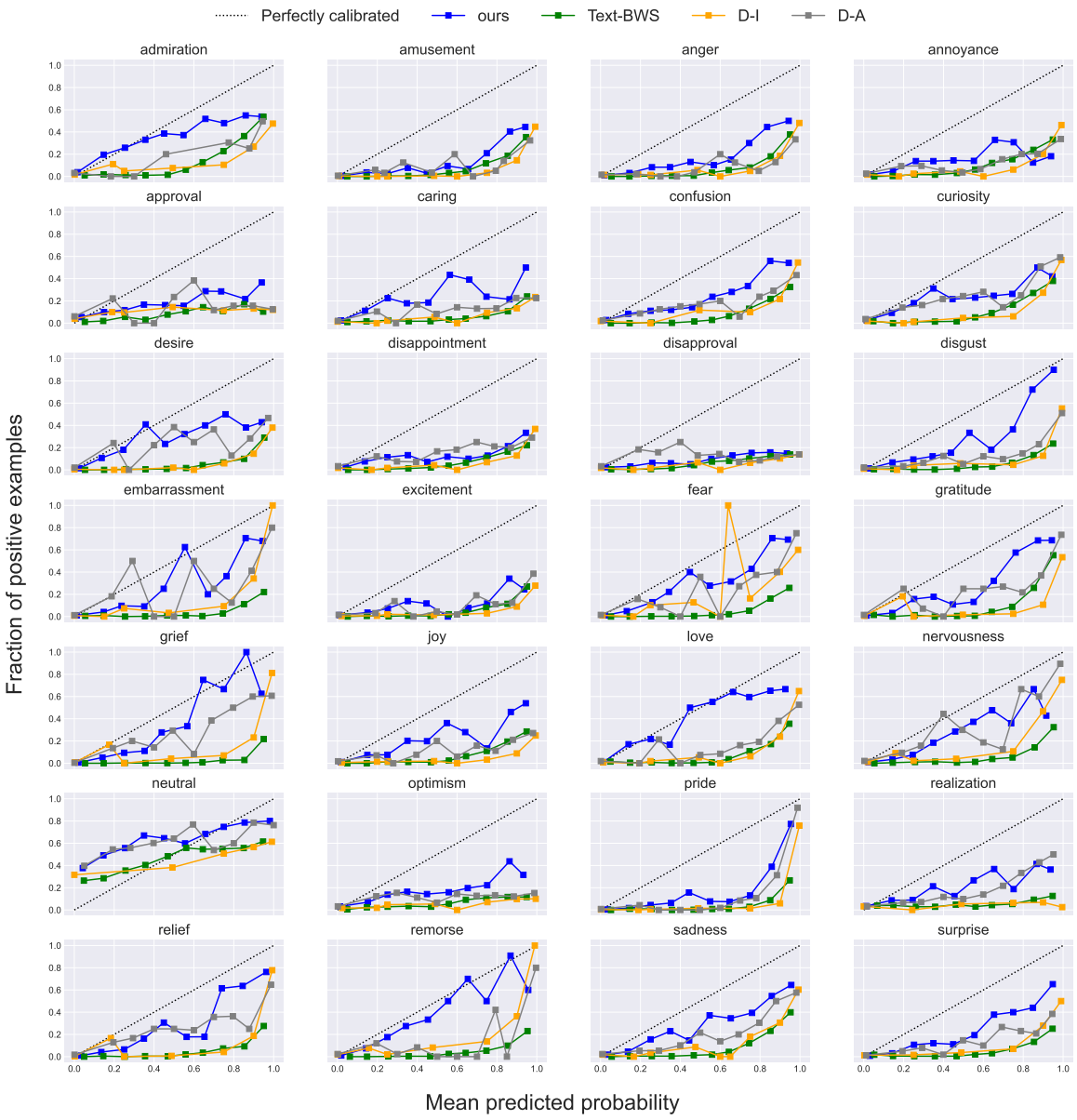

Figure 3: Calibration curves for predictions of all 28 labels in GoEmotions on our test set obtained by a) our approach, b) Text-BWS (Bagdon et al., 2024), c) D-I, and d) D-A, all obtained with the 32B model. The dotted line describes perfect calibration for reference.

sonable values than the baselines here: It also expresses a clear tendency towards positive emotions, especially *joy*, but provides more nuanced scores. Regarding the gold standard value, it is questionable why only *pride* was selected while emotions such as *joy* or *relief* were disregarded. The example thus also points to a crucial shortcoming of TER evaluation in general, namely the subjectivity inherent to human emotion annotations.

Method	Top 5 predictions	$\geq .5$
D-A	joy (1.)	1
D-I	joy (1.), realization (1.), relief (1.) admiration (.99), excitement (.99)	11
Text-BWS	relief (.92), pride (.88), realization (.88), joy (.83), love (.83)	14
ours	joy (0.84), relief (0.47), pride (0.27) gratitude (0.16), amusement (0.12)	1
(gold standard)	pride	

Table 12: All methods' (32B model, no rescaling) top 5 predictions for the example text *This just made me feel so much better about myself :)*. The column $\geq .5$ reports for how many of the 28 labels a value greater than or equal .5 is predicted.