Who’s Laughing Now? Humor Classification by Genre and Technique

Ryan Rony Dsilva and Nidhi Bhardwaj
CLEF 2024 JOKER Lab

Introduction

- Humor can be understood differently depending on who you ask.

- What one person considers humorous, another may not, and even an individual’s sense of humor can change depending on their mood or recent experiences.

Task Definition

We participated in Task 2, multi-class classification where the goal was to identify in a target text the particular technique used for generating humour.

- **IR**: Irony
- **SC**: Sarcasm
- **EX**: Exaggeration
- **AID**: Incongruity/Absurdity
- **SD**: Self-deprecating
- **WS**: Wit/Surprise

We used 3 different approaches for this task.

Our Methodology

Guided Annotation

- Developed an annotation codebook with explicit guidelines for categorizing the text
- Assigned pseudo names to humor categories to minimize bias and ensure objective classification
- Outlined specific characteristics and markers for each humor type
- Two annotators worked independently to categorize sentences, with final decisions based on agreement

Our Methodology

Guided Annotation - Codebook Construction

Humor Category	Definitions with explicit identification features
Wit	Humor involving an unexpected twist or element
Incongruous or Absurd	Unrealistic or nonsensical situations, often with a bipartite structure
Self Deprecation	Speaker highlights their own flaws or weaknesses
Exaggeration	Dramatic overstatement or hyperbolic descriptions
Sarcasm	Literal meaning is different from the intended meaning, often with contempt
Irony	Difference between the literal meaning and the implied meaning

Our Methodology

Multi-Class Classification with DeBERTa

- Fine-tuned DeBERTa-v3-large on the training set and conduct two runs
- First Run: Raw, imbalanced dataset (no class balancing)
- Second Run: Under-sampling strategy to address class imbalance
    - Majority classes capped at $n = 250$ samples

Our Methodology

Prompting with LLMs

- Utilized GPT-4o, by OpenAI using few-shot prompting technique.
- One example per class to serve as a template for desired output, along with instructions to format the output
- Seed is set to a constant value and temperature to 0, to reduce the variability in the model’s output

Results & Analysis

	Class	Precision	Recall	F-Score	Support
Guided Annotation	SD	0.6667	0.6667	0.6667	12
	WS	0.3333	1.0000	0.5000	3
	EX	0.2000	0.1250	0.1538	8
	IR	0.6842	0.7647	0.7222	17
	SC	1.0000	0.7727	0.8718	22
	AID	0.8000	0.8000	0.8000	25
DeBERTa	SD	0.8600	0.9430	0.8996	228
	WS	0.4797	0.5680	0.5201	125
	EX	0.5074	0.3286	0.3988	210
	IR	0.8226	0.7556	0.7877	356
	SC	0.5892	0.8765	0.7047	162
	AID	0.9837	0.9511	0.9671	634
DeBERTa$_{sampled}$	SD	0.7700	0.9693	0.8583	228
	WS	0.5305	0.6960	0.6021	125
	EX	0.5257	0.6333	0.5745	210
	IR	0.8393	0.6601	0.7390	356
	SC	0.7487	0.9012	0.8179	162
	AID	0.9795	0.8281	0.8974	634
GPT4o	SD	0.1905	0.2281	0.2076	228
	WS	0.2500	0.3120	0.2776	125
	EX	0.2530	0.4000	0.3100	210
	IR	0.7863	0.2893	0.4230	356
	SC	0.4803	0.8272	0.6077	162
	AID	0.6599	0.5662	0.6095	634

Train

	Class	Precision	Recall	F-Score	Support
Guided Annotation	SD	0.7500	0.6000	0.6667	5
	WS	0.5714	0.6667	0.6154	6
	EX	0.5000	0.1429	0.2222	7
	IR	0.2727	0.7500	0.4000	4
	SC	1.0000	0.8182	0.9000	11
	AID	0.8333	0.8333	0.8333	12
DeBERTa	SD	0.6777	0.9011	0.7736	91
	WS	0.4464	0.5102	0.4762	49
	EX	0.4255	0.1887	0.2614	106
	IR	0.5946	0.5986	0.5966	147
	SC	0.5000	0.8305	0.6242	59
	AID	0.9206	0.8593	0.8889	270
DeBERTa$_{sampled}$	SD	0.6833	0.9011	0.7773	91
	WS	0.4444	0.5714	0.5000	49
	EX	0.4144	0.4340	0.4240	106
	IR	0.6596	0.4218	0.5145	147
	SC	0.5185	0.7119	0.6000	59
	AID	0.9091	0.8519	0.8795	270
GPT4o	SD	0.2174	0.2747	0.2427	91
	WS	0.2642	0.2857	0.2745	49
	EX	0.2953	0.4151	0.3451	106
	IR	0.6716	0.3061	0.4206	147
	SC	0.4397	0.8644	0.5829	59
	AID	0.7117	0.5852	0.6423	270

Test

Results & Analysis

- Manual Annotation: 350 submitted; 98 from training set
    - Out of 17 AID samples, all the incorrect classifications (6) belonged to the WS class
    - In the EX class (out of 6), 2 were incorrect that belonged to SC
    - IR, SC and EX were mostly mixed up (out of 350; 80 were from these category, 60% took more than 25 seconds)
- 67% of the annotation which took more than 30 seconds belonged to AID and WS
- Annotators focused too much on structural patterns (bipartite pattern AID)

Discussion & Future Scope

LLMs with prompting showed poor results

Further studies could explore how the annotation codebook can be introduced into the prompt through prompting and maybe even fine-tuning to provide the LLM with precise instructions and specificity.

Weighted Annotation Codebook

The analysis of codebook suggests that there are improvements that can be made with the codebook itself by utilizing a weight matrix for the rules in the codebook where only if the weight crosses a certain threshold would the sample be classified in that particular category.

Thank You

Reach out to us at:

Ryan Rony Dsilva <ryan.rony.dsilva@gmail.com> | Nidhi Bhardwaj <nidhi.bhardwaj012@gmail.com>