Vayam Solve Kurmaha @ CLEF 2024: Task 2: Humor Classification according to Genre and Technique using BERT Embeddings and Transformers

Notebook for the JOKER Lab at CLEF 2024

Dhannya S M[1], Lakshmi Priya S[1], Shreedevi Seluka Balaji[1,†], Sai Nikitha N.S.R[1,*,†], Shwetha S[1,†], Surabhi Kamath[1,†] and Srinidhi Lakshmi Narayanan[1,†]

[1]*Sri Sivasubramaniya Nadar College of Engineering, Chennai*

Abstract

This study addresses humor classification using advanced NLP techniques. By using the LaBSE model classifier, this approach aims to enhance semantic understanding and accuracy in categorizing all the different humor types. Our approach utilizes an advanced variant of BERT, to achieve more nuanced and accurate humor classification in textual-based content. This task is more achievable with BERT since it allows understanding of context from both previous and subsequent words in a sentence, leading to a deeper comprehension of context-dependent nuances inherent in humor. Our model ranked 28th among 54 submissions showing the robustness and accurate nature of our approach. The proposed model demonstrates improved performance in the task of humor classification.

Keywords

BERT embeddings, tokenizers, vectors, classification, wordplay, humor, sarcasm, humor detection

1. Introduction

Social media platforms have changed the nature of communication by granting previously unheard-of freedoms of expression and democratizing the production and dissemination of content. But these advantages have also brought with them difficulties, especially when it comes to policing content, especially comedy, which frequently has nuanced undertones and context-dependent implications.

Because humor is subjective and situational, it presents special challenges for categorization and moderation. Examples of these challenges include satire, sarcasm, and irony. Because of this difficulty, sophisticated Natural Language Processing (NLP) methods are required to properly categorize amusing content and promote a more inclusive and secure online community.

This paper delves into classifying humor within textual content, as outlined in the JOKER lab at CLEF 2024. Leveraging insights from previous works such as those by Ermakova et al. (2024)[1] and Palma Preciado et al. (2024)[1], this study aims to contribute to the advancement of humor analysis by employing state-of-the-art NLP techniques.

The objectives of this study are to enhance semantic understanding and classification accuracy across diverse textual datasets, focusing on six specific categories of humor: Sarcasm (SC), Exaggeration (EX), Wit and Surprise (WS), Incongruity and Absurdity (AID), Irony (IR), and Self-deprecating humor (SD). Our approach builds upon existing research and methodologies, incorporating novel techniques to address the intricacies of humor classification [2].

The remainder of this paper is organized as follows: Section 2 talks about works related to ours, and Section 3 describes the dataset and its features. Section 4 delves into the approach and methods used, detailing the techniques employed for humor classification. Section 5 discusses the results and

CLEF 2024: Conference and Labs of the Evaluation Forum, September 09–12, 2024, Grenoble, France

*Corresponding author.

†These authors contributed equally.

dhannyasm@ssn.edu.in (D. S. M); lakshmipriyas@ssn.edu.in (L. P. S); shreedevi2210389@ssn.edu.in (S. S. Balaji); sainikitha2210401@ssn.edu.in (S. N. N.S.R); shwetha2210210@ssn.edu.in (S. S); surabhi2210196@ssn.edu.in (S. Kamath); srinidhi2210142@ssn.edu.in (S. L. Narayanan)

the conclusions drawn from it. Finally, Section 6 concludes the paper and suggests avenues for future research.

2. Related Works

Humor Classification is considered to be one of the more difficult NLP tasks as humor is quite subjective and this aspect can make it hard to classify. In recent times, humor classification has gained popularity due to its impact on downstream tasks such as sentiment analysis and opinion mining.

Recent studies, as done by Hossain et al., 2020 [3] show them testing hypotheses for their classification task by measuring distances between GloVe vectors but fell short when it came to detecting humor in non-uniformly labeled data. Another study by Meaney et al., 2021 [4] details how many teams used different approaches to detect both offense and humor yet participating teams did not meet the threshold that was expected concerning ground truth humor values, making their humor predictions weak. Zhang and Liu, 2014 [5] recognized humorous content using Federated Learning to make it more personalized. This approach is not entirely useful as it is highly personalised. Ren et al., 2021 [6] proposed ABML, an attention-based multi-task learning model that jointly performs humor recognition and pun detection, demonstrating the effectiveness of multi-task learning approaches in humor-related tasks, however, the ABML model's effectiveness in humor detection and pun recognition may be constrained by its domain-specific performance on datasets like SemEval2017 Task7 and the 16000 one-liner dataset, thereby limiting generalizability across diverse humor types. In a paper published in 2020 [7], the authors talk about using machine learning approaches like using word embeddings to detect self-deprecating humor. This falls short in areas where other types of humor need to be detected. Mihalcea and Strapparava, 2005 [8] made significant contributions to linguistic feature engineering for humor prediction also known as humor detection. Their approach concentrated on extracting linguistic features from text, such as word embeddings, syntactic structures, and sentiment analysis, to capture the information necessary for humor prediction. However, their approach is limited by certain surface-level features. Reyes et al., 2012 [**?**] explored humor detection in tweets by analyzing stylistic and semantic features, emphasizing the role of irony and sarcasm in humorous texts and yet it may face limitations due to potential biases in user-generated tags from social media like Twitter, which could affect the generalizability of humor and irony classification beyond specific cultural and linguistic contexts. Work done by Faruqi, Shrivastava [9] uses a Neural Language Model called RNN (Recurrent Neural Network) with an LSTM (Long Short-term Memory) layer to capture long-term dependencies, but only to classify as humorous and non-humorous. Chiara Bucaria, 2004 [10] analyzed some forms of lexical and linguistic ambiguity in English in a specific register, i.e. newspaper headlines. However, it may be limited by its reliance on a specific corpus of newspaper headlines, potentially constraining the findings of other forms of written or spoken humor beyond this particular genre. Another paper published by Fahim, Khan, et al. [11] delves into a comparison of classical machine learning models such as Logistic Regression, Random Forest and Support Vector Machines concluding that an ensemble model gives them the best accuracy in detecting whether the sentence is humorous or not. Again, this is just to classify as humorous and non-humorous whereas our task is to detect different types of humor and label them accordingly.

3. Dataset

The datasets for both the training and test data were provided by the organizers and is a mixture of existing corpora on irony and sarcasm, COVID-19 humor, public sources as well as the JOKER 2023 corpora [12, 13, 14, 15, 16]. To start, on performing an analysis of the training dataset [1] that was provided, given below are the observations:

- The training dataset consisted of 2,454 entries.
- Each entry had an ID, the text to be evaluated, and the Class label

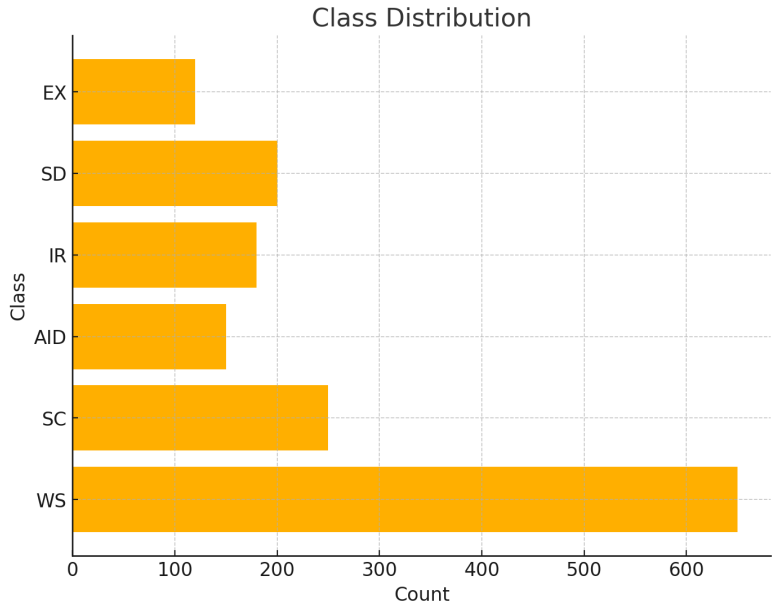

Figure 1: Bar graph displaying the counts of each class in the training dataset

- The testing dataset that was given to us consisted of 6,642 entries.
- The Class labels were:
  - SC: Sarcasm
  - EX: Exaggeration
  - WS: Wit and Surprise
  - AID: Incongruity and Absurdity
  - IR: Irony
  - SD: Self-deprecating humor

4. Approach

4.1. Pre-processing

Data preprocessing is a crucial step that is done before the data is passed on to the machine learning model. The raw data is processed to improve the quality and accuracy of our machine-learning model. Noise refers to any extraneous information in the data that does not help with the performance of our model. Data preprocessing removes noise from the dataset. This paper's approach uses the following data processing steps:

- Converting to lower: All the characters in the text are converted to lowercase to standardize the text. This makes our analysis case insensitive as now the words 'Apple' and 'apple' will be considered the same by the model.
- Removing punctuations: Punctuations constitute noise in the dataset as they often do not carry much meaning. Punctuations are removed using a list of punctuations from the string library.
- Removing stop words: Stop words are filler words that do not significantly contribute to the semantics of the text. Words like 'a', 'the', and 'is' are examples of stopwords. They occur very frequently and therefore removing them significantly reduced the number of words making the dataset more manageable.
- Stemming and Lemmatization: Stemming reduces a word to its base form and lemmatization reduces it to its dictionary form called a 'lemma'. This helps with reducing the size of tokens without changing the context of the text while also improving the vocabulary of the model.

4.2. Effects of Preprocessing on Sentence Embedding

Using the preprocessed text helped us generate embeddings that were more consistent and clear, as the standardized and cleaned text (lowercase, no punctuation, no stopwords) helped reduce noise and redundancy in the embeddings. Preprocessing helps pick out only the words that matter, which improves the semantic richness of the generated embeddings. As discussed earlier, stemming and lemmatization make the vocabulary smaller and more focused, thus increasing the efficiency and effectiveness of the model. Embeddings generated using consistent and clean data help the model perform better with new, unseen data as the model is trained to generalize better, making the model more attuned to performing better with humor detection.

4.3. BERT

BERT is a state-of-the-art deep learning model that uses a transformer-based neural network to understand human-like language. It was developed by researchers at Google AI Language in 2018. BERT stands for Bidirectional Encoder Representations from Transformers. BERT uses the encoder architecture of a transformer to learn contextual relations between the tokens in the input sequence [17].

During training, BERT processes an input sequence of tokens and generates a sequence of vectors corresponding to each token. The vectors provide a contextualized representation of the words. It then employs tasks like the Masked Language Model and Sentence Prediction to define its training objectives. After training, the model augmented by a classification layer [18] is fine-tuned on task-specific data. Finally, the model is usable for our classification task. The approach in this work involves setting the number of training epochs to 3 and using a batch size of 16. Additionally, an early stopping technique is employed.

Early stopping is a regularization technique that is used to prevent overfitting of data. It observes the evaluation metric and stops the training process if it stops improving or starts deteriorating. The patience parameter tells the model how many epochs to wait after it starts noticing a dip or stagnation in the evaluation metric. The delta parameter defines the minimum positive change in the metric to consider it as an improvement. In this model, the early stopping patience is set to 3 and the delta to 0.01.

4.4. Feature Extraction

Feature Extraction using BERT was done using LaBSE, Language Agnostic BERT Sentence Embeddings, as they generate high-quality embeddings that capture the subtle contextual clues in humor as well as wordplay.

Since the LaBSE model has been pre-trained on large corpora of multilingual text, it has learned rich multilingual text that performs better even though our dataset is monolingual.

LaBSE is also pre-trained to be better at generalizing across tasks which made it more efficient in humor classification as humor classification often involves a vast range of linguistic constructs.

Most importantly, LaBSE provides contextual embeddings that take into account the context of their surrounding words and sentences. This was a game changer for humor classification as humor is largely contextual.

5. Results

The model was trained and evaluated with an 80-20 train-test split. The results of the classification on the test set are as follows:

The table given below summarizes the performance metrics from the CLEF 2024 JOKER task results, where our team ranked 28th.

Table 1
Classification Report for the Test Set of Training Data

Class	Precision	Recall	F1-Score	Support
WS	0.68	0.89	0.78	112
SC	0.59	0.56	0.57	79
AID	0.64	0.49	0.55	47
IR	0.18	0.20	0.19	41
SD	0.67	0.50	0.57	32
EX	0.42	0.26	0.32	38
Accuracy		57.59%		
Macro Average	0.53	0.48	0.50	349
Weighted Average	0.57	0.58	0.56	349

Table 2
Performance Metrics from CLEF 2024 JOKER Task Results

Metric	Precision	Recall	F1-Score	Support
SC	0.41	0.18	0.25	106
WS	0.53	0.37	0.43	49
EX	0.41	0.18	0.25	106
IR	0.52	0.63	0.57	147
SD	0.45	0.76	0.57	59
AID	0.74	0.84	0.78	270
Macro Average	0.54	0.53	0.51	722
Weighted Average	0.59	0.60	0.58	722

5.1. Comparison of Results

From a comparison of the results obtained from our trained model with the performance metrics from the CLEF 2024 JOKER task, several observations can drawn. Our model performed better in certain categories such as WS (Wit and Surprise) and SD (Self-deprecating Humor) however, it fell short in others like IR (Irony) and EX (Exaggeration). The macro and weighted average precision, recall, and F1-scores are also marginally lower in our trained model compared to the CLEF 2024 results. However, the overall accuracy of our trained model was 57.59%, which is equivalent to the 60% accuracy achieved in the CLEF 2024 JOKER [2] task.

5.2. Detailed Analysis of Test Results

Our fine-tuned model using BERT embeddings and transformers with a train batch size of 16, achieved an overall accuracy of 57.59% on the test set. The model's performance reflects its ability to classify the types of humor based on genre and technique. The classification report in Table 1 provides a breakdown of precision, recall, and F1-score for each humor category.

The model performed well in identifying Wit and Surprise (WS) and Self-deprecating (SD) humor, achieving F1-scores of 0.78 and 0.57, respectively. Sarcasm (SC) humor's F1-score also stood at 0.57. These results indicate that the model successfully understands and captures linguistic nuances associated with these humor types. However, the model struggled with categories such as Irony (IR) and Exaggeration (EX), where F1-scores were notably lower at 0.19 and 0.32, respectively.

5.3. Performance Comparison and Ranking

Comparing our results with the performance metrics from the CLEF 2024 JOKER [12] task (Table 2), the model ranked 28th out of 54 submissions. The CLEF 2024 results show higher average precision, recall,

and F1-scores across most categories compared to the analysis performed on the run of the train-test split performed on the training dataset, as results were calculated on 722 entries of the 6,642 in the test dataset. Thereby, using this fine-tuned BERT model, predictions were made on the actual dataset, achieving an overall accuracy of 0.60.

As mentioned in the overview paper [12], our model did take a hit when it came to making predictions in the WS (Wit and Surprise) category and the EX (Exaggeration) category. Wit and Surprise is a category that is a combination of two types of humor, making it difficult to predict [12]. Our model performed very well in the AID (Absurdity and Incongruity) model with an F1 score of 0.78.

Following the CLEF 2024 JOKER [12] task results, our model's ranking among submissions has been incorporated into the abstract of our paper. This ranking reflects our model's competitive standing within the field of humor classification using advanced NLP techniques.

Although there are areas needing improvement, the insights from our analysis pave the way for future research to refine model performance across different humor categories.

6. Conclusion

Our approach provides a robust model that works well on humor classification. The results demonstrate that our approach utilizing BERT embeddings and transformers, specifically the LaBSE model, achieves competitive performance in humor classification. The high precision and recall values indicate that the model effectively distinguishes between different humor types, including subtle nuances like irony and wordplay. Future work could include extending this model to train and predict entries in other languages as this approach uses LaBSE which is language agnostic is pre-trained on multiple languages and could span cultural contexts.

Our approach of using BERT for classification along with LaBSE for feature extraction shows that pre-trained models work well for humor classification tasks such as this.

References

[1] L. Ermakova, A.-G. Bosser, T. Miller, T. Thomas, V. M. P. Preciado, G. Sidorov, A. Jatowt, CLEF 2024 JOKER Lab: Automatic Humour Analysis, in: N. Goharian, N. Tonellotto, Y. He, A. Lipani, G. McDonald, C. Macdonald, I. Ounis (Eds.), Advances in Information Retrieval, Springer Nature Switzerland, Cham, 2024, pp. 36–43.

[2] V. M. P. Preciado, et al., Overview of the CLEF 2024 JOKER task 2: Humour classification according to genre and technique, in: G. Faggioli, et al. (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024), CEUR Workshop Proceedings, CEUR-WS.org, 2024.

[3] N. Hossain, J. Krumm, M. Gamon, H. A. Kautz, SemEval-2020 Task 7: Assessing Humor in Edited News Headlines, CoRR abs/2008.00304 (2020). URL: https://arxiv.org/abs/2008.00304. arXiv:2008.00304.

[4] J. A. Meaney, S. Wilson, L. Chiruzzo, A. Lopez, W. Magdy, SemEval 2021 task 7: HaHackathon, detecting and rating humor and offense, in: A. Palmer, N. Schneider, N. Schluter, G. Emerson, A. Herbelot, X. Zhu (Eds.), Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), Association for Computational Linguistics, Online, 2021, pp. 105–119. URL: https://aclanthology.org/2021.semeval-1.9. doi:10.18653/v1/2021.semeval-1.9.

[5] X. Guo, H. Yu, B. Li, H. Wang, P. Xing, S. Feng, Z. Nie, C. Miao, Federated learning for personalized humor recognition, ACM Trans. Intell. Syst. Technol. 13 (2022). URL: https://doi.org/10.1145/3511710. doi:10.1145/3511710.

[6] L. Ren, B. Xu, H. Lin, L. Yang, Abml: attention-based multi-task learning for jointly humor recognition and pun detection, Soft Computing 25 (2021) 14109–14118.

[7] A. Kamal, M. Abulaish, Self-deprecating humor detection: A machine learning approach, in: L.-M. Nguyen, X.-H. Phan, K. Hasida, S. Tojo (Eds.), Computational Linguistics, Springer Singapore, Singapore, 2020, pp. 483–494.

[8] R. Mihalcea, C. Strapparava, Making computers laugh: Investigations in automatic humor recognition, in: Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, 2005, pp. 531–538.
[9] F. Faruqi, M. Shrivastava, "is this a joke?": A large humor classification dataset, in: Proceedings of the 15th International Conference on Natural Language Processing, 2018, pp. 104–109.
[10] C. Bucaria, Lexical and syntactic ambiguity as a source of humor: The case of newspaper headlines, Humor-international Journal of Humor Research - HUMOR 17 (2004) 279–309. doi:10.1515/humr.2004.013.
[11] N. I. Fahim, R. Khan, S. Rahman, N. Akter, M. N. Huda, Humor detection using machine learning approach, in: 2024 6th International Conference on Electrical Engineering and Information Communication Technology (ICEEICT), 2024, pp. 1217–1222. doi:10.1109/ICEEICT62016.2024.10534417.
[12] L. Ermakova, T. Miller, A.-G. Bosser, V. M. P. Preciado, G. Sidorov, A. Jatowt, Overview of CLEF 2024 JOKER track on automatic humor analysis, in: L. Goeuriot, P. Mulhem, G. Quénot, D. Schwab, L. Soulier, G. M. D. Nunzio, P. Galuščáková, A. G. S. de Herrera, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Fifteenth International Conference of the CLEF Association (CLEF 2024), Lecture Notes in Computer Science, Springer-Verlag, 2024.
[13] L. Ermakova, T. Miller, A. G. Bosser, V. M. P. Preciado, G. Sidorov, A. Jatowt, Overview of JOKER – CLEF-2023 track on automatic wordplay analysis, in: A. Arampatzis, E. Kanoulas, T. Tsikrika, S. Vrochidis, A. Giachanou, D. Li, M. Aliannejadi, M. Vlachos, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction, volume 14163 of *Lecture Notes in Computer Science*, Springer Nature Switzerland, Cham, 2023, pp. 397–415. doi:10.1007/978-3-031-42448-9_26.
[14] S. Frenda, A. Pedrani, V. Basile, S. M. Lo, A. T. Cignarella, R. Panizzon, C. Marco, B. Scarlini, V. Patti, C. Bosco, D. Bernardi, EPIC: Multi-perspective annotation of a corpus of irony, in: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics, volume 1, Association for Computational Linguistics, 2023, pp. 13844–13857. doi:10.18653/v1/2023.acl-long.774.
[15] F. Galatolo, G. Martino, M. Cimino, C. Tommasi, et al., Dense information retrieval on a latin digital library via labse and latinbert embeddings (2023).
[16] N. R. Bogireddy, S. Suresh, S. Rai, I'm out of breath from laughing! i think? a dataset of covid-19 humor and its toxic variants, in: Companion Proceedings of the ACM Web Conference 2023, Association for Computing Machinery, New York, NY, 2023, pp. 1004–1013. doi:10.1145/3543873.3587591.
[17] I. Annamoradnejad, G. Zoghi, Colbert: Using bert sentence embedding in parallel neural networks for computational humor, Expert Systems with Applications 249 (2024) 123685.
[18] R. Gupta, Bidirectional Encoders to State-of-the-Art: A Review of BERT and Its Transformative Impact on Natural Language Processing, Informatika. Ekonomika. Upravlenie-Informatics. Economics. Management 3 (2024) 0311–0320.