PICT at CLEF 2025 JOKER Track: Humour-Aware Information Retrieval using BERT-Enhanced Ensemble Methods

Notebook for the JOKER Lab at CLEF 2025

Tanish Chaudhari[1], Ansh Vora[1], Sanjeev Hotha[1] and Sheetal Sonawane[1]

[1]*Pune Institute of Computer Technology (PICT), Pune, India*

Abstract

The CLEF 2025 JOKER Task 1 (Humour-aware information retrieval in EN) focuses on the automatic and efficient retrieval of humorous texts that are relevant to a given query. The nuanced nature of the retrieval process is introduced with the detection of wordplay and literary devices used for humour, enabling the identification of jokes revolving around the query. For this task, we employed an ensemble pipeline architecture, including traditional text analytics methods, composite indices and reranking using BERT models with configurable weighted scores for each stage to create a sophisticated information retrieval system.

Keywords

Pipeline, Ensemble, Tokenization, TF-IDF, BM25, RM3, ColBERT

1. Introduction

Information Retrieval (IR) systems have seen considerable progress; we no longer rely on straightforward keyword matching [1]. In recent times, we have been able to include contextual awareness, understand semantics, and decipher user intent [2] [3]. However, a domain of information retrieval that has not yet been explored in-depth is humour—which forms a significant part of human communication in various forms such as sarcasm, irony, wit, and wordplay. These instances are deeply embedded in different forms of digital content, from social media posts and forum discussions to news articles and creative writing. For traditional IR systems, humour can be a considerable obstacle to accurate understanding of text and subsequently, effective retrieval. Humour can lead these traditional IR systems to misinterpret text, fail query satisfaction and result in a general lack of nuance in how the information is presented to the user.

The difficulty of effectively retrieving information while considering humour is its characteristic ambiguity and its derivation from shared cultural knowledge. Adding to this, humour is often delivered through subtle linguistic cues that may defy literal interpretation—something a traditional IR system is built around. Due to this, the system may miss the true underlying intent or meaning of a phrase, which it may have interpreted literally. On the flip side however, a query looking for humorous phrases might retrieve inappropriate or unfunny results, due to the system's failure to identify and categorize comedic elements. This shortcoming underscores an urgent requirement for humour-aware information retrieval (HAIR) systems that have the ability to identify humour as well as understand its qualities and relevance to a user's query.

In order to develop effective HAIR systems, we must ensure that models can distinguish complex linguistic patterns, identify nuances in cultural settings, and separate genuine humor from other non-literal language. Initial approaches to detect humour were founded on rule-based systems and traditional machine learning models had limited success but could not cope with scalability, generalization and the dynamic nature of humour. The rise of deep learning, particularly advances in the field of transformer-based architectures, have introduced new methods of natural language understanding, helping us process complex linguistics like humour.

CLEF 2025 Working Notes, 9 – 12 September 2025, Madrid, Spain

We propose a unique approach to humour-aware information retrieval through a sophisticated ensemble framework. Our pipeline utilizes the strengths of classical IR methods with modern deep learning models; we include features that help the system detect and leverage wordplay to enhance performance. In the beginning, we performed robust indexing and initial retrieval, post which we reranked the shortlisted texts that leverage neural models and humour-specific features. It was observed that considerable improvements in IR performance can be achieved by explicitly considering humour and by modeling the system around the same. This paper discusses the construction of a pipelined IR system that can take into account the literal intent of queries and documents alike, but additionally work with the nuances of humour.

2. Related Work

The JOKER corpus [4] is a large parallel pun dataset used for Humour information retrieval [5] and translation in English and Portuguese language. Another dataset used for similar tasks is the HaHackathon corpus (Meaney et al., 2021) [6] that contains 10K humour annotated tweets that is used to establish testbeds for humour intensity labels. These datasets form the foundation for most current humour retrieval systems.

Humour recognition and wordplay detection in text generally involves IR re-ranking or filtering as explored in earlier research at CLEF. Gepalova et al. (2024) [7] used fine-tuned T-5 models to detect single]-word puns while Dsilva (2023) [8] used prompt based LLMs for pun detection and BERT tokenization. Xie et al. (2020) [9] introduce GPT-2-based uncertainty for humor-sensitive reranking signals. These studies showed that transformer architecture models perform well at identifying textual ambiguity, though neural approaches have varying effectiveness. Schuurman et al. (2024) [10] trained separate pun detecting classifiers that demonstrated greater quality of humour retrieval. They observed that neural rankers perform inferiorly, and humour detectors can suppress false positives.

Beyond lexical matching, another approach to humour retrieval is semantic reranking. Schuurman et al. (2024) [10] used a zero-shot MS-MACRO encoder to rerank top 25 hits for humour queries, but it also boosted non-humorous relevance. University of Split and Malta (2024) used a cross encoder trained on humour classification to rerank candidate jokes. Annamoradnejad and Zoghi (2024) [11] used embeddings using parallel BERT encoders and dense embedding models for humour retrieval to index jokes by semantics. Gupta et al. [12] demonstrated the use of transformer ensembles for humour and offensiveness in text for sensitive systems. Ao et al. (2022) [13] used multi-encoder architectures to improve parody detection for composite humour features. Tang et al. [14] introduced the NaughtyFormer that is trained on reddit humor for more nuanced classification of humour subtypes. Al Omari et al. (2021) [15] showed transformer ensembles like BERTweet and RoBERTa excel and are very efficient at capturing humour-based IR, but these semantic approaches often struggle with contextual understanding especially when compared to lexical methods.

Ensemble methods are hybrid methods that are viable in the long run as they improve accuracy by increasing recall values. Baguian and Huynh (2024) [16] combined TFIDF with Logistic Regression and showed significant improvement while Arampatzis's group (2024) experimented with over ten different models and architectures including random forest, LSTMs and XGBoost. No other model performed as well as the ensemble methods in accuracy and results. Another method used was query expansion—Gepalova et al. (2024) [7] expanded queries via WordNet Synonyms to match the pun-based language and applied a similarity threshold for detection. UvA (2024) applied relevance feedback (RM3) to BM25 and significantly boosted recall results. Zhao et al. (2023) [17] present the RDVI framework that combined SimCSE-based retrieval and irony detection. Berger's humour-based stacking ensemble [18] achieved high F-1 scores using SVM and Random Forest. Thus, ensemble methods that merge lexical matching, semantic features and humour-specific classifiers perform consistently well at humour retrieval. Fusion architecture can also have a multi-stage pipeline as seen in Baguian and Huynh's hybrid design. Our approach builds on these findings by combining multiple retrieval strategies in a unified framework.

3. System Description

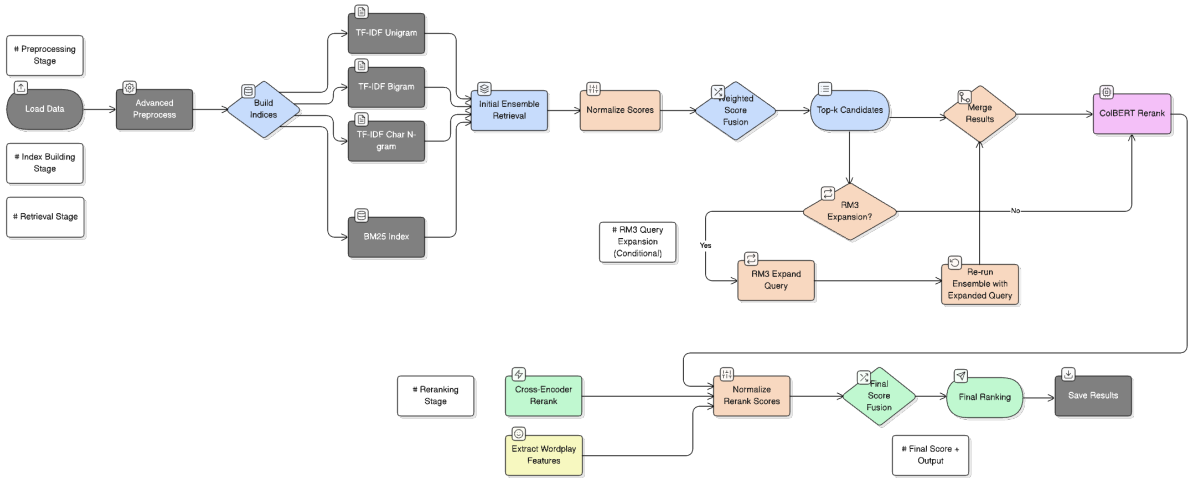

Figure 1: Architecture diagram of the BERT-enhanced ensemble pipeline

The proposed BERT-enhanced ensemble IR system is a hybrid system designed to perform efficient and humour-aware retrieval of documents from a sizeable corpus of 77,656 documents.

The corpus comprises a mix of humorous and non-humorous sentences of various types. The majority of sentences are definitional and declarative in nature, with some narratives (dialogues or quotes) and a few outliers in terms of explicit sentence structure. Most non-humorous sentences seem encyclopedic in the corpus:

- "Bill refers to a proposed law that is presented for discussion and approval in a legislative body, such as Congress in the United States or Parliament in the United Kingdom."
- "In the 7th century, the Middle East and North Africa came under caliphal rule with the Arab conquests."

Humour is observed to not be the prevailing theme in the corpus with only a small fraction of the dataset being humorous—Tom Swifties (jokes involving dialogue and adverbs), puns and double meanings. Few examples of humorous texts involving different sentence structures:

- "The farmer was surprised when his pumpkin won a blue ribbon at the State Fair. He shouted, 'Oh, my gourd.'"
- "Sellers of dried grapes are always raising awareness."

The aim was to achieve this by combining multiple retrieval methods and sophisticated reranking strategies (Encoder Hybrid—ColBERT with late interaction) [19]. A multi-method ensemble approach is utilized for initial retrieval—instead of relying on a single retrieval model, the pipeline employs multiple models and fuses the strengths of BM25 and TF-IDF to present a variety of initial "relevant" documents. We worked on improving the preprocessing stage of the system, especially focusing on patterns found in humorous texts. Specialized BERT models are used for reranking, contributing to a better overall understanding of semantics and improved performance on HAIR. The architecture of the pipeline is shown in Figure 1.

The pipeline has seven main phases—Data Loading and Preprocessing, Index Building, Initial Ensemble Retrieval, RM3 Query Expansion, Re-retrieval with Expanded Query (Optional Merge), ColBERT-based Reranking and finally result formatting and output using weighted ensemble scores which may be fine-tuned to give optimized results.

Preprocessing is a crucial stage that has a profound influence on the performance of information retrieval systems. Generally, the text preprocessing step involves normalization of documents for feature extraction (such as lowercasing, stemming, tokenization, etc.), noise reduction (removal of stop words).

For our focus on humour-aware information retrieval, we preserve cues to recognize humour during preprocessing. This requires us to handle contractions (e.g., "'ll", "n't") and possessives (e.g., "'s") by expanding or retaining them. Some specific stop words may also serve as important flags for wordplay and must be dealt with carefully. For the ensemble pipeline, data was also ensured in a format suitable for different models such as the n-gram TF-IDF and BM25.

Once all the documents in the corpus have been preprocessed in a format friendly to the diverse range of models, the intermediate stages of the ensemble pipeline are introduced to build indices (offline) using different types of TF-IDF and BM25 (as depicted in the architecture figure). Based on these indices, the system performs initial retrievals—working with queries for the first time in the process. Post-initial retrieval, reranking by different approaches ensures various linguistic mechanisms revolving around humor.

The indices are built offline on the documents before queries are processed. The model creates the TF-IDF Unigram Index for dealing with the significance of individual words, a Bigram Index for capturing common pairings of words to understand phrasing, and the TF-IDF Char N-gram Index (range 3-5) to detect wordplay patterns and phonetic similarities. To effectively recognize humorous intent, especially in the N-gram Index, we use the unstemmed/untokenized document text.

The pipeline also utilizes Okapi BM25 to create indices on the stemmed/tokenized corpus to aid relevance matching in the subsequent stages and the final ensemble scoring. The BM25 score of a document with respect to a query is calculated as:

$$\text{score}(q, D) = \sum_{i=1}^{n} \text{IDF}(q_i) \cdot \frac{f(q_i, D) \cdot (k_1 + 1)}{f(q_i, D) + k_1 \cdot \left(1 - b + b \cdot \frac{|D|}{\text{avgdl}}\right)} \tag{1}$$

Where:

- $f(q_i, D)$ is the frequency of term q_i in document D.
- $|D|$ is the length of document D in terms (words).
- avgdl is the average document length in the collection.
- k_1 and b are hyperparameters, typically $k_1 \in [1.2, 2.0]$ and $b \in [0.5, 0.8]$.
- $\text{IDF}(q_i)$ is the inverse document frequency of term q_i, calculated as:

$$\text{IDF}(q_i) = \log\left(\frac{N - n(q_i) + 0.5}{n(q_i) + 0.5} + 1\right) \tag{2}$$

Where:

- N is the total number of documents in the corpus.
- $n(q_i)$ is the number of documents containing term q_i.

Working with the pre-built indices, at this stage the model performs initial retrieval for each query. The query receives documents from each of the indices and each set of scores from all four methods is normalized (min-max scaling). To merge the normalized scores, the system employs a "Weighted Score Fusion" using predefined ensemble_weights, which in turn produces a list of "Top-k Candidates" for further reranking.

This concludes the "preparation and initial processing" phase of the corpus. The architecture diagram depicts the flow of the pipeline from this phase to the next processes handling wordplay features, RM3 expansion and reranking.

Using the initial k documents from the previous stage, the system performs the Relevance Model 3 (RM3) query expansion. The most frequent terms from the top candidates are extracted after preprocessing and then added to the original query. This enhances the scope of the query in terms of

Table 1
ensemble_weights for Initial Ensemble Retrieval

Index	Weight
TF-IDF Unigram	0.25
TF-IDF Bigram	0.25
TF-IDF Char N-gram	0.15
Okapi BM25	0.35

vocabulary to retrieve more relevant documents, which may not have been possible with the original, shorter query.

The optional re-run comes into play if the query was indeed expanded by RM3; if so, then the previous stage of ensemble retrieval is carried out again on the expanded query and the results are stored in "Merged Results". The system assigns a higher weight to the original result than the expanded query's result to maintain refinement.

Following the conditional RM3 query expansion stage in the pipeline, we introduce the BERT model to work on the refined "Top-k Candidates" to be reranked.

- The system makes use of ColBERT (Contextualized Late interaction over BERT), a breakthrough in the sphere of Document Retriever Models, which dealt with the effectiveness-computational cost trade-off particularly well. The query and document are encoded separately into embeddings using a SentenceTransformer model that calculates cosine similarity scores. The two embeddings act as vectors with attributes, say A and B, the cosine similarity, $\cos(\theta)$, is represented using a dot product and magnitude as

$$\text{cosine\_similarity}(A, B) = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}} \tag{3}$$

  Where:

  - A_i and B_i are the i-th components of $\mathbf{A}$ and $\mathbf{B}$, respectively.

  The two sets of encodings (query and document) together come up with a score indicative of relevance for each query-document pair.
- Another rerank is performed by a Cross-Encoder which processes the query-document pairs and outputs a fine-grained relevance score taking into consideration context.
- Wordplay features such as text length, word count, quotes, exclamation/question marks, dialogue indicators and unique words and their repetitions and ratio are also considered. Even alliteration contributes to all these features and makes up a "Wordplay Score".

Table 2
Score Weights for Reranking

Score	Weight
Cross-Encoder	0.5
ColBERT	0.3
Wordplay Features	0.2

Throughout the IR system, there have been multiple methods with final weighted scores to output a comprehensive conclusion to the respective stage's processing. In order to utilize each method's strengths, the system had weights associated with each method, distributing influence over the final ensemble/composite score among the diverse methods.

In the initial ensemble retrieval stage, various indices had contributed to an initial retrieval score which was then passed on to the RM3 stage which expanded the query whenever applicable. The updated

retrieval score was yet again combined with the original in a weighted fashion to maintain refinement and avoid complete dominance of one stage's output/determination over the other. This helped the system, pragmatically avoiding complete collapses associated with a single method's drawbacks as well as incorporating strengths and biases of each method in a controlled manner.

Finally, even the reranking methods had scores that were weighted as they covered different aspects of information retrieval with the focus of capturing humour as well as maintaining relevance between the query and the documents. The final composite score merged not only semantics gathered from dense embeddings, but also fine-grained relevance (which was further reinforced by Cross-Encoder reranking) and explicit humour indicators.

The weights can be fine-tuned further to optimize performance and gain better results.

The composite scores and ranks observed with train qrels against performance metrics suggested by the lab organizers served to be the primary method of evaluation and selection of methods and weights for ensemble scores at each stage of the IR system.

4. Metrics Used

1. **Mean Average Precision (MAP)**
   MAP is a key evaluation metric for ranked retrieval systems. It calculates average precision across the relevant documents for a query and then averages this value over all queries. It helps by rewarding systems that retrieve relevant documents. A higher MAP indicates better overall precision across queries.

$$\text{MAP} = \frac{1}{Q} \sum_{q=1}^{Q} \left(\frac{1}{|R_q|} \sum_{k=1}^{n} P(k) \cdot \text{rel}(k) \right) \tag{4}$$

   where Q is the number of queries, R_q is the set of relevant documents for query q, $P(k)$ is the precision at rank k, and $\text{rel}(k)$ is 1 if the document at rank k is relevant.

2. **Geometric Mean Average Precision (GMAP)**
   GMAP computes the geometric mean of the average precision in all queries. It penalizes low scores more heavily than MAP:

$$\text{GMAP} = \left(\prod_{q=1}^{Q} \text{AP}(q) \right)^{1/Q} \tag{5}$$

3. **Binary Preference (BPref)**
   Binary Preference measures how many times the relevant documents are ranked higher than the non-relevant ones. It's defined as:

$$\text{BPref} = \frac{1}{R} \sum_{r=1}^{R} \left(1 - \frac{|n \text{ ranked higher than } r|}{R} \right) \tag{6}$$

   where R is the number of relevant documents. A Bpref of 0.0 here suggests incomplete judgments or relevance annotations incompatible with this metric.

4. **Precision at k (P@k)**
   Precision at rank k is the fraction of relevant documents among the top k retrieved:

$$P@k = \frac{\text{Number of relevant documents in top } k}{k} \tag{7}$$

   Higher values essentially indicate that most relevant documents appear early in the ranking. We report P@5, P@10, P@15, P@20, P@30, P@100, P@200, P@500, and P@1000.

5. **R-Precision (Rprec)**
   R-Precision is the precision at the R-th rank, where R is the total number of relevant documents for a query:

$$\text{Rprec} = \frac{\text{Number of relevant documents in top } R}{R} \tag{8}$$

6. **Mean Reciprocal Rank (MRR)**
   MRR measures the inverse of the rank at which the first relevant document is found:

$$\text{MRR} = \frac{1}{Q} \sum_{q=1}^{Q} \frac{1}{\text{rank}_q} \tag{9}$$

   where rank_q is the rank position of the first relevant document for query q.

7. **Normalized Discounted Cumulative Gain (NDCG@k)**
   NDCG measures the usefulness of documents based on their position in the result list, with gains discounted logarithmically:

$$\text{NDCG@k} = \frac{DCG@k}{IDCG@k} \tag{10}$$

   where:

$$DCG@k = \sum_{i=1}^{k} \frac{2^{rel_i} - 1}{\log_2(i+1)} \tag{11}$$

   NDCG measures ranking quality by including graded relevance and position sensitivity. Relevance scores are calculated logarithmically according to their increasing rank and $IDCG@k$ is the ideal DCG for the top k documents. We report NDCG@5, NDCG@10, NDCG@15, NDCG@20, NDCG@30, NDCG@100, NDCG@200, NDCG@500, and NDCG@1000. These metrics indicate the effectiveness over increasing amounts of retrieval.

8. **Retrieval Statistics**
   We also track standard retrieval statistics for reporting purposes:
   - **num_ret**: Total number of documents retrieved.
   - **num_rel**: Number of known relevant documents.
   - **num_rel_ret**: Number of relevant documents retrieved.
   - **num_q**: Total number of queries used for evaluation.

5. Results

5.1. Task 1: Humour-Aware Information Retrieval

We evaluated our BERT-Enhanced Ensemble pipeline using train qrels, but blind tests were carried out through test qrels which were available to run against on Codabench. During multiple runs, we were able to understand the importance of limiting the diversity of methods for a specific purpose to achieve maximum performance.

A high weight or absolute scoring often led to detrimental changes in performance, leading to a composite scoring system that was able to capture the strengths of each method while avoiding complete bias toward an individual method. The run submissions and their analysis compared to the overall JOKER laboratory are also documented [20].

5.2. Result Analysis

RM3 allowed us to fetch more relevant documents that might not have been linked to the original query term by expanding the query's representation. This enabled us to access a larger volume of candidates for ranking. Subsequent re-retrieval helped us refine the candidate set.

The ColBERT reranking improved performance of the pipeline slightly; more fine-tuning is needed to deal with the sensitive nature of texts. The late ColBERT interaction pushed humorous texts up in rankings, observed with the first query "change" where a humorous text "I wanted change, but all I got was coins". The sentence (docid: 38) ranked 35 after reranking, but ranked 134 without it.

Another query "deal" had a corresponding humorous sentence that read "She was only a Coal dealer's daughter, but, oh, where she had bin." which was found in the reranking pipeline (albeit with a low score) but was absent without BERT.

- **MAP** remained around 0.14. This was highly sensitive to the ranking of relevant documents and heavily penalized highly relevant documents appearing lower in ranking. The observation was further enhanced by significant deviations caused by changes in scores' weights.
- **GMAP** was lower than MAP, suggesting that there was an observable variability in performance, depending on the query. Contextual understanding, as well as cultural knowledge—understanding connections between different words in a certain context to form a joke—could improve this metric to accommodate a wider variety of queries. For example, "chemistry" and "reaction" in "I tried to make a chemistry joke, but there was no reaction."
- **MRR** of 0.3369 indicated that on average at least one relevant document was placed high in the search results; the upper end of results performed fairly well.
- **R-Precision** of 0.1629 failed to meet standards, with the model searching (and selecting) a large amount of irrelevant documents before it could meet the threshold for relevant documents.
- **Precision** showed an uptick from P@5 to P@10, however, it depreciated at higher K values. The system was relatively successful in assessing relevance within the top results but declined as it went down the rankings. Compared to scores from other participants, the model struggled to put emphasis on humorous relevant documents—opting to weigh heavily on relevance for ranking sentences. A higher bias/weight for humour-specific features would improve the pipeline in this regard.
- **NDCG** values demonstrated a steady growth alongside K; the system was able to find relevant documents and could contribute positively to cumulative gain. Most of the relevant documents were found around K=500 as visible by a plateau around K=500,1000.

6. Conclusion

The results of our research are very promising and show the success of ensemble methods in the field of humour detection. Our BERT-enhanced ensemble system achieved a MAP of 0.14 and MRR of 0.3369, and ranked among the top scores in the track. The scores indicate a strong performance in placing relevant documents in the top results. The GMAP score is a bit low, suggesting a high degree of query-dependent variability. The ensemble method with its multistage architecture leveraged the strengths of TF-IDF, BM25 and ColBERT at various levels along with n-gram indexing for effective detection of wordplay patterns.

The greatest strength of our proposed method lies in its ability to capture diversity in humour. This is indicated through the weighted score fusion that prevents over-reliance on any single method in the architecture. At the same time, our model also highlights the challenges while dealing with humour understanding—cultural specificity, contextual dependency, and subjective interpretations. These challenges extend beyond the confines of traditional IR, and while our model demonstrates feasible results, the scores also highlight the need for humor-specific language models.

Declaration on Generative AI

During the preparation of this work, the author(s) used eraser.io for figure 1 to generate an image. After using this tool/service, the author(s) reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] K. A. Hambarde, H. Proença, Information retrieval: Recent advances and beyond, IEEE Access 11 (2023) 76581–76604. doi:10.1109/ACCESS.2023.3295776.

[2] Z. A. MERROUNI, B. FRIKH, B. OUHBI, Toward contextual information retrieval: A review and trends, Procedia Computer Science 148 (2019) 191–200. URL: https://www.sciencedirect.com/science/article/pii/S1877050919300365. doi:https://doi.org/10.1016/j.procs.2019.01.036, tHE SECOND INTERNATIONAL CONFERENCE ON INTELLIGENT COMPUTING IN DATA SCIENCES, ICDS2018.

[3] M. Kamil, D. Çakır, Advances in transformer-based semantic search: Techniques, benchmarks, and future directions, Turkish Journal of Mathematics and Computer Science 17 (2025) 145–166. doi:10.47000/tjmcs.1633092.

[4] L. Ermakova, A.-G. Bosser, T. Miller, R. Campos, Clef 2025 joker lab: Humour in the machine, in: C. Hauff, C. Macdonald, D. Jannach, G. Kazai, F. M. Nardini, F. Pinelli, F. Silvestri, N. Tonellotto (Eds.), Advances in Information Retrieval, Springer Nature Switzerland, Cham, 2025, pp. 389–397.

[5] L. Ermakova, A.-G. Bosser, A. Jatowt, T. Miller, The joker corpus: English-french parallel data for multilingual wordplay recognition, in: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '23, Association for Computing Machinery, New York, NY, USA, 2023, p. 2796–2806. URL: https://doi.org/10.1145/3539618.3591885. doi:10.1145/3539618.3591885.

[6] J. A. Meaney, S. Wilson, L. Chiruzzo, A. Lopez, W. Magdy, Semeval 2021 task 7: Hahackathon – detecting and rating humor and offense, in: Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), Association for Computational Linguistics, 2021, pp. 105–119.

[7] A. Gepalova, A. Chifu, S. Fournier, Clef 2024 joker task1: Exploring pun detection using the t5 transformer model, in: Working Notes of CLEF 2024 – Conference and Labs of the Evaluation Forum, CEUR-WS, 2024. Vol.3740.

[8] R. R. Dsilva, Augmenting large language models with humor theory to understand puns, Master's thesis, Purdue University, 2024.

[9] Y. Xie, J. Li, P. Pu, Uncertainty and surprisal jointly deliver the punchline: Exploiting incongruity-based features for humor recognition, arXiv preprint arXiv:2011.01120 (2020).

[10] E. Schuurman, M. Cazemier, L. Buijs, J. Kamps, University of amsterdam at the clef 2024 joker track, in: Working Notes of CLEF 2024 – Conference and Labs of the Evaluation Forum, CEUR-WS, 2024. Vol.3740.

[11] I. Annamoradnejad, G. Zoghi, Colbert: Using bert sentence embedding in parallel neural networks for computational humor, Expert Systems with Applications 249 (2024). doi:10.1016/j.eswa.2024.123685.

[12] D. Gupta, S. Chawla, R. A. Sheikh, H. Dutta, Identifying offensive and humorous posts using fine-tuned transformer ensembles, in: Working Notes of CLEF 2021 - Conference and Labs of the Evaluation Forum, volume 2936 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2021. URL: http://ceur-ws.org/Vol-2936/paper-81.pdf.

[13] X. Ao, D. Sánchez Villegas, D. Preoțiuc-Pietro, N. Aletras, Combining humor and sarcasm for improving political parody detection, arXiv preprint arXiv:2205.05505 (2022).

[14] K. Tang, P. Bhattacharyya, Naughtyformer at semeval-2022 task 6: Transformer with fine-grained classification head for humor detection, in: Proceedings of the 16th International Workshop on

Semantic Evaluation (SemEval-2022), Association for Computational Linguistics, 2022, pp. 673–679. URL: https://aclanthology.org/2022.semeval-1.91/.

[15] H. Al-Omari, I. AbedulNabi, R. Duwairi, Dljust at semeval-2021 task 7: Linking humor and offense using transformer ensembles, in: Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), Association for Computational Linguistics, 2021, pp. 1062–1068.

[16] H. Baguian, N. A. Huynh, Joker track @ clef 2024: the jokesters' approaches for retrieving, classifying, and translating wordplay, in: Working Notes of CLEF 2024 – Conference and Labs of the Evaluation Forum, CEUR-WS, 2024. Vol.3740, pp.1811–1817.

[17] T. Zhao, et al., Rdvi: A retrieval–detection framework for verbal irony detection, Electronics 12 (2023) 4830. doi:10.3390/electronics12234830.

[18] J. Bielaniewicz, P. Kazienko, An automatic humor identification model with novel features from berger's typology and ensemble models, Decision Analytics Journal 11 (2024) 100450. doi:10.1016/j.dajour.2024.100450.

[19] R. Jha, B. Wang, M. Günther, G. Mastrapas, S. Sturua, I. Mohr, A. Koukounas, M. K. Akram, N. Wang, H. Xiao, Jina-colbert-v2: A general-purpose multilingual late interaction retriever (2024). URL: https://arxiv.org/abs/2408.16672. arXiv:2408.16672.

[20] L. Ermakova, Overview of the clef 2025 joker task 1: Humour-aware information retrieval, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2025), CEUR Workshop Proceedings, CEUR-WS.org, 2025.