International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(1s) 2026 pp. 1-9

https://doi.org/10.70917/ijcisim-2026-1966
Article

Controlled Emoji Effects in Fine-Grained Tweet Valence: A Multi-Metric Benchmark of Transformer Checkpoints

Sapandeep Singh Sandhu[1], Amanpreet Kaur Sandhu[2]

[1] University Institute of Computing (UIC), Chandigarh University, Mohali, Punjab, India.
Email: sapandeepsinghsandhu@gmail.com
ORCID: 0000-0002-6865-7493
[2] University Institute of Computing (UIC), Chandigarh University, Mohali, Punjab, India.
Email: amanpreet.e9962@cumail.in

Abstract: — Transformer sentiment models are routinely benchmarked on social media text; however, performance claims regarding emoji robustness are frequently confounded by cross-dataset shifts in domain, label space, and annotation policy.This paper introduces a controlled evaluation protocol that isolates emoji presence as a variable within a fixed dataset and label space. We benchmark nine transformer checkpoints on SemEval-2018 Valence-oc (English), a fine-grained seven-class ordinal task, using a shared fine-tuning recipe and a multi-metric suite (Accuracy, Macro-F1, micro-ROC-AUC). Findings indicate that DEBERTA-V2-XLARGE-MNLI achieves the strongest performance (Acc 0.492, Macro-F1 0.385) on this task. By partitioning the fixed test set into emoji-containing (n = 251) and non-emoji (n = 686) subsets at evaluation time, we demonstrate that emoji effects are heterogeneous and model-dependent rather than uniformly beneficial. Furthermore, we show that single-metric accuracy reporting masks critical minority-class failures in ordinal sentiment,necessitating the use of Macro-F1 for valid model selection. The proposed protocol serves as a reusable methodology for measuring variable-specific effects without conounding factors.
Keywords: Sentiment analysis, Transformers, Multi-class classification, Emojis, Benchmarking, ROC-AUC

1. Introduction

Sentiment analysis on social media must handle short, informal language that mixes tokens, slang, hashtags, and emojis.Transformer models are routinely benchmarked on tweet-like data; however, two recurring methodological weaknesses limit the interpretability of published comparisons. First, many studies report a single summary metric (often accuracy), which can conceal severe class-imbalance failures in nuanced multi class or ordinal sentiment tasks [1]. Second, emoji-related con-clusions are frequently drawn from cross-dataset comparisons (different domains, label spaces, preprocessing, and annotation policies), where multiple variables change simultaneously;under such confounding, performance differences cannot be attributed to emoji presence with causal validity [2], [3].

Recent work evaluates PLMs on tweet benchmarks, but rarely isolates the emoji effect from domain shifts. This paper benchmarks modern Transformers on emoji handling via a controlled, within-dataset evaluation.Why this matters,These weaknesses lead to model-selection guidance that does not transfer to fine-grained valence settings and to unsupported claims about “emoji robustness.” In practice, a model can appear strong under accuracy while collapsing under class-balanced metrics (e.g., Macro-F1), and emoji effects can be over-claimed when they are entangled with dataset shift rather than measured under control.

Novel (protocol-level contribution). Let Dtest = {(xi,yi)}n i=1 be a fixed labeled test set, and let E(x) ∈ {0,1} denote an emoji-presence predicate (implemented by a specified detector). Define the within-dataset partition

DE = {(x,y) ∈ Dtest : E(x) = 1}and D¯E = {(x,y) ∈Dtest : E(x) = 0}.For a trained checkpoint θ (obtained under a fixed fine-tuningrecipe) and any evaluation metric P(θ,S) computed on subset S ⊆Dtest, we define the emoji sensitivity estimand.

$$\Delta_E^P(\theta) \;=\; P(\theta, D_E) \;-\; P(\theta, D_{\bar{E}}). \qquad (1)$$

Proposition 1 (Internal-control property). For fixed Dtest,labels y, checkpoint θ, detector E(·), and metric P(·,·), the contrast ΔP E (θ) compares performance on two subsets drawn from the same test set, and thus does not introduce cross-dataset or cross-label confounding by construction.

Proof. By definition, DE ⊆Dtest and D¯E ⊆Dtest, and both inherit the same label space and annotation policy because they are subsets of the same labeled set Dtest. The checkpoint θ is fixed, and P(θ,·) is evaluated using the same code path on both subsets. Therefore, the only explicit selection criterion separating the two evaluations is membership in the partition nduced by E(x), and no cross-dataset or cross-label variation is introduced when forming ΔP E (θ).

Interpretation. Equation (1) is an internally controlled within-dataset association measure of emoji sensitivity: it quantifies how model performance differs between emoji-present and emoji-absent instances under fixed data, labels, checkpoint,and evaluation procedure. It is not, by itself, a claim of causal effect or cross-domain generalization. In this paper we report ΔAcc E (θ); the same protocol applies to Macro-F1 and micro-AUC

Novelty and contributions We contribute a protocol that prioritizes internal validity and measurement logic over leaderboard chasing. We: (1) benchmark nine transformer checkpoints on a fine-grained ordinal task; (2) show that standard accuracy reporting masks minority-class failures, validating the need for Macro-F1; and (3) quantify Δemoji via a controlled split, revealing that emoji impact is heterogeneous and model-dependent.

Research questions We structure the study around: (RQ1) Which widely used transformer checkpoints perform best on a nuanced 7-class valence task under a shared fine-tuning recipe? (RQ2) Holding the dataset and label space constant, how does emoji presence in the input text change performance for each checkpoint? Why SemEval and how to interpret SST-2 SemEval Valence-oc is an ordinal, tweet-native benchmark aligned with nuanced affect modeling. Results on SST-2 (binary movie reviews) are omitted to prioritize in-domain theoretical fit, as cross-domain comparisons introduce si- multaneous confounds. TweetEval sentiment is included to demonstrate that the controlled emoji-split protocol is portable to another tweet dataset.

2. Related Work

SemEval-2018 Task 1 (Affect in Tweets) provides benchmark subtasks for emotion and valence, including ordinal classification for valence intensity [4]. Transformer architectures such as RoBERTa [5], DeBERTa [6], XLNet [7], and ALBERT[8] are commonly fine-tuned for sentiment. Domain-specific tweet pre-training, e.g., BERTweet [9], improves robustness to Twitter syntax and emoji usage. TweetEval [10] provides a unified benchmark for tweet classification tasks and is frequently used for sentiment evaluation.Prior Twitter sentiment studies using deep learning in applied settings also report strong dataset- and preprocessing dependence, reinforcing the need for controlled evaluation protocols.

a) Benchmark-summary expectations.: Prior work on strong encoder families (e.g., RoBERTa, DeBERTa) reports strong results on standard sentiment benchmarks (e.g., SST- 2/IMDb) [5], [6]. Separate emoji-rich guidance emphasizes that emoji robustness depends on tokenization and pre-training domain, motivating tweet-specialized checkpoints such as Twitter-RoBERTa, BERTweet, and Twitter-XLM-R [9]. Our study tests how these expectations transfer to a fine-grained seven-class tweet-valence setting and quantifies emoji effects under a controlled within-dataset split.

3. Task And Data

A. SemEval Valence-oc (7-class) We benchmark on SemEval-2018 Valence-oc (English), a seven-class ordinal classification task with labels in {−3,−2,−1,0,+1,+2,+3}[4]. The dataset split used in our experiments contains 1,181 training samples, 449 validation samples, and 937 test samples.

B. Emoji Subset Construction

To isolate emoji presence, we partition the SemEval test set into two disjoint subsets: (i) with emojis (n = 251) and

(ii) without emojis (n= 686). Emoji detection is performed using the Python emoji library (v2.8.0). We classify a text as containing an emoji (E = 1) if the library detects at least one character defined by the library's Unicode emoji data, including ZWJ sequences and skin-tone modifiers; all other texts are labeled as E = 0. The split is applied only at evaluation time; the fine-tuning dataset remains unchanged.

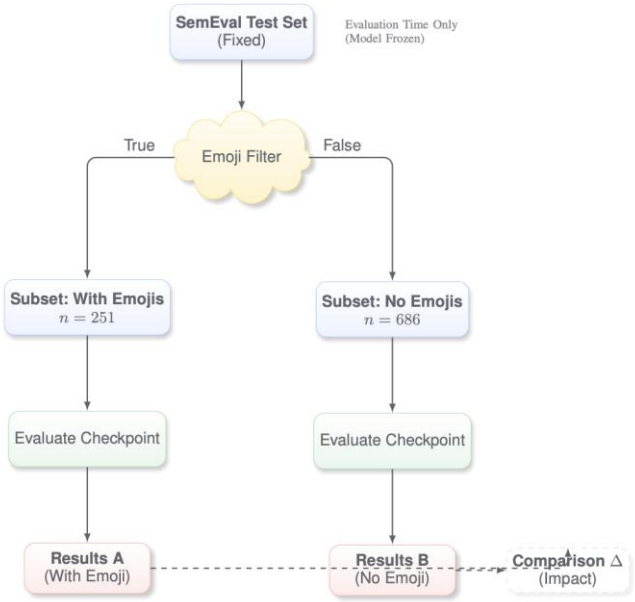

Fig. 1. Controlled within-dataset design for isolating the emoji variable: train once, split the same test set (Train 1,181 / Test 937 split into Emoji n=251 vs. No-Emoji n=686) into emoji vs. non emoji subsets, and evaluate the same checkpoint on both subsets.

4. Models And Experimental Setup

A. Models

We evaluate nine transformer checkpoints spanning general-purpose and tweet-oriented pre-training:1 DEBERTA-V2-XLARGE-MNLI [6], DEBERTA-V3-LARGE [6], ROBERTA-LARGE [5], BERTWEET-BASE [9], a BERTweet sentiment-tuned checkpoint, TWITTER-ROBERTA (SENTIMENT-LATEST),TWITTER-XLM-ROBERTA (SENTIMENT) [11],ALBERT-XXLARGE-V2 [8], and XLNET-LARGE-CASED[7].

B. Model choice rationale (theory-driven)

We choose nine checkpoints to span three orthogonal factors that theory suggests govern emoji-sensitive sentiment behavior:

F1) Pre-training domain. Models pretrained on Twitter like corpora are expected to encode emoji usage patterns and informal syntax more directly than general-domain pre-training. Therefore, we include tweet-specialized checkpoint (BERTweet; Twitter-RoBERTa; Twitter-XLM-R) alongside general-purpose models. This enables a controlled comparison between domain-exposed vs. domain-agnostic pretraining under the same downstream task.

(F2) Tokenization and Unicode coverage. Emoji handling depends on how tokenizers represent Unicode symbols. Byte-level BPE tokenizers [5] covers a wide range of Unicode without UNK tokens, whereas other tokenizers may fragment emojis or map them inconsistently. Including RoBERTa-style and tweet tokenizers allows us to observe whether representation differences align with model-dependent emoji effects under the same evaluation protocol.

(F3) Capacity and inductive bias. Larger or structurally enhanced encoders (e.g., DeBERTa variants) may learn effective emoji–sentiment associations during fine-tuning even when emoji exposure during pretraining is limited. In-cluding DeBERTa-v2/v3, RoBERTa-large, XLNet-large, and ALBERT-xxlarge provides a capacity- and architecture-diverse set, often evaluated on common English sentiment benchmarks (e.g., SST-2/IMDb) [12], allowing us to test whether those "binary leaderboard expectations" transfer to a 7-class ordinal tweet-valence setting.

C. Fine-tuning Protocol

All models are fine-tuned for 2 epochs with learning rate 2×10−5 under a fixed recipe. The best checkpoint is selected using validation Macro-F1 and then evaluated on the test set.Training and evaluation are implemented with the Hugging Face Transformers library [13].

D. Evaluation Metrics

We report: Accuracy (overall correctness), Macro-F1 (class-balanced performance), and micro-averaged ROC- AUC (one-vs-rest) to measure class separability from probability scores. Micro-AUC aggregates decisions across all classes and can overweight frequent classes; we therefore interpret it jointly with Macro-F1.

5. Results

A. What we found on SemEval Valence-oc (7-class) Figure 3 and Table I summarize the test-set behavior of nine Transformer checkpoints in a 7-class valence

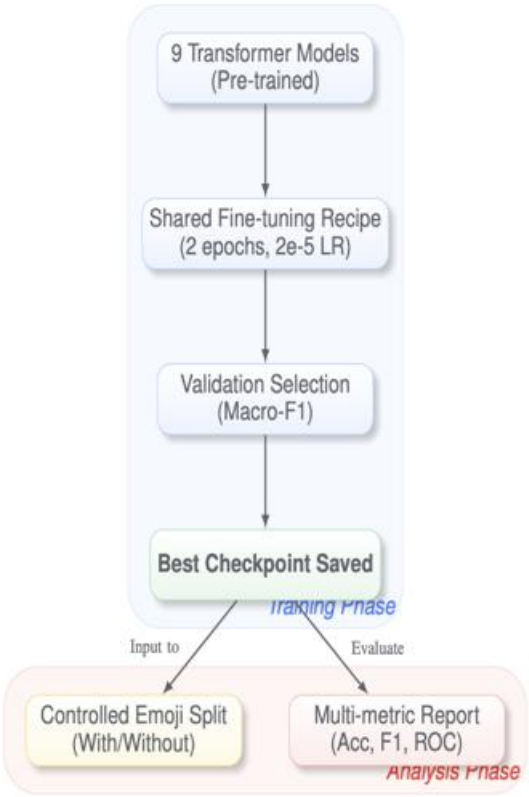

Fig. 2. Model selection and evaluation workflow: standardized fine-tuning,validation-based checkpoint selection, multi-metric testing, and controlled emoji-split analysis.

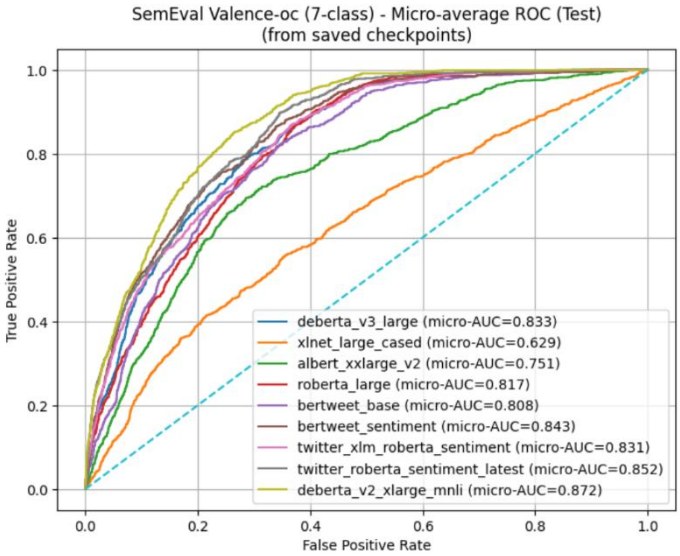

Fig. 3. SemEval Valence-oc (7-class) micro-average ROC on the test set from saved checkpoints. Higher curves indicate better discrimination under micro-averaging (aggregating all class decisions). DEBERTA-V2-XLARGE-MNLI achieves the highest micro-AUC (0.872), while XLNET-LARGE-CASED is lowest (0.629), indicating poor separability in the 7-class tweet-valence setting.

setting. Across all metrics, DEBERTA-V2-XLARGE-MNLI ranks best, achieving the highest micro-AUC (0.872) and the strongest overall Macro-F1 (0.385), indicating superior global separability and more balanced performance across classes.In contrast, XLNET-LARGE-CASED exhibits weak class discrimination (micro-AUC 0.629) and near-failure under Macro-F1 (0.062), despite being considered competitive in common binary sentiment leaderboards. The strong performance of DEBERTA-V2-XLARGE-MNLI suggests that NLI fine-tuning may improve sentence-level semantics and calibration, even when transferred to fine-grained valence tasks. We treat this as an empirical finding under a fixed recipe; it does not imply MNLI supervision is universally optimal for valence,but it suggests transfer benefits for fine-grained sentence- level semantics. Confirming this mechanism would require an ablation comparing MNLI vs non-MNLI initialization under identical conditions. This directly illustrates that leaderboard strength on binary sentiment tasks does not reliably transfer to fine-grained tweet-valence classification. We therefore do not recommend accuracy-only model selection for 7-class valence

TABLE I

SEMEVAL VALENCE-OC (7-CLASS) TEST PERFORMANCE. BEST RESULTS ARE BOLDED. ΔACC EMOJI IS THE DIFFERENCE IN ACCURACY: ACCWITH-EMOJI− ACCNO-EMOJI ; DELTAS SHOWN AS DECIMALS; FIG. 4 SHOWS THE SAME VALUES IN PERCENT.

Model	Acc	Macro-F1	Micro-AUC	$\Delta^{\text{Acc}}_{\text{emoji}}$
DeBERTa-v2-xlarge-MNLI	**0.492**	**0.385**	**0.872**	+0.063
Twitter-RoBERTa (sentiment-latest)	0.467	0.345	0.852	+0.058
BERTweet (sentiment)	0.472	0.348	0.843	-0.013
DeBERTa-v3-large	0.449	0.326	0.833	+0.099
Twitter-XLM-R (sentiment)	0.460	0.353	0.831	+0.063
RoBERTa-large	0.393	0.284	0.817	+0.040
BERTweet-base	0.396	0.236	0.808	-0.018
ALBERT-xxlarge-v2	0.367	0.226	0.751	-0.061
XLNet-large-cased	0.280	0.062	0.629	-0.099

RJ

B. The emoji effect is model-dependent under a controlled split

To isolate emoji impact, we partition the fixed test set into tweets containing at least one emoji (n = 251) and tweets without emojis (n= 686), and evaluate the same saved check-points on both subsets. The resulting accuracy deltas (ΔAcc E) show that emoji presence can either improve or degrade performance depending on the model. For instance, DEBERTA-V3-LARGEimproves substantially on emoji-containing tweets (+0.099), whereas XLNET-LARGE-CASED degrades by a comparable magnitude (-0.099). Tweet-specialized checkpoints (Twitter-RoBERTa and Twitter-XLM-R) show consistent gains (+0.058 and +0.063), while some models degrade (e.g., BERTweet sentiment: -0.013). These results support a practical conclusion: emoji robustness should be validated empirically within the target label space and domain, rather than assumed from generic benchmark rankings.

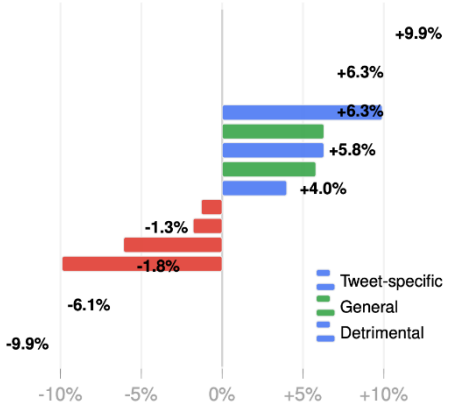

Fig. 4. The Emoji Effect Landscape: Diverging accuracy deltas (ΔAccE) reveal strongheterogeneity. While some tweet-specialized models (green) and high-capacity general models (blue) effectively leverage emojis (+5-10%), others(red) are confused by them (-6-10%), degrading performance relative to plain text.

C. Additional Benchmarks: TweetEval (Descriptive Only) TweetEval is used only to replicate the controlled emoji-split protocol within another tweet dataset; absolute numbers are reported descriptively and are not compared across datas

D. TweetEval Emoji Subset Analysis (Controlled Within-Dataset)To further test the model-dependency of emoji effects,we evaluate pre-trained sentiment checkpoints on TweetEval sentiment test data split by emoji presence. The TweetEvaltest set contains n= 809 tweets with emojis and n= 11,475 tweets without emojis. Table III shows that Twitter-RoBERTa improves with emojis, whereas a BERTweet sentiment checkpoint degrades under the same split.

TABLE II

DESCRIPTIVE PERFORMANCE ON TWEETEVAL (SENTIMENT). SST-2 IS OMITTED TO PRIORITIZE IN-DOMAIN THEORETICAL FIT

Model	Acc	Macro-F1
microsoft/deberta-v3-large	0.7620	0.7488
roberta-large	0.7605	0.7450
vinai/bertweet-base	0.7565	0.7401
cardiffnlp/twitter-roberta-base-sentiment-latest	0.7650	0.7548

TABLE III

TWEETEVAL SENTIMENT TEST SPLIT BY EMOJI PRESENCE. COLUMNS GROUPED BY METRIC TO FACILITATE DIRECT "WITH VS NO" COMPARISON

Model	Accuracy		Macro-F1	
	With	No	With	No
twitter-roberta-sentiment-latest	0.7454	0.7202	0.7246	0.7185
bertweet-base-sentiment-analysis	0.6984	0.7223	0.6966	0.7197
twitter-xlm-roberta-sentiment	0.6959	0.6810	0.6749	0.6796

6. Discussion

We quantify emoji sensitivity using a within-dataset partition (emoji-present vs. emoji-absent), allowing for an evaluation where the dataset, labels, and fine-tuning recipe remain fixedReusable artifact. This protocol is model-agnostic and applicable to fixed-label text classification datasets where emoji presence can be operationalized consistently, making it a reusable methodology for future emoji-focused evaluations.

A. Metric Validity: Why Accuracy Misleads

In 7-class ordinal tasks, accuracy is dominated by the majority class (e.g., neutral) [1]. XLNET's failure (Table I) illustrates this: moderate accuracy (0.280) but near-zero Macro-F1 (0.062) implies it simply predicts common classes. Therefore,we treat accuracy as insufficient for model selection in 7-class valence and prioritize Macro-F1. Macro-F1 is thus required to expose minority-class failures.

B. Design Principles for Emoji Evaluation

Based on our findings, we propose the following principles for future emoji-sentiment research:

1) Control the Split: Estimate emoji effects (ΔP E) only within fixed dataset/label boundaries.

2) Multi-Metric Reporting: Report Acc, Macro-F1, and micro-AUC together to catch class-imbalance failures.

3) Hypothesis Generation: Treat cross-dataset comparisons as exploratory, not confirmatory.

C. Interpreting Micro-AUC vs. Accuracy

ROC-AUC in Fig. 3 characterizes separability from probability scores [14], which can be high even when hard-label accuracy is moderate. For imbalanced multi-class tasks, Micro- AUC and Macro-F1 are often more diagnostic than accuracy alone, as they reveal whether a model can separate nuanced sentiment levels and whether it behaves comparably across minority classes [15].

D. Why the Emoji Effect Is Model-Dependent

We highlight three plausible drivers:

1) Pre-training domain. Tweet-pretrained models (e.g.,BERTweet, Twitter-RoBERTa) observe emoji usage patterns during pre-training, which can be leveraged during inference or fine-tuning.

2) Tokenization strategy. Byte-level BPE (RoBERTa variants) can represent arbitrary Unicode symbols, whereas other tokenizers may fragment emojis. Some pipelines normalize emojis into textual descriptors; mismatches between pre-training and evaluation tokenization can alter behavior.

3) Model capacity and inductive bias. High-capacity architectures (e.g., DeBERTa variants) may learn effective emoji–sentiment associations during fine-tuning even if emoji exposure is limited in pre-training.

The controlled splits reported in Table I and Table III demonstrate that emoji presence can either help or harm depending on model design and training history. The sign flips in ΔP E are consistent with differences in (F1) tweet-

domain exposure and (F2) tokenization behavior, supporting the claim that emoji effects are model-dependent rather than universal.

7. Limitations And Validity Considerations

This section outlines key validity considerations and study limitations to support accurate interpretation of the reported emoji-effect estimates.

A. Statistical Conclusion Considerations

SemEval Valence-oc is moderate in size; thus, fine-tuning variance (seed/optimization noise) can affect small performance differences and model rank order. We fix a shared fine-tuning recipe for comparability and emphasize robust qualitative patterns (metric discordance and model-dependent emoji deltas) rather than marginal rank swaps. Future work should add multi-seed runs with uncertainty estimates (e.g., bootstrap confidence intervals) and paired tests on predictions when applicable.

B. Internal Validity Considerations (Variable Isolation)

The evaluation-time partition isolates emoji presence while holding model parameters and the dataset/label space constant. However, emoji presence may be correlated with other properties (informality, topic, intensity, sarcasm markers, or author style). Therefore, ΔP E should be interpreted as the effect of "being in the emoji-containing subset" under the dataset distribution, not as a guaranteed causal effect of the emoji characters alone. Stronger isolation would require controlled interventions such as matched-pair edits (remove/replace emojis while preserving text) or propensity-score matching on non-emoji covariates; we position these as future extensions.

C. Measurement and Construct Considerations

We operationalize emoji presence using Unicode emoji characters. This may miss emoticons (e.g., ":-)"), platform-specific glyph variants, or sequences that are tokenized in- consistently. In addition, "emoji effect" is not decomposed by emoji identity, count, position, repetition, or emoji text incongruity. The current study estimates an aggregate effect of emoji presence; future work should stratify by emoji types and structural features, and distinguish semantic emojis from purely decorative ones.

D. Metric Considerations

The task is ordinal (7-level valence), yet the primary metrics (Accuracy, Macro-F1, micro-ROC-AUC) do not explicitly encode ordinal distance. Micro-AUC can overweight frequent classes and may remain high even when hard-label decisions are poor for minority classes. We mitigate this by reporting Macro-F1 alongside Accuracy and micro-AUC, and by using ROC curves as a diagnostic rather than a sole ranking criterion [16], [17]. Future work should add ordinal-aware metrics (e.g., MAE over label indices, quadratic-weighted kappa [18]–[20]) and per-class analyses (confusion matrices, per-class AUC) to characterize where errors concentrate.

E. Generalization Considerations

Findings are grounded in English tweet-like datasets (SemEval; and TweetEval for a protocol replication). Generalization to other languages, other platforms (e.g., Instagram comments), or other sentiment formulations depends on changes in emoji usage norms and annotation policies. Moreover, model behavior depends on pre-training corpora and tokenization; thus, the same protocol should be re-run when the deployment domain, tokenizer, or normalization pipeline differs.

F. Reproducibility

Our conclusions rely on controlling the training recipe and evaluating saved checkpoints under a fixed protocol. To ensure reproducibility, future artifact releases should include: exact model identifiers and versions, preprocessing and emoji- detection rules, train/validation/test splits, random seeds, and scripts for computing all metrics and plots.

8. Conclusion

We presented a standardized benchmark of nine transformer models on a seven-class valence task and introduced a controlled within-dataset protocol to test the emoji question without confounds. DEBERTA-V2-XLARGE-MNLI emerges as the strongest overall performer under multi-metric evaluation,while emoji impact varies sharply across models on both SemEval and TweetEval splits. Methodologically, our results support a simple

recommendation: if the research question concerns the effect of emojis, the experiment should hold the dataset and label space constant and vary only emoji presence at evaluation time. In addition, we recommend reporting suites of metrics (at minimum, Macro-F1 and ROC-AUC alongside accuracy) for nuanced multi-class sentiment tasks.

References

1. M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," Information Processing & Management, vol. 45, no. 4, pp. 427–437, 2009.
2. W. R. Shadish, T. D. Cook, and D. T. Campbell, Experimental and Quasi-Experimental Designs for Generalized Causal Inference. Boston, MA, USA: Houghton Mifflin, 2002.
3. M. A. Hern´ an and J. M. Robins, Causal Inference: What If. Boca Raton, FL, USA: Chapman & Hall/CRC, 2020.
4. S. Mohammad, F. Bravo-Marquez, M. Salameh, and S. Kiritchenko,"Semeval-2018 task 1: Affect in tweets," in Proceedings of the 12th International Workshop on Semantic Evaluation. Association for Computational Linguistics, 2018, pp. 1–17.
5. Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," CoRR, vol. abs/1907.11692, 2019. [Online]. Available: https://arxiv.org/abs/1907.11692
6. P. He, X. Liu, J. Gao, and W. Chen, "Deberta: Decoding-enhanced bert with disentangled attention," CoRR, vol. abs/2006.03654, 2020.[Online]. Available: https://arxiv.org/abs/2006.03654
7. Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, and Q. V. Le, "Xlnet: Generalized autoregressive pretraining for language understanding," CoRR, vol. abs/1906.08237, 2019. [Online]. Available: https://arxiv.org/abs/1906.08237
8. Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," in International Conference on Learning Representations, 2020. [Online]. Available: https://arxiv.org/abs/1909.11942
9. D. Q. Nguyen, T. Vu, and A. T. Nguyen, "Bertweet: A pre-trained language model for english tweets," in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. Association for Computational Linguistics, 2020, pp.9–14.
10. F. Barbieri, J. Camacho-Collados, L. Espinosa Anke, and L. Neves, "Tweeteval: Unified benchmark and comparative evaluation for tweet classification," in Findings of the Association for Computational Linguistics: EMNLP 2020. Association for Computational Linguistics,2020, pp. 1644–1650.
11. A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzm´ an, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, "Unsupervised cross-lingual representation learning at scale," in Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2020, pp. 8440– 8451.
12. R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2013, pp. 1631–1642.
13. T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer,P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. Le Scao, S. Gugger,M. Drame, Q. Lhoest, and A. M. Rush, "Transformers: State-of-the-art natural language processing," in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. Association for Computational Linguistics, 2020, pp. 38–45.
14. T. Fawcett, "An introduction to roc analysis," Pattern Recognition Letters, vol. 27, no. 8, pp. 861–874, 2006.
15. D. J. Hand and R. J. Till, "A simple generalisation of the area under the roc curve for multiple class classification problems," Machine Learning,vol. 45, pp. 171–186, 2001.
16. J. S. Cardoso and R. Sousa, "Measuring the performance of ordinal classification," International Journal of Pattern Recognition and Artifi-cial Intelligence, 2011.
17. E. Amig´o et al., "An effectiveness metric for ordinal classification: Formal and experimental results," in Proceedings of ACL, 2020.
18. J. Cohen, "Weighted kappa: Nominal scale agreement with provision for scaled disagreement or partial credit," Psychological Bulletin, vol. 70, no. 4, pp. 213–220, 1968.
19. J. L. Fleiss and J. Cohen, "The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability," Educational and Psychological Measurement, vol. 33, no. 3, pp. 613–619, 1973.
20. M. J. Warrens, "Cohen's quadratically weighted kappa is higher thanlinearly weighted kappa for tridiagonal agreement tables," Statistical Methodology, 2012.