

Penerapan *Fine-Tuning* Model Bert Untuk Efektivitas Klasifikasi Teks

Mhd Syafiq Hasan Jambak^{1*}, Mahardika Abdi Prawira Tanjung

¹ Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Muhammadiyah Sumatera Utara, Medan, Indonesia

Article Info	ABSTRAK
Article history: Received Oktober 2, 2026 Revised Oktober 2, 2026 Accepted Oktober 2, 2026 Kata Kunci: Analisis Sentimen, BERT, Fine-Tuning, Feature Extraction, Pemrosesan Bahasa Alami Keywords: <i>Sentiment Analysis, BERT, Fine-Tuning, Feature Extraction, Natural Language Processing</i>	Klasifikasi sentimen teks menghadapi kompleksitas semantik dan ambiguitas konteks. BERT menawarkan representasi kontekstual dua arah, tetapi strategi adaptasi antara feature extraction dan end-to-end fine-tuning memerlukan pembuktian empiris dari sisi akurasi dan efisiensi komputasi. Penelitian ini menganalisis Model A (feature extraction) dan Model B (end-to-end fine-tuning) berbasis bert-base-uncased pada Stanford Sentiment Treebank (SST-2) dengan 872 sampel held-out test set. Eksperimen dijalankan pada enam random seed dan dievaluasi menggunakan akurasi, precision, recall, F1-score, uji McNemar, Wilcoxon Signed-Rank, Bootstrap 95% Confidence Interval, Cohen's d, konsumsi VRAM, latensi inferensi, dan ketahanan linguistik. Model B mencapai akurasi $92,78\% \pm 0,71\%$ dan F1-score $92,94\% \pm 0,79\%$, sedangkan Model A mencapai $86,05\% \pm 0,69\%$ dan $86,66\% \pm 0,43\%$. Perbedaan signifikan pada Wilcoxon ($p=0,03125$) dan McNemar ($\chi^2=38,1635$; $p=6,51 \times 10^{-10}$), dengan Cohen's $d=10,11$. Latensi GPU hampir identik, sedangkan Model A lebih hemat VRAM hingga 68,7% pada clean state. Model B juga lebih kuat pada ulasan panjang dan negasi biner. Hasil ini menunjukkan fine-tuning lebih efektif untuk akurasi dan pemahaman konteks, sementara feature extraction lebih sesuai pada lingkungan dengan keterbatasan memori GPU. ABSTRACT <i>Text sentiment classification faces semantic complexity and contextual ambiguity. BERT provides powerful bidirectional contextual representations, but selecting between feature extraction and end-to-end fine-tuning requires empirical evidence regarding predictive accuracy and computational efficiency. This study compares Model A (feature extraction) and Model B (end-to-end fine-tuning) using bert-base-uncased on the Stanford Sentiment Treebank (SST-2) with 872 held-out test samples. Experiments were conducted across six random seeds and evaluated using accuracy, precision, recall, F1-score, McNemar's test, Wilcoxon Signed-Rank test, Bootstrap 95% Confidence Interval, Cohen's d, GPU VRAM consumption, inference latency, and linguistic robustness. Model B achieved $92.78\% \pm 0.71\%$ accuracy and $92.94\% \pm 0.79\%$ F1-score, while Model A achieved $86.05\% \pm 0.69\%$ and $86.66\% \pm 0.43\%$, respectively. The difference was significant under Wilcoxon ($p=0.03125$) and McNemar ($\chi^2=38.1635$; $p=6.51 \times 10^{-10}$), with Cohen's $d=10.11$. GPU inference latency was nearly identical, whereas Model A reduced VRAM allocation by up to 68.7% under a clean state. Model B also showed higher robustness on long reviews and binary negation. These results establish fine-tuning as more effective for predictive accuracy and contextual understanding, while feature extraction remains suitable for memory-constrained deployment.</i>

This is an open access article under the [CC BY](#) license



Corresponding Author:

Mhd Syafiq Hasan Jambak
Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Muhammadiyah Sumatera Utara,
Medan, Indonesia
Email: syafiqhsn012@gmail.com

1. PENDAHULUAN

Klasifikasi teks merupakan salah satu tugas fundamental dalam Natural Language Processing (NLP) yang bertujuan mengelompokkan unit teks ke dalam kategori tertentu berdasarkan makna semantiknya [1]. Pendekatan klasik berbasis fitur seperti TF-IDF dan algoritma pembelajaran mesin konvensional memiliki keterbatasan dalam menangkap urutan kata dan hubungan konteks yang kompleks. Terobosan besar terjadi ketika arsitektur Transformer diperkenalkan dengan mekanisme self-attention [2], kemudian dikembangkan menjadi Bidirectional Encoder Representations from Transformers (BERT) yang memanfaatkan representasi kontekstual dua arah dan paradigma transfer learning [3]. BERT menunjukkan kemampuan yang kuat pada berbagai benchmark NLP, termasuk GLUE [4].

Dalam penerapannya, model pra-terlatih BERT dapat diadaptasikan melalui dua strategi utama, yaitu frozen feature extraction dengan mempertahankan bobot backbone dan hanya melatih classifier, serta end-to-end fine-tuning dengan memperbarui seluruh bobot model. Penelitian terdahulu menunjukkan efektivitas BERT dan fine-tuning untuk klasifikasi teks [5], [6], sedangkan studi mengenai dinamika fine-tuning menunjukkan bahwa proses pembaruan bobot mengubah representasi internal model [9]. Namun, sebagian besar studi tersebut tidak membandingkan feature extraction dan fine-tuning secara terkontrol dengan konfigurasi dataset dan parameter yang simetris.

Kesenjangan lain terletak pada validasi statistik dan evaluasi sumber daya komputasi. Peningkatan metrik prediktif tanpa pengujian signifikansi berpotensi mencerminkan variasi acak [7], sedangkan masalah komparabilitas dan reliabilitas masih menjadi perhatian dalam eksperimen klasifikasi teks [8]. Evaluasi yang hanya berfokus pada akurasi juga belum cukup untuk menjelaskan trade-off antara peningkatan performa dan biaya komputasi. Karena itu, penelitian ini membandingkan Model A (BERT sebagai feature extraction) dan Model B (BERT dengan end-to-end fine-tuning) secara terkontrol pada dataset SST-2, dengan evaluasi metrik prediktif, signifikansi statistik, konsumsi VRAM, latensi inferensi, ketahanan terhadap fenomena linguistik, dan implementasi aplikasi web komparator real-time.

Tujuan penelitian adalah mengukur perbedaan performa klasifikasi kedua strategi berdasarkan akurasi, precision, recall, dan F1-score; menganalisis signifikansi perbedaan dengan uji McNemar, Wilcoxon Signed-Rank, Bootstrap Confidence Interval, dan Cohen's d; mengevaluasi trade-off performa dan biaya komputasi; serta mengimplementasikan aplikasi web berbasis FastAPI dan React untuk membandingkan inferensi sentimen dan menampilkan statistik benchmark.

2. METODE

2.1. Desain Penelitian dan Dataset

Penelitian menggunakan desain eksperimen komparatif terkontrol terhadap dua konfigurasi model BERT pada tugas klasifikasi sentimen biner. Dataset yang digunakan adalah Stanford Sentiment Treebank 2 (SST-2), bagian dari benchmark GLUE. Data latih resmi SST-2 sebanyak 67.349 sampel dipartisi ulang menjadi train set internal sebanyak 60.614 sampel dan validation set internal sebanyak 6.735 sampel. Data validasi resmi GLUE sebanyak 872 sampel digunakan murni sebagai held-out test set agar evaluasi akhir terisolasi dari proses pelatihan dan pemilihan model [4], [11].

Tabel 1. Distribusi Data pada Dataset SST-2

Subset Data	Jumlah Sampel	Fungsi
Train internal	60.614	Pelatihan dan pembaruan bobot
Validation internal	6.735	Validasi iterasi dan early stopping
Held-out test set	872	Evaluasi akhir dan uji statistik
Total digunakan	68.221	Total sampel berlabel aktif

Untuk mengendalikan variasi stokastik, eksperimen dijalankan pada enam random seed, yaitu 42, 123, 777, 999, 1234, dan 2024. Pra-pemrosesan menggunakan BertTokenizerFast dengan panjang urutan maksimum 128 token. Kedua model menggunakan arsitektur bert-base-uncased, batch size 32, CrossEntropyLoss, optimizer AdamW, weight decay 0,01, warmup ratio 0,1, dan early stopping berdasarkan Validation F1 dengan patience tiga epoch. Implementasi model menggunakan pustaka Transformers [20].

2.2. Konfigurasi Model

Model A menggunakan BERT sebagai feature extraction. Seluruh parameter backbone BERT dibekukan dan representasi token [CLS] berdimensi 768 diteruskan ke classification head linear 768→2. Jumlah parameter yang dilatih sekitar 1.538 parameter. Learning rate ditetapkan 1×10^{-3} dan maksimum pelatihan 10 epoch. Model B menggunakan BertForSequenceClassification dengan end-to-end fine-tuning sehingga seluruh parameter model dapat diperbarui. Learning rate Model B adalah 2×10^{-5} dengan maksimum lima epoch. Konfigurasi tersebut menjaga arsitektur dasar, tokenizer, ukuran batch, dan fungsi loss tetap sama sehingga perbedaan hasil terutama merefleksikan strategi adaptasi model.

2.3. Teknik Evaluasi dan Analisis Statistik

Performa prediktif dievaluasi menggunakan accuracy, precision, recall, dan F1-score. F1-score dipakai sebagai ukuran yang menyeimbangkan precision dan recall [12]. Stabilitas hasil diamati pada enam seed, sejalan dengan kebutuhan memperhitungkan varians pada benchmark machine learning [13]. Signifikansi perbedaan prediksi berpasangan pada held-out test set dianalisis menggunakan uji McNemar [14]. Distribusi F1-score antar-seed dibandingkan menggunakan Wilcoxon Signed-Rank [15], sedangkan ketidakpastian selisih performa dianalisis dengan Percentile Bootstrap 10.000 resamples [16]. Besarnya pengaruh perbedaan dihitung menggunakan Cohen's d [17].

Efisiensi komputasi dievaluasi melalui latensi inferensi dan konsumsi memori GPU. Pengukuran VRAM menggunakan peak allocated memory dan peak reserved memory, sedangkan evaluasi robustness linguistik dibagi menjadi lima kategori: tanpa negasi, negasi biner, ironi/sarkasme dan negasi majemuk, review panjang lebih dari 40 token, serta ambiguitas tinggi. Kategori ambiguitas menggunakan dukungan leksikon AFINN [21]. Evaluasi perilaku berbasis kategori digunakan untuk melihat kegagalan model di luar metrik agregat, sejalan dengan pendekatan behavioral testing NLP [18].

2.4. Implementasi Aplikasi Web

Hasil eksperimen diintegrasikan ke dalam BERT Sentiment Lab, aplikasi web komparator real-time dengan backend FastAPI dan frontend React 18/Vite yang mendukung Progressive Web App. Sistem menyediakan komparator inferensi dua model, dashboard benchmark statistik, riwayat prediksi, autentikasi berbasis peran, dan mekanisme dual-backend hybrid yang memungkinkan failover GPU ke CPU. Pengujian sistem meliputi unit testing backend, integration testing, failover testing, serta evaluasi kebolehgunaan menggunakan System Usability Scale (SUS) [19].

3. HASIL DAN PEMBAHASAN

3.1. Performa Prediktif Model

Pada validation set internal sebanyak 6.735 sampel, Model A memperoleh rerata Validation Accuracy $86,87\% \pm 0,12\%$ dan Validation F1-Score $88,15\% \pm 0,05\%$, sedangkan Model B memperoleh $95,44\% \pm 0,05\%$ dan $95,83\% \pm 0,05\%$. Evaluasi akhir kemudian dilakukan pada held-out test set sebanyak 872 sampel untuk setiap seed dengan data uji yang sama. Tabel 2 merangkum rerata performa prediktif dari enam seed.

Tabel 2. Rerata Performa Prediktif pada Held-out Test Set (N=872, n=6 seed)

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Model A - Feature Extraction	$86,05 \pm 0,69$	$84,51 \pm 1,62$	$88,96 \pm 1,03$	$86,66 \pm 0,43$
Model B - Fine-Tuning	$92,78 \pm 0,71$	$92,45 \pm 1,04$	$93,48 \pm 2,16$	$92,94 \pm 0,79$
Selisih (Δ)	+6,73	+7,94	+4,52	+6,28

Model B mengungguli Model A secara konsisten di seluruh metrik prediktif. Peningkatan rerata akurasi mencapai 6,73 poin persentase dan F1-score meningkat 6,28 poin persentase. Hasil ini menunjukkan bahwa pembaruan seluruh bobot BERT memungkinkan representasi internal menyesuaikan diri dengan pola semantik pada SST-2. Temuan tersebut konsisten dengan studi fine-tuning BERT untuk klasifikasi teks [5], [10] dan analisis dinamika pembelajaran BERT yang menunjukkan perubahan representasi internal selama fine-tuning [9].

3.2. Signifikansi Statistik

Uji McNemar pada prediksi berpasangan seed 42 menghasilkan $\chi^2=38,1635$ dengan $p=6,51 \times 10^{-10}$, sehingga perbedaan proporsi kesalahan kedua model sangat signifikan. Pada enam seed, uji Wilcoxon dua sisi menghasilkan $W=0,0$ dan $p=0,03125$. Percentile Bootstrap pada tingkat sampel memberikan 95% confidence interval selisih F1-score sebesar +4,67% sampai +8,85%, sehingga interval tidak melintasi nol. Cohen's d sebesar 10,11 menunjukkan ukuran efek yang sangat besar. Hasil lengkap dirangkum pada Tabel 3.

Tabel 3. Ringkasan Pengujian Statistik Inferensial

Metode	Statistik/Hasil	Interpretasi
McNemar	$\chi^2=38,1635$; $p=6,51 \times 10^{-10}$	Perbedaan error berpasangan signifikan
Wilcoxon Signed-Rank	$W=0,0$; $p=0,03125$	Perbedaan F1 antar-seed signifikan
Bootstrap 95% CI	+4,67% s.d. +8,85%	Interval positif dan tidak melintasi nol
Cohen's d	$d=10,11$	Ukuran efek sangat besar

Analisis Jackknife leave-one-out pada enam kombinasi subset 5-of-6 seed menghasilkan $W=0,0$ pada seluruh subset dengan $p=0,0625$. Nilai p tersebut merupakan batas resolusi minimum uji Wilcoxon dua sisi untuk $n=5$, sehingga robustness ditunjukkan oleh konsistensi arah perbedaan dan statistik W minimum pada seluruh subset, bukan oleh pencapaian $p<0,05$. Dengan demikian, keunggulan Model B tidak bergantung pada satu seed tertentu.

3.3. Efisiensi Komputasi

Latensi inferensi GPU kedua model hampir identik, yaitu sekitar 1,82 ms per sampel untuk Model A dan 1,83 ms untuk Model B. Perbedaan utama terdapat pada konsumsi VRAM. Pada fase inferensi, Model A mencatatkan rerata peak allocated VRAM sebesar $2.812,93 \pm 891,6$ MB, sedangkan Model B

sebesar $3.173,13 \pm 4,4$ MB. Pada initial clean GPU state seed 42, Model A hanya mengalokasikan 992,93 MB, sedangkan Model B 3.170,63 MB, sehingga penghematan Model A mencapai sekitar 68,7%.

Tabel 4. Ringkasan Efisiensi Komputasi

Indikator	Model A	Model B	Selisih
Latensi inferensi GPU (ms)	$1,82 \pm 0,01$	$1,83 \pm 0,01$	0,01
Peak allocated VRAM inferensi (MB)	$2.812,93 \pm 891,6$	$3.173,13 \pm 4,4$	360,20
Initial clean run VRAM (MB)	992,93	3.170,63	2.177,70
Peak allocated training VRAM (MB)	$3.093,98 \pm 886,9$	$5.483,50 \pm 2,9$	2.389,52

Pada fase pelatihan, perbedaan lebih besar karena Model B harus menyimpan gradient tensor dan optimizer state untuk sekitar 109,48 juta parameter yang dapat dilatih, sedangkan Model A hanya memperbarui classification head. Rerata peak allocated training VRAM Model A adalah $3.093,98 \pm 886,9$ MB dan Model B $5.483,50 \pm 2,9$ MB. Hasil ini memperlihatkan trade-off yang jelas: fine-tuning memberikan peningkatan akurasi yang besar tanpa penalti latensi inferensi yang berarti, tetapi membutuhkan alokasi memori pelatihan dan inferensi yang lebih tinggi.

3.4. Analisis Kesalahan Linguistik

Evaluasi pada lima kategori fenomena linguistik menunjukkan bahwa Model B unggul pada seluruh kategori. Peningkatan terbesar terlihat pada negasi biner dan review panjang. Pada negasi biner, akurasi Model B mencapai 93,6% dibandingkan Model A 80,3%. Pada review panjang lebih dari 40 token, Model B mencapai 89,5%, sedangkan Model A 77,6%. Tabel 5 menyajikan rincian hasil berdasarkan kategori.

Tabel 5. Hasil Analisis Kesalahan Linguistik

Kategori	N	Model A (%)	Model B (%)	Δ (poin %)
Tanpa negasi	674	87,2	93,3	+6,1
Negasi biner	173	80,3	93,6	+13,3
Ironi/sarkasme & negasi majemuk	149	79,2	89,9	+10,7
Review panjang >40 token	76	77,6	89,5	+11,9
Ambiguitas tinggi	29	82,8	89,7	+6,9

Kesalahan Model A lebih banyak muncul pada struktur yang membutuhkan pemahaman konteks lintas-token. Pada konstruksi negasi, representasi statis dari backbone yang dibekukan membatasi kemampuan penyesuaian pola attention, sedangkan Model B dapat memperbarui seluruh bobot untuk menyesuaikan hubungan kontekstual. Pada review panjang, fine-tuning juga lebih mampu mempertahankan dependensi jarak jauh melalui mekanisme self-attention. Pola ini mendukung temuan bahwa evaluasi model NLP perlu melampaui akurasi agregat dan memeriksa perilaku pada kategori linguistik tertentu [18].

3.5. Implementasi dan Pengujian Aplikasi Web

BERT Sentiment Lab berhasil diimplementasikan menggunakan FastAPI pada backend dan React 18/Vite pada frontend PWA. Pengujian backend menghasilkan code coverage 86,5% dengan 15 dari 15 test case lulus, integration testing mencapai 100% pada lima modul, dan mekanisme failover

otomatis GPU ke CPU membutuhkan 1,24 detik. Evaluasi System Usability Scale terhadap 11 responden valid dari 15 total responden menghasilkan skor 70,23, berada di atas ambang acceptable 68. Aplikasi menyediakan perbandingan inferensi real-time, dashboard statistik benchmark, riwayat prediksi, autentikasi berbasis peran, serta dukungan instalasi PWA.

3.6. Pembahasan Komparatif

Secara keseluruhan, eksperimen menunjukkan bahwa end-to-end fine-tuning lebih efektif ketika tujuan utama adalah akurasi prediktif dan pemahaman konteks semantik. Hasil tersebut memperkuat penelitian yang melaporkan peningkatan performa BERT melalui transfer learning dan fine-tuning [5], [6], [10]. Kontribusi penelitian ini terletak pada perbandingan terkontrol antara feature extraction dan fine-tuning dengan arsitektur dasar, dataset, dan skema evaluasi yang sama, kemudian dilengkapi pengujian statistik inferensial, analisis konsumsi VRAM, evaluasi error linguistik, dan implementasi sistem aplikasi web. Pendekatan multi-seed juga mengurangi risiko menarik kesimpulan dari satu run yang dipengaruhi variasi stokastik [13].

Di sisi lain, feature extraction masih relevan untuk lingkungan yang dibatasi kapasitas memori GPU. Model A menggunakan parameter terlatih yang jauh lebih sedikit dan mengonsumsi VRAM lebih rendah, terutama pada clean state dan fase pelatihan. Oleh karena itu, pemilihan strategi adaptasi BERT perlu disesuaikan dengan prioritas implementasi: Model B untuk kebutuhan performa prediktif dan Model A ketika efisiensi memori menjadi batas utama. Penelitian ini tetap terbatas pada satu arsitektur BERT dan satu dataset sentimen biner berbahasa Inggris, sehingga generalisasi lintas model dan lintas domain belum diuji.

4. KESIMPULAN

Penerapan End-to-End Fine-Tuning pada BERT terbukti lebih efektif daripada Feature Extraction untuk klasifikasi sentimen SST-2. Model B mencapai rerata Accuracy $92,78\% \pm 0,71\%$ dan F1-Score $92,94\% \pm 0,79\%$, dibandingkan Model A sebesar $86,05\% \pm 0,69\%$ dan $86,66\% \pm 0,43\%$. Perbedaan ini signifikan berdasarkan McNemar ($\chi^2=38,1635$; $p=6,51 \times 10^{-10}$), Wilcoxon ($W=0,0$; $p=0,03125$), Bootstrap 95% CI yang seluruhnya positif, dan Cohen's $d=10,11$. Model B juga lebih kuat pada negasi biner dan review panjang, sementara Model A lebih hemat VRAM hingga 68,7% pada initial clean GPU state dan membutuhkan memori pelatihan yang lebih rendah. Latensi inferensi GPU kedua model hampir identik. Aplikasi BERT Sentiment Lab berhasil mengintegrasikan komparator inferensi dan dashboard statistik dengan failover GPU ke CPU 1,24 detik dan skor SUS 70,23. Dengan demikian, Model B lebih sesuai ketika prioritas utama adalah akurasi dan pemahaman konteks, sedangkan Model A relevan untuk lingkungan dengan keterbatasan memori GPU. Penelitian selanjutnya dapat memperluas jumlah seed, membandingkan varian Transformer lain, melakukan pencarian hyperparameter yang lebih luas, dan mengevaluasi dataset berbahasa Indonesia.

REFERENSI

- [1] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, and J. Gao, "Deep learning-based text classification: A comprehensive review," *ACM Computing Surveys*, vol. 54, no. 3, pp. 1–40, 2021.
- [2] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008.
- [3] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL-HLT*, 2019, pp. 4171–4186, doi: 10.18653/v1/N19-1423.

- [4] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, "GLUE: A multi-task benchmark and analysis platform for natural language understanding," in Proc. EMNLP Workshop BlackboxNLP, 2018, doi: 10.18653/v1/W18-5446.
- [5] R. Qasim, W. H. Bangyal, M. A. Alqarni, and A. Ali Almazroi, "A fine-tuned BERT-based transfer learning approach for text classification," Journal of Healthcare Engineering, vol. 2022, Art. no. 3498123, 2022.
- [6] H. Zaman-Khan, M. Naeem, R. Guarasci, U. Khalid, M. Esposito, and F. Gargiulo, "Enhancing text classification using BERT: A transfer learning approach," Computación y Sistemas, vol. 28, no. 4, pp. 2279–2295, 2024, doi: 10.13053/cys-28-4-5290.
- [7] R. Dror, G. Baumer, S. Shlomov, and R. Reichart, "The hitchhiker's guide to testing statistical significance in natural language processing," in Proc. 56th Annual Meeting of the Association for Computational Linguistics, 2018, pp. 1383–1392.
- [8] L. Galke, A. Diera, B. X. Lin, B. Khera, T. Meuser, T. Singhal, F. Karl, and A. Scherp, "Are we really making much progress in text classification? A comparative review," ACM Computing Surveys, 2024.
- [9] Y. Hao, L. Dong, F. Wei, and K. Xu, "Investigating learning dynamics of BERT fine-tuning," in Proc. AACL-IJCNLP, 2020, pp. 87–92.
- [10] C. Sun, X. Qiu, Y. Xu, and X. Huang, "How to fine-tune BERT for text classification?" in Chinese National Conference on Computational Linguistics, 2019, pp. 194–206.
- [11] 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 Proc. EMNLP, 2013, pp. 1631–1642, doi: 10.18653/v1/D13-1170.
- [12] 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, doi: 10.1016/j.ipm.2009.03.002.
- [13] X. Bouthillier et al., "Accounting for variance in machine learning benchmarks," Proceedings of Machine Learning and Systems, vol. 3, pp. 747–769, 2021.
- [14] Q. McNemar, "Note on the sampling error of the difference between correlated proportions or percentages," Psychometrika, vol. 12, no. 2, pp. 153–157, 1947, doi: 10.1007/BF02295996.
- [15] F. Wilcoxon, "Individual comparisons by ranking methods," Biometrics Bulletin, vol. 1, no. 6, pp. 80–83, 1945, doi: 10.2307/3001968.
- [16] B. Efron, "Bootstrap methods: Another look at the jackknife," The Annals of Statistics, vol. 7, no. 1, pp. 1–26, 1979, doi: 10.1214/aos/1176344552.
- [17] J. Cohen, Statistical Power Analysis for the Behavioral Sciences, 2nd ed. Lawrence Erlbaum Associates, 1988.
- [18] M. T. Ribeiro, T. Wu, C. Guestrin, and S. Singh, "Beyond accuracy: Behavioral testing of NLP models with CheckList," in Proc. 58th Annual Meeting of the Association for Computational Linguistics, 2020, pp. 4902–4912, doi: 10.18653/v1/2020.acl-main.442.
- [19] J. Brooke, "SUS: A quick and dirty usability scale," in Usability Evaluation in Industry, P. W. Jordan, B. Thomas, B. A. Weerdmeester, and I. L. McClelland, Eds. Taylor & Francis, 1996, pp. 189–194.
- [20] T. Wolf et al., "Transformers: State-of-the-art natural language processing," in Proc. 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, 2020, pp. 38–45, doi: 10.18653/v1/2020.emnlp-demos.6.
- [21] F. Å. Nielsen, "A new ANEW: Evaluation of a word list for sentiment analysis in microblogs," in Proc. ESWC 2011 Workshop on Making Sense of Microposts, 2011, pp. 93–98.