

Anonymous Submission Title

Anonymous submission

Abstract

Zero-shot text classification (ZSC) offers the promise of eliminating costly task-specific annotation by matching texts directly to human-readable label descriptions. While early approaches have predominantly relied on cross-encoder models fine-tuned for natural language inference (NLI), recent advances in text-embedding models and rerankers have challenged the dominance of NLI-based architectures. Existing evaluations, such as MTEB, often probe embedding models with supervised classifiers atop frozen embeddings, leaving true zero-shot capabilities underexplored. To address this, we introduce **BTZSC**, a comprehensive benchmark of 22 public datasets spanning sentiment, topic, intent, and emotion classification, capturing diverse domains, class cardinalities, and document lengths. Leveraging BTZSC, we conduct a systematic comparison across three major model families, NLI cross-encoders, embedding models, and rerankers, encompassing 31 public and custom checkpoints. Our results show that: (i) modern rerankers, exemplified by *Qwen3-Reranker-8B*, set a new state-of-the-art with macro $F_1 = 0.72$; (ii) strong embedding models such as *GTE-large-en-v1.5* substantially close the accuracy gap while offering the best trade-off between accuracy and latency; (iii) NLI cross-encoders plateau even as backbone size increases; and (iv) scaling primarily benefits rerankers over embedding models. BTZSC and accompanying evaluation code are publicly released to support fair and reproducible progress in zero-shot text understanding.

Introduction

Text classification is a foundational problem in Natural Language Processing (NLP), finding broad applications across diverse domains, including topic categorization of news articles, intent detection in conversational agents, sentiment analysis of product reviews, and emotion recognition in mental health support systems (Sebastiani 2002; Kowsari et al. 2019). Formally, the task involves assigning one or more predefined labels to textual data based solely on the content of the text (Sebastiani 2002). However, the supervised approach to text classification necessitates the creation of large-scale, high-quality annotated datasets, a process that is often prohibitively expensive, particularly in specialized domains requiring expert annotators (Settles 2012).

Text zero-shot classification (ZSC) addresses this challenge by enabling models to predict labels that have not

been explicitly observed during training (Yin, Hay, and Roth 2019). The core principle underlying ZSC methods is the exploitation of semantic relationships between input texts and candidate labels. This relationship is typically captured using pretrained language models, which encode semantics based on extensive pretraining on large textual corpora (Yin, Hay, and Roth 2019; Brown et al. 2020). One straightforward approach involves prompting large autoregressive language models (LLMs) directly with textual inputs and candidate label descriptions. While effective, this method entails considerable computational cost and latency, limiting its feasibility in real-time deployment scenarios (Brown et al. 2020).

A widely adopted, more computationally efficient alternative involves fine-tuning pretrained encoder models on Natural Language Inference (NLI) datasets, reframing classification tasks as entailment problems. Specifically, the input text acts as a premise and each candidate label as a hypothesis sentence (Yin, Hay, and Roth 2019; Bowman et al. 2015; Williams, Nangia, and Bowman 2018). NLI datasets, including SNLI (Bowman et al. 2015) and MultiNLI (Williams, Nangia, and Bowman 2018), contain sentence pairs annotated with labels indicating entailment, contradiction, or neutrality. By fine-tuning encoders on these corpora, models learn to discern semantic compatibility, thus enabling effective reuse in ZSC scenarios. Despite their success and lower computational demands relative to generative LLMs, improvements in NLI-based cross-encoder methods have plateaued in recent years.

Concurrent to this, significant advances have occurred in the domain of text-embedding models (Reimers and Gurevych 2019; Gao, Yao, and Chen 2021; Muennighoff et al. 2023). Embedding models learn mappings, $f : \text{text} \rightarrow \mathbb{R}^d$, from textual inputs to dense vector representations, ensuring semantically related texts are closely situated in the embedding space. This characteristic facilitates efficient similarity-based retrieval, and in principle, supports ZSC through nearest-neighbor matching to candidate label embeddings (Reimers and Gurevych 2019; Gao, Yao, and Chen 2021). The Massive Text Embedding Benchmark (MTEB) systematically evaluates embedding models across various tasks, encompassing 58 datasets categorized into eight families (Muennighoff et al. 2023). However, classification performance within MTEB is primarily assessed through lin-

ear probes trained on labeled data atop frozen embeddings, thereby leaving the genuine zero-shot capabilities of embedding models untested (Muennighoff et al. 2023).

Another promising class of models, rerankers, originally cross-encoder or sequence-to-sequence architectures designed to refine the ranking of query-document pairs (e.g., MonoT5 (Nogueira, Jiang, and Lin 2020)), can similarly be adapted for ZSC by treating textual inputs as queries and label descriptions as retrievable documents. However, the comparative performance and potential advantages of rerankers in zero-shot classification contexts remain under-explored.

Furthermore, the distinction between encoder-based and generative approaches is becoming increasingly blurred, as modern embedding models frequently leverage distilled or instruction-tuned variants of generative LLMs (e.g., Sentence-T5 (Ni et al. 2021), E5 (Wang et al. 2024)). Given these rapid developments, a systematic comparison between NLI cross-encoders, contemporary embedding models, and reranker architectures, particularly in genuine zero-shot settings across diverse classification tasks, remains an open research question.

To address this gap, we present a comprehensive benchmark study evaluating a diverse selection of models, including NLI-based cross-encoders, embedding models, and rerankers, across 22 datasets that span four major classification categories (sentiment, topic, intent, and emotion). This benchmark systematically explores the relative strengths, limitations, and transferability of these approaches, offering a comparative analysis to guide future research directions in zero-shot text classification.

Related Work

To our knowledge, the proposed benchmark, **BTZSC**, is the first to comprehensively compare NLI cross-encoders, embedding-based models, and reranker architectures in a true ZSC setting. Previous benchmarks for ZSC have typically been limited in scope, often restricted to evaluating a single model family, a narrow task category, or a handful of datasets. For instance, Yin, Hay, and Roth (2019) introduced a foundational NLI-based ZSC benchmark but evaluated exclusively cross-encoder models on only three datasets. Chalkidis et al. (2020) examined zero-shot learning specifically within multi-label classification but confined their analysis to three hierarchical datasets. Gretz et al. (2023) proposed TTC23, evaluating prompt-based methods solely for topic classification and omitted contemporary embedding and reranking models from their analysis. Lepagnol et al. (2024) further explored the performance of smaller language models (100M–1B parameters) across 15 datasets, yet their work excluded comparisons with embedding and reranker architectures. The Massive Text Embedding Benchmark (MTEB), alongside its multilingual counterpart, has established a mature, broad-ranging evaluation platform covering numerous datasets. However, MTEB assesses classification performance via supervised linear probes trained atop frozen embeddings, thereby leaving unanswered the question of embedding models’ genuine zero-shot capability (Muennighoff et al. 2023; Enevoldsen et al. 2025; Chung

et al. 2025). Consequently, this fragmented state of evaluation has hindered a clear understanding of cross-family comparative capabilities among these diverse model types.

Zero-Shot Text Classification

Zero-shot text classification fundamentally involves assigning labels unseen during training by assessing semantic compatibility between input texts and candidate labels, typically expressed in natural language. Unlike supervised approaches, ZSC methods avoid task-specific finetuning by leveraging pretrained models’ semantic representations. A common parallel in vision tasks is zero-shot image recognition with language-aligned models like CLIP (Radford et al. 2021), though textual classification benefits directly from the intrinsic expressivity and flexibility of natural language documents.

NLI-based cross-encoders represent one of the earliest and most prominent paradigms for zero-shot text classification. Such methods recast the classification problem into an entailment task, where each candidate label is paired with the input text as a hypothesis-premise pair scored by an NLI model (Yin, Hay, and Roth 2019). This approach has been operationalised effectively by public checkpoints like `facebook/bart-large-mnli` (Lewis et al. 2020), which powers the widely used zero-shot pipeline of HuggingFace Transformers (Wolf et al. 2020). More recent advances, including stronger encoder backbones like DeBERTa-v3 (He, Gao, and Chen 2023) and improved label verbalization techniques, have incrementally enhanced performance. Nonetheless, these improvements have plateaued when compared with rapid advancements from increasingly large generative language models (LLMs).

Text-embedding models have subsequently emerged as a highly active research domain, evolving significantly from early sentence embedding techniques such as InferSent (Conneau et al. 2017) and Google’s Universal Sentence Encoder (USE) (Cer et al. 2018). Contemporary embedding frameworks, notably E5 (Wang et al. 2024), GTE (Li et al. 2023), BGE (Chen et al. 2024), and Qwen3-Embedding (Zhang et al. 2025), have substantially raised performance standards. These models integrate sophisticated training strategies including billion-scale contrastive pretraining, multilingual supervision, multi-stage data scaling, and instruction fine-tuning. For example, E5 uses an instruction-tuned approach with massive-scale contrastive learning, GTE emphasizes data-scale expansion over parameter scale, and BGE combines dense, sparse, and multi-vector encoding techniques into a multilingual framework capable of handling extensive context lengths. Compared to foundational architectures such as SBERT (Reimers and Gurevych 2019), these advancements have resulted in improvements on standard benchmarks such as MTEB, demonstrating enhanced performance in semantic representation tasks (Muennighoff et al. 2023). Additionally, embedding models increasingly incorporate distillation from or joint-training with large generative models, effectively blurring distinctions between encoder-based and generative paradigms.

Reranker models, originally developed for information

retrieval tasks, represent another promising approach for ZSC. Early reranker architectures leveraged cross-encoder models like BERT (Devlin et al. 2019), DPR’s combined bi-encoder and cross-encoder architecture (Karpukhin et al. 2020), and late-interaction models such as ColBERT (Khattab and Zaharia 2020). These methods typically assign relevance scores to a set of candidate documents with respect to a given input query, enabling them to be ranked accordingly. Sequence-to-sequence reranker variants such as MonoT5 have further extended this paradigm by scoring pairs through generative token likelihood estimation, demonstrating effective transferability to new tasks (Nogueira, Jiang, and Lin 2020). Recent embedding model families like BGE now provide integrated reranker checkpoints, inheriting their multi-stage training procedures (Chen et al. 2024).

Benchmark for Textual Zero-Shot Classification (BTZSC)

BTZSC presents a comprehensive, task-balanced evaluation suite for zero-shot text classification, aiming to serve as a benchmark for diverse model architectures. The datasets underpin five key criteria to ensure robustness and real-world relevance. First, ensuring task diversity by including at least two datasets for each of sentiment, topic, intent, and emotion classification, mirroring the four most prominent application families. Second, to probe the impact of class granularity, BTZSC covers binary, medium-sized (such as *ag-news* with four labels), and high-cardinality settings (for instance, *banking77* with 77 labels). Third, we prioritized domain diversity, drawing from sources spanning news, social media, product reviews, encyclopedic content, and political discourse to assess model robustness under domain shift. Fourth, we incorporated a wide spectrum of document lengths, from micro-texts (under 20 tokens) to longer articles (over 250 tokens). The benchmark is limited to English datasets; multilingual evaluation is left for future work. The datasets overlap to large extent with the datasets used by (Laurer et al. 2023) for transfer learning in zero-shot classification. Table A.1 in the technical appendix provides the source and further details for each dataset.

BTZSC comprises 22 English datasets encompassing the aforementioned task types. As summarized in Table 1, each dataset is characterized by its number of classes, average token length¹, and domain label (such as news, review, or social media). To quantify lexical overlap and domain similarity between datasets, we follow (Thakur et al. 2021) and compute weighted Jaccard similarity by measuring token distribution overlaps for each dataset pair. The resulting 22×22 similarity matrix, shown in Figure 1, highlights low overlap between different task types, reflecting strong lexical diversity across tasks. At the same time, we observe that datasets derived from similar sources tend to cluster more together, for example, all Wikipedia-based datasets form a distinct group, as do the biasframes-related datasets, demonstrating modest intra-source lexical similarity.

¹computed with the `answerdotai/ModernBERT` tokenizer

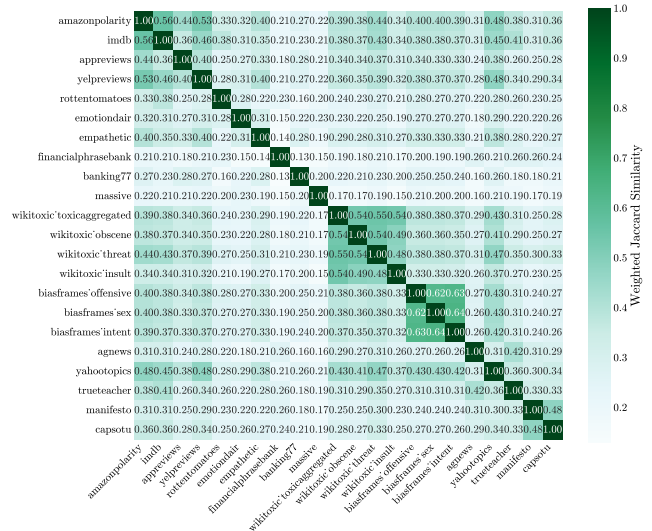


Figure 1: Pairwise weighted Jaccard similarity between datasets.

Evaluation Metrics

To make results comparable across all BTZSC tasks and model families we adopt a *single, task-agnostic primary metric*: **macro F₁**. Macro averaging gives equal weight to every class irrespective of its frequency, making it appropriate for both binary and multi-class datasets with varying label set cardinalities (Sokolova and Lapalme 2009). We additionally report (micro) **accuracy**, since it remains the most common headline number in the classification literature and is straightforward to interpret.

Finally, to probe whether success on natural-language inference transfers to zero-shot classification, we evaluate each model on standard NLI benchmarks and report the **AU-ROC**. AUROC is threshold-free and does not require calibrated probabilities; because cosine-similarity scores lie in $[-1, 1]$ rather than representing probabilities, AUROC lets us test whether entailment pairs consistently receive higher similarity than neutral/contradiction pairs.

Model Types

We categorize the models evaluated in this study according to their underlying architecture and training strategies.

Transformer Base Models. As a baseline, we include transformer-based encoder models that have not been further fine-tuned for any specific downstream task. For these models, the final *[CLS]* token representation is extracted and cosine similarity is used to compute the relevance between the input text and each candidate label. The base models considered in this category are the original BERT (*bert-large-uncased* (Devlin et al. 2019)), the increasingly adopted ModernBERT (*ModernBERT-large* (Warner et al. 2024)), and DeBERTa-v3 (*deberta-v3-large* (He, Gao, and Chen 2023)), a popular and robust modification of BERT that has demonstrated strong performance on a variety of NLP benchmarks.

Domain	Dataset	Num Classes	Avg Token Count
Emotion			
dialogue	empathetic	32	132
social-media	emotiondair	6	20
Intent			
banking	banking77	72	13
social-media	biasframes_intent	2	27
Sentiment			
apps	appreviews	2	49
e-commerce	amazonpolarity	2	103
finance	financialphrasebank	3	29
local-business	yelpreviews	2	164
movies	imdb	2	293
movies	rottentomatoes	2	26
Topic			
assistant	massive	59	8
education	trueteacher	2	282
news	agnews	4	54
politics	capsotu	21	44
politics	manifesto	56	45
qa-forum	yahootopics	10	137
social-media	biasframes_offensive	2	27
social-media	biasframes_sex	2	28
wikipedia	wikitoxic_insult	2	93
wikipedia	wikitoxic_obscene	2	91
wikipedia	wikitoxic_threat	2	99
wikipedia	wikitoxic_toxicaggregated	2	86

Table 1: Summary statistics of BTZSC datasets.

NLI-based Cross-Encoders. These models are trained on NLI datasets and perform classification by assessing the degree of entailment between an input text and each candidate label, formulated as a premise–hypothesis pair. *BART-Large-MNLI* is included as the canonical representative, being the first widely used NLI-based cross-encoder for zero-shot classification. We also consider *NLI-RoBERTa-base* as well as a set of custom-trained cross-encoders using *BERT*, *DeBERTa-v3*, and *ModernBERT* backbones. Both base and large versions are evaluated to analyze the effect of model scale, and two loss variants are tested to assess the impact of training objectives. Full details of the training procedure are provided in the technical appendix. In total, 11 NLI-based cross-encoders are benchmarked, covering the most widely used configurations in the literature.

Embedding Models. This category comprises models optimized to produce fixed-size vector representations of text for a range of downstream tasks, including classification. As a canonical embedding model, *all-MiniLM-L6-v2* (Reimers and Gurevych 2019) is serves as a baseline for this model family. Additionally, we evaluate both base and large variants of BGE, GTE, and E5, all of which use variations of transformer encoders as backbones. To provide contrast, we also include embedding models that leverage large language model architectures, such as Qwen3-Embedding and e5-mistral-7b-instruct; for Qwen3-Embedding, both 0.6B and

8B parameter variants are tested to study the effect of scale. Overall, the embedding model category comprises 11 distinct models.

Rerankers. Reranker models are typically employed in information retrieval, where they re-score candidate documents for relevance to a given query. The *ms-marco-MiniLM-L6-v2* model serves as the reranker counterpart to *all-MiniLM-L6-v2* and is used as the baseline for this group. Similarly, *gte-reranker-modernbert-base* and *bge-reranker-base/large* serve as reranking counterparts to their respective embedding models. We further include *Qwen3-Reranker*, which outputs a relevance score between a document and a query by prompting the model to decide if the document is relevant. The probability assigned to the "yes" token (computed from the model's vocabulary distribution using a softmax, with all other tokens masked out, except for "yes" and "no") is used as the final relevance score. Both the 0.6B and 8B variants of *Qwen3-Reranker* are evaluated to analyze the impact of model size. In total, 6 reranker models are benchmarked.

Table A.2 in the technical appendix summarizes the models included in the experiments, listing their architecture, training data, and parameter count. In total, the benchmark covers 31 models.

Experimental Setup

For our custom NLI-based cross-encoders, we follow the methodology of Laurer et al. (2023) and train models on a mixture of MNLI (Williams, Nangia, and Bowman 2018), ANLI (Nie et al. 2020), WANLI (Liu et al. 2022), FEVER-NLI (Thorne et al. 2018), and LingNLI (Parrish et al. 2021), datasets, deliberately omitting SNLI due to concerns regarding data quality and label bias. Appendix A.3 in the technical appendix provides further details on the training procedure.

To facilitate zero-shot classification, each class label is verbalized as a short, semantically clear, and context-rich description. For example, in the Amazon Polarity dataset, the positive class is verbalized as “The overall sentiment within the Amazon product review is {label},” where “label” is substituted with either “positive” or “negative” depending on the ground truth.

For reranker models, the text to be classified serves as the query, while the verbalized label descriptions are treated as candidate “documents” to be reranked according to their predicted relevance.

For nli-based cross-encoders we take the entailment logit and attribute the label with the highest logit as the predicted label. For embedding models, we compute the cosine similarity between the text embedding and each label embedding, selecting the label with the highest similarity score as the predicted label. For rerankers, we use the relevance score assigned to each label description to determine the predicted label.

Results and Analysis

In this section, we present and analyze the performance of all evaluated models on the BTZSC benchmark. Table 2 summarizes results across all datasets, grouped by task type, and reports (macro) F1 scores averaged within each task as well as overall, in addition to average (micro) accuracy. Standard deviations are included in parentheses to reflect variability across datasets.

Base Transformer Encoders. Models that are not further fine-tuned or trained on specific semantic matching objectives perform poorly on zero-shot classification tasks. Their inability to align input texts with candidate label descriptions underscores the necessity of explicit training for semantic compatibility.

NLI-based Cross-Encoders. Models fine-tuned on NLI data exhibit clear benefits over their off-the-shelf counterparts. Training on a diverse set of NLI datasets, including MNLI, ANLI, WANLI, FEVERNLI, and LINGNLI, yields consistently stronger performance compared to models such as *bart-large-mnli* and *nli-roberta-base*, with multi-dataset models achieving an average improvement of +6 F1 points across all tasks. Scaling model size further enhances performance: large variants outperform their base counterparts by an average of +3.5 F1 points. Figure 2 highlights this difference on a more granular level. Task difficulty remains a dominant factor: sentiment classification is relatively easy (median F1 \approx 0.88–0.9), topic and intent classification are of intermediate difficulty (F1 \approx 0.4–0.55), and emotion detection proves most challenging (F1 \approx 0.25–0.35). Larger

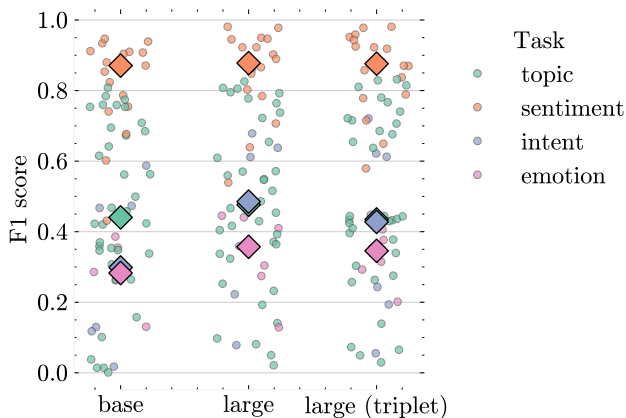


Figure 2: Performance of NLI-based cross-encoders on BTZSC. Points are individual datasets; diamonds mark task-wise medians. Comparison against model size (base vs. large) and loss type (binary vs. three-way cross-entropy).

models deliver the greatest benefit for more difficult tasks, with performance gains especially pronounced in topic and intent classification. The choice of loss function, whether binary cross-entropy with neutral-collapsed or standard three-way cross-entropy, has minimal impact; the three-way cross-entropy variant closely tracks the regular large model with no consistent additional gain. Notably, within this family, *deberta-v3-large-nli-triplet* achieves the highest overall performance, surpassing both the original BERT and ModernBERT variants, corroborating findings from Warner et al. (2024), that *deberta-v3* is still a challenging baseline for various NLP tasks.

Reranker Models. Among rerankers, the baseline *ms-marco-MiniLM-L6-v2* does not match the performance of NLI cross-encoders (average F1: 0.42), consistent with the historical view that NLI fine-tuning is advantageous for zero-shot tasks. However, more recent rerankers close the gap substantially. For example, *gte-reranker-modernbert-base* achieves an average F1 of 0.58, just two points below the best NLI cross-encoder (*deberta-v3-large-nli-triplet*), and with lower variance. The strongest reranker, *Qwen3-Reranker-8B*, achieves an average F1 of 0.72 and outperforms all other models, including NLI cross-encoders, by significant margins (+12 F1 and +15 accuracy points). This model is the top overall performer on the benchmark, ranking first in three out of four task categories and second in emotion classification. It should be noted, however, that its size (8B parameters) far exceeds that of NLI cross-encoders (typically around 300M parameters). Importantly, even the much smaller *Qwen3-Reranker-0.6B* delivers competitive results, surpassing NLI cross-encoders in accuracy and achieving the second-highest overall score (0.64), underscoring the strength of the reranker approach even at moderate scale.

Embedding Models. The canonical embedding baseline, *all-MiniLM-L6-v2*, attains an average F1 of 0.37, supporting prior observations that rerankers generally outperform embedding models in retrieval, albeit at higher computational

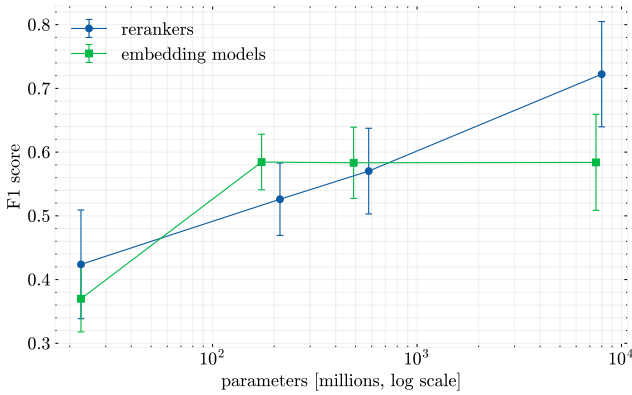


Figure 3: Effect of scale on zero-shot performance. Macro- F_1 (BTZSC) versus parameter count (log scale). Error bands show 95% confidence intervals.

cost. However, newer embedding models such as *e5-large-v2*, *gte-modernbert-base*, and *gte-large-en-v1.5* achieve substantially higher F1 scores (0.60, 0.58, and 0.62, respectively), placing them on par with or even surpassing the best NLI cross-encoders. Notably, these embedding models lack cross-attention between documents and labels yet still deliver strong results at similar model sizes. For instance, *gte-large-en-v1.5* ranks as the second-best model overall, but its performance still lags behind the top-ranked *Qwen3-Reranker-8B* by about 10 F1 points. Scaling up embedding models does not yield the same improvements seen in rerankers; for example, *Qwen3-Embedding-8B* only slightly outperforms its 0.6B variant (F1: 0.59 vs 0.58).

Figure 3 further elucidates scaling trends. Reranker models benefit substantially from larger scales, surpassing F1 of 0.70 at the highest parameter count (8B), while embedding models plateau at approximately 0.60 beyond 300M parameters. In the 100–300M range, performance between families is similar and variance is high, but at larger scales, rerankers gain a decisive advantage. Thus, rerankers are preferable when computational resources permit, whereas embedding models remain attractive for lightweight or latency-sensitive applications. Figure 4 plots model F1 score against normalized inference speed (1/wall time) on a standard test set. The upper right quadrant, bounded by the medians of both metrics, highlights models that best balance accuracy and efficiency. The majority of the models in this region are embedding models, indicating they offer the most favorable trade-off between performance and latency for practical deployments, with *gte-reranker-modernbert-base* as the only reranker achieving comparable efficiency.

NLI Performance as a Proxy for Zero-Shot Classification

We also examine whether NLI task performance predicts zero-shot classification effectiveness. As shown in Figure 5, for NLI-tuned cross-encoders, there is a strong linear relationship: improvements in NLI AUROC directly translate into higher F1 on BTZSC, reflecting the transfer of entail-

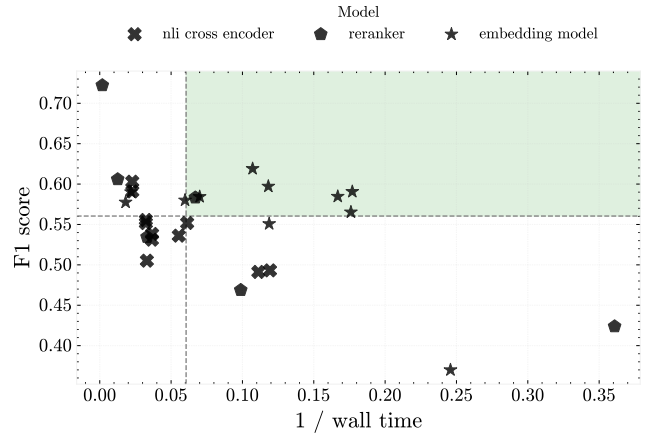


Figure 4: Trade-off between model performance and inference speed. Macro- F_1 score (BTZSC) is plotted against normalized inference throughput (1/wall time) on a standard test set. The upper right quadrant, defined by the medians of both metrics, highlights models with the best balance of accuracy and efficiency.

ment supervision. Rerankers, despite not being fine-tuned on NLI, also show a positive trend, indicating that a robust relevance or semantic-matching mechanism supports zero-shot classification. Notably, some rerankers achieve strong classification despite moderate NLI performance, highlighting their ability to capture discriminative task signals not present in standard NLI benchmarks. Embedding models, on the other hand, show tightly clustered NLI scores but a wider spread in classification F1, suggesting that well-structured embedding spaces can capture fine-grained topical distinctions that traditional NLI metrics may miss.

Conclusion and Future Work

This work presents the first comprehensive evaluation of zero-shot text classification across NLI cross-encoders, embedding models, and rerankers, leveraging the BTZSC benchmark to examine the strengths and limitations of each paradigm. Our findings demonstrate that reranker models achieve the highest overall accuracy, while strong embedding models offer the most favorable balance between speed and accuracy. In contrast, NLI cross-encoders, trail behind in both accuracy and efficiency. Scaling analyses reveal that rerankers uniquely benefit from larger model sizes, with significant gains observed beyond one billion parameters, whereas embedding models tend to plateau, suggesting different optimization ceilings across model families. Looking forward, several directions remain open for the community. Extending BTZSC to multilingual domains would facilitate evaluation under broader distribution shifts. Further investigation into the interplay between label verbalization strategies and model architectures could yield insights into improving zero-shot classification. Finally, exploring the performance and limitations of models that exceed the 8B parameter scale may provide a deeper understanding of scaling dynamics in this setting.

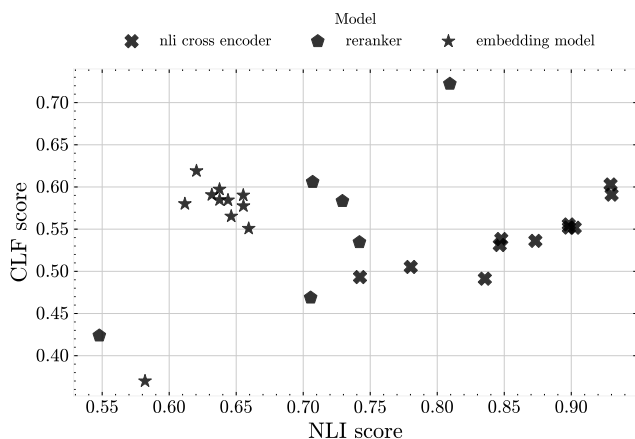


Figure 5: Relationship between NLI ability and zero-shot classification. Each dot is one model; x-axis shows AUROC on standard NLI benchmarks, y-axis macro- F_1 on BTZSC.

References

- Bowman, S. R.; Angeli, G.; Potts, C.; and Manning, C. D. 2015. A Large Annotated Corpus for Learning Natural Language Inference. In *Proc. EMNLP*, 632–642.
- Brown, T. B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; and *et al.* 2020. Language Models are Few-Shot Learners. *arXiv preprint arXiv:2005.14165*.
- Cer, D.; Yang, Y.; yi Kong, S.; Hua, N.; Limtiaco, N.; et al. 2018. Universal Sentence Encoder. In *Proceedings of EMNLP*.
- Chalkidis, I.; Fergadiotis, M.; Kotitsas, S.; Malakasiotis, P.; Aletras, N.; and Androustopoulos, I. 2020. An Empirical Study on Large-Scale Multi-Label Text Classification Including Few and Zero-Shot Labels. In *Proceedings of EMNLP*.
- Chen, J.; Xiao, S.; Zhang, P.; Luo, K.; Lian, D.; and Liu, Z. 2024. BGE M3-Embedding: Multi-Lingual, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation. *arXiv preprint arXiv:2402.03216*.
- Chung, I.; Kerboua, I.; Kardos, M.; Solomatin, R.; and Enevoldsen, K. 2025. Maintaining MTEB: Towards Long Term Usability and Reproducibility of Embedding Benchmarks. *arXiv preprint arXiv:2506.21182*.
- Conneau, A.; Kiela, D.; Schwenk, H.; Barrault, L.; and Bordes, A. 2017. Supervised Learning of Universal Sentence Representations from Natural Language Inference Data. In *Proceedings of EMNLP*.
- Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of NAACL-HLT*.
- Enevoldsen, K.; Chung, I.; Kerboua, I.; Kardos, M.; Mathur, A.; et al. 2025. MMTEB: Massive Multilingual Text Embedding Benchmark. *arXiv preprint arXiv:2502.13595*.
- Gao, T.; Yao, X.; and Chen, D. 2021. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In *Proc. EMNLP*, 6894–6910.
- Gretz, S.; Halfon, A.; Shnayderman, I.; Toledo-Ronen, O.; Spector, A.; Dankin, L.; Katsis, Y.; Arviv, O.; Katz, Y.; Slonim, N.; and Ein-Dor, L. 2023. Zero-shot Topical Text Classification with LLMs – an Experimental Study. In *Findings of EMNLP*, 9647–9676.
- He, P.; Gao, J.; and Chen, W. 2023. DeBERTaV3: Improving DeBERTa using ELECTRA-Style Pre-Training with Gradient-Disentangled Embedding Sharing. In *Proc. International Conference on Learning Representations (ICLR)*. ArXiv:2111.09543.
- Karpukhin, V.; Oğuz, B.; Min, S.; Lewis, P.; Wu, L.; Edunov, S.; Chen, D.; and tau Yih, W. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of EMNLP*.
- Khattab, O.; and Zaharia, M. 2020. ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT. In *Proceedings of SIGIR*.
- Kowsari, K.; Meimandi, K. J.; Heidarysafa, M.; Mendu, S.; Barnes, L. E.; and Brown, D. E. 2019. Text Classification Algorithms: A Survey. *Information*, 10(4): 150.
- Laurer, M.; van Atteveldt, W.; Casas, A.; and Welbers, K. 2023. Building Efficient Universal Classifiers with Natural Language Inference. ArXiv:2312.17543 [cs].
- Lepagnol, P.; Gerald, T.; Ghannay, S.; Servan, C.; and Rosset, S. 2024. Small Language Models are Good Too: An Empirical Study of Zero-Shot Classification. In *Proceedings of LREC-COLING*.
- Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; and Zettlemoyer, L. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proceedings of ACL*.
- Li, Z.; Zhang, X.; Zhang, Y.; Long, D.; Xie, P.; and Zhang, M. 2023. Towards General Text Embeddings with Multi-stage Contrastive Learning. *arXiv preprint arXiv:2308.03281*.
- Liu, A.; Swayamdipta, S.; Smith, N. A.; and Choi, Y. 2022. WANLI: Worker and AI Collaboration for Natural Language Inference Dataset Creation. In *Findings of EMNLP*, 7080–7097.
- Muennighoff, N.; Tazi, N.; Magne, L.; and Reimers, N. 2023. MTEB: Massive Text Embedding Benchmark. In *Proc. EACL*, 2014–2037.
- Ni, J.; Ábrego, G. H.; Constant, N.; Ma, J.; Hall, K. B.; Cer, D.; and Yang, Y. 2021. Sentence-T5: Scalable Sentence Encoders from Pre-trained Text-to-Text Models. *arXiv preprint arXiv:2108.08877*.
- Nie, Y.; Williams, A.; Dinan, E.; Bansal, M.; Weston, J.; and Kiela, D. 2020. Adversarial NLI: A New Benchmark for Natural Language Understanding. In *Proceedings of ACL*, 4885–4901.
- Nogueira, R.; Jiang, Z.; and Lin, J. 2020. Document Ranking with a Pretrained Sequence-to-Sequence Model. *arXiv preprint arXiv:2003.06713*.

Parrish, A.; Huang, W.; Agha, O.; Lee, S.-H.; Nangia, N.; Warstadt, A.; Aggarwal, K.; Allaway, E.; Linzen, T.; and Bowman, S. R. 2021. Does Putting a Linguist in the Loop Improve NLU Data Collection? In *Findings of EMNLP*, 4886–4901.

Radford, A.; Kim, J.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; Krueger, G.; and Sutskever, I. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*. PMLR.

Reimers, N.; and Gurevych, I. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proc. EMNLP*.

Sebastiani, F. 2002. Machine Learning in Automated Text Categorization. *ACM Computing Surveys*, 34(1): 1–47.

Settles, B. 2012. Active Learning Literature Survey. Technical Report 1648, University of Wisconsin–Madison.

Sokolova, M.; and Lapalme, G. 2009. A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4): 427–437.

Thakur, N.; Reimers, N.; Rücklé, A.; Srivastava, A.; and Gurevych, I. 2021. BEIR: A Heterogenous Benchmark for Zero-shot Evaluation of Information Retrieval Models. arXiv:2104.08663.

Thorne, J.; Vlachos, A.; Christodoulopoulos, C.; and Mittal, A. 2018. FEVER: A Large-scale Dataset for Fact Extraction and VERification. In *Proceedings of NAACL–HLT*, 809–819.

Wang, L.; Yang, N.; Huang, X.; Yang, L.; Majumder, R.; and Wei, F. 2024. Multilingual E5 Text Embeddings: A Technical Report. *arXiv preprint arXiv:2402.05672*.

Warner, B.; Chaffin, A.; Clavié, B.; Weller, O.; Hallström, O.; Taghadouini, S.; Gallagher, A.; Biswas, R.; Ladhak, F.; Aarsen, T.; Cooper, N.; Adams, G.; Howard, J.; and Poli, I. 2024. Smarter, Better, Faster, Longer: A Modern Bidirectional Encoder for Fast, Memory Efficient, and Long Context Finetuning and Inference. arXiv:2412.13663.

Williams, A.; Nangia, N.; and Bowman, S. 2018. A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference. In *Proc. NAACL–HLT*, 1112–1122.

Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; Davison, J.; Shleifer, S.; von Platen, P.; Ma, C.; Jernite, Y.; Plu, J.; Xu, C.; Scao, T. L.; Gugger, S.; Drame, M.; Lhoest, Q.; and Rush, A. M. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 38–45. Online: Association for Computational Linguistics.

Yin, W.; Hay, J.; and Roth, D. 2019. Benchmarking Zero-shot Text Classification: Datasets, Evaluation and Entailment Approach. In *Proc. EMNLP–IJCNLP*, 3914–3923.

Zhang, Y.; Li, M.; Long, D.; Zhang, X.; Lin, H.; et al. 2025. Qwen3 Embedding: Advancing Text Embedding and Reranking Through Foundation Models. *arXiv preprint arXiv:2506.05176*.

Model	Topic	Sentiment	Intent	Emotion	Avg F1	Avg Acc
Base encoders						
<u>bert-large-uncased</u>	0.30 (0.22)	0.38 (0.07)	0.22 (0.29)	0.09 (0.12)	0.30 (0.20)	0.40 (0.26)
deberta-v3-large	0.28 (0.24)	0.34 (0.02)	0.23 (0.31)	0.05 (0.07)	0.27 (0.20)	0.36 (0.26)
ModernBERT-large	0.28 (0.24)	0.36 (0.05)	0.20 (0.24)	0.03 (0.03)	0.27 (0.20)	0.35 (0.24)
NLI cross-encoders						
bart-large-mnli	0.36 (0.21)	0.84 (0.19)	0.47 (0.23)	0.40 (0.06)	0.51 (0.28)	0.53 (0.28)
nli-roberta-base	0.40 (0.24)	0.79 (0.15)	0.31 (0.31)	0.32 (0.03)	0.49 (0.28)	0.51 (0.27)
bert-base-uncased-nli	0.43 (0.26)	0.76 (0.17)	0.30 (0.40)	0.24 (0.16)	0.49 (0.29)	0.51 (0.28)
bert-large-uncased-nli	0.49 (0.26)	0.79 (0.10)	0.35 (0.38)	0.28 (0.22)	0.54 (0.27)	0.58 (0.27)
bert-large-uncased-nli-triplet	0.49 (0.25)	0.78 (0.12)	0.34 (0.40)	0.25 (0.06)	0.53 (0.27)	0.56 (0.26)
deberta-v3-base-nli	0.48 (0.25)	0.86 (0.10)	0.30 (0.24)	0.33 (0.08)	0.55 (0.28)	0.58 (0.26)
deberta-v3-large-nli	0.47 (0.25)	<u>0.90 (0.07)</u>	0.52 (0.23)	0.43 (0.03)	0.59 (0.27)	0.62 (0.26)
<u>deberta-v3-large-nli-triplet</u>	0.50 (0.26)	<u>0.90 (0.07)</u>	0.48 (0.34)	0.43 (0.03)	0.60 (0.28)	0.62 (0.26)
modernbert-base-nli	0.47 (0.26)	0.83 (0.14)	0.29 (0.25)	0.27 (0.02)	0.54 (0.29)	0.56 (0.29)
modernbert-large-nli	0.47 (0.24)	0.86 (0.16)	0.43 (0.29)	0.29 (0.02)	0.56 (0.28)	0.60 (0.27)
modernbert-large-nli-triplet	0.45 (0.24)	0.88 (0.12)	0.40 (0.30)	0.35 (0.04)	0.55 (0.29)	0.58 (0.27)
Rerankers						
ms-marco-MiniLM-L6-v2	0.38 (0.16)	0.59 (0.16)	0.42 (0.27)	0.19 (0.01)	0.42 (0.19)	0.46 (0.21)
gte-reranker-modernbert-base	0.51 (0.13)	0.82 (0.17)	0.49 (0.22)	0.42 (0.07)	0.58 (0.20)	0.62 (0.19)
bge-reranker-base	0.42 (0.13)	0.61 (0.15)	0.49 (0.01)	0.30 (0.02)	0.47 (0.16)	0.49 (0.14)
bge-reranker-large	0.44 (0.17)	0.77 (0.15)	0.58 (0.01)	0.36 (0.05)	0.53 (0.21)	0.55 (0.20)
Qwen3-Reranker-0.6B	<u>0.54 (0.23)</u>	0.80 (0.20)	0.56 (0.11)	0.46 (0.07)	0.61 (0.23)	0.64 (0.21)
<u>Qwen3-Reranker-8B</u>	0.66 (0.17)	0.93 (0.06)	0.72 (0.04)	<u>0.49 (0.01)</u>	0.72 (0.19)	0.77 (0.15)
Embedding models						
all-MiniLM-L6-v2	0.41 (0.11)	0.35 (0.04)	0.45 (0.03)	0.13 (0.02)	0.37 (0.12)	0.44 (0.14)
e5-base-v2	0.50 (0.18)	0.83 (0.19)	0.61 (0.00)	0.40 (0.05)	0.59 (0.23)	0.62 (0.21)
e5-large-v2	0.50 (0.16)	0.86 (0.17)	0.57 (0.00)	0.41 (0.05)	0.60 (0.22)	0.62 (0.20)
e5-mistral-7b-instruct	0.43 (0.22)	0.88 (0.13)	<u>0.66 (0.03)</u>	0.50 (0.00)	0.58 (0.26)	0.62 (0.24)
bge-base-en-v1.5	0.46 (0.19)	0.82 (0.20)	0.62 (0.03)	0.35 (0.09)	0.57 (0.24)	0.59 (0.23)
bge-large-en-v1.5	0.42 (0.19)	0.84 (0.19)	0.61 (0.08)	0.40 (0.06)	0.55 (0.25)	0.59 (0.24)
gte-base-en-v1.5	0.49 (0.21)	0.83 (0.18)	0.64 (0.03)	0.38 (0.07)	0.58 (0.24)	0.61 (0.22)
<u>gte-large-en-v1.5</u>	<u>0.54 (0.20)</u>	0.85 (0.18)	0.62 (0.04)	0.38 (0.03)	<u>0.62 (0.23)</u>	<u>0.64 (0.21)</u>
gte-modernbert-base	0.46 (0.20)	0.87 (0.12)	0.64 (0.01)	0.42 (0.04)	0.58 (0.24)	0.61 (0.23)
Qwen3-Embedding-0.6B	0.49 (0.13)	0.81 (0.17)	0.55 (0.15)	0.42 (0.09)	0.58 (0.20)	0.61 (0.18)
<u>Qwen3-Embedding-8B</u>	0.46 (0.16)	<u>0.90 (0.09)</u>	0.55 (0.24)	<u>0.49 (0.08)</u>	0.59 (0.24)	<u>0.64 (0.20)</u>

Table 2: Zero-shot classification results on BTZSC. We report macro-averaged F1 per task family and overall (Avg F1) and micro accuracy (Avg Acc). Standard deviations across datasets are in parentheses. Bold denotes the best and underlining the second-best score in each column. Best model in each family is underlined.

Reproducibility Checklist

Instructions for Authors:

This document outlines key aspects for assessing reproducibility. Please provide your input by editing this .tex file directly.

For each question (that applies), replace the “Type your response here” text with your answer.

Example: If a question appears as

```
\question{Proofs of all novel claims  
are included} {(yes/partial/no)}  
Type your response here
```

you would change it to:

```
\question{Proofs of all novel claims  
are included} {(yes/partial/no)}  
yes
```

Please make sure to:

- Replace **ONLY** the “Type your response here” text and nothing else.
- Use one of the options listed for that question (e.g., **yes**, **no**, **partial**, or **NA**).
- **Not** modify any other part of the `\question` command or any other lines in this document.

You can `\input` this .tex file right before `\end{document}` of your main file or compile it as a stand-alone document. Check the instructions on your conference’s website to see if you will be asked to provide this checklist with your paper or separately.

1. General Paper Structure

- 1.1. Includes a conceptual outline and/or pseudocode description of AI methods introduced (yes/partial/no/NA) [NA](#)
- 1.2. Clearly delineates statements that are opinions, hypothesis, and speculation from objective facts and results (yes/no) [yes](#)
- 1.3. Provides well-marked pedagogical references for less-familiar readers to gain background necessary to replicate the paper (yes/no) [yes](#)

2. Theoretical Contributions

- 2.1. Does this paper make theoretical contributions? (yes/no) [no](#)

If yes, please address the following points:

- 2.2. All assumptions and restrictions are stated clearly and formally (yes/partial/no) [Type your response here](#)

- 2.3. All novel claims are stated formally (e.g., in theorem statements) (yes/partial/no) [Type your response here](#)
- 2.4. Proofs of all novel claims are included (yes/partial/no) [Type your response here](#)
- 2.5. Proof sketches or intuitions are given for complex and/or novel results (yes/partial/no) [Type your response here](#)
- 2.6. Appropriate citations to theoretical tools used are given (yes/partial/no) [Type your response here](#)
- 2.7. All theoretical claims are demonstrated empirically to hold (yes/partial/no/NA) [Type your response here](#)
- 2.8. All experimental code used to eliminate or disprove claims is included (yes/no/NA) [Type your response here](#)

3. Dataset Usage

- 3.1. Does this paper rely on one or more datasets? (yes/no) [yes](#)

If yes, please address the following points:

- 3.2. A motivation is given for why the experiments are conducted on the selected datasets (yes/partial/no/NA) [yes](#)
- 3.3. All novel datasets introduced in this paper are included in a data appendix (yes/partial/no/NA) [yes](#)
- 3.4. All novel datasets introduced in this paper will be made publicly available upon publication of the paper with a license that allows free usage for research purposes (yes/partial/no/NA) [yes](#)
- 3.5. All datasets drawn from the existing literature (potentially including authors’ own previously published work) are accompanied by appropriate citations (yes/no/NA) [yes](#)
- 3.6. All datasets drawn from the existing literature (potentially including authors’ own previously published work) are publicly available (yes/partial/no/NA) [yes](#)
- 3.7. All datasets that are not publicly available are described in detail, with explanation why publicly available alternatives are not scientifically satisfying (yes/partial/no/NA) [NA](#)

4. Computational Experiments

- 4.1. Does this paper include computational experiments? (yes/no) [yes](#)

If yes, please address the following points:

- 4.2. This paper states the number and range of values tried per (hyper-) parameter during development of

the paper, along with the criterion used for selecting the final parameter setting (yes/partial/no/NA) [NA](#)

- 4.3. Any code required for pre-processing data is included in the appendix (yes/partial/no) [yes](#)
- 4.4. All source code required for conducting and analyzing the experiments is included in a code appendix (yes/partial/no) [yes](#)
- 4.5. All source code required for conducting and analyzing the experiments will be made publicly available upon publication of the paper with a license that allows free usage for research purposes (yes/partial/no) [yes](#)
- 4.6. All source code implementing new methods have comments detailing the implementation, with references to the paper where each step comes from (yes/partial/no) [yes](#)
- 4.7. If an algorithm depends on randomness, then the method used for setting seeds is described in a way sufficient to allow replication of results (yes/partial/no/NA) [NA](#)
- 4.8. This paper specifies the computing infrastructure used for running experiments (hardware and software), including GPU/CPU models; amount of memory; operating system; names and versions of relevant software libraries and frameworks (yes/partial/no) [yes](#)
- 4.9. This paper formally describes evaluation metrics used and explains the motivation for choosing these metrics (yes/partial/no) [yes](#)
- 4.10. This paper states the number of algorithm runs used to compute each reported result (yes/no) [yes](#)
- 4.11. Analysis of experiments goes beyond single-dimensional summaries of performance (e.g., average; median) to include measures of variation, confidence, or other distributional information (yes/no) [yes](#)
- 4.12. The significance of any improvement or decrease in performance is judged using appropriate statistical tests (e.g., Wilcoxon signed-rank) (yes/partial/no) [no](#)
- 4.13. This paper lists all final (hyper-)parameters used for each model/algorithm in the paper's experiments (yes/partial/no/NA) [NA](#)