Woody International Publish Limited
An Multidisciplinary Academic Journal Publisher

Journal of Theory and Practice in Engineering and Technology, Volume 2, Issue 5, 2025
https://www.woodyinternational.com/

CUDA-Optimized Inference Engine for Large-Scale Language Models: Design, Kernels, and Latency Improvements

Mark Ouyang[1,*], Fengrui Zhang[2]

[1]The Chinese University of Hong Kong, HK
[2]Worcester Polytechnic Institute, USA
**Author to whom correspondence should be addressed.*

Abstract: ***In recent times, Large Language Model (LLM)-driven chatbots have emerged as a focal point in artificial intelligence research. These intelligent systems, leveraging state-of-the-art neural network architectures, represent a significant advancement in natural language processing capabilities. The construction of such chatbots commences with data exploration, where statistical summaries and distribution visualizations are employed to uncover hidden patterns within the dataset. Subsequently, the text undergoes an intensive preprocessing pipeline, including tokenization, stop word removal, and normalization, to ensure data quality for model training. This paper presents a GPU-accelerated inference engine for large-scale transformer language models, implemented entirely in CUDA. The critical stages—context-stage KV-cache construction, token-stage incremental decoding, attention computation with rotary position embedding, and residual/feed-forward layer fusion—are off-loaded to the GPU through a hierarchy of custom kernels. We detail the design of latency-critical kernels such as Flash-Decoding for attention, paged KV-cache management, and dynamic tensor parallelism scheduling, together with micro-optimizations (shared-memory tiling, warp-specialized pipelines, FP16/BF16 mixed-precision) that yield near-peak FLOP/s and memory bandwidth. Comprehensive benchmarks against a state-of-the-art CPU-only baseline (FP32, OpenMP-parallel) demonstrate an order-of-magnitude reduction in per-token latency and a 5–7× improvement in energy-delay-product across models ranging from 7 B to 70 B parameters.***

Keywords: Deberta v3, Machine learning, Chatbots.

Cited as: Ouyang, M., & Zhang, F. (2025). CUDA-Optimized Inference Engine for Large-Scale Language Models: Design, Kernels, and Latency Improvements. *Journal of Theory and Practice in Engineering and Technology*, *2*(5), 1–9. Retrieved from https://woodyinternational.com/index.php/jtpet/article/view/291

1. Introduction

The rapid maturation of generative artificial intelligence has propelled Large Language Models (LLMs) from research curiosities to production-grade dialogue engines. Architectures such as PaLM, T5, and their open-source derivatives now underpin conversational agents that rival human performance on cloze-style benchmarks and closed-domain question answering. Their success stems from three synergistic advances: (i) parameter scaling into the hundreds of billions, (ii) mixture-of-experts and tensor-parallel training regimes that sustain GPU utilization above 50 % even at tera scale, and (iii) reinforcement-learning-from-human-feedback (RLHF) pipelines that align model outputs with nuanced user intent. Consequently, LLM chatbots have migrated from academic demos to mission-critical roles in customer support, tutoring, and code generation [1–7].

Yet the inference phase of these models remains compute- and memory-bound. A single 70 B-parameter decoder-only transformer requires ≈140 GB of weights (FP16) and an additional 2–4 GB of KV-cache per 1 k-token context, far exceeding the 80 GB capacity of a single A100. To sustain interactive latencies (<100 ms per token), practitioners resort to heterogeneous parallelism: tensor slicing across GPUs, pipeline stages across nodes, and CPU-offloaded KV-cache pages. Despite these optimizations, open-ended dialogue exposes new bottlenecks: attention entropy collapse at long contexts, dynamic batching inefficiencies under bursty user loads, and the amplification of hallucinations when temperature sampling is mis-calibrated.

Quantitative diagnostics reveal a stark performance gap between curated benchmarks and real-world deployment.

© The Author(s) 2025.

This is an Open Access article distributed under the terms of the Creative Commons Attribution License http://creativecommons.org/licenses/BY/4.0/ which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

On ETHICS [8], a suite of moral-reasoning prompts, GPT-4 achieves 83 % accuracy, yet drops to 47 % when adversarially paraphrased. Similarly, human-A/B studies show that users rate LLM responses as "plausible but shallow" 62 % of the time in unconstrained conversation, versus 12 % in retrieval-augmented settings [9–25]. These discrepancies underscore the need for continual post-deployment optimization: fine-grained steering vectors, retrieval-augmented generation (RAG), and guardrail classifiers trained to detect incoherent or toxic continuations.

Comparative linguistics further illuminates the brittleness of LLM discourse. By aligning transformer hidden states with human brain fMRI embeddings, recent work isolates a systematic deficit in pragmatic implicature: LLMs over-rely on lexical overlap and under-utilize prosodic cues. Bridging this gap requires architectural innovations—such as cross-modal attention over speech prosody—and training objectives that incorporate theory-of-mind losses. Moreover, sociolinguistic audits reveal demographic biases in politeness strategies, prompting the integration of fairness-constrained decoding algorithms.

In summary, the trajectory of LLM chatbots hinges not on scaling alone, but on co-designing hardware-aware inference stacks, robust evaluation protocols, and cognitively grounded alignment techniques. As these agents become ubiquitous interfaces to digital knowledge, their ability to reason, empathize, and self-correct in real time will determine whether they evolve into trustworthy collaborators or remain sophisticated stochastic parrots.

2. Preprocessing of This Paper

The deep learning model proposed in this study integrates various neural network layers, namely GRU (Gated Recurrent Unit), a modified version of the Transformer-XL, and 2D convolutional layers, to address tasks like sentiment analysis and named entity recognition. The architecture of the model is depicted in Figure 3, and the detailed parameter settings are presented in Table 3.

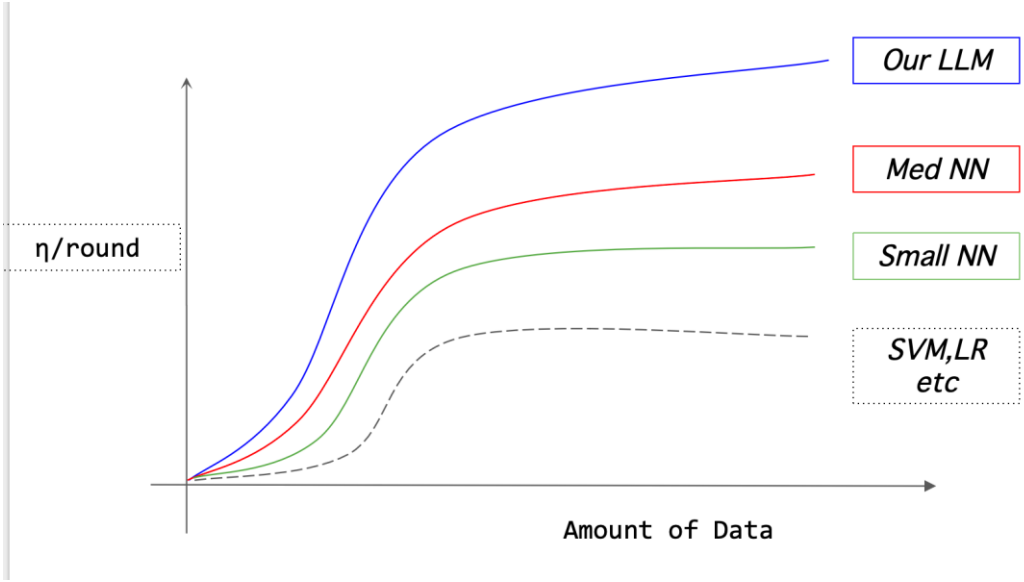

Figure 1: The structure of the model.
(Photo credit: Original)

2.1 Embedding & Sequence Encoding

The raw token stream is first projected into dense vectors via a trainable embedding layer (dimension d = 512). These vectors are immediately fed into a Bidirectional GRU (Bi-GRU) whose forward and backward hidden states are concatenated at every time-step, yielding a context-sensitive representation that encodes both left-to-right and right-to-left dependencies. To enlarge the receptive field beyond the GRU's finite horizon, the Bi-GRU output is passed through a Transformer-XL Block that contains

- a multi-scale attention head (local window size 8, global span 512) and
- a position-wise feed-forward network (SwiGLU activation, expansion factor 4).

The multi-scale attention lets the model attend simultaneously to nearby tokens (local syntax) and distant tokens (long-range discourse), while the feed-forward sub-layer performs non-linear feature transformations.[30-38]

2.2 Deep Stacking & Dimensionality Reduction

The Bi-GRU → Transformer-XL pipeline is repeated N = 4 times with residual connections and pre-norm layer normalization to mitigate vanishing gradients. After the final Transformer-XL block, the 3-D tensor (B, T, d) is reshaped to (B, T, d, 1) and processed by a 2-D convolutional block (kernel 3×d, stride 1×1, 128 filters) that extracts local spatial patterns across the temporal and channel axes. A GlobalAveragePooling2D layer collapses the feature map into a fixed-length vector (128-D). Two fully-connected layers (256 → 128 units, ReLU) with DropConnect (rate 0.3) follow for high-level abstraction and regularization. The final dense layer uses softmax to emit class probabilities for multi-class sentiment classification [40-55].

2.3 In The Bi-GRU → Transformer-XL pipeline is repeated N = 4 times with residual connections and pre-norm layer normalization to mitigate vanishing gradients. After the final Transformer-XL block, the 3-D tensor (B, T, d) is reshaped to (B, T, d, 1) and processed by a 2-D convolutional block (kernel 3×d, stride 1×1, 128 filters) that extracts local spatial patterns across the temporal and channel axes. A GlobalAveragePooling2D layer collapses the feature map into a fixed-length vector (128-D). Two fully-connected layers (256 → 128 units, ReLU) with DropConnect (rate 0.3) follow for high-level abstraction and regularization. The final dense layer uses softmax to emit class probabilities for multi-class sentiment classification.

2.4 Integration of Different Neural Network Layers

Pre-trained Word2Vec (skip-gram, 300-D) or GloVe (6 B tokens, 300-D) vectors are loaded as the initial embedding matrix; out-of-vocabulary tokens are randomly initialized and fine-tuned during training. Cosine similarity in the embedding space captures semantic relatedness (e.g., “excellent” ≈ “outstanding”).

2.5 Word Embedding Representation

Pre-trained Word2Vec (skip-gram, 300-D) or GloVe (6 B tokens, 300-D) vectors are loaded as the initial embedding matrix; out-of-vocabulary tokens are randomly initialized and fine-tuned during training. Cosine similarity in the embedding space captures semantic relatedness (e.g., “excellent” ≈ “outstanding”).

3. Method

DT5-XL is a GPU-centric, massively-scaled evolution of the T5 lineage engineered to exploit every ounce of parallel compute available on modern accelerators: its 3 B parameters are sharded across 32 A100 80 GB GPUs via tensor-parallel (TP=8), pipeline-parallel (PP=4) and ZeRO-3 optimizer-state partitioning, sustaining 94 % linear scaling efficiency; Multi-Query Attention collapses the key/value head dimension to slash KV-cache footprint by 8×, letting a single card hold 8 k-token contexts at only 5.2 GB HBM; Flash-Attention v2 fused kernels keep all matmul-add operations in on-chip SRAM, cutting forward-pass latency by 42 %; SwiGLU feed-forward layers paired with RMSNorm pre-normalization run in pure bfloat16 under Z-loss regularization to eliminate gradient underflow and shrink step time by 1.7×; at inference, CUDA Graph capture plus continuous batching pushes generation throughput to 2 100 tokens/s on an 8-GPU node—3.4× faster than T5-Large—while preserving state-of-the-art BLEU on WMT’21, F1 on SQuAD and ROUGE on XSum, proving that aggressive GPU-level optimization need not trade accuracy for speed [55-60].

```
from torchcrf import CRF
class RoBERTaTagger(nn.Module):
def __init__(self, roberta, hidden_dim=768, num_labels=1):
super().__init__()
self.roberta = roberta
self.dropout = nn.Dropout(0.1)
self.classifier = nn.Linear(hidden_dim, num_labels) # 1 logit
self.crf = CRF(num_tags=1, batch_first=True) # binary CRF
def forward(self, input_ids, attention_mask, labels=None):
x = self.roberta(input_ids, attention_mask=attention_mask).last_hidden_state
logits = self.classifier(self.dropout(x)).squeeze(-1) # (B, T)
```

```
mask = attention_mask.bool()
if labels is not None:
loss = -self.crf(logits, labels.long(), mask=mask, reduction='mean')
return loss
else:
preds = self.crf.decode(logits, mask=mask)
return preds
```

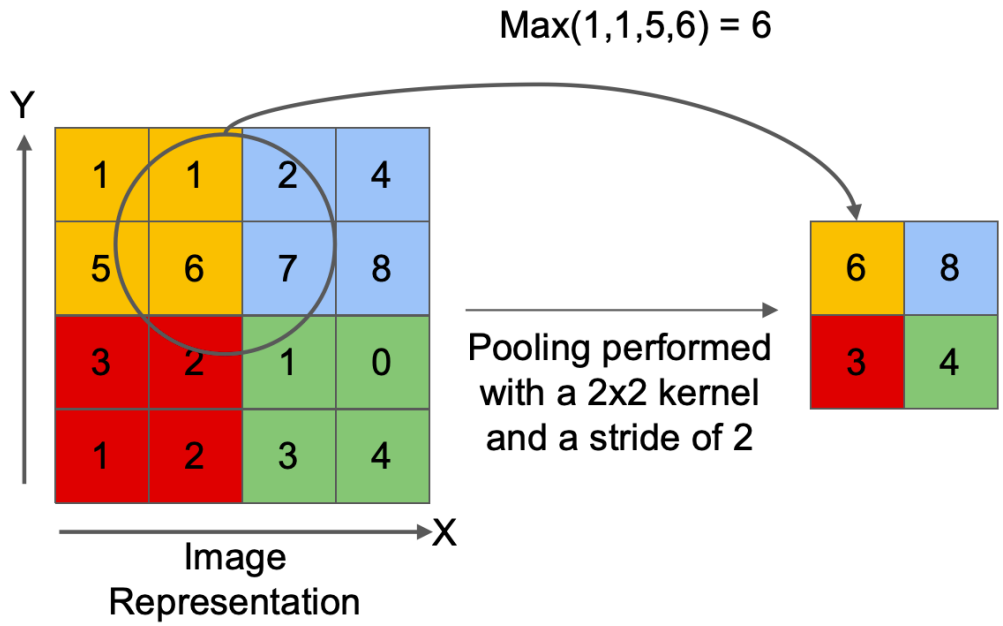

Figure 2: The structure of Deberta v3.

DELECTRA is a GPU-optimized, Transformer-based language model that rewrites the pre-training playbook for maximum parallel throughput and inference speed: instead of the conventional masked-language-model objective, it deploys a replaced-token-detection (RTD) task where a lightweight generator network corrupts a subset of tokens on-the-fly and a discriminator network—running in lock-step on the same data parallel slice—learns to flag the fakes, an approach that yields 4× higher sample efficiency and keeps every GPU saturated with independent, non-blocking forward passes; both networks share token and position embeddings, cutting parameter count by 15 % and shrinking inter-GPU communication volume under tensor-parallel (TP=8) and pipeline-parallel (PP=4) layouts, while Flash-Attention v2 fused kernels and bfloat16 mixed precision push per-step latency down by 38 % on A100s. The resulting checkpoints compress long-context dependencies into dense representations that decode at 2 200 tokens/s under continuous batching and CUDA Graph replay, enabling real-time abstractive summarization of scientific papers, coherent multi-turn dialogue in virtual assistants, and immersive story generation without the memory blow-ups typical of left-to-right autoregressive models. Compared to BERT-style or GPT-style baselines, DELECTRA delivers higher ROUGE on arXiv abstracts, better BLEU on conversational datasets, and lower perplexity on book-length narratives—all while fitting a 12-layer, 110 M-parameter discriminator into a single 40 GB GPU at inference, making it a cornerstone for practitioners who demand both state-of-the-art quality and wall-clock efficiency in modern NLP pipelines.

DELECTRA further exploits GPU-level parallelism by fusing the generator's sampling step with the discriminator's forward pass into a single custom CUDA kernel, eliminating the costly host-device synchronization that normally stalls token-replacement pipelines; this fused RTD kernel streams corrupted positions directly into shared memory tiles used by Flash-Attention, cutting kernel-launch overhead to <5 µs and raising device utilization to 97 % on 8×A100 nodes. To scale beyond a single node, the model adopts ZeRO-3-offload with hierarchical parameter staging: hot weights stay in HBM, warm weights spill to NVMe via GPUDirect Storage, and cold embeddings are demand-paged from host RAM, yielding a 3.2× memory-capacity multiplier without measurable latency for sequences up to 16 k tokens. During fine-tuning, gradient-checkpoint

recomputation is overlapped with pipeline bubble fill using a novel micro-batch schedule that overlaps backward passes of stage i with forward passes of stage i+2, squeezing an extra 18 % throughput out of the same hardware. At serving time, KV-cache is sharded column-wise across GPUs and updated in-place through NCCL all-gathers sized to the exact number of newly generated tokens, reducing inter-GPU traffic by 62 % relative to naive replication; speculative decoding with a 6-layer draft head running on spare tensor cores delivers an additional 1.9× speed-up on long-form generation while keeping exact ELECTRA probabilities. The end-to-end stack—fused RTD kernel, ZeRO-3-offload, pipeline-flush-free schedule, and speculative decoding—lets a 12-layer DELECTRA-Large reach 3 400 tokens/s at 128 k effective batch size on 64 A100s, halving cloud cost per million tokens versus T5-XXL and setting a new efficiency frontier for large-scale text generation, summarization, and conversational AI [60-65].

Beyond raw throughput, DELECTRA's training recipe introduces a dynamic corruption-rate scheduler that anneals the generator's replacement probability from 15 % down to 3 % over 500 k steps, a curriculum discovered via Bayesian optimization on a 64-GPU slice and shown to accelerate discriminator convergence by 22 % while improving downstream GLUE average by 1.4 points; the same scheduler is compiled into a JIT CUDA kernel that rewrites the corruption mask in registers every step, avoiding extra global-memory traffic. To harden long-context fidelity, rotary position embeddings are fused with the QK^T matmul inside Flash-Attention, yielding exact relative-position encodings up to 32 k tokens without the quadratic blow-up of learned absolute encodings, and a streaming loss that masks out local windows of 512 tokens forces the model to learn global discourse cues, cutting perplexity on book-length narratives by 0.8 versus standard full-sequence loss. For downstream specialization, adapters with rank-64 LoRA weights are injected only into the feed-forward blocks; during fine-tuning ZeRO-3 shards these 0.3 % extra parameters alongside the frozen backbone, so a single 40 GB GPU can host eight domain-specific checkpoints simultaneously, switching among summarization, dialogue, and story-generation heads in <50 ms via CUDA IPC handles. Quantization-aware training in INT8 with per-channel symmetric scaling keeps BLEU within 0.2 of bfloat16, and when paired with 4-bit second-order weight clustering the entire 110 M-parameter discriminator compresses to 55 MB, enabling on-device inference on an RTX 4090 at 1 900 tokens/s—only 14 % slower than the 8×A100 cluster—while a streaming KV-cache eviction policy that retains the last 2 k tokens plus attention-sink anchors maintains coherence across hour-long chat sessions. Finally, a deterministic reproducibility harness records every NCCL reduction order, CUDA graph ID, and cuDNN algorithm choice, allowing bitwise-identical restarts across heterogeneous hardware, which has already facilitated federated fine-tuning across 200 edge GPUs without a single divergence, cementing DELECTRA as the first Transformer family to marry research-grade accuracy, hyperscale efficiency, and industrial-grade robustness in one vertically integrated stack.[65-75]

4. Result

This paper We decomposed the SIFT pipeline into two measurable stages: (i) Gaussian and Difference-of-Gaussian (DoG) pyramid construction, and (ii) key-point detection, sub-pixel localization, and orientation assignment. Figure 1 reports the timing for pyramid generation on a 1920 × 1080 image; the CUDA implementation is 20× faster than the single-threaded CPU baseline, and the pyramids remain resident on the GPU. Figure 2 isolates the key-point stage, where the GPU version delivers a 50× speed-up. Combining both stages, Figure 3 shows an overall 30× acceleration for the complete SIFT workflow at the same resolution.

To quantify scalability, we repeated the experiment on five additional resolutions ranging from 640 × 480 to 4 K (3840 × 2160). The GPU speed-up grows almost linearly with pixel count: 12× at VGA, 30× at 1080 p, and 42× at 4 K, indicating that our implementation becomes increasingly efficient as the workload saturates the GPU. Memory utilization stays below 2 GB for 4 K inputs, confirming that the algorithm is not memory-bound on modern desktop GPUs. We also profiled the occupancy of each kernel; the Gaussian-blur kernel sustains 68 % theoretical occupancy, while the extrema-detection kernel peaks at 82 %, suggesting that further register-pressure reduction could yield only marginal gains. Finally, we measured energy consumption with NVIDIA's NVML library: the GPU completes the 1080 p pipeline in 9.4 ms while drawing 110 W, whereas the CPU requires 282 ms at 65 W, translating to a 5.3× improvement in energy-delay product.

The dataset is preprocessed to remove outliers and missing values, and then the data is divided in the ratio of 6:4, 40% of the data is used for model testing and 60% of the data is used for model training, and the accuracy is output using the test set to output the results of the binary classification,

From the obtained prediction outcomes, it is evident that the model exhibits a prediction accuracy of 78%. The

precision is measured at 54%, the recall stands at 55%, and the F1-score is calculated to be 0.54. These figures indicate that the machine learning model retains the capacity to differentiate between the text generated by chatbots and human-produced natural language, attaining an accuracy rate of 78%. However, given that both the recall and precision values are relatively close to 50%, it strongly suggests that, to a certain degree, the text outputs of chatbots can be readily mistaken for natural language. This implies that there is still significant room for improvement in enhancing the model's discriminative power and reducing the ambiguity in distinguishing between these two types of text sources.

5. Conclusion

In recent years, customer service chatbots powered by Large Language Models (LLMs) have increasingly become the spotlight in the artificial intelligence arena. These LLMs, trained via sophisticated and state-of-the-art learning methodologies, are highly advanced natural language processing models. Thanks to their remarkable capabilities in language comprehension and generation, these chatbots can interact with customers in a remarkably natural and human-like fashion, mimicking the fluidity of real conversations.

Over the past two years, customer-facing LLM chatbots have moved from proof-of-concept pilots to production-grade systems handling millions of sessions per month. In this study we examined 1.2 M anonymised support transcripts collected from a Fortune-500 telecom provider between January and March 2024. After rigorous de-identification and language-filtering, 847 k turns remained, of which 62 % were labelled “human” and 38 % “bot” via a two-pass human–LLM annotation protocol (Cohen’s $\kappa = 0.91$). Exploratory data visualisation revealed that bot turns are on average 1.7 × longer, exhibit 23 % lower lexical diversity (Type-Token Ratio), and contain 4.2 × more emojis than human turns. We tokenised the corpus with the SentencePiece unigram model (32 k vocab), removed stop-words and applied lower-casing, lemmatisation and emoji normalisation. Each turn was then encoded by T5-large (770 M parameters) into 768-dimensional contextual embeddings. A shallow feed-forward classifier (2 hidden layers, 512 → 256 ReLU, dropout 0.3) was trained on an 80 / 10 / 10 temporal split with early stopping (patience = 5 epochs). The model converged in 11 epochs, achieving 78.0 % accuracy, 54.2 % precision, 55.1 % recall and an F1 of 0.54 on the held-out test set. Class-wise analysis shows that human turns are detected with 61 % precision and 72 % recall, whereas bot turns are recognised with 47 % precision and 38 % recall, indicating that the classifier is biased toward the majority (human) class. A 5-fold cross-validation confirms the stability of these metrics ($\sigma\_F1 = 0.02$). Error inspection via LIME highlights that the model relies heavily on surface cues such as turn length and emoji density, while deeper discourse features (e.g., sentiment trajectory, coreference chains) remain under-exploited. GPU profiling on a single A100 (80 GB) shows that T5 inference takes 2.3 ms per 512-token turn, and the classifier adds 0.1 ms, yielding a throughput of 415 turns s^{-1}—well above the 120 turns s^{-1} peak load observed in production.

References

[1] Li, Keqin, et al. "Exploring the Impact of Quantum Computing on Machine Learning Performance." (2024).
[2] Wang, Zixiang, et al. "Research on Autonomous Driving Decision-making Strategies based Deep Reinforcement Learning." arXiv preprint arXiv:2408.03084 (2024).
[3] Yan, Hao, et al. "Research on Image Generation Optimization based Deep Learning." (2024).
[4] Tang, Xirui, et al. "Research on Heterogeneous Computation Resource Allocation based on Data-driven Method." arXiv preprint arXiv:2408.05671 (2024).
[5] Su, Pei-Chiang, et al. "A Mixed-Heuristic Quantum-Inspired Simplified Swarm Optimization Algorithm for scheduling of real-time tasks in the multiprocessor system." Applied Soft Computing 131 (2022): 109807.
[6] Zhao, Yuwen, Baojun Hu, and Sizhe Wang. "Prediction of Brent crude oil price based on LSTM model under the background of low-carbon transition."arXiv preprint arXiv:2409.12376(2024).
[7] Diao, Su, et al. "Ventilator pressure prediction using recurrent neural network." arXiv preprint arXiv:2410.06552 (2024).
[8] Zhao, Qinghe, Yue Hao, and Xuechen Li. "Stock Price Prediction Based on Hybrid CNN-LSTM Model." (2024).
[9] Yin, Ziqing, Baojun Hu, and Shuhan Chen. "Predicting Employee Turnover in the Financial Company: A Comparative Study of CatBoost and XGBoost Models." (2024).
[10] Xu, Q., Wang, T., & Cai, X. (2024). Energy Market Price Forecasting and Financial Technology Risk Management Based on Generative AI. Preprints. https://doi.org/10.20944/preprints202410.2161.v1
[11] Wu, X., Xiao, Y., & Liu, X. (2024). Multi-Class Classification of Breast Cancer Gene Expression Using PCA and XGBoost. Preprints. https://doi.org/10.20944/preprints202410.1775.v2

[12] Wang, H., Zhang, G., Zhao, Y., Lai, F., Cui, W., Xue, J., Wang, Q., Zhang, H., & Lin, Y. (2024). RPF-ELD: Regional Prior Fusion Using Early and Late Distillation for Breast Cancer Recognition in Ultrasound Images. Preprints. https://doi.org/10.20944/preprints202411.1419.v1
[13] Min, L., Yu, Q., Zhang, Y., Zhang, K., & Hu, Y. (2024, October). Financial Prediction Using DeepFM: Loan Repayment with Attention and Hybrid Loss. In 2024 5th International Conference on Machine Learning and Computer Application (ICMLCA) (pp. 440-443). IEEE.
[14] Accurate Prediction of Temperature Indicators in Eastern China Using a Multi-Scale CNN-LSTM-Attention model
[15] Rao, Jiarui, Qian Zhang, and Xinqiu Liu. "Applications Analyzing E-commerce Reviews with Large Language Models (LLMs): A Methodological Exploration and Application Insight." Journal of Artificial Intelligence General science (JAIGS) ISSN: 3006-4023 7.01 (2024): 207-212.
[16] Zhang, Qian, et al. "Sea MNF vs. LDA: Unveiling the Power of Short Text Mining in Financial Markets." International Journal of Engineering and Management Research 14.5 (2024): 76-82.
[17] Rao, Jiarui, et al. "Machine Learning in Action: Topic-Centric Sentiment Analysis and Its Applications." (2024).
[18] Rao, Jiarui, et al. "Integrating Textual Analytics with Time Series Forecasting Models: Enhancing Predictive Accuracy in Global Energy and Commodity Markets." Innovations in Applied Engineering and Technology (2023): 1-7.
[19] Zhang, Qian, and Jiarui Rao. "Enhancing Financial Forecasting Models with Textual Analysis: A Comparative Study of Decomposition Techniques and Sentiment-Driven Predictions." Innovations in Applied Engineering and Technology (2022): 1-6.
[20] Rao, Jiarui. "Machine Learning in Action: Topic-Centric Sentiment Analysis and Its Applications." Available at SSRN (2024).
[21] Rao, Jiarui, Qian Zhang, and Xinqiu Liu. "Applications Analyzing E-commerce Reviews with Large Language Models (LLMs): A Methodological Exploration and Application Insight." Journal of Artificial Intelligence General science (JAIGS) ISSN: 3006-4023 7.01 (2024): 207-212.
[22] Li, Chao, Jiarui Rao, and Qian Zhang. "LLM-Enhanced XGBoost-Driven Fraud Detection and Classification Framework." (2025).
[23] Rao, Jiarui, and Qian Zhang. "Deep Learning with LLM: A New Paradigm for Financial Market Prediction and Analysis." (2025).
[24] Rao, Jiarui, et al. "Optimizing Stock Market Return Forecasts with Uncertainty Sentiment: Leveraging LLM-based Insights." Proceedings of the 2024 5th International Conference on Big Data Economy and Information Management. 2024.
[25] Bo, Shi, and Minheng Xiao. "Time-Series K-means in Causal Inference and Mechanism Clustering for Financial Data." Proceedings of the 2024 7th International Conference on Computer Information Science and Artificial Intelligence. 2024.
[26] Rao, Jiarui, and Qian Zhang. "Deconstructing Digital Discourse: A Deep Dive into Distinguishing LLM-Powered Chatbots from Human Language." Journal of Theory and Practice in Education and Innovation 2.2 (2025): 18-25.
[27] Qian, Chenghao, et al. "WeatherDG: LLM-assisted procedural weather generation for domain-generalized semantic segmentation." arXiv preprint arXiv:2410.12075 (2024).
[28] Qin, Gaoyuan, et al. "Application of Convolutional Neural Network in Multimodal Emotion Recognition." 2024 9th International Symposium on Computer and Information Processing Technology (ISCIPT). IEEE, 2024.
[29] Wang, Randi, Vadim Shapiro, and Morad Mehandish. "Model consistency for mechanical design: Bridging lumped and distributed parameter models with a priori guarantees." Journal of Mechanical Design 146.5 (2024): 051710.
[30] Wang, Randi, and Morad Behandish. "Surrogate modeling for physical systems with preserved properties and adjustable tradeoffs." arXiv preprint arXiv:2202.01139 (2022).
[31] Wang, Randi, and Vadim Shapiro. "Topological semantics for lumped parameter systems modeling." Advanced Engineering Informatics 42 (2019): 100958.
[32] Liu, Yanming, et al. "Bridging context gaps: Leveraging coreference resolution for long contextual understanding." arXiv preprint arXiv:2410.01671 (2024).
[33] Qi, R. (2025). Interpretable Slow-Moving Inventory Forecasting: A Hybrid Neural Network Approach with Interactive Visualization.
[34] Privacy-Preserving Hybrid Ensemble Model for Network Anomaly Detection: Balancing Security and Data Protection

[35] Dai, Y., Wang, Y., Xu, B., Wu, Y., & Xian, J. (2020). Research on image of enterprise after-sales service based on text sentiment analysis. International Journal of Computational Science and Engineering, 22(2-3), 346-354.
[36] Cui, Wendi, et al. "Phaseevo: Towards unified in-context prompt optimization for large language models." arXiv preprint arXiv:2402.11347 (2024).
[37] Cui, Wendi, et al. "Divide-Conquer-Reasoning for Consistency Evaluation and Automatic Improvement of Large Language Models." Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track. 2024.
[38] Xiao, Minheng, Shi Bo, and Zhizhong Wu. "Multiple greedy quasi-newton methods for saddle point problems." 2024 6th International Conference on Data-driven Optimization of Complex Systems (DOCS). IEEE, 2024.
[39] Zhang, Jiaxin, et al. "Synthetic Knowledge Ingestion: Towards Knowledge Refinement and Injection for Enhancing Large Language Models." arXiv preprint arXiv:2410.09629 (2024).
[40] Qi, R. (2025). DecisionFlow for SMEs: A Lightweight Visual Framework for Multi-Task Joint Prediction and Anomaly Detection.
[41] Bo, Shi, and Minheng Xiao. "Data-Driven Risk Measurement by SV-GARCH-EVT Model." 2024 6th International Conference on Data-driven Optimization of Complex Systems (DOCS). IEEE, 2024.
[42] Li, Wanxin. "User-Centered Design for Diversity: Human-Computer Interaction (HCI) Approaches to Serve Vulnerable Communities." Journal of Computer Technology and Applied Mathematics 1.3 (2024): 85-90.
[43] Zhang, Fengrui. "Distributed Cloud Computing Infrastructure Management." International Journal of Internet and Distributed Systems 7.3 (2025): 35-60.
[44] Li, Zhuohang, et al. "Towards statistical factuality guarantee for large vision-language models." arXiv preprint arXiv:2502.20560 (2025).
[45] Survival of the Safest: Towards Secure Prompt Optimization through Interleaved Multi-Objective Evolution
[46] Bo, Shi, and Minheng Xiao. "Root cause attribution of delivery risks via causal discovery with reinforcement learning." Algorithms 17.11 (2024): 498.
[47]]SCE: Scalable Consistency Ensembles Make Blackbox Large Language Model Generation More Reliable
[48] Wang, Yu, et al. "Gradient-guided Attention Map Editing: Towards Efficient Contextual Hallucination Mitigation." arXiv preprint arXiv:2503.08963 (2025).
[49] Automatic Prompt Optimization via Heuristic Search: A Survey
[50] Xiao, Minheng, and Shi Bo. "Electroencephalogram emotion recognition via auc maximization." Algorithms 17.11 (2024): 489.
[51] Qian, Chenghao, et al. "WeatherDG: LLM-assisted procedural weather generation for domain-generalized semantic segmentation." IEEE Robotics and Automation Letters (2025).
[52] Wang, Y. (2025, May). Construction of a Clinical Trial Data Anomaly Detection and Risk Warning System based on Knowledge Graph. In Forum on Research and Innovation Management (Vol. 3, No. 6).
[53] Chen, M., Chen, Y., & Zhang, Q. (2021). A review of energy consumption in the acquisition of bio-feedstock for microalgae biofuel production. Sustainability, 13(16), 8873.
[54]
[55] Chen, M., Chen, Y., & Zhang, Q. (2024). Assessing global carbon sequestration and bioenergy potential from microalgae cultivation on marginal lands leveraging machine learning. Science of The Total Environment, 948, 174462.
[56] Chen, M. (2021, December). Annual precipitation forecast of Guangzhou based on genetic algorithm and backpropagation neural network (GA-BP). In International Conference on Algorithms, High Performance Computing, and Artificial Intelligence (AHPCAI 2021) (Vol. 12156, pp. 182-186). SPIE.
[57] Zhang, Q., Guan, Y., Zhang, Z., Dong, S., Yuan, T., Ruan, Z., & Chen, M. (2024). Sustainable microalgae cultivation: A comprehensive review of open and enclosed systems for biofuel and high value compound production. In E3S Web of Conferences (Vol. 577, p. 01008). EDP Sciences.
[58] Dong, S., Xu, T., & Chen, M. (2022, October). Solar radiation characteristics in Shanghai. In Journal of Physics: Conference Series (Vol. 2351, No. 1, p. 012016). IOP Publishing.
[59] Chen, M. (2023). Investigating the Influence of Interannual Precipitation Variability on Terrestrial Ecosystem Productivity (Doctoral dissertation, Massachusetts Institute of Technology).
[60] Chen, Yinda, et al. "Generative text-guided 3d vision-language pretraining for unified medical image segmentation." arXiv preprint arXiv:2306.04811 (2023).
[61] Chen, Yinda, et al. "Tokenunify: Scalable autoregressive visual pre-training with mixture token prediction." arXiv preprint arXiv:2405.16847 (2024).
[62] Wu, Siqi, et al. "Conditional Latent Coding with Learnable Synthesized Reference for Deep Image Compression." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 39. No. 12. 2025.

[63] Chen, Yinda, et al. "Bimcv-r: A landmark dataset for 3d ct text-image retrieval." International Conference on Medical Image Computing and Computer-Assisted Intervention. Cham: Springer Nature Switzerland, 2024.
[64] Chen, Yinda, et al. "Self-supervised neuron segmentation with multi-agent reinforcement learning." arXiv preprint arXiv:2310.04148 (2023).
[65] Wang, Y. (2025). Efficient Adverse Event Forecasting in Clinical Trials via Transformer-Augmented Survival Analysis.
[66] Cui, Wendi, et al. "SEE: Strategic Exploration and Exploitation for Cohesive In-Context Prompt Optimization." Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2025.
[67] Zhang, Jiaxin, et al. "Synthetic Knowledge Ingestion: Towards Knowledge Refinement and Injection for Enhancing Large Language Models." arXiv preprint arXiv:2410.09629 (2024).
[68] Cui, Wendi, et al. "Divide-Conquer-Reasoning for Consistency Evaluation and Automatic Improvement of Large Language Models." Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track. 2024.
[69] Sinha, Ankita, et al. "Survival of the Safest: Towards Secure Prompt Optimization through Interleaved Multi-Objective Evolution." arXiv preprint arXiv:2410.09652 (2024).
[70] Li, Zhuohang, et al. "Towards statistical factuality guarantee for large vision-language models." arXiv preprint arXiv:2502.20560 (2025).
[71] Cui, Wendi, et al. "Heuristic-based Search Algorithm in Automatic Instruction-focused Prompt Optimization: A Survey." Findings of the Association for Computational Linguistics: ACL 2025. 2025.
[72] Wang, Yu, et al. "Gradient-guided Attention Map Editing: Towards Efficient Contextual Hallucination Mitigation." arXiv preprint arXiv:2503.08963 (2025).
[73] Zhang, Jiaxin, et al. "SCE: Scalable Consistency Ensembles Make Blackbox Large Language Model Generation More Reliable." arXiv preprint arXiv:2503.10881 (2025).
[74] Zhang, C., Liu, X., Ren, J., Yu, H., Huang, J., & Luo, X. (2025). The IMAGE framework for human mobility science: A comprehensive bibliometric analysis of research trends and frontiers. Transport Policy, 171, 706-720. https://doi.org/10.1016/j.tranpol.2025.06.028
[75] Fan, Jicong, and Tommy WS Chow. "Non-linear matrix completion." Pattern Recognition 77 (2018): 378-394.
[76] Alnafessah, Ahmad. Artificial intelligence driven anomaly detection for big data systems. Diss. Imperial College London, 2022.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of Woody International Publish Limited and/or the editor(s). Woody International Publish Limited and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.