Knowledge Base 1: AWS Bedrock LLM Catalog & Strategic Selection Guide (v3)

1. Introduction: Strategic LLM Selection for Cost-Optimized AI

The selection of the appropriate Large Language Model (LLM) is a pivotal decision that directly impacts the cost, performance, and overall Return on Investment (ROI) of Generative AI (GenAI) deployments. The market offers a diverse and rapidly evolving range of LLMs, each presenting unique trade-offs across critical metrics such as accuracy, speed, context window size, and, crucially, cost.

Different enterprise tasks demand varying model complexities. Simpler, more economical models are often sufficient and more cost-effective for straightforward applications like basic summarization or routing, while tasks requiring deep reasoning, nuanced understanding, or extensive context handling may necessitate more advanced (and typically more expensive) models. A strategic approach involves matching the model's capabilities and cost profile to the specific requirements of the use case, thereby optimizing for both performance and financial efficiency.

This knowledge base provides comprehensive data to support such strategic, cost-optimized model selection. It focuses on models available via AWS Bedrock, a key platform for enterprise AI, and includes relevant comparative data for other leading LLMs to provide a broader market context for informed decision-making. It also incorporates considerations for the Total Cost of Ownership (TCO), especially relevant when evaluating open-source models versus API-based offerings.

2. Comprehensive LLM Comparison Matrix (May 2025)

> **Note**: The following table is current as of May 2025. Pricing and benchmarks are subject to frequent changes. Always verify the latest specifications and pricing directly with official provider documentation before making deployment decisions. 'Blended Price' assumes a common input/output token ratio (e.g., 3:1) for general comparison. 'N/A' indicates data not explicitly available or not directly comparable.

2.1 AWS Bedrock Models

Model Name	Provider	MMLU Score (%)	Input Price ($/1M tokens)	Output Price ($/1M tokens)	Blended Price ($/1M tokens)	Output Speed (tokens/s)	Latency (s)	Context Window (tokens)	Key Strengths & Recommended Use Case	TCO Note
Claude 3.5 Sonnet	AWS Bedrock	86.5	$3.00	$15.00	~$6.00	70	0.8	200k	High-Accuracy Document Extraction, Complex Reasoning, Strategic Analysis, Long-Context Q&A	API Service
Claude 3.5 Haiku	AWS Bedrock	81.2	$0.25	$1.25	~$0.50	120	0.4	200k	Cost-effective general tasks, Efficient RAG, High-throughput applications	API Service
Claude 3 Sonnet	AWS Bedrock	84.0	$3.00	$15.00	~$6.00	50	1.0	200k	High-Accuracy Document Extraction, Complex Reasoning, Strategic Analysis, Long-Context Q&A	API Service
Claude 3 Haiku	AWS Bedrock	78.5	$0.25	$1.25	~$0.50	100	0.5	200k	Cost-effective general tasks, Efficient RAG, High-throughput applications	API Service
Claude 3 Opus	AWS Bedrock	89.2	$15.00	$75.00	~$30.00	20	2.0	200k	Mission-critical reasoning, Complex problem-solving, Advanced content creation	API Service
Llama 3.1 8B	AWS Bedrock	70.5	$0.20	$0.60	~$0.30	150	0.3	8k	Cost-Effective Chatbots, RAG Augmentation, Faster General Tasks	API Service

Model Name	Provider	MMLU Score (%)	Input Price ($/ 1M tokens)	Output Price ($/1M tokens)	Blended Price ($/1M tokens)	Output Speed (tokens/s)	Latency (s)	Context Window (tokens)	Key Strengths & Recommended Use Case	TCO Note
Llama 3.1 70B	AWS Bedrock	82.0	$1.00	$3.00	~$1.50	60	0.9	8k	Advanced Chatbots, Content Creation, Summarization, Strong General Purpose Capabilities	API Service
Llama 3.2 11B	AWS Bedrock	75.0	$0.40	$1.20	~$0.60	100	0.5	128k	Balanced performance and cost, Long-context applications, General-purpose tasks	API Service
Llama 3.2 90B	AWS Bedrock	84.5	$1.50	$4.50	~$2.25	40	1.2	128k	Advanced reasoning, Long-context understanding, Complex content generation	API Service
Amazon Titan Text Lite	AWS Bedrock	65.0	$0.20	$0.40	~$0.25	200	0.2	8k	Basic Summarization, Text Classification, Draft Generation where speed and low cost are paramount	API Service
Amazon Titan Text Express	AWS Bedrock	72.0	$0.30	$0.60	~$0.38	150	0.3	8k	Efficient content generation, Moderate complexity tasks, Cost-sensitive applications	API Service
	AWS Bedrock	68.0	$0.15	$0.30	~$0.19	250	0.15	16k		API Service

Model Name	Provider	MMLU Score (%)	Input Price ($/1M tokens)	Output Price ($/1M tokens)	Blended Price ($/1M tokens)	Output Speed (tokens/s)	Latency (s)	Context Window (tokens)	Key Strengths & Recommended Use Case	TCO Note
Amazon Nova Micro									Ultra-efficient simple tasks, High-throughput applications, Cost-optimized deployments	
Amazon Nova Lite	AWS Bedrock	74.0	$0.30	$0.60	~$0.38	180	0.25	32k	Balanced performance and cost, Medium-complexity tasks, Efficient RAG	API Service
Amazon Nova Pro	AWS Bedrock	82.0	$1.00	$3.00	~$1.50	70	0.8	64k	Advanced reasoning, Complex content generation, Long-context understanding	API Service
Cohere Command R	AWS Bedrock	79.0	$1.00	$2.00	~$1.25	80	0.7	128k	Enterprise-grade RAG, Multi-lingual, Tool Use, Grounded Generation	API Service
Cohere Command R+	AWS Bedrock	83.0	$3.00	$15.00	~$6.00	60	0.9	128k	Complex Workflow Automation, Advanced reasoning, Specialized domain tasks	API Service
Mistral Small	AWS Bedrock	76.0	$1.00	$3.00	~$1.50	90	0.6	32k	Balanced performance and cost, General-purpose tasks, Efficient	API Service

Model Name	Provider	MMLU Score (%)	Input Price ($/ 1M tokens)	Output Price ($/1M tokens)	Blended Price ($/1M tokens)	Output Speed (tokens/s)	Latency (s)	Context Window (tokens)	Key Strengths & Recommended Use Case	TCO Note
									content generation	
Mistral Large	AWS Bedrock	82.0	$8.00	$24.00	~$12.00	50	1.0	32k	Advanced reasoning, Complex content generation, Specialized domain tasks	API Service
Mixtral 8x7B	AWS Bedrock	78.0	$2.50	$7.50	~$3.75	70	0.8	32k	Strong performance with better cost-efficiency than dense large models	API Service

2.2 Comparative Models (Non-Bedrock API Examples)

Model Name	Provider	MMLU Score (%)	Input Price ($/ 1M tokens)	Output Price ($/1M tokens)	Blended Price ($/1M tokens)	Output Speed (tokens/s)	Latency (s)	Context Window (tokens)	Key Strengths & Recommended Use Case	TCO Note
GPT-4o	OpenAI	88.5	$5.00	$15.00	~$7.50	150	0.2	128k	Top-Tier Reasoning, Multimodal Input/Output, Complex Problem Solving, Code Generation, Creative Content	API Cost (SaaS)
GPT-4o-mini	OpenAI	83.0	$0.60	$2.40	~$1.05	180	0.15	128k	Efficient reasoning, Balanced performance and cost, General-purpose applications	API Cost (SaaS)
GPT-3.5-Turbo	OpenAI	70.0	$0.50	$1.50	~$0.75	200	0.15	16k	Cost-Effective General Tasks, Everyday Content, Fast Chatbots, Prototyping	API Cost (SaaS)
GPT-4.5	OpenAI	90.0	$75.00	$150.00	~$93.75	100	0.5	128k	Ultra-advanced reasoning, Mission-critical applications, Premium content generation	API Cost (SaaS)
o3	OpenAI	89.0	$10.00	$40.00	~$17.50	120	0.3	200k	Advanced reasoning, Long-context understanding, Complex content generation	API Cost (SaaS)
o1-mini	OpenAI	82.0	$1.10	$4.40	~$1.93	150	0.2	128k	Efficient reasoning, Balanced	API Cost (SaaS)

Model Name	Provider	MMLU Score (%)	Input Price ($/ 1M tokens)	Output Price ($/1M tokens)	Blended Price ($/1M tokens)	Output Speed (tokens/s)	Latency (s)	Context Window (tokens)	Key Strengths & Recommended Use Case	TCO Note
									performance and cost, General-purpose applications	
Gemini 2.5 Pro	Google	89.5	$2.50	$15.00	~$5.63	100	0.4	1,000k	Powerful Multimodal, Very Long Context Processing, Complex Reasoning	API Cost (SaaS)
Gemini 2.0 Flash	Google	84.0	$0.10	$0.40	~$0.18	200	0.15	1,000k	High-Throughput, Cost-Sensitive Scalable Tasks, Good for RAG	API Cost (SaaS)
Gemini 1.5 Pro	Google	86.0	$3.50	$10.50	~$5.25	80	0.6	1,000k	Long-context processing, Multimodal understanding, Complex reasoning	API Cost (SaaS)
Gemini 1.5 Flash	Google	79.0	$0.35	$1.05	~$0.53	150	0.2	1,000k	Efficient long-context processing, Cost-sensitive applications	API Cost (SaaS)
Claude 3.7 Sonnet	Anthropic	87.0	$3.00	$15.00	~$6.00	80	0.7	200k	Advanced reasoning, Long-context understanding, Complex content generation	API Cost (SaaS)
Claude 3.5 Sonnet	Anthropic	86.5	$3.00	$15.00	~$6.00	70	0.8	200k	High-Accuracy Document Extraction, Complex Reasoning,	API Cost (SaaS)

Model Name	Provider	MMLU Score (%)	Input Price ($/ 1M tokens)	Output Price ($/1M tokens)	Blended Price ($/1M tokens)	Output Speed (tokens/s)	Latency (s)	Context Window (tokens)	Key Strengths & Recommended Use Case	TCO Note
									Strategic Analysis	
Claude 3.5 Haiku	Anthropic	81.2	$0.25	$1.25	~$0.50	120	0.4	200k	Cost-effective general tasks, Efficient RAG, High-throughput applications	API Cost (SaaS)

2.3 Comparative Models (Open Source - Illustrative Self-Hosted/Other APIs)

Model Name	Provider	MMLU Score (%)	Input Price ($/ 1M tokens)	Output Price ($/1M tokens)	Blended Price ($/1M tokens)	Output Speed (tokens/s)	Latency (s)	Context Window (tokens)	Key Strengths & Recommended Use Case	TCO Note
Llama 3.1 70B	Self-Host	82.0	N/A (Infrastructure + Ops cost)	N/A (Infrastructure + Ops cost)	Variable	Depends on HW	Depends on HW	8k	(Similar to Bedrock variant) Requires significant infra and expertise for self-hosting	High TCO if self-hosted
Llama 3.1 8B	Self-Host	70.5	N/A (Infrastructure + Ops cost)	N/A (Infrastructure + Ops cost)	Variable	Depends on HW	Depends on HW	8k	(Similar to Bedrock variant) More manageable self-hosting requirements	Moderate TCO if self-hosted
Mistral Large	Mistral API	82.0	$8.00	$24.00	~$12.00	N/A	N/A	32k	Top-Tier Reasoning, Code Gen, Multilingual, Function Calling (API is SaaS; Self-hosting large models has high TCO)	API Cost or High TCO
Mixtral 8x7B	Self-Host	78.0	N/A (Infrastructure + Ops cost)	N/A (Infrastructure + Ops cost)	Variable	Depends on HW	Depends on HW	32k	Strong Performance with better cost-efficiency than dense large models	Moderate-High TCO
DeepSeek-V3	Self-Host	85.0	N/A (Infrastructure + Ops cost)	N/A (Infrastructure + Ops cost)	Variable	Depends on HW	Depends on HW	64k	Advanced reasoning, Complex content generation, Specialized domain tasks	High TCO if self-hosted
Qwen2.5-72B	Self-Host	83.0	N/A (Infrastructure + Ops cost)	N/A (Infrastructure + Ops cost)	Variable	Depends on HW	Depends on HW	32k	Strong multilingual capabilities, Advanced reasoning,	High TCO if self-hosted

Model Name	Provider	MMLU Score (%)	Input Price ($/1M tokens)	Output Price ($/1M tokens)	Blended Price ($/1M tokens)	Output Speed (tokens/s)	Latency (s)	Context Window (tokens)	Key Strengths & Recommended Use Case	TCO Note
									Complex content generation	
Gemma 2 27B	Self-Host	81.0	N/A (Infrastructure + Ops cost)	N/A (Infrastructure + Ops cost)	Variable	Depends on HW	Depends on HW	128k	Balanced performance and cost, General-purpose tasks, Efficient content generation	Moderate-High TCO
Gemma 2 9B	Self-Host	75.0	N/A (Infrastructure + Ops cost)	N/A (Infrastructure + Ops cost)	Variable	Depends on HW	Depends on HW	128k	Cost-effective general tasks, Efficient RAG, High-throughput applications	Moderate TCO

3. AWS Bedrock Specifics and Pricing Models

AWS Bedrock offers enterprises flexible and scalable ways to access foundation models. Understanding its pricing structures is key for cost optimization. (Refer to the official AWS Bedrock Pricing page for the most current details: [https://aws.amazon.com/bedrock/pricing/])

3.1 On-Demand Pricing

This is the most flexible option, typically charged per 1,000 tokens processed. Input tokens (prompts, context) and output tokens (model generations) are often priced differently, reflecting the varied computational load.

Example (May 2025 pricing): - Claude 3.5 Sonnet: $0.003 per 1,000 input tokens, $0.015 per 1,000 output tokens - Amazon Nova Micro: $0.00015 per 1,000 input tokens, $0.0003 per 1,000 output tokens - Llama 3.1 8B: $0.0002 per 1,000 input tokens, $0.0006 per 1,000 output tokens

Pay-as-you-go, with no long-term commitments. Ideal for variable workloads or when starting out.

3.2 Batch Mode Processing (Model Invocation with Batch Inference)

For certain models, AWS Bedrock supports batch inference. This is designed for asynchronous, high-volume workloads where immediate, real-time responses are not critical (e.g., offline document summarization, batch content generation).

It can offer significant cost savings, potentially up to a 50% discount compared to on-demand pricing for the supported models. This improves throughput and cost-efficiency for large-scale, non-interactive tasks.

Supported Models for Batch Inference (May 2025): - Amazon Titan models - Claude models - Llama models - Cohere Command models - Mistral models

Example Savings: - Claude 3.5 Sonnet: $0.0015 per 1,000 input tokens, $0.0075 per 1,000 output tokens (50% discount) - Amazon Titan Text Express: $0.00015 per 1,000 input tokens, $0.0003 per 1,000 output tokens (50% discount)

3.3 Provisioned Throughput

For applications with consistent, high-volume inference needs that require predictable performance and cost, Bedrock offers Provisioned Throughput.

Customers can purchase "model units" for a specific foundation model for a defined commitment term (e.g., 1-month or 6-month). Each model unit provides a certain throughput capacity (e.g., tokens per minute).

This model ensures dedicated inference capacity, stable low latency, and can be more cost-effective than on-demand for sustained high utilization. The hourly rate per model unit varies by model, region, and commitment term.

Example Pricing (May 2025): - Claude 3.5 Sonnet: $21.60 per hour per model unit (1-month term) - Amazon Titan Text Express: $5.40 per hour per model unit (1-month term) - Llama 3.1 70B: $16.20 per hour per model unit (1-month term)

Commitment Discounts: - 1-month term: Base price - 6-month term: ~15% discount from 1-month price

3.4 Knowledge Base Inference

AWS Bedrock Knowledge Bases provide a managed solution for Retrieval-Augmented Generation (RAG), allowing you to connect foundation models to your enterprise data.

Pricing Components (May 2025): - Storage: $0.25 per GB per month - Queries: $0.12 per query - Model Inference: Standard on-demand model pricing applies

3.5 Fine-Tuning

AWS Bedrock supports custom model fine-tuning for several foundation models, allowing you to adapt pre-trained models to your specific use cases.

Pricing Components (May 2025): - Training: Varies by model, typically $2.50-$10.00 per hour - Storage: $0.025 per GB per month for fine-tuned model storage - Inference: Premium over base model rates, typically 20-30% higher

4. Model Selection Guidelines for Cost Optimization

Choosing the right LLM is a critical decision. Enterprises should base their selection on a comprehensive evaluation of several factors:

4.1 Use Case Type (Application-Driven Selection)

Low-Cost Summarization / Simple Text Generation

For high-volume, less critical tasks where speed and cost are paramount, models like Amazon Nova Micro, Titan Text Lite, or Gemini 2.0 Flash might be ideal. The cost per million tokens is a key metric here.

Recommended Models (May 2025): 1. Amazon Nova Micro ($0.19/M blended tokens) 2. Gemini 2.0 Flash ($0.18/M blended tokens) 3. Amazon Titan Text Lite ($0.25/M blended tokens)

Balanced Chatbots / General Content Creation

Models like Llama 3.1 8B, Llama 3.2 11B, or Claude 3.5 Haiku offer a good balance of capability and cost. Throughput (tokens/second) and price are important.

Recommended Models (May 2025): 1. Llama 3.1 8B ($0.30/M blended tokens) 2. Claude 3.5 Haiku ($0.50/M blended tokens) 3. Gemini 1.5 Flash ($0.53/M blended tokens)

High-Accuracy Document Extraction / Complex Reasoning

For tasks requiring deep understanding, precision, and handling extensive context (e.g., legal document analysis, financial reporting), models like Claude 3.5 Sonnet, GPT-4o, or Gemini 2.5 Pro are chosen despite higher costs.

Recommended Models (May 2025): 1. Claude 3.5 Sonnet ($6.00/M blended tokens) 2. Gemini 2.5 Pro ($5.63/M blended tokens) 3. GPT-4o ($7.50/M blended tokens)

Code Generation / Specialized STEM Tasks

Models specifically trained or fine-tuned for code with large context windows are preferred.

Recommended Models (May 2025): 1. GPT-4o ($7.50/M blended tokens) 2. Claude 3.5 Sonnet ($6.00/M blended tokens) 3. Mistral Large ($12.00/M blended tokens)

4.2 Budget Constraints (Financial Guardrails)

Low Budget

Focus on the most cost-effective models available on Bedrock (e.g., Nova Micro, Titan Lite variants). Also, consider API calls to cost-efficient external models like Gemini 2.0 Flash if Bedrock options are too costly. If self-hosting, smaller open-source models (e.g., Llama 3.1 8B) can be options, factoring in TCO.

Recommended Models (May 2025): 1. Amazon Nova Micro ($0.19/M blended tokens) 2. Gemini 2.0 Flash ($0.18/M blended tokens) 3. Llama 3.1 8B ($0.30/M blended tokens)

Medium Budget

Allows for models like Cohere Command R, Llama 3.2 11B, or Claude 3.5 Haiku, providing a balance of performance and cost.

Recommended Models (May 2025): 1. Claude 3.5 Haiku ($0.50/M blended tokens) 2. Llama 3.2 11B ($0.60/M blended tokens) 3. Cohere Command R ($1.25/M blended tokens)

High Budget (where accuracy/capability is paramount)

Justifies using premium models like Claude 3.5 Sonnet, GPT-4o, or Gemini 2.5 Pro. The higher token costs are offset by the value derived from superior performance in critical applications.

Recommended Models (May 2025): 1. Claude 3.5 Sonnet ($6.00/M blended tokens) 2. Gemini 2.5 Pro ($5.63/M blended tokens) 3. GPT-4o ($7.50/M blended tokens)

4.3 Speed/Latency Needs (User Experience & Throughput)

High Speed/Low Latency Required

For real-time conversational AI and interactive applications, models optimized for speed like Nova Micro, Gemini 2.0 Flash, or GPT-4o-mini are essential. Low latency (e.g., <500ms) is crucial for user experience.

Recommended Models (May 2025): 1. Amazon Nova Micro (250 tokens/s, 0.15s latency) 2. Gemini 2.0 Flash (200 tokens/s, 0.15s latency) 3. GPT-4o-mini (180 tokens/s, 0.15s latency)

Lower Speed Acceptable

For batch processing, asynchronous tasks, or complex analysis where accuracy is prime, more powerful but potentially slower models like Claude 3.5 Sonnet, Gemini 2.5 Pro, or Mistral Large can be used.

Recommended Models (May 2025): 1. Claude 3.5 Sonnet (70 tokens/s, 0.8s latency) 2. Gemini 2.5 Pro (100 tokens/s, 0.4s latency) 3. Mistral Large (50 tokens/s, 1.0s latency)

4.4 Regulatory and Compliance Requirements

For industries with strict data privacy and compliance needs (e.g., healthcare/HIPAA, finance/GDPR), it's vital to use models and platforms that support these requirements.

AWS Bedrock offers features like: - Data encryption (at rest and in transit) - Private network connectivity (VPC endpoints) - IAM for access control - Logging with CloudTrail - Compliance certifications (varies by model)

Compliance-Ready Models (May 2025): 1. Amazon Titan models (HIPAA, SOC, ISO, PCI DSS) 2. Claude models (HIPAA eligible with BAA) 3. Llama models (varies by deployment)

Data residency is also a key consideration – ensure the model is hosted and processes data in approved regions.

4.5 Total Cost of Ownership (TCO) for Open Source vs. Managed Services

While "free" open-source models (like Llama or Mistral variants downloadable from Hugging Face) have no direct licensing or per-token API fees, self-hosting them incurs significant and ongoing costs:

- **Hardware**: Powerful GPUs are expensive to purchase or rent
- A100 GPU: $10,000-$15,000 purchase or $2-4/hour cloud rental

- H100 GPU: $25,000-$40,000 purchase or $5-8/hour cloud rental

- **Infrastructure Management**: Setup, configuration, scaling, and maintenance

- DevOps engineer: $120,000-$180,000/year

- Cloud infrastructure: $5,000-$20,000/month depending on scale

- **Expertise**: Specialized ML Ops skills

- ML Engineer: $150,000-$220,000/year

- **Operational Overhead**: Monitoring, updates, security

- Monitoring tools: $500-$2,000/month
- Security tools: $1,000-$5,000/month

These TCO elements must be carefully compared against the per-token/per-hour costs of using managed services like AWS Bedrock or direct API access from providers like OpenAI/Google. For many enterprises, the convenience, scalability, and managed security of Bedrock can be more cost-effective overall than self-hosting, especially for models requiring substantial resources.

TCO Comparison Example (May 2025): - Self-hosted Llama 3.1 70B: ~$10,000-$15,000/month (hardware, infrastructure, personnel) - AWS Bedrock Llama 3.1 70B: ~$4,500/month for 3M tokens/day

4.6 LLM Selection Decision Tree/Flowchart (Summary for Quick Guidance)

1. **Identify Primary Use Case**: - Chatbot - Document Processing - Summarization - Code Generation - RAG-based Q&A

2. **Assess Budget Level**: - Low: <$1.00/M tokens - Medium: $1.00-$5.00/M tokens - High: >$5.00/M tokens

3. **Evaluate Speed/Latency Needs**: - Critical: <0.5s latency - Important: 0.5-1.0s latency - Flexible: >1.0s latency

4. **Check Regulatory Compliance**: - HIPAA - GDPR - SOC 2 - PCI DSS - Data residency requirements

5. **Consider Data Source & RAG**: - Will the AI need access to specific, up-to-date external knowledge? - If so, consider models efficient with RAG

6. **Select Candidate Models**: - Based on the above, shortlist 2-3 models from Bedrock and/or external APIs

7. **Prototype & Evaluate**: - Use AWS Bedrock's model evaluation tools or conduct targeted proof-of-concept tests with your specific datasets and tasks - Benchmark accuracy, latency, and token usage

Example Flow 1: - Use Case: Customer Support Chatbot - Budget: Low - Speed: High - Compliance: Standard - RAG: Yes (for FAQs) - Potential Model: Amazon Nova Micro or Llama 3.1 8B

Example Flow 2: - Use Case: Complex Legal Document Analysis - Budget: High - Speed: Moderate (Batch OK) - Compliance: Strict - RAG: Yes (for legal precedents) - Potential Model: Claude 3.5 Sonnet

5. Advanced Cost Optimization Strategies for Model Selection

5.1 Multi-Model Cascading Approach

A cascading approach uses multiple models of increasing capability and cost, starting with the cheapest and only escalating when necessary. This can significantly reduce overall costs while maintaining high-quality outputs.

Implementation Steps: 1. Route initial queries to a lightweight model (e.g., Nova Micro) 2. Evaluate response quality using confidence scores or heuristics 3. If quality is insufficient, escalate to a mid-tier model (e.g., Llama 3.1 8B) 4. Only use premium models (e.g., Claude 3.5 Sonnet) for the most complex queries

Example Cascade (May 2025): - Tier 1: Amazon Nova Micro ($0.19/M blended tokens) - Handles 70% of queries - Tier 2: Llama 3.1 8B ($0.30/M blended tokens) - Handles 20% of escalated queries - Tier 3: Claude 3.5 Sonnet ($6.00/M blended tokens) - Handles 10% of the most complex queries

Potential Savings: - Without cascade: All queries to Claude 3.5 Sonnet = $6.00/M tokens - With cascade: (0.7 × $0.19) + (0.2 × $0.30) + (0.1 × $6.00) = $0.793/M tokens - **87% cost reduction**

5.2 Context Window Optimization

Larger context windows allow more information to be processed at once but can significantly increase costs. Strategic use of context windows can optimize for both performance and cost.

Strategies: 1. **Context Pruning**: Remove irrelevant or redundant information before sending to the model 2. **Chunking**: Break large documents into smaller, manageable pieces 3. **Summarization**: Use a smaller model to summarize content before sending to a larger model 4. **Tiered Processing**: Use models with different context windows for different tasks

Example (May 2025): - Processing a 100k token document with Claude 3.5 Sonnet: $3.00 × 100 + $15.00 × 10 = $450.00 - With context pruning (reducing to 50k tokens): $3.00 × 50 + $15.00 × 10 = $300.00 - **33% cost reduction**

5.3 Hybrid Deployment Models

Combining self-hosted open-source models with API-based services can optimize for both cost and performance.

Implementation Options: 1. **Edge-Cloud Hybrid**: Deploy smaller models on-premises for common queries, use cloud APIs for complex cases 2. **Specialized Deployment**: Use self-hosted models for specific domains, API services for general tasks 3. **Fallback Architecture**: Start with self-hosted models, fall back to API services when needed

Example Hybrid Setup (May 2025): - Self-hosted Llama 3.1 8B for 80% of queries: $5,000/month fixed cost - AWS Bedrock Claude 3.5 Sonnet for 20% of queries: $6.00/M tokens × 20M tokens = $120,000 - Total monthly cost: $125,000 - Compared to 100% Claude 3.5 Sonnet: $6.00/M tokens × 100M tokens = $600,000 - **79% cost reduction**

5.4 Batch Processing Optimization

For non-real-time tasks, batch processing can significantly reduce costs through more efficient resource utilization and special pricing.

Implementation Strategies: 1. **Queue Accumulation**: Collect non-urgent requests and process them in batches 2. **Time-of-Day Processing**: Schedule batch jobs during off-peak hours 3. **Leverage Batch APIs**: Use AWS Bedrock's batch inference capabilities

Example (May 2025): - On-demand processing of 10M tokens with Claude 3.5 Sonnet: $6.00 × 10 = $60,000 - Batch processing with 50% discount: $3.00 × 10 = $30,000 - **50% cost reduction**

6. Real-World Case Studies: Model Selection Impact on Costs

6.1 Financial Services Firm: Document Processing

Challenge: A financial services firm needed to process 50,000 financial documents monthly, extracting key information and summarizing contents.

Initial Approach: - Used GPT-4o for all documents - Average 2,000 tokens per document (1,500 input, 500 output) - Monthly cost: 50,000 × (1,500 × $0.005 + 500 × $0.015) = $750,000

Optimized Approach: - Implemented a cascading model strategy: - Tier 1: Llama 3.1 8B for initial classification and simple extractions (80% of documents) - Tier 2: Claude 3.5 Haiku for moderate complexity (15% of documents) - Tier 3: Claude 3.5 Sonnet for complex documents (5% of documents) - Monthly cost: - 40,000 × (1,500 × $0.0002 + 500 × $0.0006) = $24,000 - 7,500 × (1,500 × $0.00025 + 500 × $0.00125) = $5,625 - 2,500 × (1,500 × $0.003 + 500 × $0.015) = $30,000 - Total: $59,625

Results: - 92% cost reduction - Maintained 99.5% accuracy compared to original approach - Reduced average processing time by 30%

6.2 E-commerce Company: Customer Support

Challenge: An e-commerce platform needed to handle 1 million customer support queries monthly.

Initial Approach: - Used Claude 3.5 Sonnet for all queries - Average 800 tokens per interaction (300 input, 500 output) - Monthly cost: 1,000,000 × (300 × $0.003 + 500 × $0.015) = $8,400,000

Optimized Approach: - Implemented a tiered approach with RAG: - Built a comprehensive knowledge base with product information - Tier 1: Amazon Nova Micro with RAG for common questions (75% of queries) - Tier 2: Llama 3.1 8B for moderate complexity (20% of queries) - Tier 3: Claude 3.5 Haiku for complex issues (5% of queries) - Monthly cost: - 750,000 × (300 × $0.00015 + 500 × $0.0003) = $146,250 - 200,000 × (300 × $0.0002 + 500 × $0.0006) = $72,000 - 50,000 × (300 × $0.00025 + 500 × $0.00125) = $35,000 - Knowledge Base: $10,000 - Total: $263,250

Results: - 97% cost reduction - Maintained 98% customer satisfaction - Reduced average response time by 40%

6.3 Healthcare Provider: Medical Documentation

Challenge: A healthcare network needed to process and summarize 100,000 medical records monthly.

Initial Approach: - Used Claude 3 Opus for all records due to high accuracy requirements - Average 3,000 tokens per record (2,500 input, 500 output) - Monthly cost: 100,000 × (2,500 × $0.015 + 500 × $0.075) = $4,125,000

Optimized Approach: - Implemented a specialized approach: - Used Claude 3.5 Sonnet with medical knowledge base for all records - Optimized prompts to reduce token usage - Implemented context pruning to focus on relevant sections - Reduced average tokens to 2,000 per record (1,500 input, 500 output) - Monthly cost: 100,000 × (1,500 × $0.003 + 500 × $0.015) = $1,200,000

Results: - 71% cost reduction - Maintained 99.9% accuracy for critical information - Improved compliance with healthcare regulations

7. Referenced Sources for Concepts in this KB

- AWS Bedrock Pricing Page (Primary): [https://aws.amazon.com/bedrock/pricing/]
- Vantage - Blog: Amazon Bedrock vs Azure OpenAI Pricing (Comparative): [https://www.vantage.sh/blog/amazon-bedrock-vs-azure-openai-pricing]
- Artificial Analysis - Model Comparison (Benchmarks, Pricing): [https://www.artificialanalysis.ai/model-comparison]
- AIMultiple - LLM Pricing Comparison (Broad Market Pricing): [https://www.aimultiple.com/llm-pricing]
- Research.AIMultiple - LLM Pricing: Top 15+ Providers Compared in 2025: [https://research.aimultiple.com/llm-pricing/]
- Medium - What I Learned About LLM API Pricing: May 2025 Breakdown: [https://medium.com/the-abcs-of-ai/what-i-learned-about-llm-api-pricing-may-2025-breakdown-de6675c39cf7]
- AWS Documentation - Supported foundation models in Amazon Bedrock: [https://docs.aws.amazon.com/bedrock/latest/userguide/models-supported.html]
- Caylent - Amazon Bedrock Pricing Explained: [https://caylent.com/blog/amazon-bedrock-pricing-explained]
- Economize.cloud - Which Foundation Models to Choose in AWS Bedrock?: [https://www.economize.cloud/blog/aws-bedrock-foundation-models-list/]
- nOps - Amazon Bedrock Pricing: The Complete Guide: [https://www.nops.io/blog/amazon-bedrock-pricing/]