**Comprehensive AI Model Recommendations for Phishing Detection**

**Executive Summary**

This document provides strategic AI model recommendations for building a comprehensive phishing detection system targeting Critical Sector Enterprises (CSEs). The recommendations balance accuracy, performance, and scalability across multiple detection vectors including visual analysis, natural language processing, behavioral pattern recognition, and infrastructure analysis.

**Model Categories & Strategic Applications**

**1. Computer Vision Models**

**Primary: CLIP (Contrastive Language-Image Pre-training)**

* **Use Case**: Logo/favicon similarity detection against CSE brands
* **Advantage**: Semantic similarity understanding between images and text descriptions
* **Implementation**: Search for "State Bank of India logo" semantically rather than pixel-perfect matches
* **Why Superior**: Handles resized, recolored, or stylistically modified logos that traditional CV approaches (SIFT/ORB) miss

**Secondary: DINO (Self-Distillation with No Labels)**

* **Use Case**: Structural layout similarity detection between legitimate CSE pages and phishing attempts
* **Advantage**: Excellent visual representations without labeled data requirements
* **Implementation**: Compare webpage screenshots to detect layout copying
* **Why Superior**: Captures structural patterns that persist even when attackers modify colors, fonts, or content

**Specialized: Structural Similarity Networks (Custom Architecture)**

* **Use Case**: DOM tree structure comparison and form hierarchy analysis
* **Advantage**: Purpose-built for detecting structural mimicry
* **Implementation**: Identify identical field arrangements (username → password → OTP) across sites
* **Why Superior**: Detects structural patterns even when field labels or styling change

**2. Natural Language Processing Models**

**Multilingual Text Analysis**

**Primary: Qwen2-7B-Instruct**

* **Use Case**: Language vs content mismatch detection
* **Advantage**: Strong multilingual capability with instruction following
* **Resource Requirements**: Single 24-48 GB GPU or 4-bit quantization on 16-24 GB
* **Alternative**: Mistral-7B-Instruct for compact, fast inference

**Enhanced: Multilingual BERT**

* **Use Case**: Keyword detection across Hindi, English, and regional languages
* **Advantage**: Handles code-mixing patterns common in Indian web
* **Implementation**: Detect suspicious keywords like "खाता अपडेट करें" or mixed-language attempts
* **Why Superior**: Single model handles India's multilingual landscape vs separate language-specific models

**Language Detection**

**Primary: fastText Language Detection**

* **Use Case**: Identifying language mismatches indicating amateur/malicious construction
* **Advantage**: Fast, lightweight detection of inconsistencies
* **Implementation**: Detect when declared language doesn't match actual content
* **Escalation Path**: Route uncertain cases to Qwen2-7B for complex analysis

**Entity Recognition**

**Primary: Named Entity Recognition (Custom Financial Domain)**

* **Use Case**: Extract financial institution names and banking terminology
* **Advantage**: Trained on Indian financial institutions and terminology
* **Implementation**: Detect unofficial domains mentioning legitimate institution names
* **Why Superior**: Handles Indian institution name variations and common misspellings

**3. Visual Analysis Models**

**Screenshot Analysis**

**Primary: Qwen2-VL-7B**

* **Use Case**: Visual phishing intent detection from full-page screenshots
* **Advantage**: Excellent UI/screenshot understanding with OCR-like text reading
* **Implementation**: Determine if page resembles legitimate banking interface
* **Alternative**: LLaVA-1.6/LLaVA-NeXT-7B for strong web UI semantics
* **High-Accuracy Option**: InternVL-2-8B for state-of-the-art OCR + layout grounding

**Optimization Strategy**: Use pHash/SSIM for initial filtering, escalate uncertain cases to VLM

**4. Social Media & Content Classification**

**Social Media Monitoring**

**Primary: RoBERTa-base (Fine-tuned)**

* **Use Case**: Classify social mentions promoting fake login/credential harvesting
* **Advantage**: Fast, accurate binary/multi-label text classification
* **Implementation**: Identify posts urging login/payment on suspicious domains
* **Multilingual Alternative**: XLM-RoBERTa-base for Indian language coverage
* **Zero-shot Fallback**: Llama-3.1-8B-Instruct when labeled data is insufficient

**Code/Paste Analysis**

**Primary: RoBERTa-base (Fine-tuned)**

* **Use Case**: Classify paste/code-sharing content for credential bait and token theft
* **Advantage**: Handles noisy, technical text efficiently
* **Alternative**: Mistral-7B-Instruct for rapidly evolving rules without labeled data

**5. Behavioral & Time Series Models**

**Temporal Pattern Analysis**

**Primary: LSTM/Transformer for Temporal Behavior**

* **Use Case**: Analyze domain behavioral patterns over time
* **Advantage**: Detect complex temporal signatures of phishing campaigns
* **Implementation**: Identify rapid DNS changes, burst traffic, short-lived activity cycles
* **Why Superior**: Captures non-linear behavioral shifts vs traditional time series models

**Anomaly Detection**

**Primary: Isolation Forest/Autoencoders**

* **Use Case**: Detect statistical anomalies across multiple dimensions
* **Advantage**: Unsupervised learning detects novel attack patterns
* **Implementation**: Flag unusual combinations (new domain + premium SSL + foreign hosting claiming to be Indian bank)
* **Why Superior**: Works without balanced datasets of known phishing examples

**Infrastructure Analysis**

**Primary: Graph Neural Networks (GNNs)**

* **Use Case**: Analyze infrastructure relationships for coordinated campaigns
* **Advantage**: Identifies relationship patterns across entire infrastructure graph
* **Implementation**: Detect shared hosting, registrars, or certificate patterns
* **Why Superior**: Catches multi-domain campaigns that individual analysis might miss

**6. Analyst Support & Reporting**

**Narrative Generation**

**Primary: Llama-3.1-8B-Instruct**

* **Use Case**: Generate analyst-facing summaries explaining why domains were flagged
* **Advantage**: Controllable, faithful summaries from structured features
* **Implementation**: Convert detection results into coherent explanations
* **Alternatives**:
  + Mistral-7B-Instruct for fast, concise rationales
  + Llama-3.1-70B-Instruct for complex multi-feature explanations

**Implementation Strategy**

**Performance Optimization**

* **Quantization**: Use 4-bit quantization (GGUF or AWQ) for efficient GPU utilization
* **Triage Policy**: Deploy deterministic features first, escalate ambiguous cases to AI models
* **Resource Management**: Balance accuracy requirements with computational constraints

**Deployment Architecture**

* **Fast Screening**: fastText language detection, pHash/SSIM visual comparison
* **Medium Processing**: RoBERTa classification models, BERT multilingual analysis
* **Deep Analysis**: VLMs, large language models, GNNs for complex cases

**Quality Assurance**

* **Evaluation Framework**: Maintain labeled dev sets (300-1,000 cases) per detection type
* **Monitoring**: Track precision/recall across different sources (social, visual, behavioral)
* **Guardrails**: Use structured feature inputs and extraction-then-summary prompts to minimize hallucinations

**Scalability Considerations**

**Model Maintenance Requirements**

* **Minimal Maintenance**: CLIP, DINO (pre-trained models)
* **Periodic Retraining**: Custom NER, Structural Similarity Networks
* **Continuous Learning**: Behavioral models requiring new attack pattern updates

**Resource Allocation Priority**

1. **High-Impact Features**: Visual similarity, behavioral analysis (prioritize accuracy)
2. **Volume Processing**: Language detection, anomaly detection (optimize for speed)
3. **Infrastructure Analysis**: GNNs (computationally intensive but essential for campaign detection)

**Recommended Implementation Pipeline**

**Phase 1: Core Detection**

1. Deploy CLIP for logo similarity
2. Implement multilingual BERT for keyword detection
3. Set up RoBERTa classifiers for social media monitoring
4. Configure fastText for language mismatch detection

**Phase 2: Advanced Analysis**

1. Integrate Qwen2-VL for screenshot analysis
2. Deploy temporal behavior models
3. Implement anomaly detection systems
4. Set up infrastructure relationship analysis with GNNs

**Phase 3: Intelligence & Reporting**

1. Configure narrative generation with Llama models
2. Implement feedback loops for model improvement
3. Deploy comprehensive monitoring and evaluation systems

This comprehensive approach ensures robust phishing detection across all attack vectors while maintaining operational efficiency and scalability for enterprise deployment.