DYNAMIC CONFIDENCE SCORING SYSTEM FOR AI RESPONSE VALIDATION

PROVISIONAL PATENT APPLICATION

Inventor: Kinan Lemberg

Address: 270 Bolton Rd, Koah, 4881, Australia

Filing Date: June 3, 2025

FIELD OF THE INVENTION

This invention relates to dynamic confidence scoring systems for artificial intelligence validation, and more specifically to adaptive confidence threshold mechanisms that adjust validation requirements based on domain, context, and historical performance patterns for AI-generated responses.

BACKGROUND OF THE INVENTION

Current AI confidence scoring systems use static thresholds that fail to account for domain-specific requirements, contextual factors, or evolving model performance. Medical advice requiring 99% confidence is treated the same as casual conversation requiring 70% confidence. This one-size-fits-all approach creates both unnecessary restrictions and dangerous permissiveness.

Current limitations include:

- No domain-specific confidence threshold adaptation
- Lack of contextual confidence adjustment based on query criticality
- Absence of learning mechanisms from user feedback on confidence accuracy
- No mathematical framework for confidence score calibration across models
- Limited integration of historical performance in confidence calculations

There exists a need for a dynamic confidence scoring system that adapts thresholds based on domain requirements, learns from feedback, and provides calibrated confidence assessments across different AI models and use cases.

SUMMARY OF THE INVENTION

The present invention provides a dynamic confidence scoring system that automatically adapts confidence thresholds based on domain, context, and historical performance, implementing machine learning algorithms to continuously improve confidence calibration.

The invention comprises:

1. Domain-Adaptive Confidence Engine - Automated adjustment of confidence thresholds based on specific domain requirements and risk profiles.

2. Contextual Confidence Calibration System - Real-time confidence adjustment based on query context, user profile, and situational factors.

3. Feedback-Based Learning Module - Machine learning system that improves confidence scoring accuracy through user feedback integration.

4. Cross-Model Confidence Normalization - Mathematical framework for normalizing confidence scores across different AI models with varying scales.

5. Explainable Confidence Breakdown - Detailed confidence component analysis providing transparency in confidence calculations.

The system provides significant advantages by dynamically adapting confidence requirements to match real-world needs while continuously improving through feedback learning.

DETAILED DESCRIPTION OF THE INVENTION

System Architecture

The Dynamic Confidence Scoring System operates as an intelligent layer that calculates, calibrates, and adapts confidence scores based on multiple factors to provide accurate validation decisions.

1. Domain-Adaptive Confidence Engine

The Confidence Engine implements sophisticated domain-specific threshold adaptation based on risk profiles and accuracy requirements.

Domain Classification Framework:

```
Domain_Risk_Profiles = {
    critical_safety: {
        domains: [medical, aviation, nuclear, pharmaceutical],
        base_threshold: 0.99,
        error_tolerance: 0.001,
        false_positive_cost: 10,
        false_negative_cost: 1000
    },
    high_stakes: {
        domains: [legal, financial, engineering, security],
        base_threshold: 0.95,
        error_tolerance: 0.01,
        false_positive_cost: 50,
        false_negative_cost: 500
    },
    professional: {
        domains: [business, technical, academic, scientific],
        base_threshold: 0.85,
        error_tolerance: 0.05,
        false_positive_cost: 20,
        false_negative_cost: 100
    },
    general_use: {
        domains: [educational, informational, creative, general],
        base_threshold: 0.75,
        error_tolerance: 0.10,
        false_positive_cost: 5,
        false_negative_cost: 20
    },
    casual: {
        domains: [entertainment, social, personal, lifestyle],
        base_threshold: 0.60,
        error_tolerance: 0.20,
        false_positive_cost: 1,
        false_negative_cost: 5
    }
}
```

Dynamic Threshold Calculation:

```
Adaptive_Threshold = Base_Threshold × Context_Multiplier × Performance_Adjustment × Risk_Factor

Context_Multiplier = f(
    query_complexity,
    user_expertise_level,
    decision_reversibility,
    time_criticality,
    downstream_impact
)

Performance_Adjustment = {
    if recent_accuracy > 0.95: 0.98,
    elif recent_accuracy > 0.90: 1.00,
    elif recent_accuracy > 0.85: 1.02,
    elif recent_accuracy > 0.80: 1.05,
    else: 1.10
}

Risk_Factor = sqrt(
    (False_Negative_Cost × False_Negative_Probability) /
    (False_Positive_Cost × False_Positive_Probability)
)
```

Sub-Domain Specialization:

```
Medical_Sub_Domains = {
    diagnosis: {threshold: 0.995, require_evidence: true},
    treatment: {threshold: 0.99, require_alternatives: true},
    drug_interaction: {threshold: 0.999, require_verification: true},
    general_health: {threshold: 0.95, require_disclaimer: true},
    wellness: {threshold: 0.90, require_context: true}
}

Financial_Sub_Domains = {
    trading_advice: {threshold: 0.98, require_disclaimer: true},
    tax_guidance: {threshold: 0.97, require_jurisdiction: true},
    investment_strategy: {threshold: 0.95, require_risk_disclosure: true},
    budgeting: {threshold: 0.85, require_personalization: true},
    general_finance: {threshold: 0.80, require_education: true}
}
```

2. Contextual Confidence Calibration System

The Calibration System adjusts confidence requirements based on real-time contextual factors and query characteristics.

Contextual Factor Analysis:

```
Query_Complexity_Score = f(
   token_count,
   concept_density,
   technical_term_frequency,
   ambiguity_level,
   multi_part_structure
)

User_Context_Profile = {
   expertise_level: Domain_Expertise_Assessment,
   risk_tolerance: Historical_Risk_Preference,
   accuracy_requirements: Stated_Accuracy_Needs,
   decision_authority: Organizational_Decision_Level,
   historical_satisfaction: Past_Confidence_Feedback
}

Situational_Factors = {
   urgency_level: Time_Criticality_Score,
   reversibility: Decision_Reversibility_Assessment,
   impact_scope: Affected_Stakeholder_Count,
   regulatory_context: Compliance_Requirements,
   market_conditions: Environmental_Volatility
}
```

Real-Time Calibration Algorithm:

```
Contextual_Confidence_Adjustment = {
    base_score: Model_Reported_Confidence,
    complexity_adjustment: -0.01 × log(Query_Complexity_Score),
    expertise_adjustment: +0.02 × User_Expertise_Level,
    urgency_adjustment: +0.03 × (1 / Time_Available),
    impact_adjustment: +0.02 × log(Impact_Scope),
    volatility_adjustment: -0.01 × Market_Volatility_Index
}

Calibrated_Confidence = min(0.99, max(0.01,
    base_score + Σ(adjustments) × Calibration_Weight
))

Calibration_Weight = Historical_Calibration_Accuracy × Model_Trust_Factor
```

Multi-Dimensional Confidence:

```
Confidence_Components = {
    factual_accuracy: Fact_Verification_Confidence,
    logical_consistency: Reasoning_Chain_Confidence,
    completeness: Response_Completeness_Confidence,
    relevance: Query_Relevance_Confidence,
    safety: Risk_Assessment_Confidence
}

Overall_Confidence = Weighted_Harmonic_Mean(
    Confidence_Components,
    weights = Domain_Specific_Component_Weights
)
```

3. Feedback-Based Learning Module

The Learning Module implements sophisticated machine learning algorithms to improve confidence calibration through user feedback.

Feedback Collection Framework:

```
Feedback_Types = {
  explicit_feedback: {
    accuracy_rating: User_Provided_Accuracy_Score,
    confidence_appropriateness: Was_Confidence_Too_High_or_Low,
    outcome_report: Actual_Decision_Outcome,
    satisfaction_score: Overall_Satisfaction_Rating
  },
  implicit_feedback: {
    acceptance_rate: Response_Acceptance_Frequency,
    modification_rate: User_Edit_Frequency,
    query_refinement: Follow_Up_Query_Pattern,
    session_completion: Task_Completion_Success
  },
  behavioral_signals: {
    dwell_time: Time_Spent_Reading_Response,
    interaction_pattern: Click_Through_Behavior,
    return_rate: Repeat_Query_Frequency,
    escalation_rate: Human_Expert_Consultation_Rate
  }
}
```

Machine Learning Calibration:

```
Calibration_Neural_Network = {
  input_features: [
    raw_confidence_score,
    domain_category,
    query_complexity,
    user_expertise,
    contextual_factors,
    model_identity,
    historical_performance
  ],
  hidden_layers: [
    Dense(128, activation='relu'),
    Dropout(0.3),
    Dense(64, activation='relu'),
    BatchNormalization(),
    Dense(32, activation='relu')
  ],
  output_layer: Dense(1, activation='sigmoid'),  # Calibrated confidence
  loss_function: 'binary_crossentropy_weighted',
  optimizer: 'adam_with_warmup'
}

Online_Learning_Pipeline = {
  batch_collection: Collect_Feedback_in_Mini_Batches,
  incremental_training: Update_Model_Every_N_Samples,
  validation_set: Hold_Out_Recent_20_Percent,
  drift_detection: Monitor_Calibration_Drift,
  retraining_trigger: Significant_Performance_Degradation
}
```

Confidence Calibration Metrics:

```
Expected_Calibration_Error = (1/N) × Σ|Accuracy_i - Confidence_i|

Brier_Score = (1/N) × Σ(Confidence_i - Actual_Outcome_i)²

Reliability_Diagram = {
  bin_confidences: [0.0-0.1, 0.1-0.2, ..., 0.9-1.0],
  actual_accuracies: Observed_Accuracy_Per_Bin,
  perfect_calibration: Diagonal_Line,
  calibration_gap: Distance_From_Diagonal
}

Adaptive_Binning = {
  equal_frequency_bins: Ensure_Equal_Samples_Per_Bin,
  adaptive_boundaries: Adjust_Bins_Based_on_Distribution,
  domain_specific_bins: Different_Binning_Per_Domain
}
```

4. Cross-Model Confidence Normalization

The Normalization system ensures confidence scores are comparable across different AI models with varying calibration characteristics.

Model Calibration Profiles:

```
Model_Calibration_Characteristics = {
  gpt_4: {
        typical_range: [0.7, 0.95],
        overconfidence_factor: 1.15,
        domain_biases: {medical: -0.05, technical: +0.03},
        temperature_sensitivity: 0.8
    },
    claude_3: {
        typical_range: [0.6, 0.9],
        overconfidence_factor: 1.05,
        domain_biases: {legal: +0.02, creative: -0.03},
        temperature_sensitivity: 0.6
    },
    gemini_pro: {
        typical_range: [0.65, 0.92],
        overconfidence_factor: 1.10,
        domain_biases: {scientific: +0.04, financial: -0.02},
        temperature_sensitivity: 0.7
    }
}
```

Normalization Algorithm:

```
Normalized_Confidence = Isotonic_Regression_Transform(
    raw_confidence,
    model_calibration_curve,
    domain_specific_adjustment
)

Cross_Model_Mapping = {
    step_1: Z_Score_Normalization(raw_confidence, model_mean, model_std),
    step_2: Quantile_Mapping_to_Reference_Distribution,
    step_3: Domain_Specific_Adjustment_Application,
    step_4: Smooth_Interpolation_for_Continuity
}

Ensemble_Confidence = {
    weighted_average: Σ(Model_Weight_i × Normalized_Confidence_i),
    confidence_variance: Variance(Normalized_Confidences),
    agreement_bonus: Bonus_for_Low_Variance_High_Agreement,
    uncertainty_penalty: Penalty_for_High_Variance_Disagreement
}
```

Calibration Transfer Learning:

```
Transfer_Calibration = {
    source_models: Well_Calibrated_Model_Set,
    target_model: New_or_Poorly_Calibrated_Model,
    transfer_method: {
        feature_extraction: Extract_Calibration_Features,
        mapping_function: Learn_Source_to_Target_Mapping,
        fine_tuning: Adjust_with_Limited_Target_Data,
        validation: Cross_Validate_on_Hold_Out_Set
    }
}
```

5. Explainable Confidence Breakdown

The Breakdown system provides detailed explanations of confidence calculations for transparency and trust.

Confidence Component Analysis:

```
Confidence_Explanation = {
    primary_factors: {
        model_certainty: Raw_Model_Confidence_Score,
        factual_support: Evidence_Strength_Score,
        logical_coherence: Reasoning_Consistency_Score,
        domain_alignment: Domain_Relevance_Score,
        safety_assurance: Risk_Mitigation_Score
    },
    contributing_factors: {
        query_clarity: Question_Unambiguity_Score,
        knowledge_coverage: Topic_Knowledge_Depth,
        recent_performance: Model_Recent_Accuracy,
        consensus_level: Multi_Model_Agreement_Score,
        data_recency: Information_Currency_Score
    },
    detracting_factors: {
        ambiguity_present: Detected_Ambiguities,
        knowledge_gaps: Identified_Unknowns,
        conflicting_evidence: Contradictory_Information,
        high_complexity: Complexity_Penalty,
        domain_mismatch: Out_of_Domain_Penalty
    }
}
```

Visual Confidence Representation:

```
Confidence_Visualization = {
    overall_gauge: Circular_Confidence_Meter,
    component_bars: Stacked_Component_Contributions,

    uncertainty_bands: Confidence_Interval_Display,

    historical_trend: Time_Series_Confidence_Plot,

    comparative_view: Model_Comparison_Radar_Chart

}


Natural_Language_Explanation = {

    high_confidence: "High confidence (X%) based on strong factual support and consistent reasoning",

    moderate_confidence: "Moderate confidence (X%) due to [primary limiting factor]",

    low_confidence: "Lower confidence (X%) because of [key uncertainty factors]",

    uncertainty_disclosure: "Main uncertainties: [list of specific unknowns]"

}
```

System Integration and Performance

Confidence System Performance Requirements:

- Calculation Latency: < 50ms for confidence scoring
- Calibration Update: < 100ms for online learning update
- Normalization Speed: < 20ms for cross-model normalization
- Explanation Generation: < 200ms for detailed breakdown

Scalability Architecture:

- Distributed Calculation: Parallel confidence processing
- Model Caching: Calibration curve caching
- Incremental Learning: Online model updates
- Explanation Templates: Pre-computed explanation patterns

ADVANTAGES OVER PRIOR ART

The present invention provides significant advantages over existing confidence scoring approaches:

1. Dynamic Adaptation: Unlike static threshold systems, the invention adapts confidence requirements based on domain and context.

2. Continuous Learning: The system improves calibration accuracy through feedback rather than fixed confidence mapping.

3. Cross-Model Normalization: Enables meaningful confidence comparison across different AI models rather than model-specific scores.

4. Explainable Confidence: Provides detailed confidence breakdowns rather than opaque single scores.

5. Domain Specialization: Implements domain-specific confidence requirements rather than universal thresholds.

6. Contextual Awareness: Adjusts confidence based on query context and user needs rather than fixed criteria.

CLAIMS

Claim 1: A dynamic confidence scoring system for AI response validation comprising:

- a domain-adaptive confidence engine configured to adjust thresholds based on domain-specific risk profiles;

- a contextual confidence calibration system implementing real-time adjustment based on query and user context;

- a feedback-based learning module using machine learning to improve confidence calibration;

- a cross-model confidence normalization system ensuring comparable scores across different AI models; and

- an explainable confidence breakdown system providing transparent confidence calculations.

Claim 2: The system of claim 1, wherein the domain-adaptive confidence engine implements different thresholds for critical safety, high stakes, professional, general use, and casual domains.

Claim 3: The system of claim 1, wherein the contextual calibration system adjusts confidence based on query complexity, user expertise, urgency, and impact scope.

Claim 4: The system of claim 1, wherein the feedback-based learning module implements neural network calibration with online learning from explicit and implicit user feedback.

Claim 5: The system of claim 1, wherein the cross-model normalization uses isotonic regression and quantile mapping for confidence standardization.

Claim 6: The system of claim 1, wherein the explainable breakdown provides component analysis of factual support, logical coherence, and domain alignment.

Claim 7: A method for dynamic confidence scoring in AI validation comprising:

- classifying queries into domain-specific risk categories;

- calculating base confidence thresholds for identified domains;

- adjusting confidence based on contextual factors;

- normalizing confidence scores across different AI models;

- learning from user feedback to improve calibration; and

- providing explainable confidence breakdowns.

Claim 8: The method of claim 7, further comprising implementing sub-domain specialization with specific thresholds for medical diagnosis, financial trading, and other critical applications.

Claim 9: The method of claim 7, wherein confidence adjustment includes query complexity, user expertise, decision reversibility, and time criticality factors.

Claim 10: The method of claim 7, wherein learning from feedback implements expected calibration error minimization and reliability diagram optimization.

ABSTRACT

A dynamic confidence scoring system automatically adapts AI validation confidence thresholds based on domain requirements, contextual factors, and continuous learning from feedback. The system

comprises: (1) domain-adaptive confidence engine with risk-based thresholds, (2) contextual calibration adjusting for query and user factors, (3) machine learning module improving calibration through feedback, (4) cross-model normalization for comparable confidence scores, and (5) explainable confidence breakdowns for transparency. The system ensures appropriate confidence thresholds for different use cases while continuously improving accuracy, providing advantages through dynamic adaptation, continuous learning, cross-model comparison, and transparent explanations.

END OF PATENT SPECIFICATION