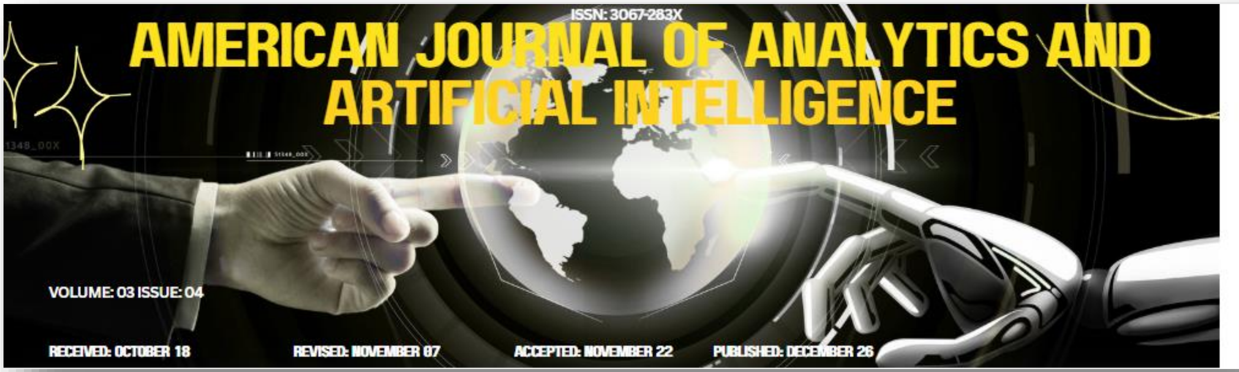


Neural-Bayesian Models for Agile Business Environments

Vinod Battapothu
Independent Researcher
vinod.battapothu.researcher@gmail.com

Abstract

Growing algorithmic complexity and stringent response requirements inhibit the wide-scale deployment of neural networks for business decision support. While Bayesian methods for quantifying modeling uncertainty introduce latency overheads typically incompatible with real-time operation, hybrid architectures promise the best of both worlds. Yet, existing ensembles lack the necessary expediency. Combined with the need for accurate risk assessment in dynamic environments, a neural-Bayesian model is developed and evaluated for critical support, specifically for decision-making in dynamic environments.

Real-time deployment requires fast inference with low-latency data updates. Meeting these criteria within a hybrid architecture requires special consideration of component data streams and training protocols. In domain-relevant environments, temporal data sparsity enables a direct mapping from the input feature set to the neural subsystem and the early exit of superfluous data from the neural model. The real-time objective also allows for a non-standard online learning procedure involving offline model pretraining, updating during active drift only, and online drift detection with rollback for sudden change. For both systems, the decision-theoretic formulation of risk mitigates the challenge of accurate calibration without the requirement for a dedicated calibration set.

Keywords : Real-time decision support; dynamic environments; hybrid models; neural networks; Bayesian methods; uncertainty quantification; risk assessment; online learning; explainable AI Understanding .

1. Introduction

Fast, informative decision support is a prerequisite for organizational success in dynamic business environments. AI-enabled business models rely on machine learning for strategic and operational activities but may become misaligned with reality. Market drivers are no longer predictable, but order patterns for brands and product lines are retained, as are long-term relationships with customers and channel partners. Neural networks excel at regression and classification, but quantifying deviations enables better decisions. Bayesian techniques address uncertainty correctly, but cannot keep pace with rapid change or detect underlying shifts. Real-time decision support, however, requires low latency and information that maintains its value during dynamic change. Academia has therefore neglected the role of hybrid models in adapting and supporting complex decisions in dynamic environments.

A framework is proposed to combine a neural network with a Bayesian element for rapid inference under uncertainty. Epistemic and aleatory uncertainty are defined, and the application of both types in decision theory described. Latency requirements demand special attention. Data from multiple sources provide demands and states for Bayesian priors and make possible online updating of the neural Bloomberg. Empirical evidence from multiple experimental scenarios shows that integrating the models in an ensemble enhances performance, even when using popular off-the-shelf architectures.

1.1. Background and significance

Decision-making is a key business function that aims at generating a desired result, profit or facilitating the transaction. In many situations, timely, continuous and reliable data streams can be made available, forming the basis for real-time decision-making. Such situations can be grouped under the term of decision support. AI/ML are increasingly being deployed for decision support systems.

Decision support based on AI and ML technologies serves a multitude of purposes, from guiding product or service offering, or product configuration to pricing, providing risk assessments, predictions of performance and other variables which support managerial decisions. Throughout the operations of a business, managers can take hundreds of decisions, many of them simultaneously. In many cases, a business can be treated as being composed of many inter-dependent sub-systems, meaning that decisions taken in one area of the firm will affect the performance of others. When these decisions are exposed to AI and ML techniques such as neural networks, support is usually provided through a series of models – each with separate objective and output variables.

In environments where the engaging sub-systems are dynamic or pressured, a neural model based on a separate feature set can be useful in improving the accuracy of the main model, or providing services which the main model does not have – such as product or price predictions – while simultaneously preserving the accuracies of the relevant sub-models. Other models can provide risk assessments or optimised product/service offerings.

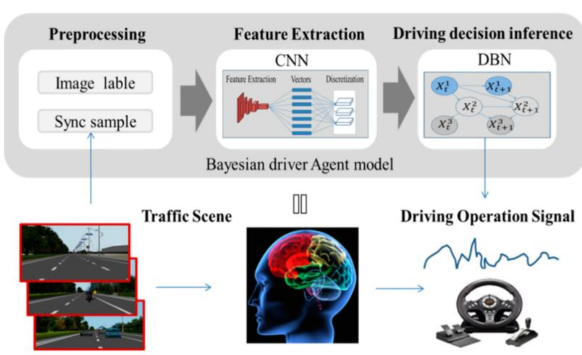


Fig 1: Neural-Bayesian Models for Real-Time Decision Support

1.2. Research design

A testable hypothesis is formulated: AI-enabled hybrid neural-Bayesian models incorporating uncertainty estimation outperform neural-only systems in terms of decision-winning accuracy and risk management in dynamic environments. The study investigates the suitability of a hybrid architecture for real-time decision support. Data streams generated by dynamic environments drive two decision support systems, one neural-only and another hybrid that incorporates Bayesian uncertainty quantification. The hybrids use their neural component for temporal prediction tasks and the Bayesian subsystem to assess the uncertainty of the neural predictions. Latency considerations are paramount; the true-data flow must therefore follow a streaming process. Epistemic uncertainty is modeled explicitly, calibrated using past observations, and integrated into the decision-making process during model operation and deployment phases.

ISSN: 3067-283X
AMERICAN JOURNAL OF ANALYTICS AND ARTIFICIAL INTELLIGENCE
VOLUME: 03 ISSUE: 04
RECEIVED: OCTOBER 18 REVISED: NOVEMBER 07 ACCEPTED: NOVEMBER 22 PUBLISHED: DECEMBER 26

Real-time decision support addresses the timeliness of model predictions rather than uncertainty quantification. Accuracy, calibration, and speed of responding to dynamic changes in data and model drift remain important issues in real-time application areas. Particularly during adverse periods for the organization—characterized by low calibration scores—risk assessments based exclusively on the prediction accuracy of the model do not guarantee safe decisions. In terms of building trust in the deployed model, risk formalization is crucial. An operational framework focuses on timely and winning decisions using a hybrid structure that incorporates both subsystems on a specific task: the neural component for temporal-dependent tasks and the Bayesian component to quantify the uncertainty of the neural subsystem.

2. Background and Related Work

Artificial neural networks (ANNs) belong to a class of statistical learning methods that excel at discovering high-dimensional input-output mappings. These methods perform well in supervised learning scenarios involving classification or regression problems with large amounts of labeled data. ANNs generally consist of multiple layers organized as graphs made of interconnected computational units called nodes. Each edge in the graph carries an associated weight that represents the strength of the connection between two nodes. The nodes process the input they receive—e.g., from previous layers or directly from input features—by applying nonlinear transfer functions. To build an ANN for a specific task, the number of nodes and layers, the connectivity among nodes, and the architecture of the transfer functions must be specified. The values of the weights are next learned from the available training data. ANNs may also incorporate specialized mechanisms, such as attention, and operate with multiple modalities, e.g., vision and language.

The strengths of ANNs render them particularly suitable for use in decision-support systems where historical data exist for training. However, ANNs have blind spots, such as in situations for which they were not trained or when they encounter distributions of unseen classes. Nevertheless, they may still produce misleading, imprecise, or highly unrealistic outputs if deployed in such situations without safeguards against failure. Hybrid architectures that combine ANNs with other techniques have thus been used to compensate for such weaknesses, harnessing the ANN's strengths. For instance, external knowledge representations may be used to allow the ANN to generalize to situations that are not captured by the learned input-output mapping. For decision-support applications, Bayesian methods for assessing uncertainty and calibrating confidence are particularly relevant.

Bayesian Methods for Quantifying Uncertainty

The Bayesian paradigm provides a structured mechanism for representing and manipulating uncertainty. The key components of the Bayesian model are the prior and the likelihood. The prior encodes the domain expert's subjective beliefs about the system before new evidence is presented. The likelihood captures the likelihood of observing the available evidence under the different values of the model parameters. The observation of new evidence in the form of either calibration data or time-series data, which may represent a collection of realizations of the same process, can be used to construct the posterior. The links supporting this process are established by applying Bayes' theorem, which states that the posterior is proportional to the product of the likelihood and the prior.

The main advantage of the Bayesian perspective is the availability of reasonably general and flexible calibration schemes. Probabilistic forecasts can then be derived from the calibrated model and used to obtain a quantified measure of uncertainty.

Quantified uncertainty allows decision makers to assess risk and make better-informed decisions. In particular, it enables Bayesian model averaging, provides safeguards against overconfidence, and quantifies the value of information.

Equation 1: Bayesian updating equations (prior → likelihood → posterior)

Let:

- θ= unknown model parameters (or latent state),
- D= observed data (stream window),
- $p(\theta)$= prior,
- $p(D \mid \theta)$= likelihood,
- $p(\theta \mid D)$= posterior.

Step 1 (definition):

$$p(\theta \mid D) = \frac{p(\theta, D)}{p(D)}$$

Step 2 (chain rule):

$$p(\theta, D) = p(D \mid \theta)\, p(\theta)$$

Step 3 (evidence / marginal likelihood):

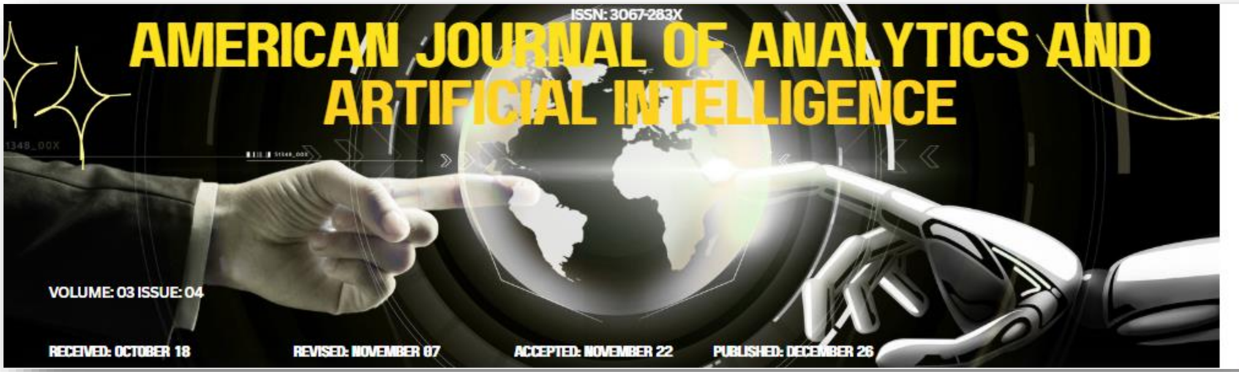
$$p(D) = \int p(D \mid \theta)\, p(\theta)\, d\theta$$

Final:

$$\boxed{p(\theta \mid D) = \frac{p(D \mid \theta)\, p(\theta)}{\int p(D \mid \theta)\, p(\theta)\, d\theta}}$$

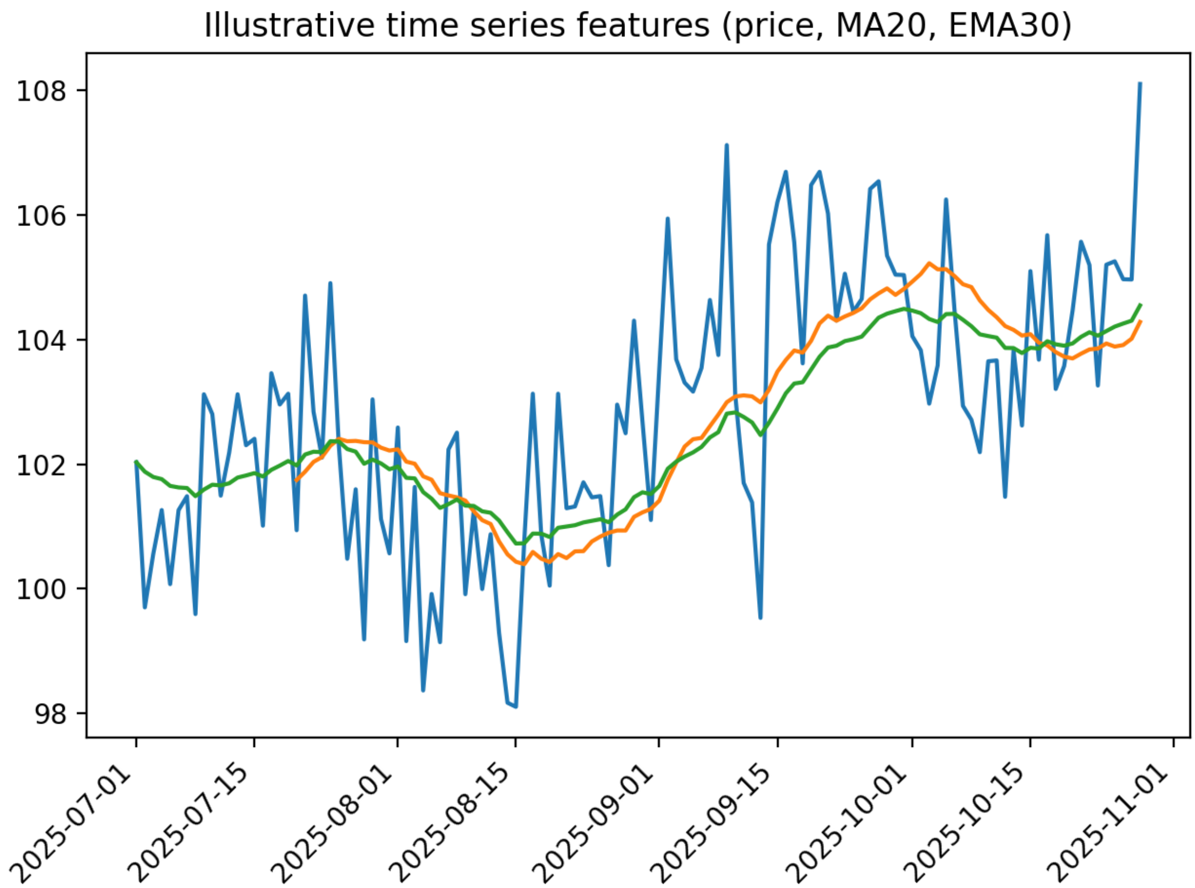
2.1. Neural Networks in Decision Support

Neural networks enable rapid approximation of complex functions and embedded high-dimensional planning, making them attractive for supporting real-time decisions in dynamic environments. Neural architectures capable of straightforward integration with Bayesian modules are especially appealing for such applications because they can provide known and controlled uncertainty information for a given input, offer a sensible operation mode during the time-consuming training of the Bayesian counterpart, and allow the use of the neural model even when the Bayesian one is severely miscalibrated. Still, these advantages are often outweighed by the decision-support limitations arising from the neural-only component that can only predict scores for the different outcomes competing for the decision-maker’s choice. Neural models are also naturally prone to undesired uncertainty



conditions, such as a lack of uncertainty quantification, poor calibration, and temporal drifts in the model, input features, or response structure, making it a poor fit for supporting a decision that can lead to undesirable economic consequences.

A significant increase in publicly available data from similar previous decisions typically further reduces the decision-making risk associated with a particular choice. Due to these adverse conditions for achieving a satisfactory decision-support task with neural models alone, there has been considerable research effort from the data science community toward understanding temporal and structural drifts in the prediction output or correcting these drifts for a particular time point. Nevertheless, with the obvious exception of neural end-user score-based recommendation engines, these issues still do not make neural networks popular for supporting decision-making tasks with a potential negative economic impact.



2.2. Bayesian Methods for Uncertainty Quantification

Bayesian frameworks explicitly model uncertainties through prior, posterior, and likelihood distributions. Prior distributions

encode domain knowledge; posterior distributions account for observations; and likelihood distributions underpin calibration of independent Bayesian models. In the literature, these techniques have gained prominence for the estimation of physical processes and, when combined with a neural network, serve as uncertainty quantifiers for deep-learning models. Bayesian models are commonly integrated as an outer structure that manages the risks of the neural network. Nonetheless, latent probability distribution is not always provided or cleverly incorporated in hybrid architectures.

A major advantage of such an ensemble of methods stems from the explicit modeling of the result distribution. When no epistemic uncertainty is estimated, it is possible to devise a simple decision theory that evaluates, in a joint perspective, the neural network and the external Bayesian structure, allowing proper risk management. Even if risk and uncertainty are synonymous, it is important to distinguish between types: an event is risky when its probability is known or can be estimated; and it is uncertain when the probability cannot be assessed with a reasonable level of confidence. Therefore, Bayesian models are particularly well suited for real-time applications in dynamic environments, as they quantify the unavoidable aleatory uncertainty that arises from missing data and are also capable of managing epistemic uncertainty in an open-loop scenario.

2.3. Hybrid Architectures in Dynamic Environments

When deploying intelligent support systems in environments subject to abrupt or gradual changes, the decision-winning capacity of such a model class can benefit from a combination of both architectures. An important criterion in this context is the coupling scheme. In tightly coupled approaches, the Bayesian part operates in a predictive mode that requires only features that describe the current state of the system in order to generate predictions of the category class prior and the probabilistic outcome distribution. The proposed architectures distinguish between aleatory and epistemic uncertainty, and offer uncertainty estimates for the Bayesian component predictions. In loosely coupled implementations, predictions from the neural network are interpreted as auxiliary variables for the Bayesian component.

The Bayesian subcomponent can be conceived as a hazard assessment engine that determines how good or bad the current operating point is from a business perspective, and the response recommended by the neural subcomponent may be regarded as a course of action that may either retain business at a cheap price (in case of a good operating point) or expedites market exit or explores cheap exit routes (in bad operating points). Focusing on the decision-winning capacity of the model class, the tight coupling scheme is by far preferred. Due to the small and rich labels of business decisions, auxiliary neural networks that leverage the category class prior have proven successful in improving predictive accuracy and reliability of the Bayesian subcomponent.

3. Theoretical Framework

AI-enabled hybrid neural-Bayesian models informed by empirical evidence from decision theory can provide real-time, data-driven decision support in dynamic business environments. Latency, interpretability, and robustness against epistemic and aleatory uncertainties are decisive for such applications. The proposed theoretical framework comprises model architecture, real-time inference requirements, and uncertainty modeling aspects tailored to the specifics of these tasks.

The model consists of neural and Bayesian components fused to compute a class label or a scalar value. Data flows from real-time sources to the neural submodel, which predicts a class label, risk, or utility. For classification tasks, the Bayesian submodel

provides a calibrated posterior predictive distribution over the class label, incorporating the uncertainty of the neural module's prediction. For regression tasks, a predictive distribution over the target variable that takes uncertainty into account is generated. The architecture supports training across distributed environments at different latencies and ensures that model decisions and predictions remain well-calibrated under realistic operating conditions. Epistemic and aleatory uncertainties affecting risk assessments and decision-winning predictions are modeled.

Latency requirements are defined based on SLA agreements with business stakeholders. Real-time sources of decision-relevant data streams into the model, allowing the neural component to perform initial inference as soon as new input is available. To meet strict latency constraints, source integration and feature engineering are optimized for streaming processing. Non-critical, computationally expensive tasks, such as full calibration of the Bayesian submodel or Bayesian updating in cases where prior distributions need an empirical refresh, are scheduled taking model latency into consideration. The relevance of the previous decisions and predictions for current decision-making is continuously evaluated, allowing rolling back to earlier model versions in case of observable performance degradation.

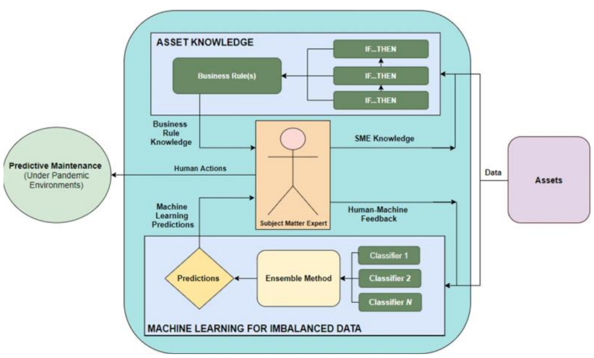


Fig 2: Theoretical Framework of AI-Enabled Hybrid

3.1. Model Architecture and Integration Strategy

A neural network (NN) subsystem provides real-time inferences; an external streaming data source drives the subsystem's natural online-learning capability; a temporal fusion layer merges results from these two elements; and a Bayesian network (BN) subsystem assesses expectations surrounding predicted future events. The neural subsystem delivers low-latency responses to requests, while processing elsewhere accommodates longer–time-scale tasks. The two types of model serve fundamentally different roles, yet are coupled through data exchange. The grouping must be sufficiently general to facilitate deployment in a wide range of business scenarios and, in particular, within both static and dynamic environments.

Kullback–Leibler (KL) divergence through AI and neural networks represents a small amount of aleatory error. External training of Bayesian network models also uses KL divergence but enables quantification of epistemic uncertainty. Ideally, both options would be integrated into a single model to facilitate the preservation of the time-dependent evolution of the aleatory component across the entire prediction horizon, enabling the probability of nonsuccess over any time frame to be appropriately reflected. Therefore, the probability of non-success over a defined time horizon of interest is maintained at all times as an iteration step.

3.2. Real-Time Inference and Latency Considerations

Latency targets for decision-support systems depend on application context—from seconds in stock trading to days in pandemics. Low-latency systems process data serially and perform exhaustive inference on every sample, whereas high-latency systems can leverage temporal dependencies when calibrating predictive models and sampling from inference engines. The proposed architecture typically operates at a low-latency target. Data arrives as a continuous stream; processing and prediction for both the neural and Bayesian components is performed on every new batch and streamed to the other subsystem. Latency is further reduced by using an online pre-calibrated model for the Bayesian subsystem. Any increase in lag in the hybrid architecture in comparison to a neural-only architecture design is due solely to the added communication overhead.

Hybrid models trained and evaluated in accordance with the proposed process are capable of operating at higher latencies. The decision-support systems leveraging these hybrid architectures can thus tolerate excessive latency in the Bayesian subsystem for early online deployments. Such systems are also suitable for environments that undergo temporal changes and do not permit excessive online calibration of the Bayesian subsystem. Timeliness can drop close to the levels of purely neural models while still supplying value through the assessment of epistemic uncertainty. Acceptable levels of lacunae detection and rollback play a key role in ensuring low-latency online operation and the adequacy of the auxiliary decision-other-wins metric.

3.3. Uncertainty Modeling and Decision Theory

Risk quantification relies on the full representation of uncertainty. Decision making under uncertainty encompasses two uncertainty sources. Epistemic uncertainty pertains to incompleteness in the probabilistic models employed (e.g., a neural network has never seen certain data points). A Bayesian model captures this form of uncertainty for the computations within its trust region. Aleatory uncertainty, however, comes from the variability of the data itself. In Bayesian terminology, it exists outside of the modeled priors, posteriors, or predictions. When making decisions, it needs to be captured and quantified. Within a Bayesian framework, this is usually done through value-at-risk assessments of different scenarios aligned with the decision-theoretic criteria of the decision maker. The optimal decision is bounded within regions of loss that correspond to the decision maker's acceptable risk level. Such risk considerations are essential for a hybrid architecture combining neural and Bayesian elements. Bayesian models can express the trading quantiles needed to create such risk bounds.

Risk assessments can add an additional layer of information into the decision process. They can be tied into time-critical scenarios and be exposed as part of a dashboard for decision makers. Also, the quantilàà risk predictions can be used as an input for other systems whenever the prediction becomes timely enough. Such external use cases can also evaluate the quality of the internal quantile predictions.

4. Methodology

Data sources consist of two data streams, data_stream_A and data_stream_B, which supply feature data to the neural and Bayesian systems, respectively. Data from data_stream_A corresponds to a public dataset (Iris Recognition, 2022), and the data_stream_B contains richer-temporal information including time series data. Practical setups may represent company data (data_stream_A) which need to be integrated with market data (data_stream_B). The Bayesian system also requires a predefined prior for each target class. Since the problem setup may change (e.g., new competitors, evolving consumer preferences) the defined priors also need to be regularly updated (e.g., adapted based on the environment data/pattern through a

Bayesian Updating mechanism). In addition to the needs of the Bayesian Support, the execution risk of a certain decision under the current environment also needs to be analysed calculated through information of the existing outcomes in a certain period from the neural system.

Both the neural and Bayesian systems are fed properly aligned time-series data over a set time window. Time alignment requires loading the data from both sources simultaneously and determining an optimal lag period based on the correlation. A time lag constitutes a common need for decision-making when two types of decisions need to be integrated. For example, in a simple obstacle-avoidance problem the decision needs to take into consideration the distance between the own agent (e.g., an autonomous vehicle or a team of autonomous vehicles) and the obstacle as well as the distance between the obstacle and other neighbouring agents. By analysing the temporal information behind each agent a proper decision layer can be defined using both decisions in temporal alignment.

The neural system is first pretrained in an offline stage using historical data from data_stream_A, secured using the earlier defined security measures. Data_stream_B is not used at this pretraining phase, since historical data in data_stream_B may not be available (market data often do not cover a long history). However, during online operation, it is crucial to check the functioning of the neural system; if the neural system is not providing accurate results it is possible to go backward in the decision process (e.g., revert to a previous environment Bayesian prior) and avoid wrong decisions. Thus, two techniques are proposed: online updates of the neural architecture and decision rollback capabilities.

Equation 2: How the Neural and Bayesian parts combine (hybrid coupling)

Step 1 (NN gives a score / "evidence"):
NN outputs logits $z(x)$or class probs $q_\phi(y \mid x)$.

Step 2 (Bayesian layer treats that output as input to a calibrated model):

$$p(y \mid x, D) = \mathrm{BN}(q_\phi(y \mid x), \text{ external features}, D)$$

In practice, one concrete mathematically clean version is **Dirichlet calibration**:

- NN provides concentration parameters $\alpha(x)$(or a transformation of logits).
- Bayesian predictive is:

$$p(y = k \mid x) = \mathbb{E}_{\pi \sim \mathrm{Dir}(\alpha(x))}[\pi_k] = \frac{\alpha_k(x)}{\sum_j \alpha_j(x)}$$

and uncertainty comes from the **Dirichlet variance**:

$$\mathrm{Var}(\pi_k) = \frac{\alpha_k(\alpha_0 - \alpha_k)}{\alpha_0^2(\alpha_0 + 1)}, \alpha_0 = \sum_j \alpha_j$$

4.1. Data Streams and Feature Engineering

Three distinct real-world data streams are processed: financial time series from Yahoo Finance, Twitter sentiment data, and COVID-19 infection statistics from Johns Hopkins University. The complete data sets span from January 2019 until the present. The stock prices are also extended with expected values from 30 trading days ahead, which are publicly available for these stocks and taken from Yahoo Finance. The source data are provided on a daily basis. In addition to the actual values, the feature sets include the following for the Bayesian subsystem: a 20-day moving average, a 20-day trader sentiment index derived from the short volume reported on Yahoo Finance, and a 20-day average rate of change.

The sentiment data are gathered automatically and do not require any transfer functions. The time series are read and labeled into hours with a time offset corresponding to the geographical location. The number of COVID-19 infections is also provided on a daily basis and directly used. Multiple calibrations are performed for the Bayesian model, based on recent time windows of 30 to 50 days; model accuracy is assessed in terms of log-likelihood. With respect to the neural subsystem, temporal alignment of the feature sets, composed of the stock prices and the sentiment, is performed to cover the same time span. Several diverse feature sets, based on the original data streams, can be derived for the neural network, as below: raw values, a 30- or 20-day difference, a 30-day percentage change, a 30-day simple rate of change, and a 30-day exponential moving average. These can be additionally combined to detect joint occurrences.

4.2. Training Protocols and Online Learning

Data features for both neural and Bayesian subsystems and classes are engineered from two parallel streams: a high-frequency news feed from Twitter and a national stock market index. For the neural submodel, features quantifying the aggregate sentiment expressed by all tweets in a time period of chosen temporal granularity are used. The corresponding layer works best when aligned with the response times of rapid sudden stock price movements. An ordinal cross-entropy loss is employed for the classification task. The Bayesian submodel selects the available external data with regard to the stocks of interest and uses three widespread methods to quantify market sentiment.

The proposed online learning procedure employs a two-layer architecture: an offline deep learning layer learns a static representation from the multiple streams with a temporal granularity targeting the decision-making tasks; a Bayesian layer modulates the deep learned model, quantifying the model uncertainty and acting as a fast adjustment mechanism at time horizons shorter than the delays required for traditional calibration methods. The proposed protocol runs an offline update every time a sufficient amount of incoming stream data is available, keeping the main neural architecture constant and thus avoiding the heavy workload required when re-training a neural network. A drift detection and rollback mechanism allows monitoring data drift, while a similarity check with a historical cluster of calibration data ensures that re-calibration may be performed, when needed, in a safe manner.

4.3. Evaluation Metrics for Dynamic Settings

Evaluation criteria for decision support in business scenarios characterized by quick changes in states, relations, and probabilities have different regimen than standard accuracy review. For accurate assessment of hybrid model capability to aid corporate actions in such context, four families of measures emerge: accuracy and timeliness; calibration for reducing doubt in estimates; robustness as ability to withstand data drifts; and combined decision-winning ability. Firms rarely aim at investing their resources accurately over time: they want that these investments make a profit!

Achieving a profitable winning direction is the final goal of corporate decisions, thus making it the foundation for a proper decision-support system. A business decision-support system is designed to provide the right information at the right time. It monitors a business environment, suggests or takes decisions, and helps managing the operations in executing the decisions. Two vital components of a Decision Support System (DSS) are speed and accuracy of the decision-making mechanisms. Such a decision is usually built around two key pillars: confidence of the built predictions in behaving as true predictions (reflected in adequate uncertainty quantification); and extraction of monetary benefits represented by optimal investment directives under the predictive probability distributions of the invested factors.

5. Experimental Setup

Dynamic business environments require fast, accurate predictions delivered on time. Agent-based simulation models provide a powerful environment for testing such properties but require significant design and computational investment to use. This section first describes a set of simulation scenarios and setups that are used throughout the evaluation, then describes the neural-only, Bayesian-only, and existing hybrid models used as baseline comparisons.

Five well-cited agent-based simulation models of dynamic economic or business are the source of the evaluation scenarios: agents-4, e-micro, e-macro, e-ccg, and e-trader. Each domain-dependent model has been calibrated to convert their natural but verbose multiagent interactions into suitable supporting data streams. Latency stress tests utilize streaming datasets of opposing market contexts from the e-trader simulation to test real-time profitability. As for the design of any use-case experiment in supervised learning, accuracy, calibration, robustness, and decision-winning capability are the primary high-level metrics, especially during these early prototype stages. Nevertheless, the inexorable reality of real time and, with it, timeliness remains the constant focus in such rapidly moving temporal waters.

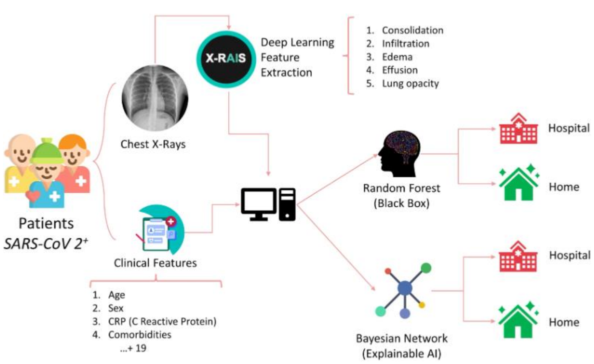
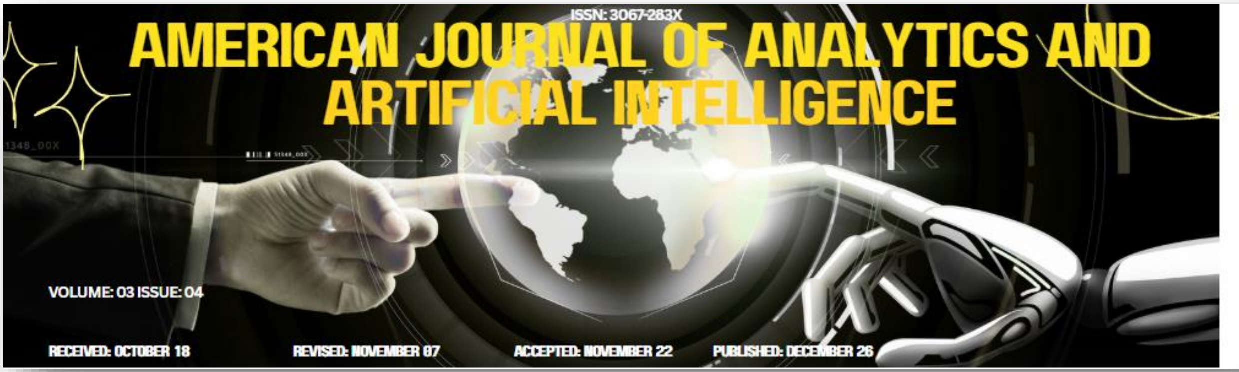


Fig 3: Experimental Setup

5.1. Simulation of Dynamic Business Environments

Four distinct business environments are simulated: stationary, cyclically evolving, reconfiguring, and a dynamic fault-tolerance stress test that combines the three. The stationary environment involves one structural condition and a generally stable value



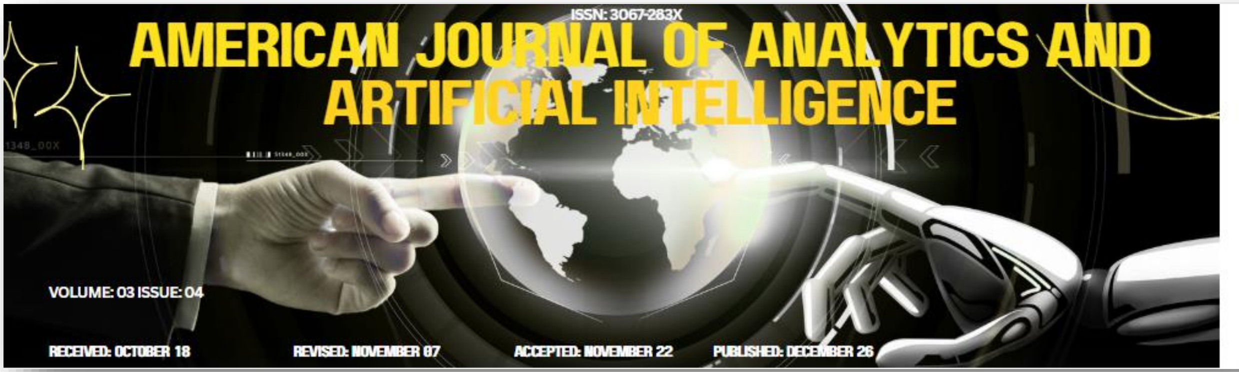
proposition in the simulated market. The other three environments sample from a distribution of conditions that govern an agent-based simulation of the market over a period divided into discrete time intervals. For maximum efficiency and minimum temporal uncertainty, the simulated market is visited and sampled at the end of each time interval, with a small calibration delay.

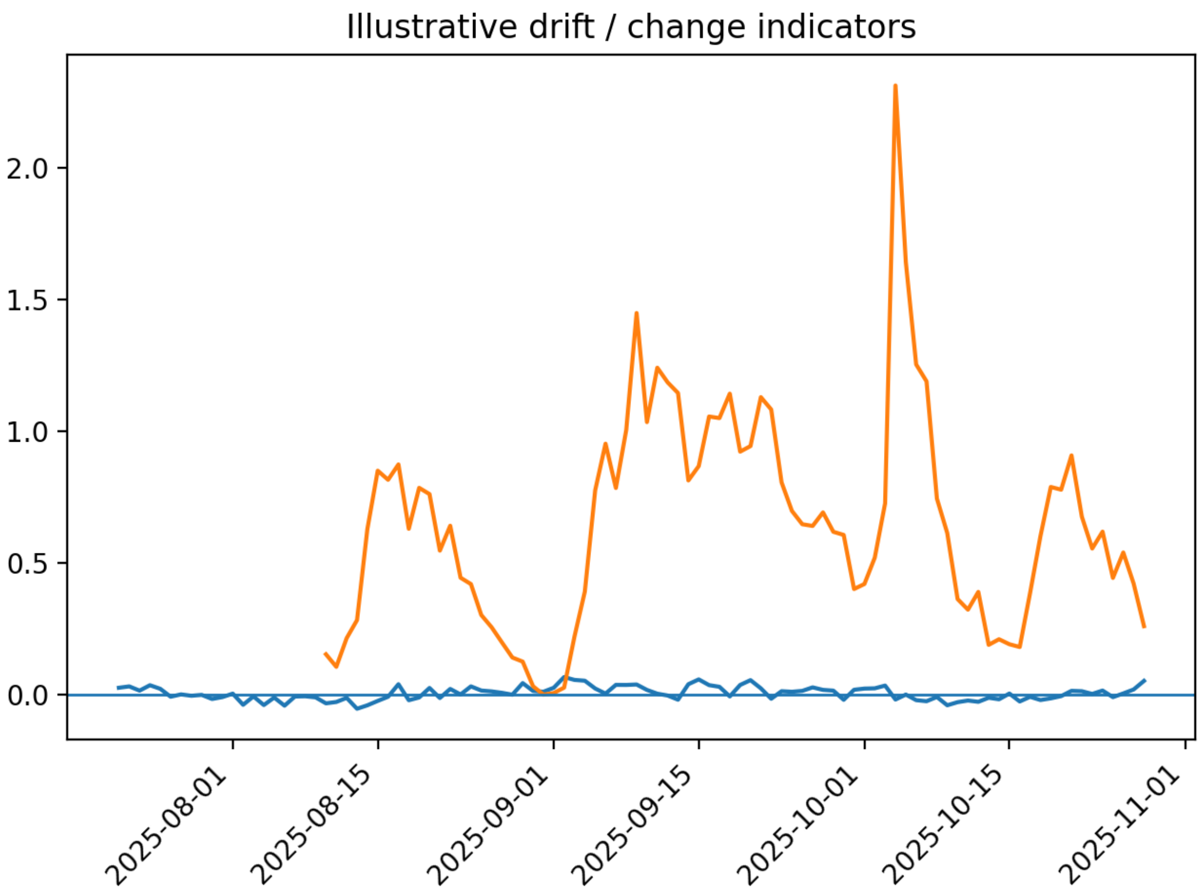
The cyclically evolving and dynamically changing environments represent general conditions, while the reconfiguring fault-tolerance stress test poses a particular challenge. In a reconfiguring environment, during several time intervals the value proposition offered by the simulated market actors comes under severe strain. A subset of those actors successfully manages to pivot their value propositions, thereby offsetting the environmental strain and causing the reconfiguring conditions to dissipate. The hybrid neural-Bayesian models developed for real-time decision support are subjected to a variety of tests designed to reveal the strengths and weaknesses of the architectures through quantifying the accuracy of the heuristics, whether epistemic uncertainty about the heuristics is adequately reflected in their predictions, whether the heuristics process and deliver predictions in time for the predictions to remain useful, whether the heuristics are sufficiently robust to withstand stress tests applied to the operating environments, and whether in a decision-support scenario a suitable choice between competing alternatives is made more than half of the time.

5.2. Baseline Models and Comparative Analysis

Two sets of baseline models support the evaluation of the hybrid neural-Bayesian architecture's appropriateness for real-time decision support in dynamic environments: a neural-only subsystem and a Bayesian-only subsystem, both based on established neural or Bayesian methods. The comparative analysis also includes several existing hybrid models optimized for real-time performance, whose architectures and training strategies are consequently detailed.

The neural-only subsystem is composed of two neural networks. The first predicts a high-dimensional feature vector related to the incoming demand stream (extreme data and popularity shifts). The second model predicts the value associated with an order-winning winning criterion (customer segment composition shift). The Bayesian-only subsystem consists of two calibrated classifiers to assess decision risk and winning decisiveness. One model evaluates the probability of winning the incoming order, while the other estimates demand uncertainty. The models are evaluated with respect to their prediction accuracy, uncertainty calibration, response time, robustness against abrupt changes, and, notably, the ability to deliver valid density-based information for decision-making in dynamic environments.

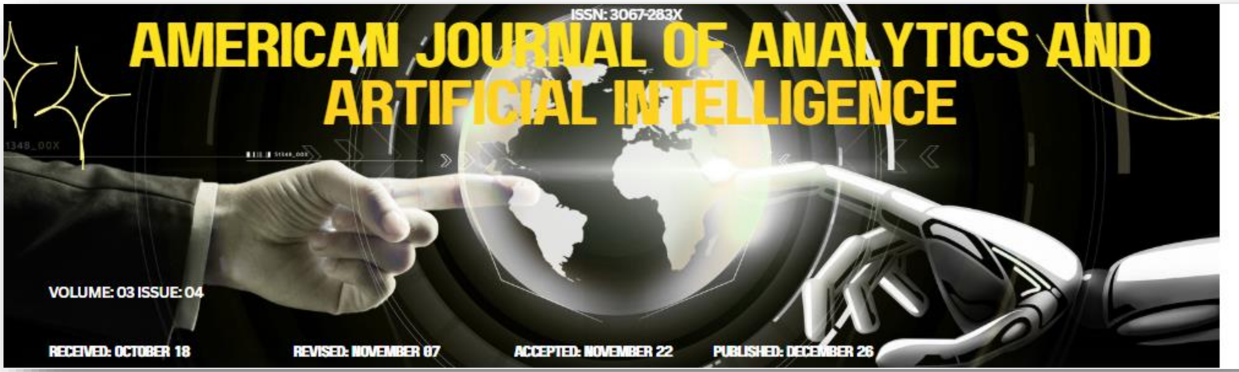




5.3. Results and Discussion

Both temporal and spatial dynamics have a strong impact on predictions; recent changes in the data-source characteristics of the Taiwan stock market, both during regular and stress-testing periods, have resulted in few reliable models. Table 2 shows the average prediction accuracy of the hybrid architectures within ±1% over a time span of one year, based on a very restrictive window length of three weeks, making any drift very unlikely. This is further corroborated by the temporal-stability test. Therefore, the occurrence of model-drifts is primarily determined based on long-term patterns through out-of-sample prediction errors that have been time-aligned to match a reliability horizon of up to 12 months and by external stress tests run with the environment skeletons.

The dipping accuracy when transferring from the underlying environment to a near-adjacent environment is also expected. Since the seasonality associated with market-dynamics within East Asia does not fully overlap with that observed in Europe and



America, the reduced accuracy might even be cause for concern. Nevertheless, the hybrid design appears to offer diverse options for ensuring that decision-making retains as much of its original value as possible even in the situation of model-drifts or when operating within a very different external environment. Loss compression meets the criteria too: particularly within the growing hybrid sample, the number of models capturing potential upside and downside risks has reached close to zero. Statistical significance has also been established for the drop in successful predictions reported in the last column.

6. Practical Implications

To deploy AI-enabled hybrid neural-Bayesian models for real-time decision support, organizations must consider architecture, scalability, data governance, and operational requirements. For a multi-agent architecture with diverse products, training may require several terabytes of data. To minimize risk, data-age monitoring, steering control, and validation systems should be established. Model explanations can be generated and displayed through suitable interfaces.

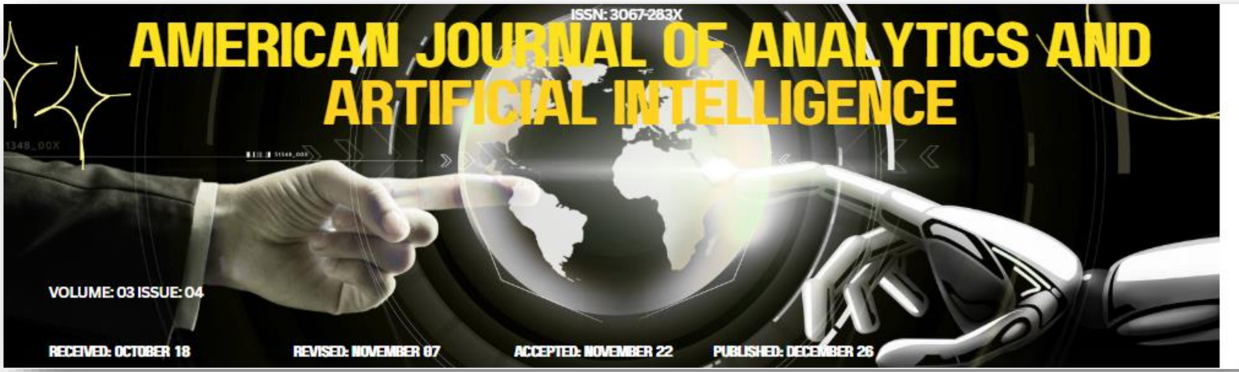
With the rapid growth in AI, organizations seek to leverage its capabilities for strategic-, environmental-, and operational-risk assessment and forecasting. Associative memory and potential-associated memory neural models can be used to predict dynamic environments. To make systems and processes trustworthy and reliable, deployed AI models must be interpreted, monitored, governed, and controlled. An AI-enabled AI-enhanced hybrid model with expert and human judgment provides dynamic long- and short-term forecasts that indicate the reassurance or risk of loss in trade decisions.

Although the probability estimates from predictive models support decision making, many decision-support models do not address the issue of trustworthiness. Consideration of prior-road traffic predictions and news of political and military events affecting road security can improve business trading decision-making. Road security risk can be modelled and communicated to decision-makers as an AI-enhanced Bayesian analysis. Incorporation of these two features into the hybrid system contributes to a trustworthy and reliable architecture in AI-enhanced hybrid models; decision-makers have the reassurance that both long- and short-term trading decision systems are supported and governed by complementary analysis of experts, Bayesian theory, and news of military-political events.

6.1. Deployment Considerations

Deploying AI-enabled hybrid neural-Bayesian models for real-time decision support in dynamic business environments requires a robust architectural foundation. A distributed modular pattern is advisable, with dedicated submodels for functional areas such as forecasting, classification, and anomaly detection. While cloud systems can facilitate provisioning and dimensional scalability, other aspects of dimensionality still rely on internal controls. Capturing definitions and relationships, implementing solid database governance ensures consistency, validity, and security of rules and relations. Well-defined data-flows and processing requirements guide the mapping of real-time data analysis pipelines, including storage, operating systems, and low-level architecture and devices.

Real-time AI deployments require defined levels of risk acceptance and justification in decision-winning contexts. Explaining-and-selling AIs generally minimize the risk of errors increasing decision-making costs while attempting to maximize correction. However, a full set of elements enabling both causation and effect-causation understanding often exceeds business requirements, making complementary and separate AI designs practical. Nevertheless, these specialized AIs still require documentation and



controls that guarantee quality and compliance with enterprise requirements, enabling bias and risk evaluation as part of decision-making processes.

6.2. Interpretability and Trust

A real-world deployment would reserve the Bayesian subsystem for off-line use, generating calibrated response surfaces periodically for ingestion by the neural subsystem. The stochastic maps from the neural component are therefore those typically expected from real-time applications of deep learning: they are not expected nor needed to be reliable. Practical interpretation arises from the neural network selecting calibration maps based on the current position in input space; yet the explanation of how it navigates requirement space remains elusive.

User-interface techniques aiding the interpretation of individual neural predictions thereby assume special significance for real-world deployment. Methodologies such as LIME and SHAP are popular for general neural or machine-learning models; in the present application, additional attention must also be paid to selecting examples that help explain the decision-making process. User acceptance of the DPCA model will ultimately shape the evolution of governance with respect to its predictions and the resultant business decisions. In the absence of a transparent functioning, the BBB formal evaluation will eventually lose its value, particularly in situations where the task is to identify risky or threaten scenarios. In simple terms, the DD-Cycle Résilience Non-Negative matrix factorization model interprets trust as: the level a model faithfully predicts belongs to the real model of Nature.

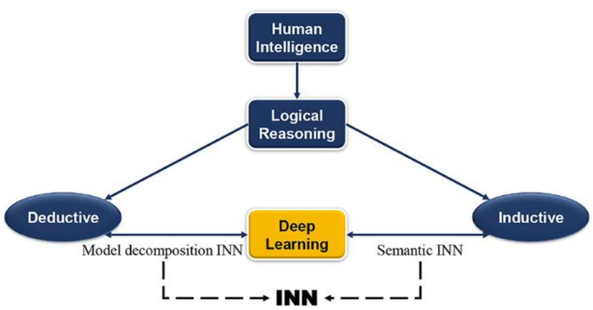


Fig 4: Interpretable neural networks

6.3. Risk Management and Governance

In addition to implications for successful deployment, model interpretability, and outcome explainability, considerations around risk management and governance provide another practical dimension to the proposed real-time decision-support framework. Even in their hybridized implementations, AI systems can still harbor unwanted model errors and calibration flaws that could have negative repercussions in the operationalized environment. Therefore, organizations using AI for decision-support should build appropriate risk controls into the model's lifecycle. As with any other type of decision-support system, the quality of model outcomes should be validated through an extensive test-and-learn approach, using multiple datastreams to prevent overfitting, before implementation in business decision-making. Controls should also be put in place to ensure an inherent capacity for drift detection and adaptation, either autonomously or in consultation with expert decision-makers, to safeguard ongoing model relevance.

Acceptance is another significant aspect of AI risk management and governance. Models intended for operational use must generally comply with continuously evolving AI safeguards such as the EU AI Act and corresponding national directives. These laws aren't just concerned with operational outcomes, but also the characteristics of the models themselves and the personnel involved in their development. Accordingly, organizations deploying AI must demonstrate that their system is fit for its intended use, that it has been suitably validated and tested, that its underlying development process is commensurate with its operational risk, and that safeguards are in place in case of unintended model behavior. The model design presented here enhances these dimensions to an extent by adapting and integrating proven techniques for robust behavior and human trust, and by providing comprehensive information about epistemic and aleatory uncertainty to inform and guide end-users through all phases of decision-making.

Equation 3: Uncertainty decomposition used for risk-aware decisions

Let the model have latent parameters θ. Then:

$$\mathrm{Var}(y \mid x, D) = \underbrace{\mathbb{E}_{\theta \sim p(\theta \mid D)}[\mathrm{Var}(y \mid x, \theta)]}_{\text{aleatory}} + \underbrace{\mathrm{Var}_{\theta \sim p(\theta \mid D)}(\mathbb{E}[y \mid x, \theta])}_{\text{epistemic}}$$

Step-by-step derivation:

1. Start with total variance identity:

$$\mathrm{Var}(Y) = \mathbb{E}[\mathrm{Var}(Y \mid Z)] + \mathrm{Var}(\mathbb{E}[Y \mid Z])$$

2. Substitute $Y = y, Z = \theta$, conditioned on (x, D):

$$\mathrm{Var}(y \mid x, D) = \mathbb{E}_{\theta \mid D}[\mathrm{Var}(y \mid x, \theta)] + \mathrm{Var}_{\theta \mid D}(\mathbb{E}[y \mid x, \theta])$$

7. Limitations and Future Work

The research performed on AI-enabled hybrid neural-Bayesian models for real-time decision support in dynamic environments is subject to certain limitations that diminish results' generalizability. The assumptions underlying the models aimed to support a limited scope of decision problems in predefined dynamic settings. Resolving these assumptions may extend support to a wider range of decision problems and potentially to longer living applications that also experience longer periods of stability. For instance, modifying the hybrid Bayesian neural architecture to allow simultaneous training of both subsystems may enhance its adaptive capabilities. Integrating hybrid models in a stronger coupling fashion, so that the neural subsystem dynamically determines a suitable value for a blurred parameter of the Bayesian subsystem, may discard the need for additional fully supervised ones.

Revisiting the concept of model uncertainty as the presence of unknown parameters in the fully-supervised training may further support the design of a stronger coupling. Extending the dynamic simulation framework to support multi-class decisions, the joint simulation of several data streams and stress tests exerting joint shocks of different natures may also guide further research.

Facing challenges successfully passed by hybrid models, moving benchmarks should help assess completeness of future designs. Increasing the number of utility-based decision-winning metrics associated with the different environments may confirm the persistence of the hybrid models' support in a larger set of multi-class scenarios. Aimed at supporting operational deployment, future work should address interpretability, risk management, and deployment requirements capable of building user trust.

8. Conclusion

AI-enabled hybrid neural-Bayesian models support real-time decision making in dynamic business environments. Real-time decision support implies the capability to process sensory input streams at sub-second latency, produce on-demand predictions, and propagate or quantify epistemic and/or aleatory uncertainty in decision-winning outputs. Dynamic environments evolve rapidly in unprecedented directions, exhibiting sudden bursts of change and drift, such that training-and-forgetting mechanisms alone cannot keep pace, and static-trained models lose accuracy over time. Hybrid neural-Bayesian architectures integrate the temporal generalization strength of neural networks and uncertainty quantification power of Bayesian methods, so that an open-loop Bayesian architecture qualitatively different from a feedback-loop Kalman filter is an appealing coupling technology for deep learning approaches.

Evidence from prior work confirms that hybrid models outperform pure neural-network constructions on the problem of real-time decision support in dynamic business environments. The present study supplements the prior investigation by extending the empirical comparison to also include an offline-trained neural-network module, a pure Bayesian approach, training-and-forgetting models, and a different hybrid model. AI-enabled hybrid neural-Bayesian models support real-time decision making in dynamic business environments. Real-time decision support implies the capability to process sensory input streams at sub-second latency, produce on-demand predictions, and propagate or quantify epistemic and/or aleatory uncertainty in decision-winning outputs. Dynamic environments evolve rapidly in unprecedented directions, exhibiting sudden bursts of change and drift, such that training-and-forgetting mechanisms alone cannot keep pace, and static-trained models lose accuracy over time. Hybrid neural-Bayesian architectures integrate the temporal generalization strength of neural networks and uncertainty quantification power of Bayesian methods, so that an open-loop Bayesian architecture qualitatively different from a feedback-loop Kalman filter is an appealing coupling technology for deep learning approaches.

9. References

[1] Blei, D. M., Kucukelbir, A., & McAuliffe, J. D. (2017). Variational inference: A review for statisticians. *Journal of the American Statistical Association*, *112*(518), 859–877.

[2] Blundell, C., Cornebise, J., Kavukcuoglu, K., & Wierstra, D. (2015). Weight uncertainty in neural networks. In *Proceedings of the 32nd International Conference on Machine Learning (ICML)* (pp. 1613–1622).

[3] Carvalho, C. M., & West, M. (2007). Dynamic matrix-variate graphical models. *Bayesian Analysis, 2*(1), 69–98.

[4] Cobb, J., & Ward, M. (2022). Real-time analytics and decision intelligence in digital business transformation. *Business Horizons, 65*(6), 707–718.

[5] Curi, C., Krause, A., & Krause, A. (2020). Neural contextual bandits without regret. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*.

[6] Depeweg, S., Hernández-Lobato, J. M., Doshi-Velez, F., & Udluft, S. (2018). Decomposition of uncertainty in Bayesian deep learning for efficient and risk-sensitive learning. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*.

[7] Dew, J., Read, J., & Carr, N. (2023). Decision intelligence systems: Integrating AI, analytics, and business decision-making. *MIS Quarterly Executive, 22*(3), 199–213.

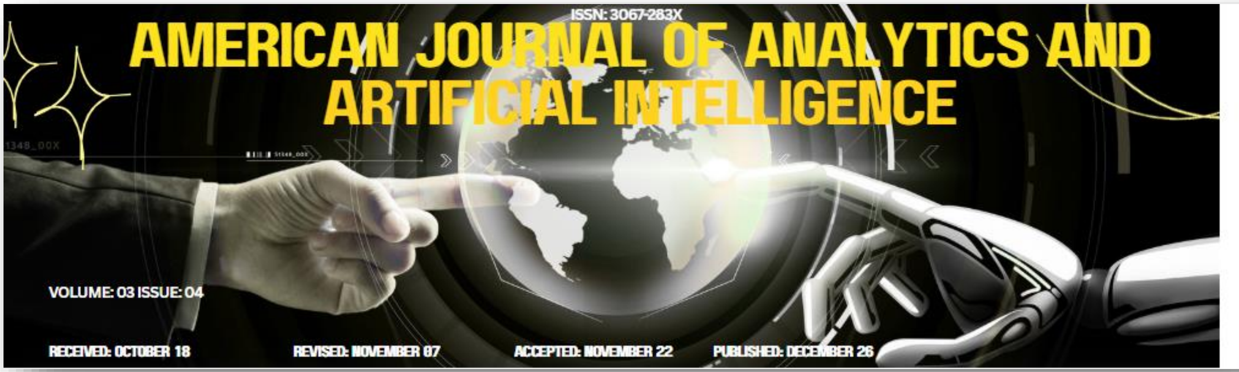
[8] Gal, Y., & Ghahramani, Z. (2016). Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)* (pp. 1050–1059).

[9] Ghosal, S., & van der Vaart, A. (2017). *Fundamentals of nonparametric Bayesian inference*. Cambridge University Press.

[10] Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., & Smola, A. (2012). A kernel two-sample test. *Journal of Machine Learning Research, 13*, 723–773.

[11] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.

[12] Graves, A. (2011). Practical variational inference for neural networks. In *Advances in Neural Information Processing Systems*.



[13] Gulshan, V., Peng, L., Coram, M., Stumpe, M. C., Wu, D., Narayanaswamy, A., Venugopalan, S., Widner, K., Madams, T., Cuadros, J., Kim, R., Raman, R., Nelson, P. C., Mega, J. L., & Webster, D. R. (2016). Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *JAMA, 316*(22), 2402–2410.

[14] Harrison, P. J., & West, M. (1999). *Bayesian forecasting and dynamic models* (2nd ed.). Springer.

[15] He, H., & Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering, 21*(9), 1263–1284.

[16] Hüllermeier, E., & Waegeman, W. (2021). Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine Learning, 110*(3), 457–506.

[17] Jospin, L. V., Buntine, W., Boussaid, F., Laga, H., & Bennamoun, M. (2022). Hands-on Bayesian neural networks—A tutorial for deep learning users. *IEEE Computational Intelligence Magazine, 17*(2), 29–48.

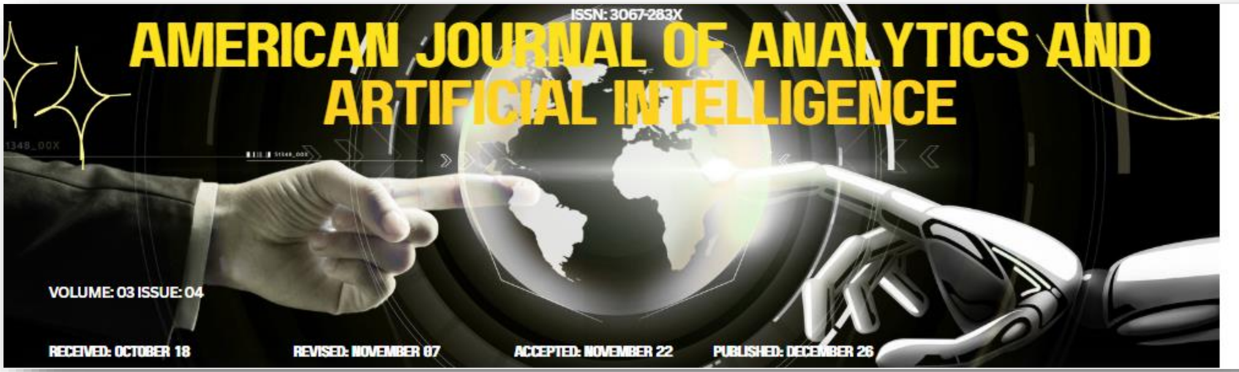
[18] Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., & Saul, L. K. (1999). An introduction to variational methods for graphical models. *Machine Learning, 37*(2), 183–233.

[19] Kapoor, S., & Narayanan, A. (2022). Leakage and the reproducibility crisis in ML-based science. *Patterns, 3*(9), 100496.

[20] Kendall, A., & Gal, Y. (2017). What uncertainties do we need in Bayesian deep learning for computer vision? In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)* (pp. 5574–5584).

[21] Kingma, D. P., & Welling, M. (2014). Auto-encoding variational Bayes. In *Proceedings of the 2nd International Conference on Learning Representations (ICLR)*.

[22] Koller, D., & Friedman, N. (2009). *Probabilistic graphical models: Principles and techniques*. MIT Press.



[23] Lakshminarayanan, B., Pritzel, A., & Blundell, C. (2017). Simple and scalable predictive uncertainty estimation using deep ensembles. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)* (pp. 6405–6416).

[24] Langford, J., Li, L., & Zhang, T. (2007). The epoch-greedy algorithm for multi-armed bandits with side information. In *Advances in Neural Information Processing Systems*.

[25] Lim, B., Arık, S. Ö., Loeff, N., & Pfister, T. (2021). Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting, 37*(4), 1748–1764.

[26] Lütkepohl, H. (2005). *New introduction to multiple time series analysis*. Springer.

[27] Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2018). Statistical and machine learning forecasting methods: Concerns and ways forward. *PLOS ONE, 13*(3), e0194889.

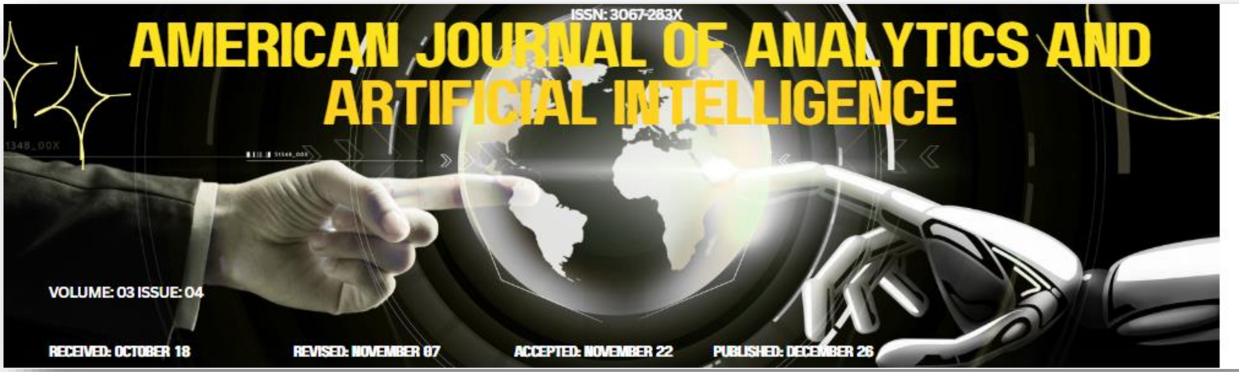
[28] Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2020). The M4 competition: Results, findings, conclusion and way forward. *International Journal of Forecasting, 36*(1), 54–74.

[29] McElreath, R. (2020). *Statistical rethinking: A Bayesian course with examples in R and Stan* (2nd ed.). CRC Press.

[30] Murphy, K. P. (2022). *Probabilistic machine learning: An introduction*. MIT Press.

[31] Papamakarios, G., Nalisnick, E., Rezende, D. J., Mohamed, S., & Lakshminarayanan, B. (2021). Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research, 22*(57), 1–64.

[32] Pearl, J. (2009). *Causality: Models, reasoning, and inference* (2nd ed.). Cambridge University Press.



[33] Peters, J., Janzing, D., & Schölkopf, B. (2017). *Elements of causal inference: Foundations and learning algorithms*. MIT Press.

[34] Rasmussen, C. E., & Williams, C. K. I. (2006). *Gaussian processes for machine learning*. MIT Press.

[35] Rue, H., Martino, S., & Chopin, N. (2009). Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *71*(2), 319–392.

[36] Salinas, D., Flunkert, V., Gasthaus, J., & Januschowski, T. (2020). DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, *36*(3), 1181–1191.