

FinIntel AI: A Confirmation-Gated, Explainable Platform for Corporate Financial Analysis, Credit Risk Assessment and Bank-Routed Lending

Dr. B. Meena Preethi

Dept. of Software Systems & AIML
Sri Krishna Knowledge Foundation, Deemed to be University
Coimbatore, India
meenapreethib@skasc.ac.in

Nadish N

Dept. of Software Systems & AIML
Sri Krishna Knowledge Foundation, Deemed to be University
Coimbatore, India
nadishn23mss028@skasc.ac.in

Abstract—Small and medium-sized enterprises seldom employ a financial analyst, and the lenders who assess them still work from documents rather than from verified figures. Statements arrive as spreadsheets, portable documents and scanned images; values are re-keyed by hand into private spreadsheets; interim statements are compared against thresholds calibrated on annual accounts; and the credit opinion that results carries neither a recorded probability nor a stated reason. This paper presents FinIntel AI, a role-based web platform that converts uploaded financial statements into confirmed, machine-readable data and derives ratio analysis, financial-health and credit-risk scoring, feature-level explanation, forecasting, simulation, anomaly detection, a grounded question-answering assistant and a bank-routed lending workflow from that single verified dataset. The design rests on a confirmation gate: no analytical module may read a statement whose extracted figures the subject company has not explicitly approved, and every stored result carries the version of the data from which it was computed, so an assessment produced before a correction is surfaced as stale rather than presented as current. A reporting-period model annualises interim statements before any comparison or scoring, removing a distortion that otherwise penalises companies reporting more frequently than annually. Financial health and corporate credit risk are modelled with gradient-boosted decision trees; every prediction is accompanied by an additive feature attribution rendered inside the application; irregular transactions are identified by an unsupervised detector supported by explicit rules; and the assistant answers only from results the platform has already computed, so it cannot invent a figure. A single application reaches one, several or all eligible banks, and each bank decides independently against the assessment snapshot taken at submission, with the aggregate status recomputed from those decisions. The platform was evaluated on held-out partitions of its training data and then exercised end to end on demonstration companies drawn from different industries, with every analytical output and every service response time recorded. Because the classifiers were trained on synthetic data, the reported measurements characterise model behaviour rather than real-world credit accuracy.

Keywords—*credit risk assessment, explainable artificial intelligence, gradient boosting, Shapley additive explanations, financial document processing, reporting-period normalisation, anomaly detection, decision support systems, financial technology.*

I. INTRODUCTION

Access to credit is one of the binding constraints on small and medium-sized enterprises. The constraint is only partly a matter of risk; a substantial part of it is informational. A lender asked to assess such a borrower receives a folder of documents rather than a dataset, and must establish from those documents

what the business earns, what it owes, whether the figures are internally consistent and whether the applicant is who it claims to be. The same information is rarely available to the business itself in any analytical form, so an applicant cannot tell in advance how a lender will read its own accounts, and learns the answer only when the application is refused.

Three problems follow from this arrangement. First, the figures that enter the assessment are re-keyed by hand, which introduces errors that no later stage can detect and for which no party is formally accountable. Second, interim statements covering a quarter or a half-year are routinely compared against thresholds calibrated on annual accounts, which understates turnover, returns and coverage for any company that reports more frequently than once a year. Third, the resulting opinion is a categorical judgement with neither a quantified probability nor a stated reason, which makes it difficult to audit, to contest or to act upon. A business told only that its application was unsuccessful has no basis on which to improve.

This paper describes FinIntel AI, a platform built around a different premise: that analysis should proceed only from figures the subject company has confirmed, and that every result should be traceable to the exact version of those figures. The system provides three portals. Companies register, submit verification evidence, upload statements, confirm the extracted values and read the resulting analysis. Bank officers see applications addressed to their own bank and branch, together with the assessment snapshot taken when the application was submitted, and record a decision with its terms and its reasons. Administrators verify companies, officers and bank accounts, maintain the bank and branch directory, and monitor documents, models and system health.

Fig. 1 shows the path through the system. A company registers and is verified by an administrator before any financial feature opens; it uploads statements, which are parsed or read optically, classified and reconciled; it confirms or corrects the extracted figures, and only the confirmed set is analysed; it then reads the ratio, health, credit and explanation screens and may apply for a loan. An application is addressed to one or more banks, each of which reviews the frozen snapshot and decides on its own.

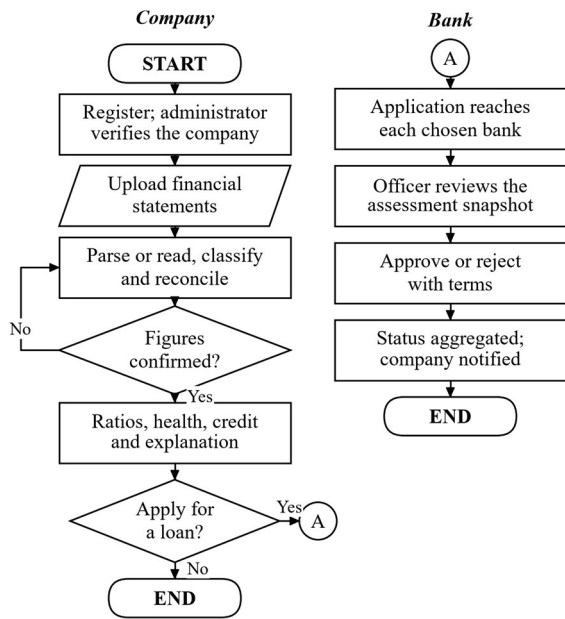


Fig. 1. System flow. The confirmation decision governs everything downstream of it; a figure that is not confirmed returns to extraction for correction, and no analysis is produced from it.

The contributions of this work are as follows. (i) A confirmation gate and an accompanying data-versioning scheme that make the provenance of every analytical result explicit and that mark results computed from superseded data as stale. (ii) A reporting-period model that annualises interim statements before any comparison, ratio computation or model input is formed. (iii) An integrated intelligence layer in which classification, explanation, forecasting, simulation and anomaly detection all read the same confirmed dataset, so that the figures quoted on a screen, in a generated report and by the assistant cannot disagree. (iv) A lending workflow in which one submission reaches several banks, each deciding independently against a frozen snapshot. (v) An end-to-end evaluation in which both the analytical outputs and the service response times were recorded.

The remainder of the paper is organised as follows. Section II reviews related work. Section III states the problem and the design principles that follow from it. Section IV describes the architecture and Section V the data pipeline. Section VI sets out the analytical and learning methods, Section VII the database design and Section VIII the implementation and interface. Sections IX and X present the experimental setup and the results. Section XI compares the approach with conventional practice, Section XII states the limitations, and Section XIII concludes.

II. RELATED WORK

A. Statistical Credit Scoring

Quantitative credit assessment begins with discriminant analysis of financial ratios [1] and with the probabilistic formulation that followed [2]. These models identified the predictors still in use today, chiefly liquidity, leverage, profitability and coverage, and they remain strong baselines against which later methods are measured. Their limitation for the present purpose is not accuracy but input: they assume that a clean, comparable set of ratios already exists, and say nothing about where it comes from or who vouches for it.

B. Machine Learning for Credit Risk

Large comparative studies report that ensemble methods outperform single classifiers on credit-scoring tasks [3], and

systematic surveys of the field reach the same conclusion while noting the recurring difficulty of comparing results obtained on different and often proprietary datasets [4]. Gradient-boosted decision trees, in the two implementations used in this work [5], [6], dominate recent applied results on tabular credit data because they handle mixed scales, missing values and non-linear interactions without extensive preprocessing. This literature almost always begins from a prepared feature matrix; how that matrix is produced from documents, and who is accountable for its values, is treated as lying outside the problem.

C. Explanation

Additive feature attribution [7] and its efficient tree-specific form [8] provide per-prediction explanations with a coherent theoretical basis, in which the contributions of the features sum to the difference between the prediction and a base value. Surveys of explainable artificial intelligence [9] and work specific to credit risk [10] argue that explanation is a precondition for use in regulated lending rather than an optional extra. The present work takes that position literally: the attribution is not an offline artefact produced for a reviewer but a screen that the borrower and the officer both see, rendered from the same stored values. A dissenting position holds that high-stakes decisions should use intrinsically interpretable models rather than post-hoc explanations of opaque ones [14]. The design here answers that objection in part by keeping the decision itself with a named officer, so that the model informs the outcome without determining it, which is also the condition under which supervisory guidance treats machine learning in credit risk as acceptable [13].

D. Anomaly Detection and Forecasting

Isolation forests [11] detect outliers without labels and scale to the transaction volumes found in bank statements, which makes them appropriate where no labelled history of irregularity exists, as is normally the case for a small business. For short, seasonal business series, decomposable additive models [12] give interpretable trend and seasonality components together with uncertainty intervals; interpretability matters more here than marginal accuracy, because the forecast is shown to a non-specialist who must decide how much weight to give it.

E. Document Processing and the Gap Addressed

Work on financial document understanding concentrates on extraction accuracy: on recognising a table, a label or a figure correctly. The gap this paper addresses lies between extraction and analysis. Even a perfect extractor produces a proposal, and in a lending context somebody must accept that proposal before it is treated as fact. No system in the surveyed literature makes that acceptance a precondition for analysis, records it against a data version, and propagates staleness to every result derived from superseded figures. That gate, together with the integration of scoring, explanation and a multi-bank workflow on top of it, is what distinguishes the present work.

III. PROBLEM FORMULATION AND DESIGN PRINCIPLES

Let a company hold a set of source documents covering one or more reporting periods. Extraction yields a proposed value for each canonical field of each document. The platform treats a proposed value as inadmissible until the company accepts or corrects it; only the accepted set, together with the version counter incremented at acceptance, constitutes the analytical dataset. Every derived artefact, whether a ratio, a score, an attribution, a forecast or a generated report, is stored with the

version it was computed from. A result whose version is lower than the current version of its company is reported as stale, and the interface offers to recompute it rather than presenting it as current.

Four principles follow, and they are applied throughout the system.

- Nothing is analysed before it is confirmed. Extraction produces a proposal; the company accepts or corrects it; responsibility for the figures therefore rests with the business that submits them.
- Nothing is decided by the platform. The models produce scores with explanations; the decision, its terms and its reasons are recorded against a named officer of a named bank.
- Comparable units only. Interim statements are annualised before any ratio, benchmark or model input is formed, so that a half-yearly filer is not penalised against an annual one.
- One dataset, many views. Screens, generated reports and the assistant read the same stored results, so the figures they present cannot diverge.

Access is governed by a verification gate that is independent of the confirmation gate, and the two are deliberately separate: the first establishes who the party is, the second establishes what its figures are. A company may register and complete its profile at any time, but financial features open only after an administrator has verified the company against submitted evidence; an officer gains a review queue only after the officer record itself has been verified against the bank directory; and a bank account may be used in an application only after it too has been verified. Table I summarises the resulting capabilities and their preconditions.

TABLE I. ROLE-BASED CAPABILITIES AND THEIR PRECONDITIONS

Capability	Company	Officer	Admin
Register and complete profile	Yes	Yes	n/a
Submit verification evidence	Yes	Yes	n/a
Upload and confirm documents	Verified	No	No
Read analytics and explanations	Verified	Scoped	No
Apply for a loan	Verified	No	No
Review an application	No	Own bank	No
Record a lending decision	No	Own bank	No
Verify companies and officers	No	No	Yes
Maintain the bank directory	No	No	Yes
Monitor models and health	No	No	Yes

IV. SYSTEM ARCHITECTURE

The platform is a layered application in which each layer depends only on the layer beneath it. A single-page client renders role-scoped routes; a stateless application programming interface authenticates every request and enforces role-based access control; a document layer validates, parses and reads uploaded files; the confirmation gate sits between that layer and everything above it; the intelligence layer holds the models and the explanation machinery; a service layer computes ratios, cash flow, tax and benchmarks and orchestrates the lending workflow; and a relational store holds both the confirmed data and the versioned results. Fig. 2 shows the arrangement.

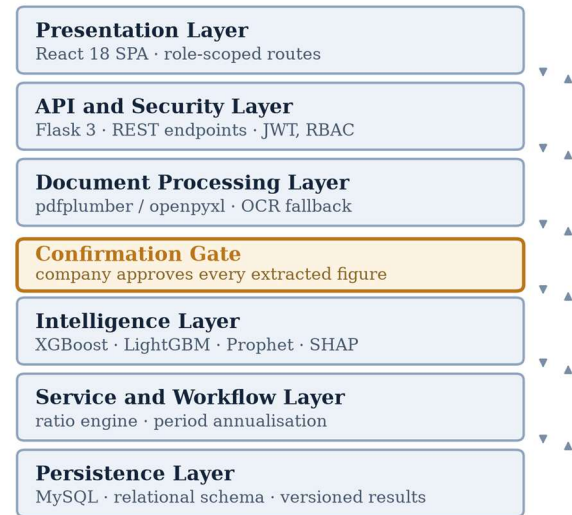


Fig. 2. Layered architecture. The confirmation gate is placed between document processing and every analytical layer above it, so that no module can read an unconfirmed figure.

Placing the gate in the architecture rather than in application logic is a deliberate choice, and it is the single decision on which the integrity of the whole system rests. A check written inside one endpoint protects that endpoint alone, and a system of any size accumulates endpoints faster than it accumulates reviewers; a gate expressed as a query predicate that every analytical service inherits protects all of them, including the services written after the gate was designed. In the implementation, the predicate restricting reads to confirmed statements of the current data version is applied in the data-access layer, and the services above it have no route around it.

The external boundary of the system is narrow, which matters for a platform holding commercially sensitive accounts. Fig. 3 gives the context-level data flow. Three human actors interact with the platform; a mail service carries one-time codes and alerts; and a large language model provider, which is optional, receives only a grounded prompt assembled from results already computed and returns prose. No external party is given access to the underlying data store, and the platform continues to function with the language model absent, falling back on templated answers assembled from the same stored values.

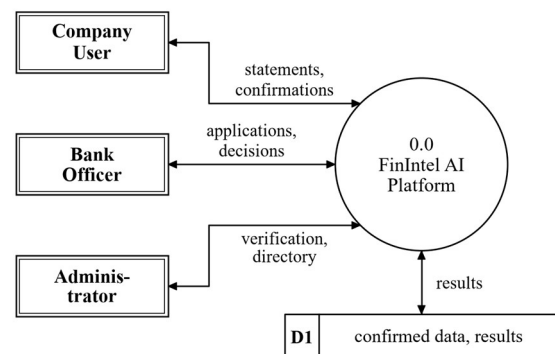


Fig. 3. Context-level data flow. One process, three human actors and the confirmed data store; the mail service and the language-model provider, both optional, are omitted because neither receives access to stored data.

V. DATA PIPELINE

A. Validation, Extraction and Classification

An uploaded file is first validated by declared type, size and signature, and a cryptographic digest is computed and stored so

that a later re-upload of the same file can be recognised. Spreadsheets and delimited files are parsed directly; portable documents are parsed for their text layer and fall back to optical character recognition only when no text layer exists, which keeps recognition error out of the common case; images are read by optical character recognition. The extractor then classifies the document into a statement type, detects its reporting period and the units in which it is denominated, and maps the recognised labels onto canonical fields.

Cross-document checks follow, and they are what turn a set of independent extractions into a coherent dataset. The closing balance of one period is reconciled against the opening balance of the next; the balance sheet is checked for equality of assets with liabilities and equity; and any figure that appears in more than one document is compared across those documents. A discrepancy does not block the upload: it is surfaced to the company at the confirmation step, where it is most cheaply resolved.

B. Document Coverage and Duplicate Control

The classifier recognises thirteen document types, among them the balance sheet, the profit and loss account, the cash-flow statement, the bank statement, the tax return and the supporting registers that a lender normally asks for. Recognition is by content rather than by file name, because a file arriving from a small business is as likely to be called a date as a statement type.

Duplicate control is handled by the stored digest. A file whose digest matches an existing document is recognised as a re-upload rather than as new evidence; a file that covers a period already on record is recognised as a new version of that document, and the earlier version is retained rather than overwritten. Versioning at the document level is what makes the correction path safe: a company that discovers an error after confirmation can upload a corrected statement without destroying the record of what was analysed before.

C. The Confirmation Gate

The output of extraction is presented to the company field by field, beside the source document and the label from which each figure was read, so that checking a value does not require opening the original file. The company accepts the proposal or corrects it, individually or for a batch of documents at once. Acceptance writes the financial statement, records an audit entry naming the user and the time, and increments the data version of the company. Withdrawal of a document reverses the effect: results derived from it are removed rather than left behind to be read later as though still valid. Analytical services read confirmed statements only, and a stored result whose version precedes the current one is displayed as stale with an explicit refresh offered.

D. Reporting-Period Normalisation

Each statement carries a period type with a known length: annual, half-yearly, quarterly or a stated interval. Flow quantities such as revenue, profit and the components of cash flow are annualised by the ratio of twelve months to the period length before any ratio is formed, any benchmark is applied or any model input is constructed. Stock quantities such as assets, liabilities and closing cash are taken as at the period end and are not scaled, because scaling a balance would be meaningless. Without this step a company filing half-yearly reports turnover at roughly half the annual figure and is scored as though its business had halved; with it, the two are comparable. The interface states on every affected screen that returns and turnover ratios of an interim period have been

annualised, so the normalisation is visible rather than hidden inside the arithmetic.

VI. ANALYTICAL AND LEARNING METHODS

A. Ratio Engine

A single engine computes the canonical liquidity, leverage, profitability, efficiency and coverage ratios from the confirmed, annualised figures. Every ratio is stored with the formula applied, the inputs used and the period, so that a displayed value can be traced back to its arithmetic without re-deriving it. One engine serves the screens, the generated reports, the assistant and the model feature builders, and that single source is what prevents two parts of the system from quoting different values for the same quantity.

Fourteen ratios are computed, grouped into the five categories of Table II. For each one the engine returns the value, the formula, the inputs it used, a health status against the benchmark target and, where the value cannot be computed, the reason: a null denominator yields an explicit unavailable status rather than a silent zero, and a negative net profit propagates correctly into the margin and return ratios instead of being suppressed.

TABLE II. RATIO CATEGORIES COMPUTED FROM CONFIRMED DATA

Category	Ratios computed	No.
Liquidity	Current, quick and cash ratios	3
Solvency	Debt ratio, debt to equity, interest coverage	3
Working capital	Working capital	1
Profitability	Gross, operating and net margin; return on assets and on equity	5
Efficiency	Inventory turnover, asset turnover	2

B. Financial Health Classification

Financial health is modelled as binary classification over engineered ratio features with an extreme gradient-boosting classifier [5]. The published score is the probability of the healthy class mapped onto a hundred-point scale and reported with a classification band, a risk level, the model version and the data version it was computed from. Training used a stratified split with a fixed seed; the model selected on the validation partition was evaluated once on the held-out test partition, and the test partition was not consulted during model selection.

C. Credit Risk Classification and Scoring

Corporate credit risk is modelled as binary classification of default over a wider feature set that adds leverage dynamics, coverage and debt-service measures to the health features, using a histogram-based gradient-boosting classifier [6]. The predicted probability of default is mapped to a credit score by the affine transformation reported in Table VII, which places the score on a scale lenders already read, and the score is then placed in a low, medium or high band. A what-if facility recomputes the score and the debt-service coverage ratio for a prospective loan amount, rate and tenure, so a company can see the effect of a borrowing decision before it makes an application.

D. Model Explanation

Each health and credit prediction is accompanied by an additive attribution computed with the tree-specific estimator [8]. The application renders the base value, the signed contribution of each feature and the reconciliation of those contributions with the model margin, so that the explanation

can be checked arithmetically rather than merely read. The same attribution drives the plain-language drivers shown on the score screen and the risk factors shown to the bank officer, so borrower and lender are given the same reasons rather than two different summaries of one model.

E. Anomaly Detection

Bank-statement transactions are scored by an isolation forest [11] over engineered features covering amount, direction, timing, counterparty repetition and balance movement. Because an unsupervised score on its own is a weak explanation, two deterministic rules run beside it to detect exact duplicates and split transfers, whose individual amounts are unremarkable and which an amount-sensitive detector therefore misses. Every flag carries a reason, and flags are recomputed from confirmed documents only, so that withdrawing a statement withdraws its flags.

F. Forecasting and Simulation

Revenue and profit are projected with a decomposable additive model [12] that returns a trend, a seasonal component and an uncertainty interval; the interval is shown rather than suppressed, because the histories available from small enterprises are short and a point forecast drawn from them would imply a confidence the data does not support. A Monte Carlo facility simulates profit paths over a chosen horizon and reports the fifth, fiftieth and ninety-fifth percentile outcomes together with the distribution, which communicates dispersion more usefully than a single projected figure.

G. Benchmarking, Cash Flow and Tax

Three service modules sit beside the models and read the same confirmed data. Benchmarking places each ratio against the median of the company's industry and size band for the financial year, so that a margin is judged against the trade it is earned in rather than against a universal threshold. The cash-flow module separates operating, investing and financing flows, derives free cash flow, closing cash, runway and debt-service coverage, and annualises interim flows before any of these are compared. The tax module computes the liability under the applicable regime and reports the effective rate, which is one of the few figures a borrower can act on quickly.

H. Advice and Goal Planning

A rule engine turns the computed results into prioritised actions. Each rule states the condition it fires on, the evidence it cites and the priority it carries, so a recommendation can be traced to the figure that produced it rather than appearing as unexplained advice. Goal planning works in the other direction: a company states a target, and the engine reports whether the target is reachable from the current cash position and what monthly contribution it implies. Because both read the confirmed dataset, advice and plan cannot contradict the analysis they are drawn from.

I. Grounded Question Answering

The assistant classifies a question into an intent, retrieves the stored results for that intent, and answers from them. When a language model is configured it receives only the retrieved values as a grounded prompt and is used to phrase the answer; when it is not, a template is used instead. The assistant has no path to the database other than the results already computed, so it cannot invent a figure, and a request outside its scope is declined rather than answered.

Table III summarises the eight intelligence modules, the method each one uses and the output it writes back to the confirmed dataset.

TABLE III. INTELLIGENCE MODULES AND THEIR OUTPUTS

Module	Method	Output
Financial health	XGBoost	Score, band, drivers
Credit risk	LightGBM	Default probability, score
Explanation	SHAP (tree)	Signed contributions
Anomalies	Isolation Forest, rules	Flags with reasons
Forecasting	Decomposable additive	Path with interval
Simulation	Monte Carlo	Percentile outcomes
Advice	Rule engine	Prioritised actions
Assistant	Intent routing	Grounded answer

VII. DATABASE DESIGN

The relational schema holds identity and verification, the bank and branch directory, documents and extractions, confirmed statements, every analytical result and the lending workflow. Three design decisions carry most of the weight.

First, confirmed statements are separate entities from the documents and extractions that produced them. This is what allows the confirmation gate to be expressed as a join rather than as a boolean column that a careless query could ignore: a service that reads statements cannot reach an unconfirmed figure, because unconfirmed figures are not statements.

Second, every result table carries both the company data version and the model version. Staleness is therefore a query and not an inference: a result is stale exactly when its recorded data version is lower than the company's current one. The same column makes it possible to answer, after the fact, which version of the data a given assessment was based on, which is the question an auditor asks first.

Third, an application and its per-bank targets are distinct entities. One submission therefore carries several independent decisions, each with its own status, terms, remarks and officer, while the aggregate status of the application remains derivable from them rather than being stored twice and risking divergence.

Sensitive values are protected at rest. Account numbers are stored encrypted and are returned to any interface in masked form only; passwords are stored as adaptive hashes; and uploaded evidence is held outside the web root with its digest recorded. Domain events are appended for every state change of a verification request, a document and an application, which gives an audit trail that does not depend on application logs and that survives a redeployment.

VIII. IMPLEMENTATION AND USER INTERFACE

The client is a single-page application; the server is a stateless interface secured with signed tokens and role-based access control; the store is a relational database accessed through an object-relational mapper. Table IV lists the stack and Table V the scale of the delivered system.

TABLE IV. IMPLEMENTATION STACK

Layer	Technology
Client	React single-page application
Interface	Flask REST API, JSON Web Tokens
Documents	Native parsers with OCR fallback
Intelligence	XGBoost, LightGBM, SHAP, Prophet
Anomalies	Isolation Forest, scikit-learn
Store	MySQL with an object-relational mapper
Verification	pytest, Playwright

TABLE V. SCALE OF THE DELIVERED SYSTEM

Measure	Count
REST endpoints	204
Interface modules	28
Relational tables	41
Automated tests	417
Test modules	36
Captured application screens	32

The interface is organised by role, and seven screens are reproduced here because each shows something the text asserts. Fig. 4 is the company dashboard: the headline figures of the latest confirmed period, the revenue and profit trend, and the attention panel that collects alerts, pending documents and overdue items. Fig. 5 is the financial-health screen, which carries not only the score but the classification, the model version, the key drivers and the model inputs, so that a reader can see what the score was computed from.

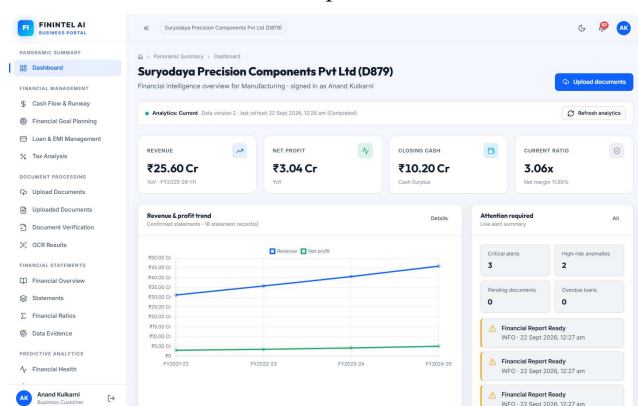


Fig. 4. Company dashboard. Every headline figure is drawn from the latest confirmed period, and the status bar states the data version behind it.

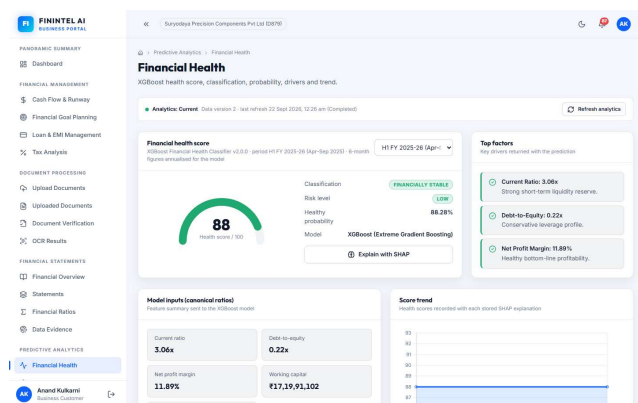


Fig. 5. Financial-health screen. The score is shown with its classification, its model version, the drivers returned by the attribution and the model inputs.

Fig. 6 is the credit-risk screen. It carries the credit score and the probability of default beside the debt context the model used, the key risk factors and the what-if form that recomputes both for a prospective loan, so a company can test a borrowing decision before making an application. Fig. 7 is the explanation screen, which renders the base value, the signed contribution of each feature and the reconciliation with the model margin; this is the screen that makes the claim of explainability checkable rather than merely asserted.

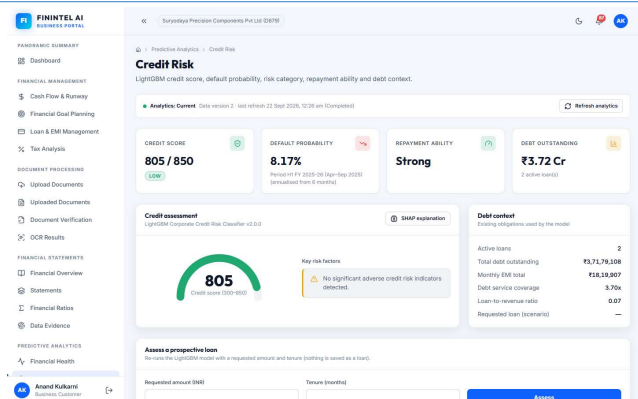


Fig. 6. Credit-risk screen. The score, the probability of default and the risk factors are shown with the what-if form for a prospective loan.

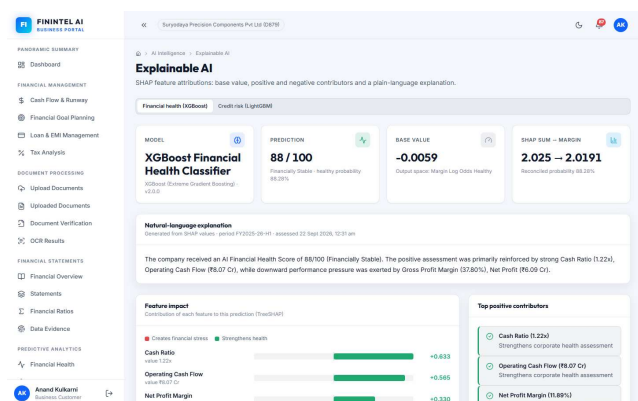


Fig. 7. Explanation screen. The contributions reconcile with the model margin, so the explanation can be checked rather than taken on trust.

The last three screens carry the workflow. Fig. 8 is the application form, on which a company chooses a verified bank account and then addresses the application to one bank, to several, or to every bank for which it is eligible; the eligibility check runs before submission, so a company is not left waiting on a bank that could never have lent to it. Fig. 9 is the bank officer's review of a single target: the assessment snapshot frozen at submission, the requested terms and the controls that record a decision. The officer sees applications addressed to the officer's own bank and branch and no others, which is enforced in the query and not only in the interface. Fig. 10 is the administrator's verification queue, where companies, officers and bank accounts are approved against submitted evidence before any of the gated capabilities in Table I open.

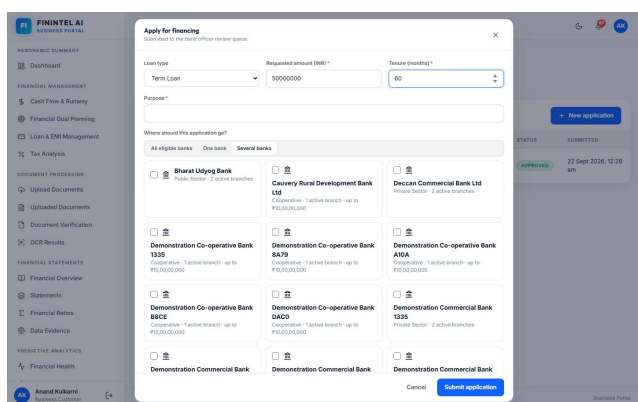


Fig. 8. Loan application. A verified bank account is selected and the application is addressed to one, several or all eligible banks.

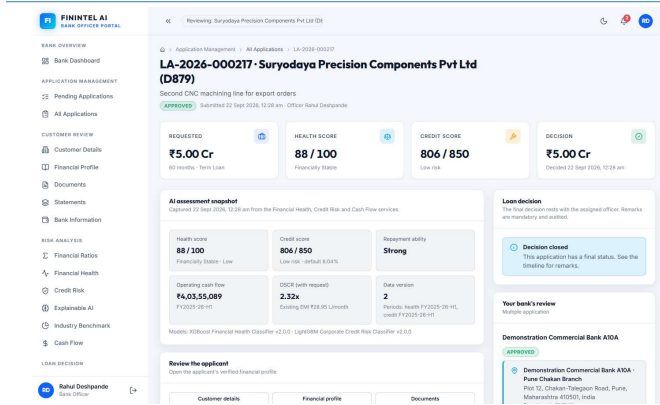


Fig. 9. Bank officer review of one application target, showing the assessment snapshot frozen at submission alongside the decision controls.

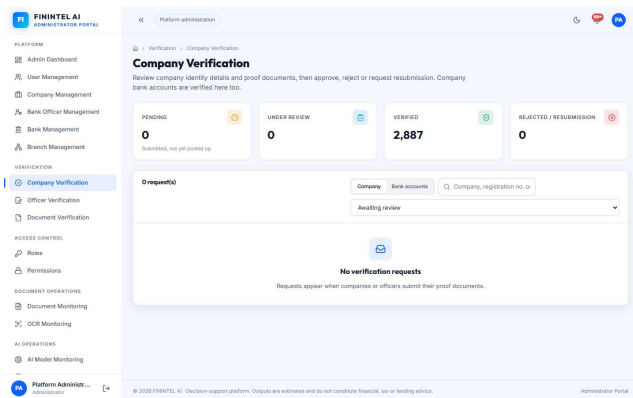


Fig. 10. Administrator verification queue. Companies, officers and bank accounts are approved against submitted evidence before the gated features open.

Two details of the interface follow directly from the design principles. Every analytical screen states the data version it was computed from and offers a refresh when that version is behind, so staleness is visible rather than implicit. Every bank account is shown masked, with an explicit notice that numbers are encrypted at rest, so the interface does not quietly contradict the storage policy.

Two output channels sit beside the screens. A report generator assembles the stored results of a chosen period into a portable document for a lender or an accountant; because it reads results rather than recomputing them, a report and the screen it was generated from cannot disagree. A notification service records every state change that a user needs to know about, among them the outcome of a verification request, the completion of document processing and each bank's decision on an application, and delivers it in the application with electronic mail as an optional second channel.

IX. EXPERIMENTAL SETUP

Two kinds of evaluation were carried out. The classifiers were measured on held-out partitions of their training data. The platform as a whole was then exercised end to end on demonstration companies, with every analytical output and every service response time recorded. Table VI states the training configuration.

TABLE VI. DATASET AND TRAINING CONFIGURATION

Property	Health	Credit
Algorithm	XGBoost	LightGBM
Model version	v2.0.0	v2.0.0
Records	6,000	6,000
Training	4,200	4,199
Validation	900	901
Test	900	900
Features	22	24
Split	70 : 15 : 15	70 : 15 : 15
Provenance	Synthetic	Synthetic

The end-to-end run used two fully documented companies: a precision manufacturer in sound financial condition and an agri-processing business under stress. Both were registered, verified through the administrator portal, given verified bank accounts, and had their complete document sets uploaded and confirmed before any analytical endpoint was called, so that no result in the run could have come from an unconfirmed figure. Three further demonstration companies, in information technology, healthcare and logistics, were used to confirm that the behaviour generalises across sectors with different cost structures. All runs used an isolated test database, and the anomaly detector was evaluated against a bank statement into which irregularities of four known types had been planted by the authors of the test pack, with a second statement containing none used as a control.

X. RESULTS AND DISCUSSION

A. Predictive Performance

Table VII reports the held-out performance of both classifiers together with the cross-validated area under the receiver operating characteristic curve. Fig. 11 shows the receiver operating characteristic and precision-recall curves, and Fig. 12 shows which features each model relies on.

TABLE VII. HELD-OUT PERFORMANCE OF BOTH CLASSIFIERS

Measure	Health	Credit
Accuracy	0.7467	0.7667
Precision	0.7278	—
Recall	0.7862	—
F1 score	0.7559	—
ROC-AUC (test)	0.8078	0.8278
ROC-AUC (5-fold CV)	0.8251 ± 0.0124	0.8215 ± 0.0165
Brier score	0.179	—
True negatives	319	493
False positives	132	86
False negatives	96	124
True positives	353	197
Score mapping	—	850 – (default probability × 550)

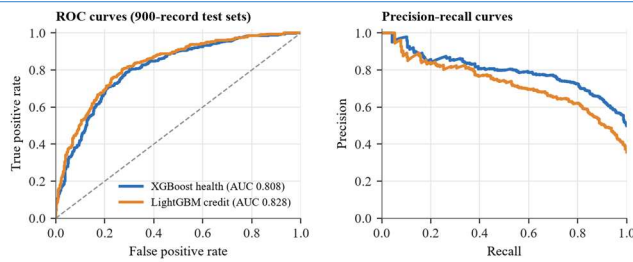


Fig. 11. Receiver operating characteristic and precision-recall curves for both classifiers on their held-out test partitions.

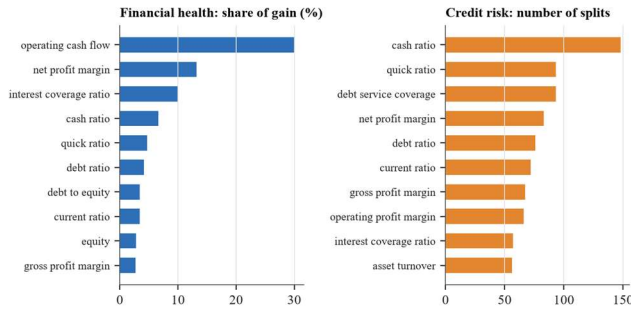


Fig. 12. Feature importance of the health model by gain and of the credit model by split count. Both rely chiefly on liquidity, leverage and coverage.

The discrimination reported here is consistent with published results for this model family on tabular credit data [3], [4]. The health model is deliberately biased towards recall: its recall exceeds its precision, so it over-identifies rather than misses companies in difficulty, which is the safer error for a screening tool whose output a human then reads. The Brier score indicates that the published probabilities are usable as probabilities and not only as a ranking, which matters because the credit score is an affine function of the predicted probability rather than of its rank.

B. Risk Band Separation

For a lending workflow the property that matters is not raw accuracy but whether the bands the model assigns separate outcomes. Table VIII and Fig. 13 give the observed default rate within each predicted band on the credit test partition.

TABLE VIII. OBSERVED DEFAULT RATE BY PREDICTED RISK BAND

Band	Companies	Observed default rate
Low	402	11.19 %
Medium	215	36.74 %
High	283	69.61 %

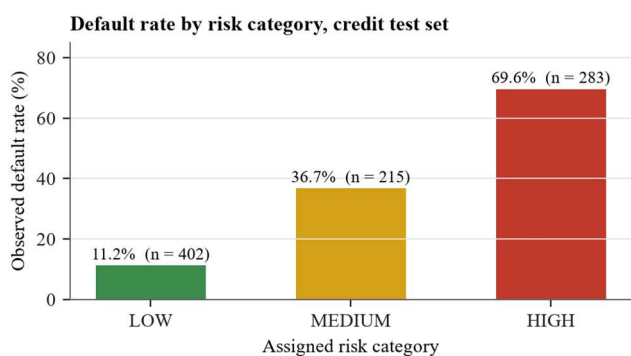


Fig. 13. Observed default rate by predicted risk band on the credit test partition.

The bands are monotone and widely separated: the high band carries roughly six times the default rate of the low band. A monotone ordering of this size is what makes the output

usable as a first filter inside a lending process, because an officer can act on the band while reading the attribution for the individual case.

C. Explanation in Practice

Fig. 14 shows the attributions behind the two assessments discussed below: the health prediction for the manufacturer and the credit prediction for the stressed company. The contributions are signed and sum to the difference between the model margin and the base value, so the figure is a decomposition of the prediction rather than a ranking of correlated variables. In both cases the features the attribution names are the ones a lender would cite: coverage and leverage for the credit decision, liquidity and margin for the health score. That correspondence is what makes the explanation usable in a conversation with the borrower.

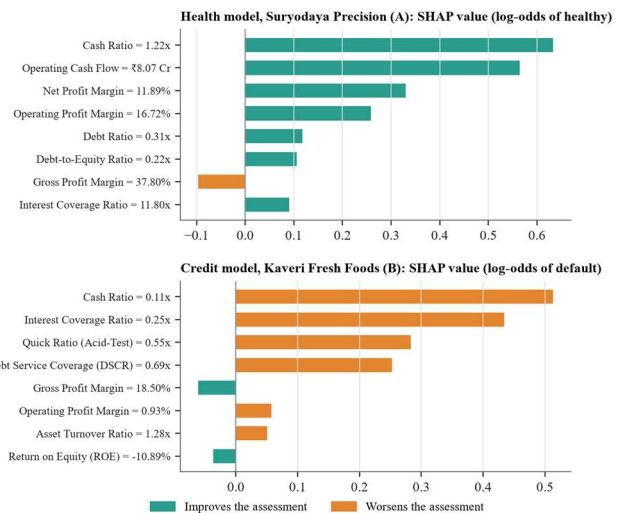


Fig. 14. Additive feature attribution for the latest health assessment of the manufacturer and the latest credit assessment of the stressed company.

D. End-to-End System Behaviour

Table IX reports the outcomes for the demonstration companies. The two profiles the platform was asked to separate are separated by a wide margin on every axis: score, band and probability of default.

TABLE IX. END-TO-END RESULTS FOR THE DEMONSTRATION COMPANIES

Company	Health	Risk	Score	Default prob.
Vyom Cloud Software	93	Low	810	7.24 %
Suryodaya Precision	88	Low	805	8.17 %
Nilgiri Diagnostics	88	Low	787	11.51 %
Sahyadri Logistics	53	Medium	572	50.58 %
Kaveri Fresh Foods	23	High	478	67.69 %

The ratio detail behind these outcomes is consistent with the scores. The manufacturer reports a current ratio of 3.06, debt to equity of 0.22, a net profit margin of 11.89 per cent and debt-service coverage of 3.70; the stressed company reports 0.88, 1.31, -2.58 per cent and 0.69 respectively. No component of either assessment contradicts another, which is the intended consequence of deriving every figure from one confirmed dataset.

Anomaly detection was evaluated against a bank statement containing four planted irregularities, with a second statement containing none used as a control. Table X reports the outcome and Fig. 15 shows the scores. Every planted transaction was

flagged and nothing else was, which leaves a reviewer with twelve records to examine instead of 595.

TABLE X. ANOMALY DETECTION ON THE EVALUATED BANK STATEMENTS

Measure	Manufacturer	Control
Transactions examined	595	125
Irregularities planted	4	0
Transactions involved	12	0
Transactions flagged	12	0
False positives	0	0
Review load	2.02 %	0.00 %

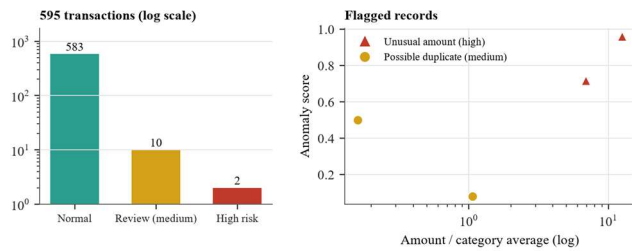


Fig. 15. Anomaly scores on the evaluated bank statement. The flagged records are separated from ordinary activity, and each carries a stated reason.

The two detectors divided the work as the hybrid design intends. The isolation forest located the two unusual amounts with high scores, because an amount far from the distribution is exactly what an unsupervised detector finds. The duplicate and split-transfer rules located the remaining cases, whose individual amounts are entirely ordinary and which no amount-sensitive detector would have reached. Neither component alone would have found all four.

E. Advice, Alerts and Assistant Behaviour

The advice and alert engines behaved differently for the two profiles, which is the behaviour they were designed for. For the manufacturer the system raised three critical alerts, two high-risk transaction anomalies and one goal at risk, with no overdue loan and no pending document, and it produced a single high-priority recommendation: to investigate the two high-severity anomalies already flagged. For the stressed company it raised seven critical alerts, and its recommendations addressed exactly the conditions that make the business a credit risk; no growth or investment advice was produced for it. Advice is therefore bounded by what the data supports rather than offered by default.

The assistant answered nine questions per company from stored results. Answers for the manufacturer took between seventeen and three hundred and forty-nine milliseconds; the two heaviest questions for the stressed company, which concerned a loan and a revenue scenario, took about one and a half seconds because they required a fresh computation rather than a lookup. Generated reports took between one hundred and forty-two and three hundred and five milliseconds to assemble.

F. Responsiveness

Every call of the end-to-end run was timed. Table XI gives the median response times. The analytical screens a user moves between are served in tens to hundreds of milliseconds; the operations that are slower are those that fit a model or assemble a multi-bank submission, and both are presented as explicit actions rather than as page loads, so the delay falls where a user expects to wait.

TABLE XI. MEDIAN SERVER RESPONSE TIMES

Operation	Median
Explanation (SHAP)	19 ms
Ratio analysis	29 ms
Monte Carlo simulation	28 ms
Credit risk assessment	126 ms
Financial health assessment	129 ms
Document upload	134 ms
Report generation	155 ms
Cash flow analysis	0.96 s
Loan submission	1.05 s
Revenue forecast	3.58 s

G. Functional Verification

Functional correctness was verified by 417 automated tests across 36 test modules, all of which pass against an isolated database, and by a browser suite of twenty-seven end-to-end checks covering the verification, banking, document and lending journeys, all of which pass. The assistant was asked nine questions per company covering health, credit, ratios, forecasting, instalments, a revenue scenario, anomalies, advice and one deliberately out-of-scope request; every in-scope answer matched the stored results and the out-of-scope request was declined rather than answered.

H. Discussion

Four observations follow from the evaluation. First, the confirmation gate changes the character of the analysis without changing its mathematics: the ratios and scores are computed as they would be anywhere, but their inputs carry an accountable signature, which is what makes the output defensible in a lending conversation. Second, period normalisation is not a cosmetic adjustment; without it, the half-yearly filer among the demonstration companies would have been scored on roughly half of its annualised turnover and would have been placed in a worse band for a reason that has nothing to do with its business.

Third, the combination of a band and an attribution is more useful to an officer than either alone, because the band orders the queue while the attribution answers the question the borrower will ask. Fourth, the multi-bank routing produced the behaviour it was designed for: a single submission reached two banks, one of which approved while the other requested further documents, and the aggregate status was recomputed from those two independent decisions rather than being set by either one of them.

XI. COMPARISON WITH CONVENTIONAL PRACTICE

Table XII sets the conventional assessment process beside the one proposed here, aspect by aspect.

TABLE XII. CONVENTIONAL PRACTICE AND THE PROPOSED APPROACH

Aspect	Conventional	Proposed
Data entry	Manual re-keying	Extraction with confirmation
Accountability	Undefined	Company confirms each figure
Interim periods	Compared as filed	Annualised before use
Risk output	Categorical opinion	Probability, score and band
Reasoning	Implicit	Per-prediction attribution
Staleness	Undetected	Version-tracked and surfaced
Applications	One bank at a time	One submission, many banks
Audit trail	Electronic mail	Append-only domain events

The comparison is not a claim that the platform decides better than an experienced officer. It is a claim about the inputs to that decision: the officer receives figures with a recorded provenance, a quantified risk with a stated reason, and a complete history of the application, in place of a folder of documents and an electronic mail thread. The judgement remains where it was.

XII. LIMITATIONS AND THREATS TO VALIDITY

The most significant limitation concerns the training data. Both classifiers were trained on synthetic records constructed to reproduce the statistical relationships documented in the credit-scoring literature. The reported measurements therefore characterise the behaviour of the models on that distribution and must not be read as real-world credit accuracy. The pipeline was built so that substituting real lending outcomes changes the training data and nothing else: the feature construction, the gate, the versioning and the workflow are independent of where the labels come from.

Four further limitations should be stated plainly. Extraction accuracy depends on document quality, and a poorly scanned statement produces a proposal that costs the company more to correct than to enter; the confirmation gate bounds the consequences of extraction error but does not remove the effort. Forecast intervals are wide where the available history is short, which is the normal case for a small enterprise, and the intervals are shown rather than hidden for that reason. The anomaly evaluation used planted irregularities of four known types, so the measured detection rate speaks to those types and not to adversarial behaviour designed to evade the detector. Finally, the platform has been exercised at demonstration scale against an isolated database, so the measured response times describe this configuration and not a production deployment under concurrent load; a formal security assessment and a load study would both precede any such deployment.

XIII. CONCLUSION AND FUTURE WORK

This paper presented FinIntel AI, a platform in which corporate financial analysis, credit-risk assessment, explanation, forecasting, anomaly detection and a bank-routed lending workflow are all derived from a single set of figures that the subject company has confirmed. The confirmation gate and the accompanying data-versioning scheme make the provenance of every result explicit and make staleness visible; the reporting-period model removes a systematic distortion in period-mixed analysis; and the separation between scoring and deciding keeps the lending judgement with a named officer. Evaluation on held-out data shows classifier performance consistent with the literature for this model family, and the credit model separates observed default rates across its bands by a wide and monotone margin, which is the property that makes the output usable inside a lending process.

The clearest direction for further work is retraining on real lending outcomes obtained under an appropriate data-sharing agreement. Beyond that, direct integration with banking and tax-filing interfaces would remove the upload step altogether; federated training across institutions would let models improve without centralising borrower data; a calibration study would establish whether the published probabilities are trustworthy in absolute terms rather than only as a ranking; and extending the period model to consolidated group accounts would widen the range of businesses the platform can serve.

REFERENCES

- [1] E. I. Altman, "Financial ratios, discriminant analysis and the prediction of corporate bankruptcy," *The Journal of Finance*, vol. 23, no. 4, pp. 589–609, 1968.
- [2] J. A. Ohlson, "Financial ratios and the probabilistic prediction of bankruptcy," *Journal of Accounting Research*, vol. 18, no. 1, pp. 109–131, 1980.
- [3] S. Lessmann, B. Baesens, H.-V. Seow, and L. C. Thomas, "Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research," *European Journal of Operational Research*, vol. 247, no. 1, pp. 124–136, 2015.
- [4] X. Dastile, T. Celik, and M. Potsane, "Statistical and machine learning models in credit scoring: A systematic literature survey," *Applied Soft Computing*, vol. 91, art. 106263, 2020.
- [5] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, 2016, pp. 785–794.
- [6] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A highly efficient gradient boosting decision tree," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 3146–3154.
- [7] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 4765–4774.
- [8] S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal, and S.-I. Lee, "From local explanations to global understanding with explainable AI for trees," *Nature Machine Intelligence*, vol. 2, no. 1, pp. 56–67, 2020.
- [9] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bénéttot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," *Information Fusion*, vol. 58, pp. 82–115, 2020.
- [10] N. Bussmann, P. Giudici, D. Marinelli, and J. Papenbrock, "Explainable machine learning in credit risk management," *Computational Economics*, vol. 57, no. 1, pp. 203–216, 2021.
- [11] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *Proc. 8th IEEE Int. Conf. on Data Mining*, 2008, pp. 413–422.
- [12] S. J. Taylor and B. Letham, "Forecasting at scale," *The American Statistician*, vol. 72, no. 1, pp. 37–45, 2018.
- [13] D. Bholat, J. Lastra, S. Maram, and K. Tanaka, "Credit risk analytics and the supervisory use of machine learning," *Journal of Banking Regulation*, vol. 21, no. 3, pp. 207–221, 2020.
- [14] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature Machine Intelligence*, vol. 1, no. 5, pp. 206–215, 2019.