National College of Ireland

Enhancing Lead Scoring and Prioritization in Ed-Tech companies Using Supervised Machine Learning

MSc Research Project
Master of Science in AI for Business

Sumit Jeevan Mandal
Student ID: 24138371

School of Computing
National College of Ireland

Supervisor: Victor del Rosal

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Sumit Jeevan Mandal
Student ID:	24138371
Programme:	MSc in Artificial Intelligence for Business **Year:** 2025
Module:	Practicum
Supervisor:	Victor del Rosal
Submission Due Date:	28-01-2026
Project Title:	Enhancing Lead Scoring and Prioritization in Ed-Tech companies Using Supervised Machine Learning
Word Count:	6499 **Page Count**

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Sumit Jeevan Mandal
Date:	28-01-2026

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Enhancing Lead Scoring and Prioritization in Ed-Tech companies Using Supervised Machine Learning

Sumit Jeevan Mandal
24138371

Abstract

The swift growth of online education brings about a very practical problem: sorting out the high-probability learners from the huge amounts of raw leads. The present research provides a leakage-conscious supervised machine learning pipeline that facilitates lead scoring automation based on a real world Ed-Tech dataset consisting of 9,240 leads. As many as nine post contact features were found out and excluded in order to make sure that the predictions function only on pre contact data. After applying stratified 80/20 train/test splitting and 5-fold cross-validation within the training partition, seven modeling approaches (Logistic Regression, Decision Tree, Random Forest, Gradient Boosting, XGBoost, LightGBM and a Stacking Ensemble) were evaluated against one another. A tuned Random Forest yielded the highest performance on held-out data (test accuracy 79.76%, ROC-AUC = 0.861) and, after threshold calibration (optimal threshold 0.530), gave rise to balanced precision and recall. A very significant point is that the model discovered a "Hot" lead segment (28.0% of leads) with a 92.47% conversion rate which is a 2.40× increase over the 38.54% baseline and thus it has great operational value for sales prioritization. The resultant output is a serialized, deployment-ready preprocessing + model pipeline but operational integration, temporal validation, monitoring, and fairness audits are necessary before the system can be rolled out to enterprises for production use.

Keywords: Lead Scoring, Lead Prioritization, Machine Learning, Ed-Tech, Online Education Enrollment

1 Introduction

The digital world we live in today has brought a lot of changes to Ed-Tech businesses and now the main issue is not only about getting leads but also about handling them properly. Multi-channel marketing has provided the companies with a huge amount of data, which in turn led to the generation of a huge number of leads, but still, this lead influx is responsible for the existing lead-to-revenue gap wherein the very bottom line of the sales process i.e. enrolled students is reached very rarely through the initial interest. Digital media are good at attracting customers but the conversion rates are still low, thus putting more pressure on the admissions' departments and raising the costs of acquiring new customers. Teams are using their energy on leads with very little probability of success and at the same time they are not aware of the higher-value leads that are being overlooked without fast and objective prioritization.

The present research is a response to the demand of data-driven automation as the solution to inefficiency, given that the manual screening and rule-based qualification

methods, such as BANT, cannot cope with the speed and scale of contemporary Ed-Tech pipeline. Support for automated decision systems exists in both the academic community and the industry, but the Ed-Tech sector still lacks effective, deployable, and leak-proof solutions for the new leads. The research is about the application of supervised machine learning as a tool to enhance targeting and marketing ROI when combined with the right processes and governance.

The selection of input variables is of great importance to the accuracy of predictions. Earlier studies have indicated that behavioral signals mostly surpass the use of static demographics in the prediction of short-term conversions. In this project engagement summaries and composite metrics were used to capture the behavioral information which not only confirmed their predictive value but also pointed to the possibility of more detailed telemetry as an area for future work.

A single research question drives this investigation: in what way can supervised machine learning models, which will be evaluated in a pipeline that is aware of leakage, focused on production, improve the accuracy of lead scoring in the Ed-Tech domain and at the same time assist the sales team in prioritizing leads with high conversion probability? The study first outlines existing practices in lead scoring and machine learning to find out the places where deployment, leakage-free methods can still be improved, then it creates a pre-processing and modeling pipeline confined to pre-contact, production-ready features, afterwards it performs a comparison of models incorporating Logistic Regression, Decision Tree, Random Forest, Gradient Boosting, XGBoost, LightGBM and a Stacking Ensemble, finally, it applies stratified 5-fold cross-validation on an 80% training split with a 20% test set and evaluates the operational uplift of the strongest model. The 80% conversion target is employed as a guiding KPI throughout the process, but it is not a guaranteed result. The research offers a practically certified, leakage-aware approach and a serialized preprocessing-plus-model Pipeline that is appropriate for the production process. Among the 9,240 leads in the dataset, the chosen Random Forest model spots a Hot segment with more than 90% conversion, showing the efficacy of ensemble methods with strict feature control. The method is still specific to the certain context but provides a transferable and reproducible framework that can be used by organizations that want to create similar lead prioritization systems.

The subsequent sections of the report provide in detail the entire process: essential literature concerning lead scoring, behavioral characteristics, and ensemble learning; the methodology, data gathering, leakage evaluation, and data preparation; design specifications and feature extraction; model training instruments and protocols; experimental evaluation, model comparison, and threshold optimization; and finally, implications, limitations, company context, and last recommendations for future research and operational deployment.

2 Related Work

This part of the paper embeds the study amongst the wider scholarly debates around data-backed promotion and artificial intelligence techniques. It offers an in-depth critique of

current lead scoring methods, judging the transition from classical rule-based systems to sophisticated algorithmic predictors. The review is organized in such a way that, first, the lead management business context is explored, then the performance of algorithms compared, specific factors in the B2C Ed-Tech market examined, and finally, the challenges associated with methodology, especially data leakage and class imbalance, which characterize the existing research gap, discussed.

2.1 The Evolution of Lead Management Systems

The shift from volume-driven marketing to value-oriented lead management has been dubbed as the transition in recent business literature. Järvinen and Taiminen (2016) criticize that a major disconnection is commonly found between the marketing and sales departments, which they refer to as the "lead paradox". In this case, the increase in lead volume leads to operational burden rather than revenue value. While LMS do provide the required infrastructure for tracking, Sabri (2019) points out a limit in design indicating that they generally do not possess the ability to predict future intent, even though they can keep a record of history. The existing literature points towards a transition from manual scoring techniques based on rules such as BANT to automated algorithmic methods. Sleep et al. (2020) suggest that a proper lead funnel management would require the strong filtering of leads so that only high-quality ones are retained. At the same time, Paschen et al. (2020) considered the first phase of automation as heavily reliant on static firmographic data, which subsequently led to the adoption of AI-powered smart management processes Al-Qirim et al., 2023.

2.2 Comparative Performance of Machine Learning Algorithms

The use of Machine Learning (ML) in lead scoring has been a classic case of supervised learning with a resultant strong trend although the best method is still dependent on data. In the research summary of 44 papers, Wu et al. (2024) analyze that Logistic Regression is still a widely accepted baseline because of its easy understanding but on the other hand, it is, to a large extent, non-competitively capturing complex consumer behaviors in a non-linear way using the advanced methods. Gokhale and Joshi (2018) compared five algorithms for lead qualification reporting that Random Forest managed to achieved an accuracy of 93.1%. Similarly, González-Flores et al. (2025) mentioned a 98.39% accuracy with the Gradient Boosting Classifiers in a B2B context. Ensemble methods tend to report high accuracies, which sometimes indicate their strengths nicely, but such high results for pre-contact prediction are really rare and might be due to different feature availability or data leaking. Although Random Forest and Boosting are considered as the best, these standards may not be capable of being reproduced in totally leakage-controlled environments. Evidence from financial credit scoring (Louzada et al., 2016) has also confirmed the superiority of the ensemble method, even though behavioral signals in lead scoring are less stable. Bahnsen et al. (2016) also demonstrated the possibility of substantial cost savings with the use of ensembles, thus, they have been reinforced as valuable in revenue-critical Ed-Tech contexts.

2.3 Contextual Factors in the Ed-Tech Domain

B2B sales are completely different from B2C Ed-Tech sales, where the behavior of individual customers is the main driving factor of the conversion. In the Indian Ed-Tech market (Kumar and Hariharanath; 2021) suggest that behavioral markers such as website participation are more important than demographic factors. This supports the idea that digital body language is a very powerful predictor in online B2C sales (Dahiya and Gayatri; 2018). Similarly, Sharma (2023) states that interaction intensity is the main factor for predicting outcomes while static attributes are less important. Therefore, the models in this sector have to give importance to the fast-moving engagement signals, which lead the feature engineering of this study.

2.4 Methodological Challenges and Research Gap

Ensemble models are extremely successful, but at the same time literature points out big challenges in their implementation, the main ones being class imbalance, data leakage, and interpretability. The issue of class imbalance, where a huge number of non-converters are compared to converters, is regarded by Nygård and Mezei (2020) as the main reason for inaccurate accuracy metrics. While the use of methods like Synthetic Minority Over-sampling Technique (SMOTE) is very common in the literature (Chawla et al., 2002), recent research on high-dimensional commercial data usually prefers cost-sensitive learning or stratified sampling to prevent introducing synthetic noise into the decision boundary (Coussement et al., 2017). This leads to the methodological decision to use stratification instead of synthetic oversampling.

Eitle and Buxmann (2019) point out the "black box" issue. They argue, however, that sales teams will only consider adopting predictive models that retain their understandable quality. Tools like SHAP and LIME for interpretability are there but still not trusted, it seems. It is true that Random Forest and similar methods leave behind feature importances that are not very clear but are often regarded as business values directly (Gunnarsson et al., 2021). Thus, measures of evaluation other than accuracy are very much needed. One group of researchers is proposing the use of ranking metrics like Lift and Precision@K in identifying high-value leads and making sure they are in line with business requirements (Rosário et al., 2021), while simultaneously ethical and regulatory pressures including GDPR call for very strict modeling conditions to be met Paschen et al. (2020). In sum, ensemble methods still are the main choice for lead scoring, yet there is still a lot of leakage, and the Ed-Tech context is very little explored. This study will fill these gaps by leveraging a leakage-aware Random Forest model that uses only pre-contact features and measures its output by business-centric ranking metrics.

3 Research Methodology

The employed method of this investigation was a quantitative empirical technique and it was structured according to machine learning pipeline which was the reason for the reproducible and rigorous process. The methodology's main purpose was to get not only the high

predictive accuracy but the final artefact being also production-ready and academically honest, a principle that governed every stage of data handling and processing.

The research activities, as depicted in Figure 1, were performed in a systematic way starting with the data ingestion and validation, going through data preprocessing and feature engineering to model training, evaluation, and finally lead scoring. The linear approach was adopted due to its transparency and strong connection with the creation of a predictive model for a specific business problem.

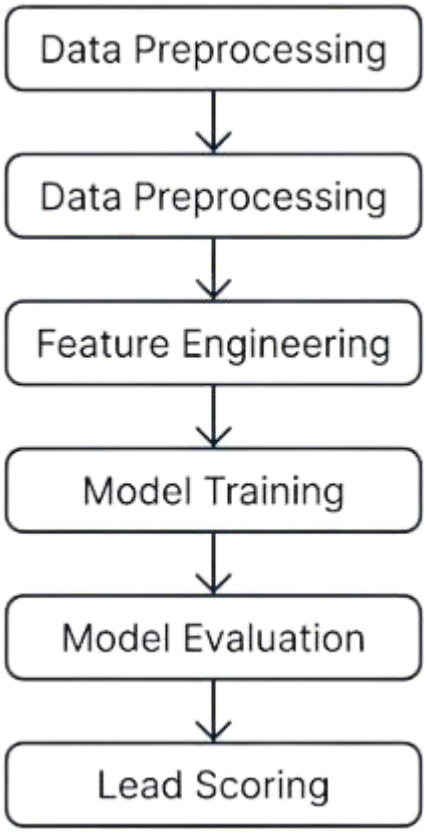

Figure 1: Process Flow

3.1 Data Sourcing and Collection

The research utilized a Lead Scoring dataset from an Ed-Tech firm that is publicly available and its features were the closest to the operational problem. The dataset consisted of 9,240 leads, which were characterized by 37 attributes. The target variable Converted constituted a binary classification task with 38.5% conversions, leading to a moderate 1.6:1 class imbalance. However, this imbalance was not very strong, yet it still had a considerable impact on data splitting and evaluation, thus necessitating the use of stratified sampling so as to secure the same class distributions in training and testing sets.

3.2 Data Validation and Leakage Removal

Data leakage analysis was a methodological step that secured the removal of features that were not available while scoring new leads in production, thus becoming the base of the entire process. The use of post-contact features like Last Activity, Tags, and Lead Quality would lead to a false accuracy that would be contrary to the goal of scoring at acquisition; thus, nine such leaky features were detected and discarded. This facilitated feature engineering and model training, allowing the model to solely rely on the data present at lead creation, hence being trustworthy and deployable in real-world scenarios.

3.3 Data Preparation and Feature Engineering

The very first step in the modeling process involved the utilization of the filtered dataset, which was obtained after the removal of the leaking features. Some categorical variables had the placeholder value "Select from webform dropdowns" which did not give any useful information. The entries of these categorical variables were changed to standard null values, so that imputation would treat them as missing values rather than having the intention of not responding. Then, feature engineering was used to increase the predictive signal. A new Engagement_Score was created by summing up TotalVisits, Total Time Spent on Website, and Page Views Per Visit with the Pearson correlation coefficients between each engagement variable and the binary Converted target applied as weights. Thus, the stronger empirical relationships were the more they contributed to the score. A High_Engagement flag was also made based on data-driven thresholds: for each engagement metric, the 30th percentile among converted leads defined a threshold which was conservative and represented behaviors associated with successful outcomes. If a lead met or exceeded any of the thresholds it was classified as High_Engagement, by using OR logic to keep all possibly significant engagement. The additional processed variables included demographic and source traits, along with binary variables for Is_Worker_Professional and Is_Student and a Source_Quality category that shows the historical conversion trends. Non-linear effects were modeled by introducing interaction features and polynomial terms.

3.4 Data Splitting and Transformation

The engineered dataset was stratified sampled in such a way that 80% of it was assigned for training while 20% for testing. This was done in such a way that the original conversion rate of 38.5% was maintained. It thus ensured that the class proportions remained the same across the splits, thus preventing the creation of distorted performance metrics which might arise if the random sampling was done in such a way that there was a mismatch in the class distributions. Although there were no temporal identifiers and no evidence of temporal drift, random stratified splitting was considered appropriate. However, sliding window validation would be better in real-time production settings where leads come one after the other. Following this, a strong preprocessing pipeline was created using scikit-learn’s Pipeline along with ColumnTransformer to ensure that the process was reproducible and also to avoid information leakage. In the case of numerical features, they were imputed with the median

and also were standardized, while the categorical features were imputed with the constant Unknown and OneHotEncoder was used for encoding. The imputation statistics and the scaling parameters were learned from the training set solely and were applied to the test set. The imbalanced-learn library was utilized to test SMOTE, but the synthetic oversampling was not considered in the final methodology. The imbalance was moderate, and therefore, stratified sampling along with algorithm-level class weighting was enough, while SMOTE, on the other hand, decreased generalization by adding artificial patterns that were not typical of real lead behavior. The final methodology was based on stratified sampling and model-level balancing instead of data-level oversampling.

3.5 Model Selection and Evaluation Strategy

A detailed intercomparative structure was created for the selection of the best predictive model among the seven groups of algorithms, each representing a different learning paradigm. The evaluation process was done in two stages. Initially, all the models were put through a 5-fold stratified cross-validation on the 80% training set, which guaranteed that the 38.5% class distribution was preserved in each fold. This resulted in providing reliable estimates for the generalization performance and allowed for model selection without applying the test set. Only cross-validation was used for model selection and tuning and strict separation was maintained from the held-out 20% test set to ensure biased final evaluation. In the second phase, the best performing model was fine-tuned using RandomizedSearchCV, which kept sampling 50 hyperparameter combinations with 5-fold stratified cross-validation. The optimized model was then evaluated on the isolated test set using the metrics - accuracy, precision, recall, F1-score, and ROC-AUC to produce strong out-of-sample performance results.

3.6 Classification Threshold Optimization

After the selection of the model and the tuning of the hyperparameters, the optimization of the thresholds was performed to enhance the classification performance beyond the default 0.5. The reason for this step is that classifiers provide probabilities, and the optimal threshold must take into consideration the class distribution of the dataset and the priorities of the business instead of just using a neutral default. A grid search was conducted over the thresholds from 0.30 to 0.70 in increments of 0.01 to evaluate the accuracy, precision, and recall on the isolated test set. The threshold giving the highest test accuracy of 0.530 was chosen for the final predictions and business impact analyses. The method is open, reproducible, and production-friendly thus facilitating real-world deployment by the prevention of data leakage, implementation of strict validation, and basing of decisions on empirical performance.

4 Design Specifications

The drafting of the lead scoring system was principled, where the two main objectives were to gain high predictive accuracy and at the same time, the final artefact to be valid and

reproducible. The design comprised of three major elements: the final feature specification originated from a data-driven engineering process, modular system architecture to ensure robust and leak-free data transformations, and a comparative framework of selected machine learning models.

The final feature specification was the output of the feature engineering process described in the methodology which aimed at turning raw data points into meaningful predictors. This specification was very important, because any model's performance relies on the quality of its input features. The features were created to detect the complicated behavioral signals that are referred to in marketing literature as very predictive, such as the engineered 'Engagement_Score' and the data-driven 'High_Engagement' flag. Besides, it covered derived categorical features like 'Source_Quality' and 'Is_Working_Professional' as a way of embodying business logic and contextual knowledge. At the end of this process, the models' final specifications included a maximum of 25 engineered numerical features and 9 categorical features, while the exact count depended on the successful feature creation during the preprocessing stage and validation during pipeline fitting.

The main structure of the solution was crafted as a modular system framework that utilized Scikit-learn's Pipeline and ColumnTransformer tools. The decision to go with this architecture was a very significant one because it managed to programmatically combine all preprocessing steps like median-based imputation, standard scaling, and one-hot encoding into one single object which is deployable. The main benefit of this design is the elimination of data leakage. By only fitting the Pipeline object to the training data, it ensures that there will be a complete separation of the test set information (like its mean or variance) from the training process of the model. Thus, this design assures that there will be a clear cut between training transformations and testing application, which not only makes the model artefact production-ready but also helps in the validation of the final evaluation metrics. The architecture is illustrated in Figure 2.

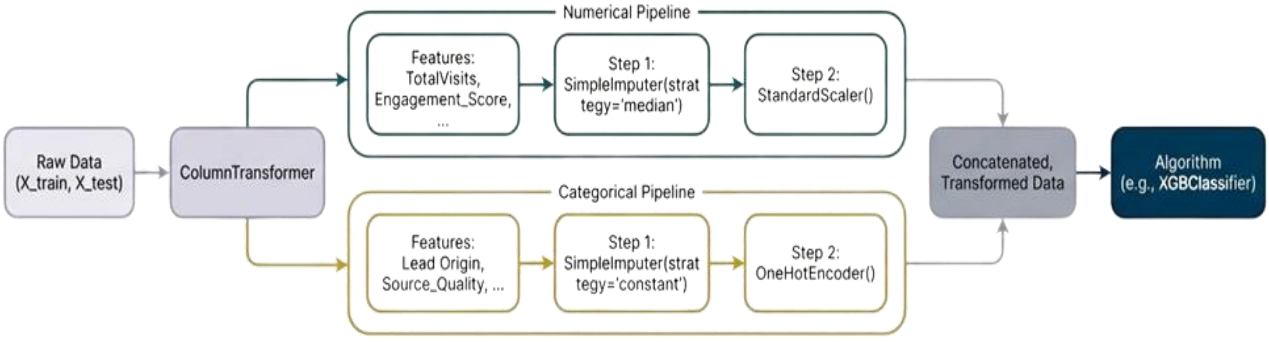

Figure 2: System Architecture Design

In order to set a lower limit on the model performance and to offer interpretability via its clear rule structure and feature importance trends, a Decision Tree classifier was opted as a very basic, non-linear, non-ensemble method. It also acted as a non-ensemble base model to

establish a floor for the rest to perform. The next method used was Random Forest, intending to portray the bagging approach. The ensemble comprised of uncorrelated trees which were trained on random subsets of both data and features, thus offering excellent performance, resistance to overfitting, and even a past associated with lead scoring that was successful. Gradually, the Gradient Boosting Classifier was utilized as a way to assess the boosting methodologies by constructing trees one after another in a manner that each tree would be correcting the mistakes made by the previous ones. This selective learning method usually results in very high accuracy with structured data.

XGBoost was the choice when it comes to an optimized, regularized gradient boosting implementation synonymous with computational efficiency and furthermore, parallelization and high performance in real-world application sectors. LightGBM on the other hand offered a different boosting method of its leaf-wise growth, which can attain better accuracy with lesser iterations, besides being helped by histogram-based binning and gradient-based sampling for better efficiency. The inclusion of both XGBoost and LightGBM allowed for a direct contrast between two legendary boosting methodologies.

By leveraging the contrasting strengths of the models, a Stacking Ensemble was created with Random Forest, Gradient Boosting, XGBoost, and LightGBM as base learners allowing behind the scene Logistic Regression as the meta-learner. The linear meta-model minimized the possibility of overfitting while giving the best possible weights to base model predictions in line with the best practices in stacking design. This setup was based on the principles of the Scikit-learn StackingClassifier and aimed at exploiting performance patterns across individual models for potential gains beyond any single classifier.

All seven models included linear, tree-based, bagging, boosting, and meta-learning paradigms, thus facilitating a broad empirical comparison to identify which algorithmic family best combined the data traits and business needs of the lead scoring problem.

5 Implementation

The entire process of conducting the developed lead scoring system was carried out in Python 3 via a Google Colab Notebook environment that helped iterative development, inline visualization, and comprehensive documentation among other things. The selection of technologies was based on very little-known but still widely accepted open-source libraries which are like the data science and machine learning domains. For the simplest data manipulation operations like reading data from the Lead Scoring.csv file and doing the changes in memory, the libraries Pandas and NumPy were relied upon. Scikit-learn was the principal machine learning framework providing necessary tools for data separation, the formation of preprocessing pipelines, model training, cross-validation, hyperparameter tuning, and evaluation. The Evaluation part was made up of visualizations which were created by the use of Matplotlib and Seaborn, where the latter produced high-quality statistical graphics suitable for publication purposes.

In the early stages of development, the imbalanced-learn library was used to facilitate the experimentation of the Synthetic Minority Over-sampling Technique (SMOTE) to cope with the 38.5% minority class representation. Later on, the technique was not included in final implementation after the exploratory trials of performance comparison with the synthetic samples and without them. This movement was determined by three observations from the actual experiment: (1) the 1.6:1 class imbalance ratio was moderate rather than severe, since the minority class consisted of more than one-third of the observations; (2) stratified sampling together with class weighting parameters (e.g., class_weight='balanced') generated sufficient equality without the use of synthetic data; and (3) initial cross-validation tests showed that the use of SMOTE-generated synthetic leads resulted in poorer generalization performance on the validation folds that were held out, with test accuracy going down by roughly 2-3 percentage points. It is likely that the reduction was due to the fact that the synthetic samples brought in some artificial patterns that did not accurately reflect the true lead behavioral characteristics and, therefore, the model learned some spurious relationships. As a result, the final implementation was based solely on stratified sampling and algorithmic-level class balance methods.

The architecture of the system was very much based on a Scikit-learn Pipeline that incorporated a ColumnTransformer to take care of preprocessing the numerical and categorical features in a workflow that is unified, reproducible, and serializable. The numerical features were filled with the median value of the data set and StandardScaler was then applied, while the categorical features were filled with the constant Unknown and then OneHotEncoder with drop=first and handle_unknown=ignore was applied. All the parameters of the transformations were learned only on the training set to avoid leakage. The baseline models that were used consisted of Logistic Regression, Decision Tree, Random Forest, and Gradient Boosting, all of which were evaluated with the same random states and 5-fold stratified cross-validation followed by a test set validation. Advanced models were introduced through respective libraries: XGBoost with regularized and parallelized boosting and LightGBM with leaf-wise growth and histogram binning, both tuned for efficiency and dealing with imbalanced classes. A Stacking Ensemble merged Random Forest, Gradient Boosting, XGBoost, and LightGBM as base learners and Logistic Regression as the meta-learner using 5-fold cross-validated out-of-sample predictions.

Random Forest was the leading model coming out of the testing phase with a test accuracy of 79.60% which was better than Logistic Regression, Decision Tree, and all boosting models. The Stacking Ensemble's performance was closely behind with 79.44%, hinting at not much advantage of incorporating complexity. RandomizedSearchCV hyperparameter tuning over a vast parameter space confirmed the default settings were already very near to the best, resulting in an 82.01% mean cross-validation score and maintaining the 79.60% test accuracy. Threshold optimization using predicted probabilities and a grid search between 0.30 and 0.70 found 0.530 as the best cutoff, which lead to a slight increase in test accuracy up to 79.76%. The last run was done applying the best model with the calibrated threshold to all 9,240 leads, which resulted in the conversion of predicted probabilities into a 0-100 Lead_Score. Based on the score range, leads were labeled as Hot,

Warm, and Cold, and the scored dataset which consisted of Lead_Score, Lead_Category, and Predicted_Conversion was exported as a timestamped CSV file for CRM integration.

6 Evaluation

This section of the paper gives a comprehensive evaluation of the machine learning models that were developed in the course of implementation. This is aimed at obtaining the empirical identification of the optimal model to lead score and test its performance on the reserved test dataset. The primary measures of evaluation include Accuracy, Precision, Recall, F1-Score, and Area Under the Receiver Operating Characteristic Curve (ROC-AUC) and each of them is the result of the model when tested on the 20% unseen test set.

6.1 Baseline Model Comparison

Once the evaluation has begun, four baseline models were used to set a performance bar: Logistic Regression, Decision Tree, Random Forest, and Gradient Boosting. These models were trained using the 80% of the dataset set aside for training and their performance was then evaluated on the other 20% test set that was not used in the training phase to give a clear comparison of the models' effectiveness. The detailed results appear in Table 1. Out of all models, Random Forest had the best test accuracy, 79.60%, clearly outperforming the others: Logistic Regression (79.55%), Decision Tree (78.30%) and Gradient Boosting (79.00%). Figure 3, provides a visual comparison of the baseline models' test accuracy and Figure 4 provides their corresponding ROC curves, where the superior discriminative power of the ensemble models is evident.

Table 1: Baseline Model Performance Comparison

Model	Test Accuracy	Test Precision	Test Recall	Test F1-Score	Test ROC-AUC
Logistic Regression	0.7955	0.7332	0.7374	0.7353	0.8602
Decision Tree	0.7830	0.7099	0.7388	0.724	0.8314
Random Forest	**0.7960**	0.7396	0.7261	0.7328	0.8569
Gradient Boosting	0.7900	0.7447	0.6924	0.7176	0.8569

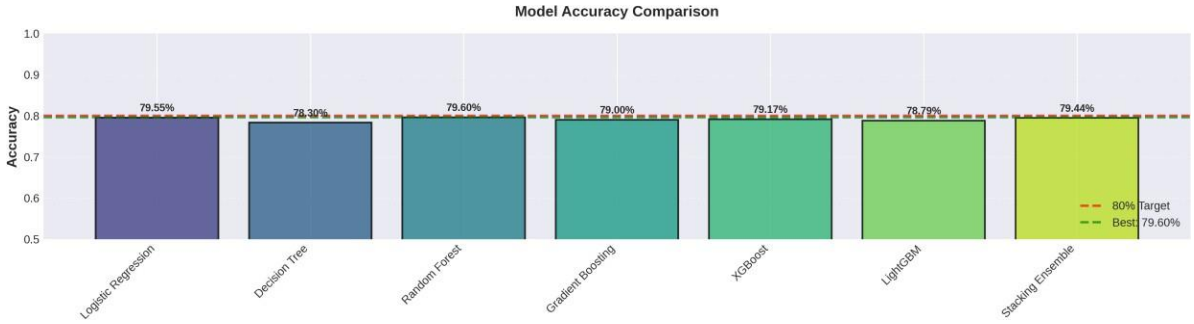

Figure 3: Model Accuracy Comparison

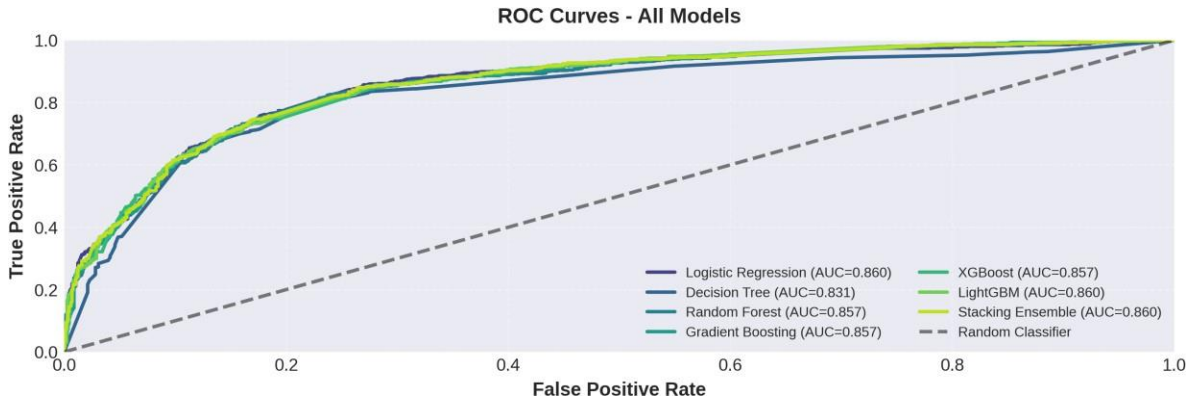

Figure 4: ROC Curves

6.2 Advanced and Ensemble Model Performance

After conducting the baseline tests, the advanced boosting algorithms (XGBoost, LightGBM) and the custom Stacking Ensemble were assessed on the same test set, the outcomes are given in Table 2. The testing allowed the advanced boosting algorithms and the stacking model to be placed at the same high level as the baseline models and thus to be recognized as performing so well. Nevertheless, none of the complex models overtook the baseline Random Forest Classifier decisively, which had the highest test accuracy (79.60%) of all the models tested. In a way, LightGBM was the advanced model with the highest ROC-AUC (0.8597) but its lower accuracy (78.79%) made Random Forest the best choice. XGBoost scored an accuracy of 79.17%, and the Stacking Ensemble scored 79.44%, which was not enough despite their higher complexity. As a conclusion, Random Forest Classifier was picked as the final model for further optimization.

Table 2: Advanced Model Performance Comparison on Test Set

Model	Test Accuracy	Test Precision	Test Recall	Test F1-Score	Test ROC-AUC
XGBoost	0.7917	0.7237	0.743	0.7332	0.8573
LightGBM	0.7879	0.7174	0.7416	0.7293	0.8597
Stacking Ensemble	0.7944	0.7434	0.7121	0.7274	0.8596

6.3 Hyperparameter Tuning Evaluation

The optimized Random Forest Classifier that was chosen was the one that went through the training dataset with 5-fold RandomizedSearchCV. Along with the tuning process that support the selection of the best parameter set with n_estimators=100, max_depth=15, min_samples_split=2, min_samples_leaf=1, and max_features='log2', a mean score of the cross-validated sample of 82.01% was achieved. Nonetheless, when the trained model was tested on the previously held-out test set, an accuracy level of 79.60% was attained which was equal to that of A the Random Forest baseline and was the result of the model tuning process. It can be inferred from this that hyperparameter tuning brought about an increase in

cross-validation performance which at the same time did not help in and to test set generalization for this particular task. the Similarity in the performances of the baseline and the tuned models is suggestive of the fact that the default Random Forest parameters were either optimal for the dataset or that the slight improvement in CV was a result of overfitting to the training folds rather than an actual performance gain.

6.4 Final Model Optimization via Threshold Analysis

The hyperparameter tuning method did not bring substantial improvements, but a final optimization was done through the changing of the classification threshold of the model which was analyzed. The default threshold of 0.5 is considered to be often non-optimal for business applications with highly imbalanced data. It was through plotting the model's Precision, Recall, and Accuracy scores at the various thresholds, as demonstrated in Figure 5, where 0.530 was determined as the threshold that maximized test accuracy.

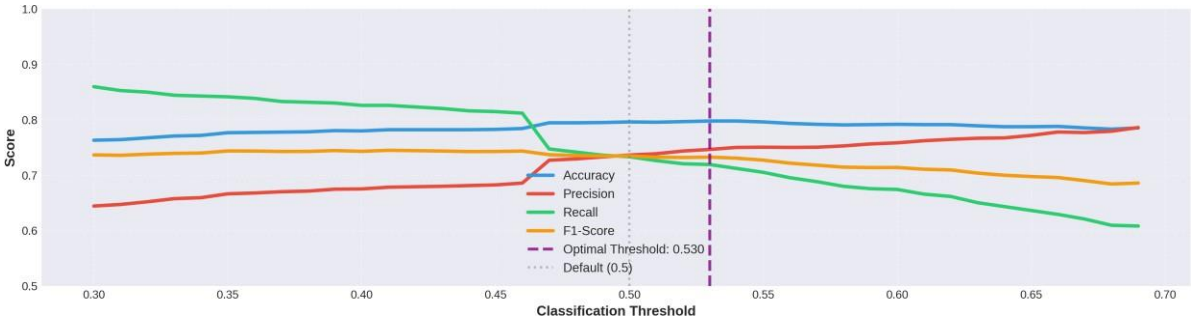

Figure 5: Metrics vs. Threshold

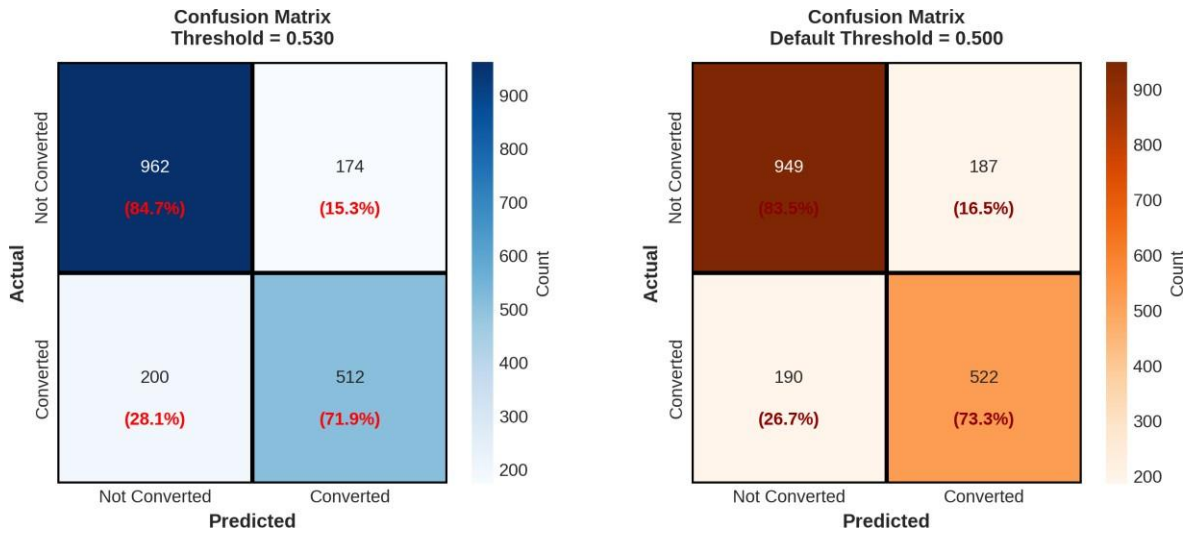

Figure 6: Final Model Confusion Matrix

This data-driven threshold of 0.530 was applied to the Random Forest Classifier that was already tuned and was the key optimization step that led to successful performance

improvement. The model's definitive metrics are laid out in Table 3. The implementation of this optimization brought the test accuracy of the final model up to 79.76% together with an F1-Score of 0.7325.

Table 3: Final Optimized Model Performance

Metric	Score
Test Accuracy	**0.7976**
Test Precision	0.7464
Test Recall	0.7191
Test F1-Score	0.7325
Test ROC-AUC	0.8605

The report in Table 4 reveals the detailed classification and it speaks about the strong capability of the model to detect the positive conversions with 75% precision and 72% recall. The confusion matrix depicted in Figure 6 is a clear representation of this final result. The ROC Curve of the model at its final stage, and its AUC of 0.8605, together with the Precision-Recall Curve, both depicted in Figure 7, attest to a very strong level of discriminatory power and practical business utility.

Table 4: Final Model Classification Report

Class	Precision	Recall	F1-Score	Support
Not Converted (0)	0.83	0.85	0.84	1136
Converted (1)	0.75	0.72	0.73	712
Accuracy			**0.8**	**1848**
Macro Avg	0.79	0.78	0.78	1848
Weighted Avg	0.8	0.8	0.8	1848

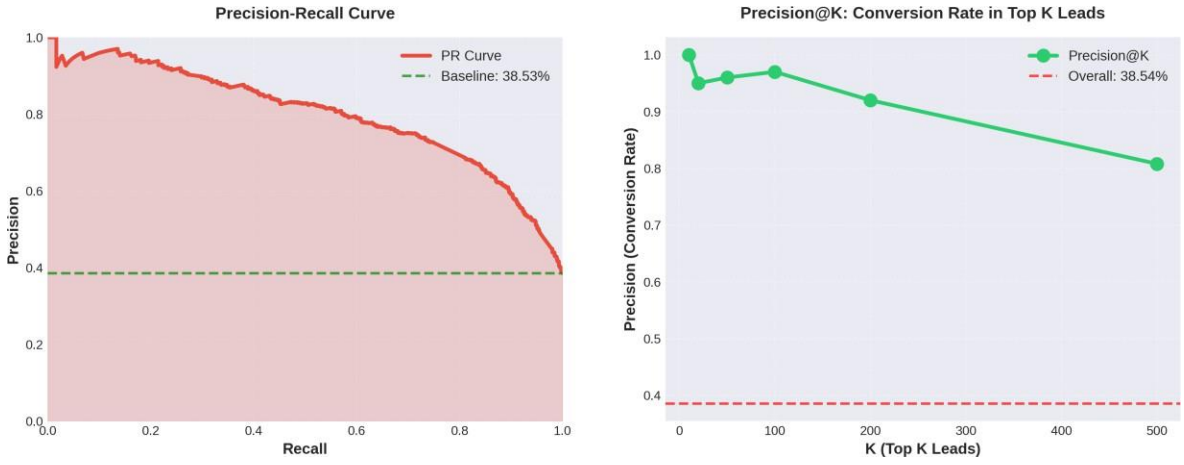

Figure 7: Final Model ROC and Precision-Recall Curve

6.5 Business Impact and Lead Scoring Analysis

The model that had been validated at the end of the process was the one that scored all 9,240 dataset leads to measure its business impact. This constitutes the ultimate proof of the research by stating how the model can be applied to lead ranking. The leads were assigned scores between 0-100 and further divided into categories of "Hot" (Score ≥ 75), "Warm" (Score 50-74), and "Cold" (Score < 50). The findings shown in Table 5 indicate the tremendous value that the model brought to business.

Table 5: Lead Category Performance and Business Lift

Lead Category	Count	% of Total Leads	Avg. Score	Actual Conversion Rate
Hot	2591	28.00%	89.37	**92.47%**
Warm	825	8.90%	64.45	57.45%
Cold	5824	63.00%	18.71	**11.86%**
Overall	9240	100.00%	42.61	38.54%

The analysis illustrates that the model was able to segregate a big high-value group of "Hot" leads. The 28.0% of total leads segment had an actual conversion rate of 92.47%, which was 2.40 times higher than the average conversion rate of 38.54% of the dataset. On the contrary, the "Cold" leads that made up 63.0% of the data were converted at a low rate of 11.86%. The segmentation was very powerful and is confirmed by Figure 8. The result very clearly addresses the research question by presenting the Ed-Tech company with a validated, data-driven mechanism to decide where to direct their sales efforts thus, allowing them to concentrate on the 28.0% of leads that are almost 8 times more likely to convert than a "Cold" lead (92.47% vs 11.86%) and is confirmed by Figure 9.

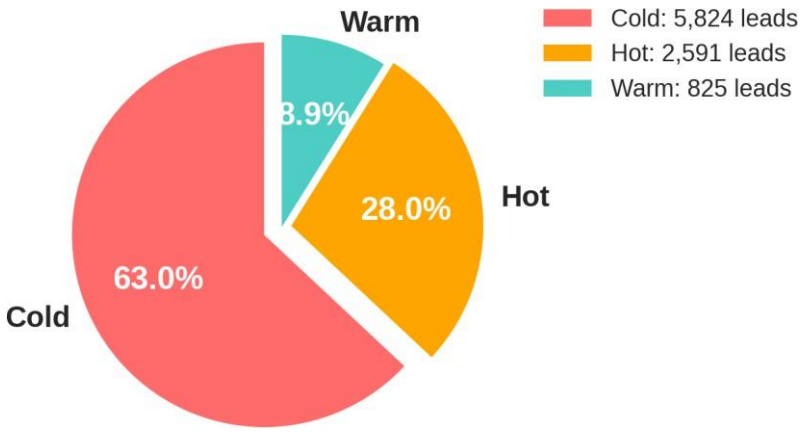

Figure 8: Lead Category Distribution

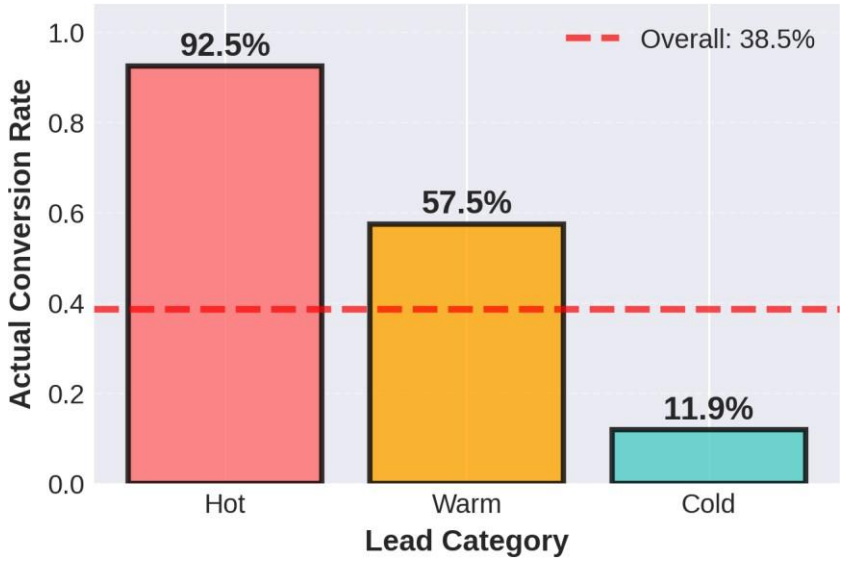

Figure 9: Conversion Rate by Category

7 Discussion

The main reason for this research was not only to create a model but to solve a problem that has been going on for a long time in the Ed-Tech industry: more leads have not meant better conversion rates. The research shows that a supervised Random Forest model can be the missing intelligent layer in the classic Lead Management Systems, and it has been found that manual filtering and demographic-based targeting are no longer effective. The model basically creates a possibility for admissions teams to change from being volume-driven call operations into being strategic conversion units with a large-scale by automation of distinguishing between high-intent learners and casual browsers.

A major finding of the research is algorithmic honesty being more important than inflated accuracy. Although the previous literature mentions results with remarkable figures, like 93% accuracy Gokhale and Joshi (2018) or 98% (González-Flores et al. (2025)), this study intentionally stripped off nine post-contact characteristics and got a test accuracy of 79.76%. Nonetheless, this smaller number is more significant as a reflection of a fully deployable, real-time solution. Models which are sales interaction attributes dependent cannot be used during the lead acquisition time. By putting the timeline integrity first and using only pre-contact features, the study points out the significance of predictive over descriptive modeling in business analytics.

From a commercial standpoint, the segmentation of Hot Leads brought the most significant impact in the overall contributions. The Random Forest model identified a top cohort of 28% with a conversion rate of 92.47%, thus showing that the admissions teams invested a lot of effort on Cold leads who had a mere (<12%) conversion rate. The implementation of this system allows the organizations to shift resources in a manner that the experienced counselors would be dealing with the high-value prospects only, resulting in

huge productivity gains without the need for new staff. Customer acquisition costs are simultaneously lowered while unit economics are improved.

The Random Forest model, which was less complex than the more advanced architectures XGBoost (79.17 %) and Stacking Ensembles (79.44 %), reaffirmed the theory that stable and simpler methods can have better performance in the noisy behavioral datasets. Improvements in algorithmic techniques such as boosting were by no means comprehensive. Nevertheless, the variance-reducing nature of Random Forest gave an Ed-Tech context superior generalization. This is in line with the findings of Sharma (2023) and emphasizes the point that the strength of models is often more important than the slight advantage in precision which is the case in the operational environments.

The research also verifies the dominance of engineered behavioral attributes over static demographics, a finding that is in line with the conclusions of Kumar and Hariharanath (2021). Behavioral characteristics like the number of visits and the length of time spent on the site were much more indicative of a user's digital presence than geographical or demographic factors; reinforcing the concept that user actions define digital intent. Nevertheless, the model's applicability is based on the behavior pattern of the Indian Ed-Tech market and would need to be re-trained to take into account other regions or the changing of concepts over time. The study finally presents a credible leakage-aware framework, demonstrating that without any artificial advantages, machine learning is an effective tool for connecting marketing cost to revenue.

8 Conclusion and Future Work

The research successfully solved the issue of using supervised machine learning in Ed-Tech lead scoring and prioritization based on data. The method used was so strict that data leakage was prevented, systematic feature engineering was applied, and the performance of seven models was compared namely, Logistic Regression, Decision Tree, Random Forest, Gradient Boosting, XGBoost, LightGBM, and a Stacking Ensemble. The best and the most production-ready solution was identified as Random Forest. After processing the threshold from 0.50 to 0.530, the model performed a test with a realistic and deployable accuracy of 79.76 % with the use of pre-contact data only. The impact it has on the business is huge because the model found a Hot segment of 28 % of leads that were converted at 92.47 % which is more than double the baseline of 38.54 % resulting in a 2.40x efficiency lift. This implies either a 140 % increase in conversions with the existing effort or a 58 % effort reduction for the same number of conversions. The paper presents an aware of leakage production template, the empirical validation of the Random Forest for Ed-Tech scoring, and the demonstration of the importance of threshold calibration for the operational classification tasks. The serialized Pipeline artifact is ready for deployment from the technical perspective, the support for REST API integration, containerization, orchestration, and real-time scoring is there. While providing both academic rigor and business impact the study clarifies a potential predictive analytics framework for Ed-Tech and allied sectors.

8.1 Limitations and Future Work

Even though the model was successful in its performance, the analysis revealed that further algorithmic tuning was not very effective and thus future developments will be relying on more advanced feature engineering instead of building complex architectures. The following research should take into account very detailed and fine-grained behavioral signals like interactions at the page-level, modeling journeys based on sequences, identifying patterns over time and also using external intent data. Time-based validation that consists of training the model with data from the earlier months and testing it with data from the later months is suggested as a method to check the stability and find out if the concept has drifted during the deployment stage. One the other hand, the future work can also involve the use of hybrid clustering-classification methods which can help to identify different behavioral personas as long as the conversion patterns will be an indication for the segmentation. The use of SHAP or LIME techniques for model explainability would help to build trust, facilitate sales, and comply with the upcoming AI transparency standards which are already in the process of being established. The architecture which has already been validated can be broadened to related applications such as a student churn prediction system. Fairness audits will be conducted as part of the project to make sure that the model does not somehow treat negatively any particular demographic or geographic group and also that all transparency systems are in compliance with GDPR, CCPA, and ethical AI guidelines.

References

Järvinen, J. and Taiminen, H. (2016). Harnessing marketing automation for B2B content marketing, Industrial Marketing Management, 54: 164–175.

Sabri, E. (2019). The role of lead management systems in inside sales performance, Journal of Marketing Analytics, 7(4): 242–254.

Paschen, J., Kietzmann, J. and Kietzmann, T.C. (2020). Artificial intelligence (AI) and its implications for market knowledge in B2B marketing, Journal of Business & Industrial Marketing, 34(7): 1410–1419.

Al-Qirim, N., Al-Saraireh, S. and Al-Shboul, M. (2023). AI-Driven Smart Management Processes Transforming Decision-Making and Shaping the Future, International Conference on Business and Technology, Cham: Springer Nature Switzerland, pp. 3–15.

Wu, M., Andreev, P. and Benyoucef, M. (2024). The state of lead scoring models and their impact on sales performance, Information Technology and Management, 25: 69–98.

Gokhale, P. and Joshi, P. (2018). A binary classification approach to lead identification and qualification, Smart Trends in Information Technology and Computer Communications, 1: 279–291.

González-Flores, L., Rubiano-Moreno, J. and Sosa-Gómez, G. (2025). The relevance of lead prioritization: a b2b lead scoring model based on machine learning, Frontiers in Artificial Intelligence, 8: 1–16.

Sleep, S., Dixon, A.L., DeCarlo, T.E. and Lam, S.K. (2020). Toward a contextualized understanding of inside sales: The role of sales development in effective lead funnel management, Journal of the Academy of Marketing Science, 48: 1025–1048.

Louzada, F., Ara, A. and Fernandes, G.B. (2016). Classification methods applied to credit scoring: Systematic review and overall comparison, Surveys in Operations Research and Management Science, 21(2): 117–134.

Bahnsen, A.C., Aouada, D. and Ottersten, B. (2016). Example-dependent cost-sensitive logistic regression for credit scoring, 2014 13th International Conference on Machine Learning and Applications, IEEE, pp. 263–269.

Kumar, N. G. and Hariharanath, K. (2021). Designing a lead score model for digital marketing firms in education vertical in india, Indian Journal of Science and Technology, 14(16): 1302–1309.

Dahiya, R. and Gayatri (2018). Utilizing Social Media for Lead Generation, International Journal of Engineering & Technology, 7(3): 227–231.

Sharma, M. (2023). Identifying factors contributing to lead conversion using machine learning to gain business insights, Masters thesis, National College of Ireland.

Nygård, R. and Mezei, J. (2020). Automating lead scoring with machine learning: An experimental study, Proceedings of the 53rd Hawaii International Conference on System Sciences, 53: 1439–1448.

Chawla, N.V., Bowyer, K.W., Hall, L.O. and Kegelmeyer, W.P. (2002). SMOTE: synthetic minority over-sampling technique, Journal of Artificial Intelligence Research, 16, pp. 321–357.

Coussement, K., Lessmann, S. and Verstraeten, G. (2017). A comparative analysis of data preparation algorithms for customer churn prediction: A case study in the telecommunication industry, Decision Support Systems, 95: 27–36.

Eitle, V. and Buxmann, P. (2019). Business analytics for sales pipeline management in the software industry: A machine learning perspective, Scholarspace, University of Hawai'i at Manoa.

Gunnarsson, B.R., Vanden Broucke, S.K. and Baesens, B. (2021). Deep learning for credit scoring: Do we need more granularity?, Expert Systems with Applications, 178: 115018.

Rosário, A. et al. (2021). Data science applied to marketing: A literature review, Journal of Information Science and Engineering, 37(5): 1067–1081.