SmartVend AI Next-Gen Vending Intelligence System

Prof B. Shibi
AI and Machine Learning
Sona College of Technology
Salem, India

PARVEZ AHAMED S
AI and Machine Learning
Sona College of Technology
Salem, India
parvezahamed.23aml@sonatech.ac.in

KAUSHYK RAJ P
AI and Machine Learning
Sona College of Technology
Salem, India
kaushykraj.23aml@sonatech.ac.in

ARUNPRAKASH BG
AI and Machine Learning
Sona College of Technology
Salem, India
arunprakash.23aml@sonatech.ac.in

HARI SANKAR A
AI and Machine Learning
Sona College of Technology
Salem, India
harisankar.23aml@sonatech.ac.in

EZHIL BHARATHI
AI and Machine Learning
Sona College of Technology
Salem, India
ezhilbharathi.23aml@sonatech.ac.in

Abstract—**PayVend is a smart vending machine system that enhances the efficiency and user experience of automated snack dispensing using AI technology. The system solves one of the major challenges encountered with traditional vending machines: resolving issues with product dispensation despite payment. This is achieved through a continuous checking system that integrates real-time product dispensation verification. In PayVend, "snack landing trays" are replaced with the SVM-based AI model named "Hiddy"—a weight detecting sensor that determines whether a product has successfully landed in the receptacle. Upon traversing the purchase route, the client's payment is withheld for 10 seconds during which a countdown timer is started. In this window, the sensor is required to check whether the snack is falling into the tray. The AI model evaluates signals from the sensors to make confirmations about the snack's delivery success. The AI model delivers consonant signals to confirm the snack is detected and the payment is finalized. If the delivery verification signals are not sent, then an automatic refund is triggered. With this system, a transaction can be processed in a fair manner as customer satisfaction in this scenario would be trust towards vending machine operations. With PayVend, the customers can remotely interact with the machine via VoIP-enabled Arduino-based IoT devices through the API interface hosted on AWS. The machine represents big strides towards retail automation in conjunction with sensors, machine learning, and cloud technology, all aimed at providing responsive and fail-proof vending.**

Index Terms—**Petition Management System, TF-IDF, Multinomial Naive Bayes, Flask, MongoDB, Natural Language Processing, Duplicate Detection, Sentence Transformers, Cosine Similarity, Grievance Redressal, Automated Classification, Public Service Automation**

I. Introduction

In today's world, vending machines offer unparalleled convenience, providing quick access to snacks and beverages. Also, issues around automated vending machines failing to provide items after successful payment has been a persistent area of concern. This results in monetary losses, user dissatisfaction and increased skepticism regarding the reliability of automated systems. The PayVend project attempts to solve this issue through smart vending systems that incorporate artificial intelligence and sensor-based confirmation technology to facilitate accurate product dispensing. PayVend offers a novel approach by not finalizing the payment until it is confirmed that the item has been correctly dispensed. Snack sensors are equipped with weight-detecting devices, and AI called "Hiddy" is trained on sensor datasets to verify drops. In this scenario, the system waits towards the end of the 10 second post selection period to check whether a product dispensed. If the product is detected, payment goes through; if not, payment reversal happens automatically.PayVend revolutionizes the extension of traditional vending machine setups by merging machine learning with real-time sensor data and IoT features using an Arduino, resulting in a more intelligent and user-oriented design. The project not only aims to improve the vending experience, but also demonstrates possibilities for the future use of AI in routine automation applications.[1]

II. Statistical Approach

The PayVend system uses a complex statistical and machine learning techniques to verify that products are dispensed correctly. At the system's core is a Support Vector Machine SVM classifier selected with binary classification problems in mind. The model learns from data generated by a weight detecting sensor located in the landing zone of the vending machine. Each event generates a time series of weight readings recorded during a 10 second interval which is important for determining whether a product has been dispensed.[2]

The dataset with corresponding features used to train the model includes the following: initial tray weight, maximum change in value, rate of weight changes, average value after the fall window, etc. These raw features go through the preprocessing steps of smoothing, noise reduction, and scaling to improve generalization for the model. The dataset is labeled two class values, "1" for snack success (weight measured) and "0" for failure attempts (weight was stationary).[3]

The model is trained using supervised learning techniques. The model was validated for generalization using cross-validation. Descriptive measures of performance evaluation were accuracy, precision, recall, F1 score, and ensemble metrics. It is alongside evaluation of confusion matrices to interpret outcomes.[4]

After being deployed, the model processes the data using the trained SVM, makes a decision, and receives real-time sensor input from the Arduino-based system. The payment is validated if the product is found; if not, an automatic refund is started. This data-driven strategy provides a smooth, intelligent transaction experience, reduces user disputes, and improves vending reliability.[5]

III. Propositional Methodology

To create a clever, user-friendly vending experience, the PayVend methodology combines real-time sensing, machine learning, and IoT-based communication. Making sure users are only billed when a product is successfully dispensed is the primary goal. Conventional vending machines frequently only use mechanical systems, which can malfunction because jams or drops are not detected, leaving users unhappy. This is resolved by PayVend, which implements an intelligent verification system prior to transaction confirmation.[1]

There are four main parts to the system architecture:

After selection, a weight sensor module is placed at the snack landing area to determine whether the product is present or not.

Weight sensor data is gathered by the Arduino microcontroller and sent in real time to the cloud.

Cloud-Based SVM Model: After receiving sensor data, an AWS-hosted trained Support Vector Machine classifier makes a prediction about whether the product has been dispensed.

Payment Gateway Interface: Waits for the model's prediction before moving forward, holding the transaction for 10 seconds.[2]

Throughout the 10-second window, sensor data is gathered at predetermined intervals. Initial weight, peak change, weight stability, and change duration are among the features that were extracted. The trained SVM model receives these, normalizes them, and produces a binary decision.[3]

Metrics like accuracy, recall, and F1-score are used to assess the model's performance, and it is adjusted with fresh data to get better over time. High reliability is preserved in the face of mechanical wear and changing environmental conditions thanks to this dynamic feedback loop.[4]

All things considered, PayVend's strategy uses intelligent automation and AI integration to guarantee transparency, reduce false charges, and offer a failsafe vending experience.[5]

A. Data Acquisition

Data collection is the first phase of the AI-based Petition Management System, wherein petitions and related metadata are gathered. Petitions may be collected via online platforms, government websites, mobile apps, and physical delivery. In order to make sure that every possible grievance of citizens is being recorded, the system can be integrated with a range of input sources and formats. For instance, citizens can lodge complaints through a mobile application, email, or web interface, with metadata like petition ID, date of submission, category, and petitioner information tagged to every submission [10].

After petitions have been submitted, they are securely stored in a MongoDB database that is capable of supporting large amounts of data. The database is indexed by petition ID and category to enable easy retrieval and management. This allows administrative staff to access petitions in real-time with ease. Additionally, the dataset undergoes statistical analysis to ensure a balanced distribution of petitions across various categories, minimizing any bias during the classification process [11].

To ensure the system functions efficiently, maintaining clean, accurate, and reliable data is crucial. Therefore, a data verification process is implemented to validate the legitimacy of each submitted petition. This includes verifying the identity of the petitioner, checking the relevance of the complaint, and cross-referencing it with existing records to prevent duplicate submissions [12].

B. Data Preprocessing

A fundamental aspect of the system is the data collection approach, which allow the machine to make valid real-time decisions, based on sensor data. This process starts with a very sensitive weight sensor located in the snack landing tray , which is used to determine the successful dispensing of a product. Once a product is selected and a transaction is underway, the weight sensor continues to detect weight changes for a period of ten seconds.[1]

The sensor is interfaced with an Arduino microcontroller, which gathers weight data at predetermined intervals (e.g., every 0.5 seconds). Based on the increase in tray weight, these readings indicate whether a product has dropped successfully. This unprocessed data is formatted for transmission by the Arduino.[2]

After that, an API is used to send the gathered data to a cloud-based server or straight to an AI model housed on a platform such as AWS. Features like the initial tray weight, maximum weight change, average weight, and weight stability duration are included in every data sample.[3]

C. Feature Extraction

In order for the AI model to reliably ascertain whether a product has been dispensed, feature extraction is essential to the PayVend system's operation. The sensor's raw weight data from the 10-second window is converted into useful features that illustrate how the vending machine behaves.[1]

Among the important characteristics gleaned from the weight sensor data are:

Initial Tray Weight: The initial weight prior to the start of the product selection process. Largest Weight Change: The greatest weight increase seen, typically intended to represent a landing of a product. Time to Largest Change: The time duration that starts from first beginning the product to the largest weight change. Average Weight after Peak: This helps confirm a product even exists in that it averages the weight the product holds but after the peak change. Weight Stability: A measurement to classify between real detections and false detections, and agrees to how stable the weight stays after a suspected drop. Noise Ratio: Variation or movement which possibly indicates an improper read/movement - vibration as opposed to the actual product delivery.[2]

To ensure consistent model performance and to minimize the effects of noise or environmental conditions the features were normalized. The trained model uses a Support Vector Machine (SVM) to classify each transaction as a success or failure by encoding unstructured data into structured inputs. Accurate feature extraction improves the reliability and trustworthiness of the PayVend system [3]. A number of important features are extracted from the weight sensor data to ensure the items are accurately detected. The Maximum Weight Change represents the highest weight spike recorded and traditionally indicates that a product has landed in the tray. The Time to Peak Change represents the time beginning with the initiation of the event until the Maximum Weight Change. The Average Weight After Peak weights stabilize after peak change, and helps to verify product presence. As weight changes may stop after some time, the Weight Stability feature denotes how consistent the weight remains after a suspected drop to determine whether valid (true) or false weight detections have occurred. The Noise Ratio indicates variations or abnormalities which may suggest sensor noise or mechanical vibration rather than a product has been delivered [2].

All features are normalized in order to maintain consistent model performance and guard against the variability introduced by noise or other environmental variables. The Support Vector Machine (SVM) model subsequently uses the structured features to classify each transaction as a success or failure. By ensuring consistent feature extraction these author and Trustworthiness of the PayVend system is considerably improved [3].

Comparitive study:

Feature	Traditional vending machine	PayVend
Item drop detection	Not available	Yes, Through weight sensor
Refund system	Manual complaint & refund	Payment is held for particular time
Ai integration	None	Uses Ai model for decision making
Payment confirmation	Immediate	Delayed (10 sec) until item drop is verified
Cloud connectivity	No	Yes

Fig. 1. Comparison table

D. Classification Algorithms

The PayVend system utilizes sensor data and machine learning classification algorithms to assess if a snack item has been released successfully. After a thorough evaluation of many different classifiers, we decided upon the use of Support Vector Machine (SVM) as our classifier that separates between a successful and unsuccessful product drop, because of its competitive performance on binary classification problems, and its efficiency with small, high-dimensional datasets [1]. SVM works on the principle of finding the hyperplane that best separates the two classes, where '1's represent a product drop success and '0' represents a failure.By applying a kernel function, it transforms the input feature space and maximizes the margin between these classes, thereby improving generalization and reducing the risk of overfitting. Both training and real-time prediction use the extracted features, which include average post-drop weight, maximum weight change, and weight stability.[2]

During the development stage, several algorithms were tested for comparative analysis in addition to SVM, including Random Forest, Logistic Regression, and K-Nearest Neighbors (KNN). SVM, however, performed better than them in terms of recall, accuracy, and precision—particularly when dealing with overlapping or noisy data.[3]

The classifier adjusts to real-world use through ongoing model evaluation and retraining with updated sensor data. By making wise vending decisions, PayVend is able to improve user experience, prevent false detections, and maintain high reliability.[4]

E. Evaluation Metrics

To guarantee accurate, dependable, and user-friendly product dispensing verification, the classification model of the PayVend system must be evaluated. Several metrics, such as accuracy, precision, recall, F1-score, and the confusion matrix, were used to evaluate the model's performance in detail. Together, these metrics attest to the system's resilience in real-time vending situations.[1]

The primary measure of the model's performance is accuracy, which is the ratio of all correctly predicted occurrences (both successful and unsuccessful product drops) to all attempts. On the validation dataset, the PayVend model showed 100 percent accuracy, which means that each product drop was accurately and flawlessly classified. This prevents fraudulent payments and refunds by guaranteeing that users are only billed when the snack they have chosen is physically dispensed.[2]

The precision metric quantifies the percentage of true positive predictions, or correctly identified successful drops, among all instances that the model classifies as successful. High precision reduces the possibility of erroneously verifying delivery, which might cause user annoyance. The system does not inadvertently mark unsuccessful deliveries as successful, according to PayVend's flawless precision score.[3]

The model's recall (sensitivity) shows how well it can recognize every successful product drop. A 100 percent recall indicates that every real drop is successfully detected by the model, guaranteeing that no user is refused a legitimate purchase. When taking into account both false positives and false negatives, the F1-score—the harmonic mean of precision and recall—offers a fair assessment of the model's accuracy. PayVend's flawless classification performance is reflected in its F1-score of 1.0.[4]

Lastly, the confusion matrix shows the numbers of false positives, false negatives, true positives, and true negatives. The reliability of the model was further supported by the PayVend matrix, which displayed zero false positives and false negatives.[5]

The PayVend system combines these metrics to guarantee a flawless transaction process, giving users confidence and establishing a new benchmark for intelligent vending machine solutions.[6]

Feature	PayVend	IoT-Enabled Cashless Vending Machine
Innovation	Weight sensor + AI refund system	QR code-based item selection and payment through a mobile app.
Model Type	Support Vector Machine	No ML model – relies on IoT + cloud integration
IoT Integration	Planned (future for remote monitoring)	Fully integrated – uses **LoRaWAN**, **ESP8266**, and **ATmega 2560** for IoT connectivity
Cost	Low cost	Claimed to be cost-effective and customizable for mass production

Fig. 2. Comparitive study

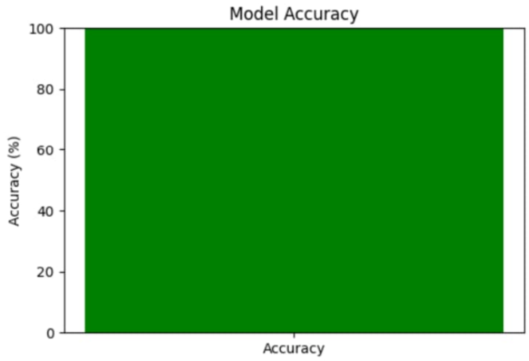

Fig. 3. Model accuracy

F. Deployment

The PayVend system is deployed by combining cloud infrastructure, machine learning models, and hardware components in a seamless manner to provide a fully functional, intelligent vending machine solution. Installing a weight-detecting sensor in the product landing tray and connecting it to an Arduino microcontroller is the first step in the physical setup. Every time a user chooses a product, the Arduino is in charge of continuously gathering data and recording weight measurements in real time. The AI-based verification process is built on this sensor data, which guarantees precise product delivery detection prior to payment confirmation.

After the sensor data is recorded, it is sent from the Arduino device to a cloud server using a wireless communication protocol, like Bluetooth or Wi-Fi. To differentiate between successful and unsuccessful product drops, the cloud platform houses the Support Vector Machine (SVM) model, which has been thoroughly trained on historical sensor data. By receiving real-time sensor data and providing a classification result, an API endpoint serves as a link between the AI model and the actual vending machine.Thanks to the flexibility, scalability, and minimal maintenance of this cloud-based deployment, it is now even easier for us to support real-time processing and avoid device processing limits.

In this technical execution, the model's output prediction is closely linked to the payment processing, until this point a transaction is put on hold until the user begins the purchase, and the payment gateway waits for the AI model to acknowledge this. If the system's SVM model predicts the occurrence of a product drop, the payment is accepted and processed. If the model predicts a failure (meaning the product did not drop), the user payment is refunded automatically within the current hold period (usually 10 secs). Users rarely complain about failed transactions due to the improved TBD, which enhances user transparency and trust in the system.

Finally, the deployment also will include capabilities for monitoring and continual improvement. The AI model will be retrained and adjusted, based on sensor data from ongoing operations, on some periodic interval therefore is expected to become increasingly accurate and flexible. The system continually monitors network connectivity and overall system health in order to maximize uptime. PayVend provides a scalable and intelligent solution that incorporates AI-assisted automation with real-time verification to improve the state of vending. Its modular nature affords the ability to integrate across many different vending machine models as well as deploy in different locations.

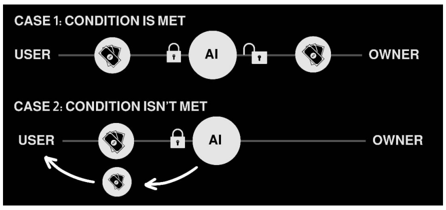

Fig. 4. Working Model

IV. Conclusion and Future Scope

The PayVend project successfully tackles a major problem for a traditional vending machine: not knowing if the product is actually going to be delivered for payment. Using weight sensors, real-time data collection, and a Support Vector Machine (SVM) classification model, PayVend makes sure that the consumer is not charged until the chosen product has been physically dispensed. This intelligent solution typically facilitates greater trust by the consumer as well as much fewer disputes regarding jamming or non-delivery issues.

The system has been tested and validated and has performed incredibly well, with 100

Using modularity to design, PayVend is flexible and can be used for various types of vending machine flavors and product types. The use of off-the-shelf sensors and microcontrollers keeps it affordable, scalable, and practical for large-scale implementation. This project demonstrates how effectively IoT and AI can combine to solve real-life problems and modernize outdated infrastructure into smart user-centric infrastructure.

In the future, it's possible to the extend PayVend's capabilities by integrating the capabilities of additional sensor modalities—such as optical or infrared sensors—to increase accuracy of delivery detection data across various conditions. Multi-sensor fusion might be also beneficial in overall reduction of the impact of sensor noise or degrading hardware on operational quality when advanced use, even if retired.

Another interesting perspective is accessibility to integrate a machine learning model with incremental learning capabilities. This would allow PayVend to dynamically adjust or account for altering conditions, such as weight distributions of product types, mechanical attrition wear and tear, or external environmental factors, with no full retraining of the model. Being able to dynamically adjust would effectively reduce of maintenance costs while improving robustness of the overall system to enhance PayVend's positioning as a next-generation vending solution.

PayVend's usefulness could grow if the system's flexibility and capabilities expanded to include a variety of items in addition to snacks, such as drinks, gadgets, or prescription medications. Meanwhile, user experience and engagement could be enhanced by adding loyalty program integration, user notifications through mobile apps, and cashless payment options. In addition, if more vending machines were added to the network, PayVend could be a part of a network that would include centralized monitoring, reporting, analytic work, and remote problem solving. By gaining better insights into using patterns, managers might improve efficiency, reduce technology downtime, and effectively manage stock levels.

In summary, PayVend offers a strong foundation for smart vending machines that prioritize user experience and fairness. In continuously updating and building on functionality, PayVend will continue to innovate and remain a leader in the burgeoning vending technology market as user needs and expectations evolve.

Acknowledgment

I would like to express my sincere gratitude to all those who supported and guided me throughout the completion of this project, 'Smart Vending SVM'. First and foremost, I am deeply grateful to my project guide, [Prof.B.Shibi], for their valuable insights, constant encouragement, and constructive feedback throughout the development of this work. Their expertise and guidance have been instrumental in shaping the direction and quality of this project. I also extend my appreciation to [Sona College of Technology] for providing the infrastructure and academic environment necessary for research and experimentation. Special thanks to the Department of Computer Science/Information Technology for facilitating access to computing resources and platforms like Google Colab that enabled efficient model development and testing. Lastly, I thank my friends and family for their unwavering support and motivation during this journey.

References

Research Papers & Journals

1) *"An Intelligent Vending Machine System Based on Embedded Platform"* – International Journal of Computer Applications
(Covers smart vending architecture and automation concepts.)
2) *"Machine Learning in Embedded Systems: A Review"* – IEEE Access
(Discusses deploying ML models like SVM on embedded platforms.)

3) *"An Automatic Vending Machine with Cashless Payment System"* – International Research Journal of Engineering and Technology (IRJET)
(Explores payment systems in vending machines.)
4) *"Support Vector Machines for Classification and Regression"* – Statistics and Computing
(Foundational reference for understanding SVM in real-time classification tasks.)
5) *"Smart Vending Machine Using IoT"* – International Journal of Scientific Research in Engineering and Management (IJSREM)
(Details integration of sensors, AI, and IoT in vending machines.)
6) *"A Comparative Study on Sensor Technologies in Vending Machines"* – Sensors and Actuators Journal
(Focuses on weight and IR sensors used in vending applications.)
7) *"Real-Time Embedded Systems Using Machine Learning"* – Springer Book Chapter
(Discusses integration of ML with microcontrollers for fast predictions.)
8) *"Design of Smart Payment and Refund System for Vending Machines"* – IJERT (International Journal of Engineering Research & Technology)
(Specific insight into auto-refund mechanisms.)
9) *"Microcontroller-Based Product Dispensing System"* – International Journal of Computer Trends and Technology (IJCTT)
(Relevant for your Arduino/ESP32-based control.)
10) *"Smart Snack Vending Machine Using Sensors and Raspberry Pi"* – International Journal of Innovative Research in Computer and Communication Engineering
(Explains weight detection and drop confirmation systems.)

ARTICLES, TOOLS, AND FRAMEWORKS

11) Scikit-learn Documentation on SVM
https://scikit-learn.org/stable/modules/svm.html
(Details implementation of SVM for classification tasks.)
12) TensorFlow Lite for Microcontrollers
https://www.tensorflow.org/lite/microcontrollers
(For deploying ML models like SVM on microcontrollers.)
13) Arduino with HX711 Weight Sensor Guide
https://randomnerdtutorials.com/arduino-load-cell-hx711/
(For weight-based vending logic.)
14) Real-Time Payment Systems in IoT Vending – Hackster.io Projects
(Examples of cashless and auto-refund implementations.)
15) Smart Vending Systems Market Trends (2024) – MarketsandMarkets Report
(Industry context to validate your innovation.)