Fraudulent Check Detection

JP Morgan & Chase

Fall, 2023

Mohamed Abdelmalek, Omar Alkhawaldeh, Alex Garcia, Cassandra Hetrick

Table of Contents

Requirements: JP Morgan Chase Fraudulent Check Detection

Omar Alkhawaldeh, Alex Garcia, Cassandra Hetrick, and Mohamed Abdelmalek

Computer Science and Engineering Students
Department of Computer Science and Engineering
University of South Florida
Tampa, FL 33620

alkhawaldeh@usf.edu, alexgarcia@usf.edu, chetrick@usf.edu, mabdelmalek@usf.edu

Version 1.03
November 26, 2023

History of document

- **Version 1.00** (September 15, 2023) – Initial document created.
- **Version 1.01** (September 22, 2023) – Revised problem statement, specified fraudulent check test cases, removed the first assumption.
- **Version 1.02** (October 21, 2023) - Modified quantities in the problem statement to match requirements. Revised types of check fraud, replaced two assumptions, added an assumption, replaced one requirement, and reworded several other requirements. Additionally, modified needs and factors to reflect new requirements.
- **Version 1.03** (November 26, 2023) - after demo, we decided to remove requirement 4 since it has to do with the training. We also modified the problem statement to not mention the cost of scanning checks since it's zero. We also took out the 'not sure' bucket because both the fraud and 'non sure' bucket are going to the same destination, manual checking. Assumption 4 was removed since our team is now checking the checks against their original transaction. Added data correction definition. Added an appendix.

Table of Contents

1. Introduction

The problem addressed by these requirements is: “Can we build an ML-based scanning system that can determine if a check is fraudulent within 3 seconds with 15% false positive rate and 1% false negative rate.”

2. Glossary

Fraudulent Check: an illegitimate check that has been tampered with, altered, or fabricated.

- Signs of fraudulent checks (refer to appendix for examples):
  - Abnormal spacing.
  - Crossed-out writing.
  - Dollar amount mismatch with database record.
  - Dollar amount not matching what is written on the amount line.
  - Bleached checks.
  - Serial Number Mismatch (on top and bottom of check).

True Positive: a check is marked as fraudulent when it is fraudulent.
True Negative: a check is marked as non-fraudulent when it is non-fraudulent.
False Positive: a check is marked as fraudulent when it is non-fraudulent.
False Negative: a check is marked as non-fraudulent when it is fraudulent.
Service Agent: an employee who's responsible for manually checking fraud in checks.
Data Correction: in machine learning context, data correction is a set of techniques, such as correcting dataset shift, that alters the training dataset in order to allow better accuracy.
Tuner: a set maximum amount of dollars for which a check will be sent to the manual review bucket regardless of the decision of the algorithm. (example $10,000)

3. Assumptions

The eight assumptions for this project are:

1) The dataset for the software is created by the team and no real data is used.
2) JP Morgan & Chase have the computing ability to run a scanning algorithm.
3) All checks fed into the algorithm are in the form of an isolated digital image.
4) The checks fed into the software do not have an imagery background.
5) The software will only focus on checking fraud activities mentioned in the glossary.
6) The handwriting in checks legible (non-cursive).
7) The manual process of reviewing a single check takes 10 seconds or more.
8) A check that meets or surpasses the tuner will be automatically sent to manual review regardless of how the false positive rate is affected.

4. Requirements

The five requirements for this project in order of priority are:

1) As a servicing agent, I want to be able to successfully flag fraudulent checks (in the cases of abnormal spacing, crossed out writing, and dollar amount mismatch with the database) so that I can reduce the amount of manually processed checks.

    - Acceptance criteria:
        a) Keep false positive below 15%.
        b) Keep false negative below 1%.
        c) True positive and false positive will be sent to the manual review bucket.
        d) True negative and false negative will be sent to the non-fraudulent bucket.

2) As a servicing agent, I want to know that customers' checks are checked for multiple types of fraud so that company and customer money is not lost.

    - Acceptance criteria:
        a) ML algorithm accuracy improves as the training dataset size increases.
        b) Diverse dataset that contains at least three cases of fraud.

3) As a service agent, I want to be able to scan a check from fraudulent activities in less time than the manual process so that I can justify using this scanning software.

    - Acceptance criteria:
        a) Keep a single check scan time under 3 seconds.

4) As a service agent I want to have a record of all the images used to train the algorithm so I can execute data correction if needed.

    - Acceptance criteria:
        a) All training images are saved in one cloud uploaded folder.
        b) All checks are named by their serial numbers.

5) As a service agent, I want the scanning software to only focus on the name box and dollar amount box so that I can use only the necessary information (refer to appendix).

5. Needs and factors

5.1 Public health needs
Public health needs are not applicable to this project. This does not apply to our project because we are collecting data and creating a system to detect fraud online, so no physical health will be affected.

5.2 Public safety needs
Public safety needs are addressed by assumption (1). This assumption shows that we are not using anyone's private data and creating the check datasets on our own to protect the financial privacy of JP Morgan customers.

5.3 Public welfare needs
Public welfare needs are addressed by requirement (1) and (3). These requirements imply easing and simplifying the work life of the agent and improving customer experience.

5.4 Global factors
Global factors are not applicable to this project. This does not apply to our project as we are creating our own datasets and not designing this project to be used outside our product owner and location.

5.5 Cultural factors
Cultural factors are not applicable to this project. This does not apply to our project as will be designed to flag if checks are fraudulent or not, thus not needing to optimize the system to work across multiple cultures as we are receiving inputs in data and outputting a flag on a scanned check.

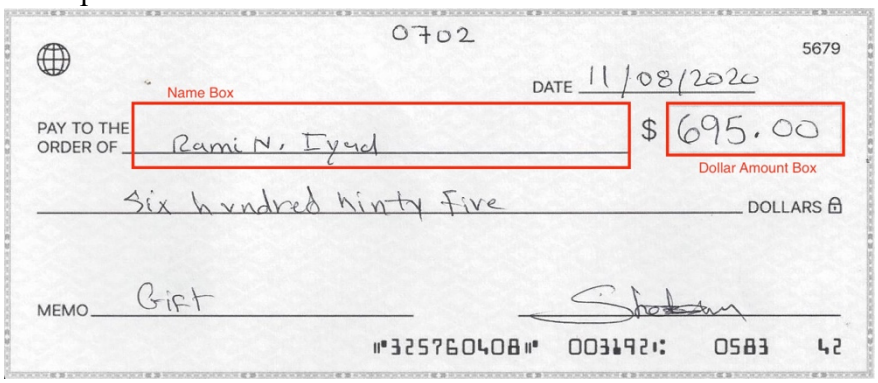
5.6 Social factors
Social factors are not applicable to this project. This does not apply to our project because the process we are creating is automated and does not connect people socially in any capacity.

5.7 Environmental factors
Environmental factors are not applicable to this project. This does not apply to our project since no part of our problem or software strains or aides the environment.

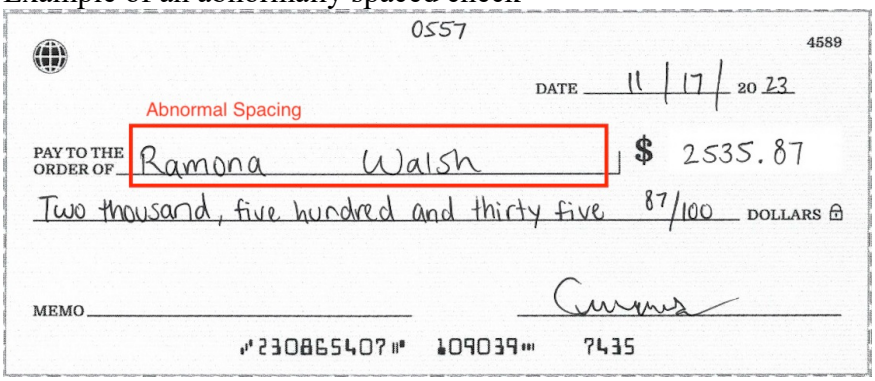
5.8 Economic factors
Economic factors are addressed by requirement (3). The requirement notes the need to protect company and customer money.

Appendix

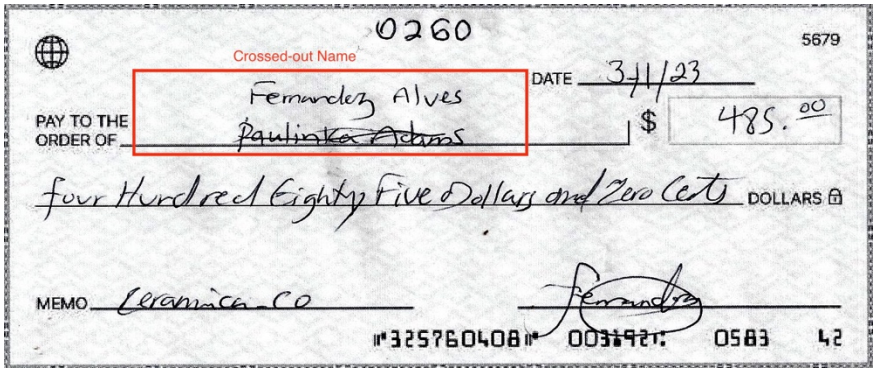
Example of a no fraud check



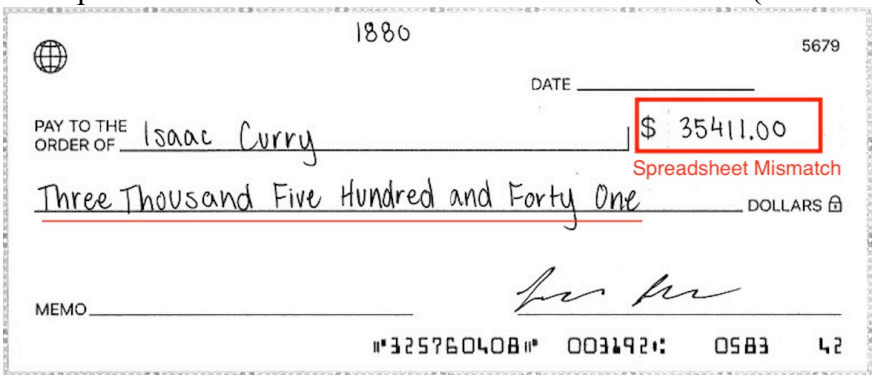
Example of an abnormally spaced check



Example of a crossed-out name check



Example of dollar amount mismatch with database record (added a one to the end of the number)



Specification: JP Morgan Chase Group 1

Omar Alkhawaldeh, Mohamed Abdelmalek,

Alex Garcia, Cassandra Hetrick

Undergraduate Students

Department of Computer Science and Engineering

University of South Florida

Tampa, FL 33620

Email: alkhawaldeh@usf.edu, mabdelmalek@usf.edu,
alexgarcia@usf.edu, chetrick@usf.edu

Version number: 1.01

November 26, 2023

History of document

- **Version 1.00** (October 12, 2023) - Initial document created.
- **Version 1.01** (November 26, 2023) – redefined cases of fraud. Increased glossary. Removed functionality constraint. Redid figure 5.1 (removed not sure bucket). Redid trade-offs. Redid traceability to requirements using the new requirements doc.

Table of Contents

1. Introduction

The issue we face arises when customers notice unauthorized withdrawals of money from their account, indicative of fraudulent checks.

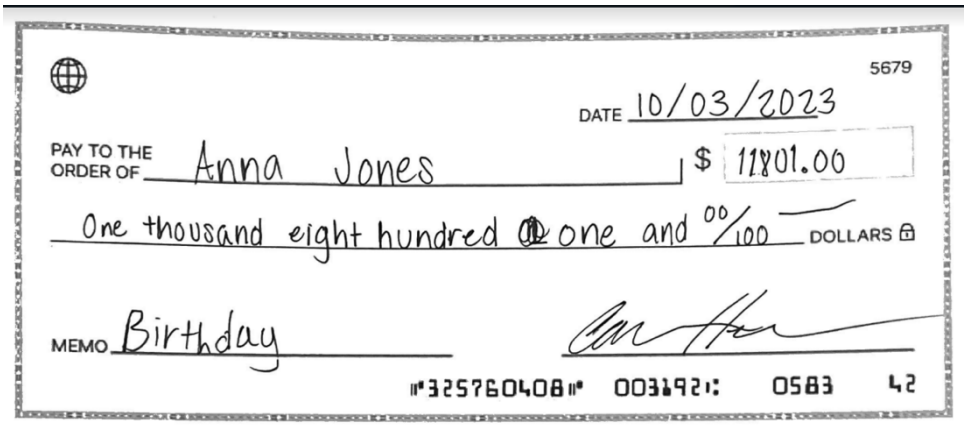
- **Current process:**
  - Fraud detection at JP Morgan Chase is done manually.
  - As time passes, scammers become better and better at their job.
  - Features introduced to customers could make security harder.

- **Concerns with the old fashion way:**
  - Time consuming.
  - Very costly.
  - Hard to scale.
  - Exposed to human error.

Fraudulent checks can come in a plethora of forms, some signs of fraud include:

- Abnormal spacing.
- Crossed-out writing.
- Dollar amount mismatch with database record.
- Dollar amount not matching what is written on the amount line.
- Bleached checks.
- Serial Number Mismatch (on top and bottom of check).

One example of a fraudulent check is displayed below where the amount is meant to be $1801.00, but in the number box a 1 has been added to the front which makes the amount look like $11,801.00. This can be detected with the two amounts displayed not being equal.



The problem addressed by these requirements is: “Can we build an ML-based scanning system that can determine if a check is fraudulent within 3 seconds with 15% false positive rate and 1% false negative rate.”

2. Glossary

Fraudulent Check: an illegitimate check that has been tampered with, altered or fabricated.

True Positive: a check is marked as fraudulent when it is fraudulent.

True Negative: a check is marked as non-fraudulent when it is non-fraudulent.

False Positive: a check is marked as fraudulent when it is non-fraudulent.

False Negative: a check is marked as non-fraudulent when it is fraudulent.

Service Agent: an employee who's responsible for manually checking fraud in checks.

Data Correction: in machine learning context, data correction is a set of techniques, such as correcting dataset shift, that alters the training dataset in order to allow better accuracy.

Tuner: a set maximum amount of dollars for which a check will be sent to the manual review bucket regardless of the decision of the algorithm. (example $10,000)

3. Constraints

This project has three main constrains:

- *Time***:** the biggest challenge of this project is the deadline. The project needs to be ready for presentation by December 1st (the end of the semester), which only gives us four months to perfect the project.

- *Cost*: the project does not include a budget from JP Morgan & Chase. Therefore, the resources of the project will be limited to open-source code and free online tools.

- *Financial Records*: without access to a database of checks provided by Chase with a greater number of samples and a wide variety of test cases, the project will be limited to a self-created data set of checks found online and filled in by group members.

4. Applicable standards

1. ISO 25010 Testing and Quality Assurance standards
    - This standard requires the project to be tested against:
        - Functional suitability: code meets all specified task and user objectives.
        - Usability: code meets specified goals with effectiveness, efficiency, and satisfaction.

        - Performance efficiency: code meets processing times when performing its functions.

2. ISO/IEC 25012:2008 Software product Quality Requirements and Evaluation.

    - It provides guidelines for our assessing and managing data quality process, which is crucial for training and testing our ML model.

3. IEEE Software Engineering Standards.

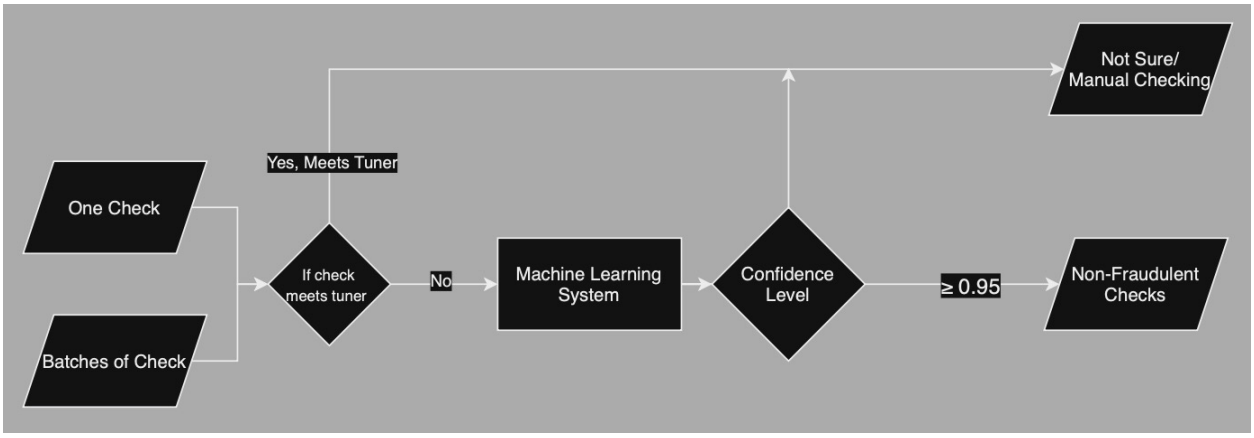
    - IEEE 730-2014, Standard for Software Quality Assurance Processes.

        - Project Planning (Clause 6.1): in our ML project, we will adapt the project planning process to include the unique aspects of ML, such as data collection, data preprocessing, model training, and evaluation to ensure that we can achieve the best quality outputs with high confidence and accuracy by choosing the appropriate ML algorithm to design and train the most effective Model.

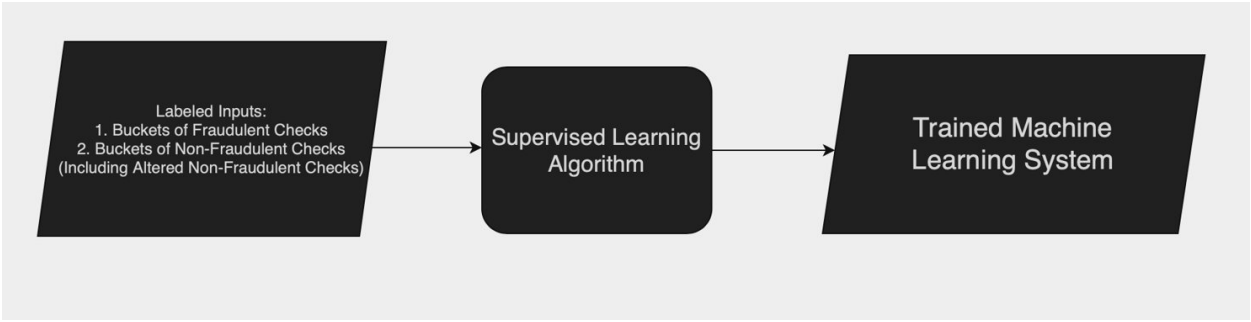
        - Verification and Validation (Clause 6.4): in our ML design, we will assess our trained model's accuracy, precision, F1-score as crucial parts of the verification and validation process to ensure that the data we used for validation is representative and cross-validation techniques are being applied as needed.

5. Design

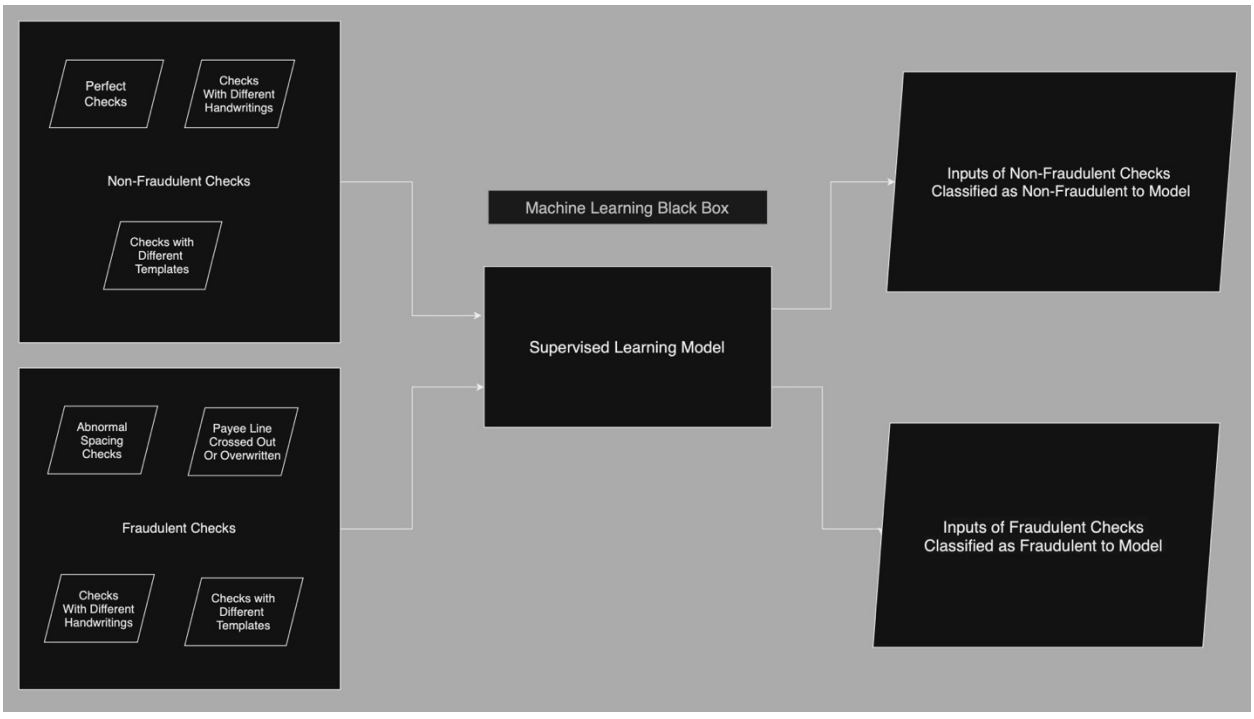
5.1: Design Flow Chart



5.2: Training Black Box Flow Chart



5.3: Inside Training Black Box Flow Chart



6. Risk analysis and Mitigation

Risk	Probability	Impact (1-10)	Mitigation
Personal emergencies: including illness and unexpected events.	Medium (50%)	8	Created a set project timeline and planning to always be ahead of schedule.
Dataset loss or corruption	Low (<50%)	8	Dataset is distributed among team members' devices and backed up on a cloud.
Technical difficulties: including loss of internet, failure in the coding tools, failure in the teams' devices.	Low (<50%)	7	Code is backed up online and the team's phone numbers are shared.
JPMC point of contact leaving the company	Low (<50%)	4	In contact with multiple employees within their team rather than just one.

7. Design trade-offs

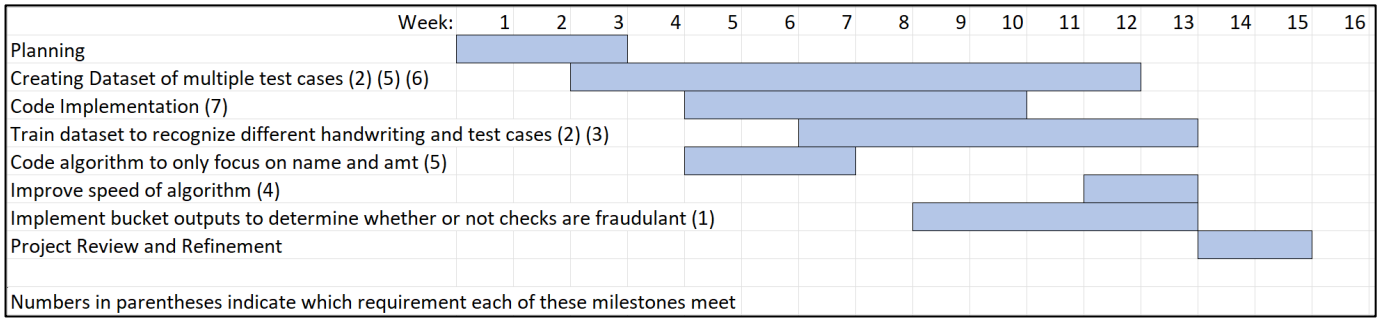
1. Extensibility vs. Functionality:

   We traded covering all potential known types of fraud (extensibility) for the functionality of the three types of fraud which we addressed. Instead of addressing all fraud types in our design, we decided to incrementally address one type of fraud at a time until we achieve satisfactory accuracy of results.

2. Accuracy vs. Legal Consideration:

   We traded accuracy for costumer data safety. In order to reduce the risk of costumer data leakage, we decided to build our own dataset of checks that is much smaller than the company's database. Using the company's dataset would increase the accuracy of the algorithm.

8. Project plan



9. Traceability to requirements

Requirement Number	Requirement Description	Design Equivalent
1	Determine if a check is fraudulent or not and split it into two different buckets	Model 5.1: based on confidence level, output is split between fraud and non-fraud buckets. Model 5.3: Supervised Learning Model -> Fraudulent or Not Fraudulent
2	Are multiple fraudulent check test cases being tested	Model 5.3: Non-Fraudulent and Fraudulent potential cases defined
3	Scan a check for fraudulent activities in less time than the manual process	Model 5.1: Shows whole process which should execute in a matter of seconds
4	The scanning software only focuses on the name box and dollar amount box of the check	Model 5.2: Train supervised learning module to focus only on name and number box
5	Have a record of all the images used to train the algorithm	Model 5.3: The two boxes on the left are saved in folders on the cloud

10. Traceability to needs and factors

Needs and factors	Design Equivalent
Public safety needs	Part of the design is using a self-generated dataset by our team, meaning no costumer data will be used nor exposed.
Public welfare needs	The goal of the design is to reduce the workload of JPMC agents so that less manual work is needed. This contributes to employees and costumers' satisfaction.
Economic factors	Our design in its essence is a way of saving the company's money from scams. In extension, the design contributes to its economic well-being.

References

- https://iso25000.com/index.php/en/iso-25000-standards/iso-25010
- https://ieeexplore.ieee.org/document/6835311
- https://www.w3.org/TR/webmachinelearning-ethics/

Test Plan: Fraudulent Check Detection

Omar Alkhawaldeh, Alex Garcia,
Cassandra Hetrick, and Mohamed Abdelmalek

Computer Science and Engineering Students
Department of Computer Science and Engineering
University of South Florida
Tampa, FL 33620

Email: alkhawaldeh@usf.edu, alexgarcia@usf.edu, chetrick@usf.edu, and mabdelmalek@usf.edu

Version 1.01
November 26, 2023

History of document

- **Version 1.00** (October 16, 2023) - Initial document created.
- **Version 1.01** (November 26, 2023) - updated fraud types. Completely changed parameters. Added details to design of testing. Added pictures to the expected output of the two test cases.

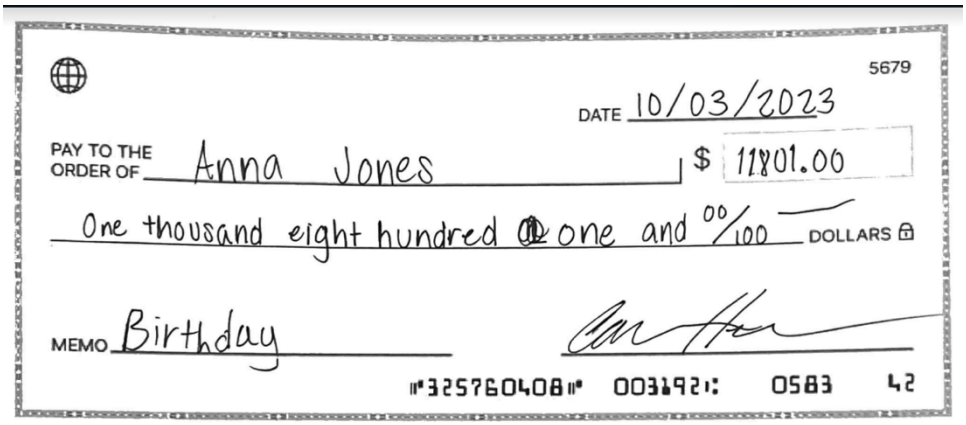
Table of Contents

1. Introduction

Fraudulent checks can come in a plethora of forms, some signs of fraud include:

- Abnormal spacing.
- Crossed-out writing.
- Dollar amount mismatch with database record.

One example of a fraudulent check is displayed below where the amount is meant to be $1801.00, but in the number box a 1 has been added to the front which makes the amount look like $11,801.00. This can be detected with the two amounts displayed not being equal.



The problem addressed by these requirements is: “Can we build an ML-based scanning system that can determine if a check is fraudulent within 3 seconds with 15% false positive rate and 1% false negative rate.”

2. Glossary

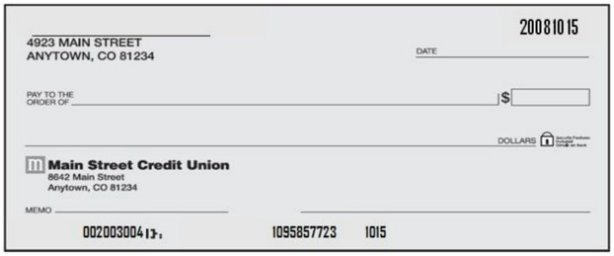
- *OCR*: Optical Character Recognition, a model which takes handwriting and translates it to text.

- *ROI*: Region of Interest, which section of the check should be focused on and read by the algorithm.

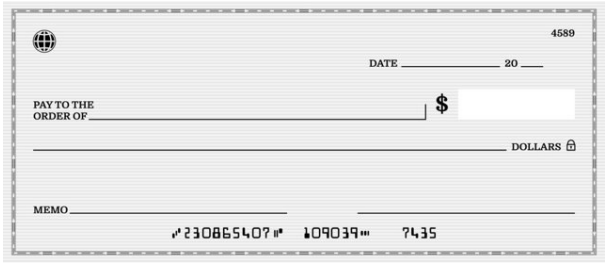
3. Parameters to be tested

- **Check Format**: 4 different check formats were included in our training/testing dataset.

  Format 1

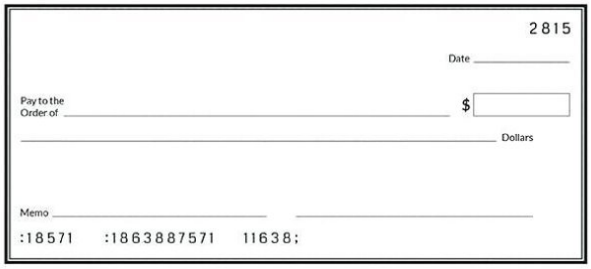


  Format 2



  Format 3

  Format 4



- **Different Handwriting**: our dataset unitizes 30 different handwriting to ensure diversity in data collection.

- **Different Dollar Amount**: our dataset includes check amounts starting at $5 and up to $25,000 to ensure inclusion of a variety of possible amounts chosen by costumers.

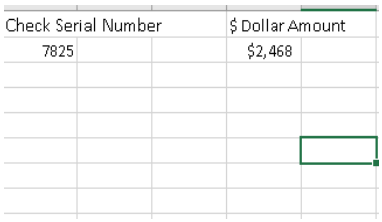
4. Design of testing

Test case 1 design:

- Step 1: scan ROI on check: Dollar amount handwritten on check.

- Step 2: utilize OCR algorithm to read to the recorded ROI as an integer.

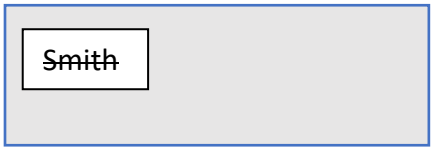
- Step 3: compare amount detected from check by amount on spreadsheet.



- Step 4: if they match, then it's determined to be non-fraudulent, fraudulent otherwise.
- Step 5: repeat steps 1-4 for all checks in the dataset.

Test case 2 design:

- Step 1: scan ROI on check: "PAY TO THE ORDER OF" name box on check.



- Step 2: run the scanned ROI through the ML model and it will determine if the name box is altered (fraud) or not (non-fraud).

- Step 3: repeat steps 1-2 for all test check of the dataset.

*Note on device used to run the algorithm: our team used an M1 MacBook Air to run both OCR and ML algorithms. We believe any computer that is made in 2015 and up will be able to the algorithms with similar timing if not better.

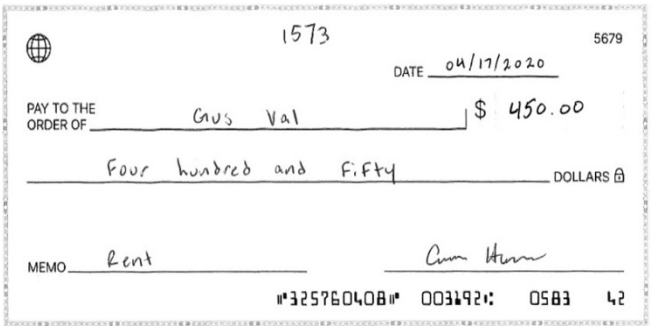
5. Test case #1

Name: Dollar Amount Mismatch

System Configuration: the tester must run the preparatory section of code to install all necessary packages. They must also upload the image to sample folder and change this line of code to the desired image name.

```
# reading the image
img = cv2.imread('image.png')
```

Exact Input: The input will be in a form of an image and a spreadsheet that keeps record of the written checks as follows:



01570	18.00		FALSE
01571	380.00		FALSE
01572	57.00		FALSE
01573	450.00		FALSE
01574	5000.00		FALSE
01575	333.00		FALSE
01576	28.00		FALSE
01577	360.00		FALSE
01578	405.00		FALSE

Exact Expected Output: the output in this scenario must determine that this check is non-fraudulent. The output will be in the form of text. The reason is that the dollar amount associated with the check's serial number matches our spreadsheet record. The unexpected output would be if the program determines this check to be fraudulent. Here is an example output:

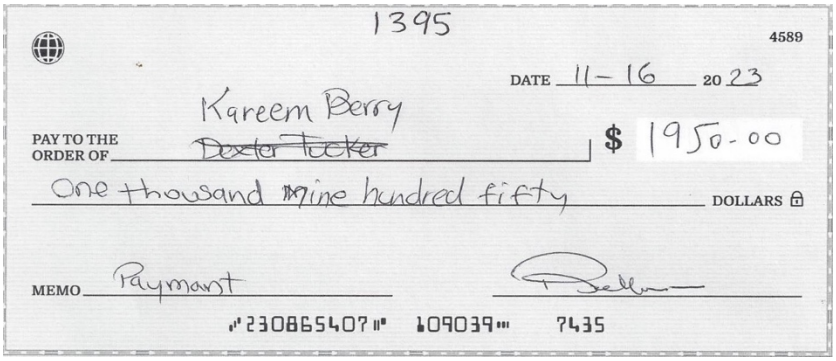
```
The amounts match!!! Here is our record for this transaction:
[1573, '450', '', 'FALSE', '', '', '', '', '', '', '', '']
```

6. Test case #2

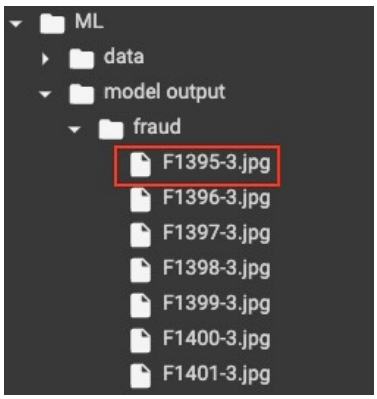
Name: Crossed-Out Name

System Configuration: the tester must run the preparatory section of code to install all necessary packages. They must also upload the image to the unscanned checks folder.

Exact Input: The input will be in a form of an image as follows:



Exact Expected Output: It's expected to flag the check as a 'fraud' meaning this check will be later sent to manual verification. The output will be placed in the fraud folder of the model output. The unexpected out of this test case is to find this check in non-fraud folder. Here is how the folder will look like after running (F in the file name stands for fraud):



Our test cases are associated with requirements which are established already. The following matrix demonstrates the connection:

Test Case ID	Test Case Name	Requirement Met
1	Dollar Amount Mismatch	1. As a servicing agent, I want to be able to successfully flag fraudulent checks so that I can reduce the amount of manually processed checks. 5. As a service agent, I want the scanning software to only focus on the name box and dollar amount box so that I can use only the necessary information .
2	Crossed-Out Name	5. As a service agent, I want the scanning software to only focus on the name box and dollar amount box so that I can use only the necessary information.

Design Review Presentation

JP Morgan Chase Group 1

**Omar Alkhawaldeh, Mohamed Abdelmalek,
Alex Garcia, Cassandra Hetrick**

alkhawaldeh@usf.edu, mabdelmalek@usf.edu,
alexgarcia@usf.edu, chetrick@usf.edu

Department of Computer Science and Engineering
University of South Florida
Tampa, FL 33620

Acknowledgments

- **Ken Christenson**
- **Hari Sangaraju**
- **Robert Andion**
- **Arun Krish**

Agenda

- **Background**
- **Problem**
- **Requirements**
- **Design**
- **Constraints**
- **Applicable standards**
- **Risk analysis and mitigation**
- **Project plan**

Background

- The issue arises when customers notice unauthorized withdrawals of money from their accounts. This means fraudulent checks.

- **Current process:**
  - Fraud detection at JP Morgan Chase is done manually.
  - As time passes, scammers become better and better their job.
  - Features introduced to costumers could make security harder.

- **Concerns with the old fashion way:**
  - Time consuming
  - Very costly
  - Hard to scale
  - Exposed to human error

Problem

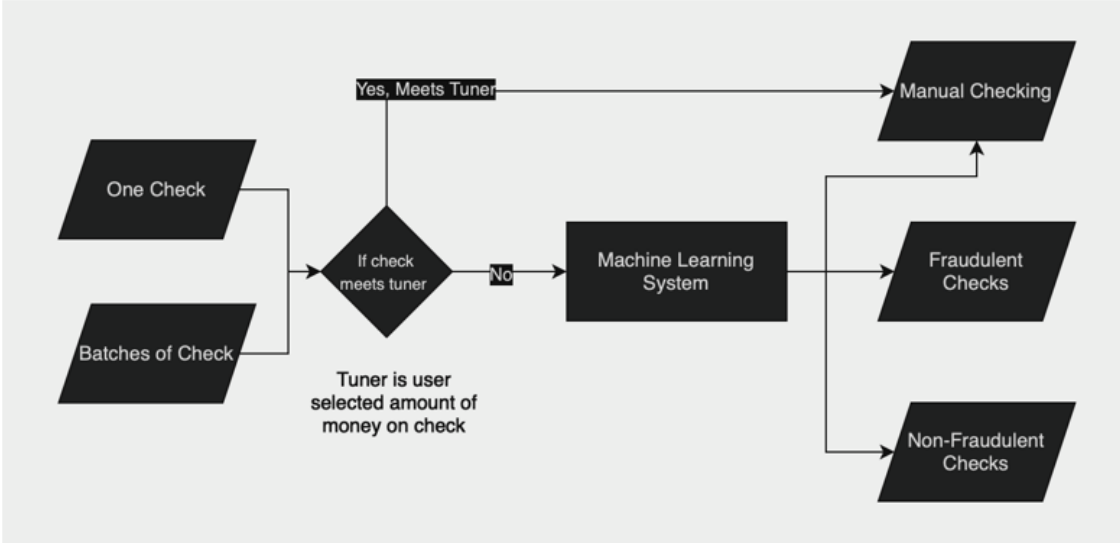
Can we build a ML-based scanning system that can determine if a check is fraudulent within 3 seconds. The scan will cost no more than 1 cent per check with a 5% false positive rate and 1% false negative rate.

Requirements

1. As a servicing agent, I want to be able to determine if a check is fraudulent or not so that I can have confidence that the system will flag fraudulent checks.
2. As a servicing agent, I want to know that customers' checks are checked for multiple fraud scenarios so that company and customer money is not lost.
3. As a service agent, I want the scanning system to be able to recognize various check appearances, themes, and different handwritings so that the system does not frequently misinterpret a check as fraudulent.
4. As a service agent, I want to be able to scan a check from fraudulent activities in less time than the manual process so that I can justify using this software.
5. As a service agent I want the scanning software to only focus on the name box and number box of the check so that I can use only necessary information.
6. As a service agent I want to have a record of all the images used to train the algorithm so I can determine the root cause of any ML error encountered.
7. As a service agent, I want the algorithm to be well documented with comments so that I can easily do maintenance to the code if future errors are encountered.

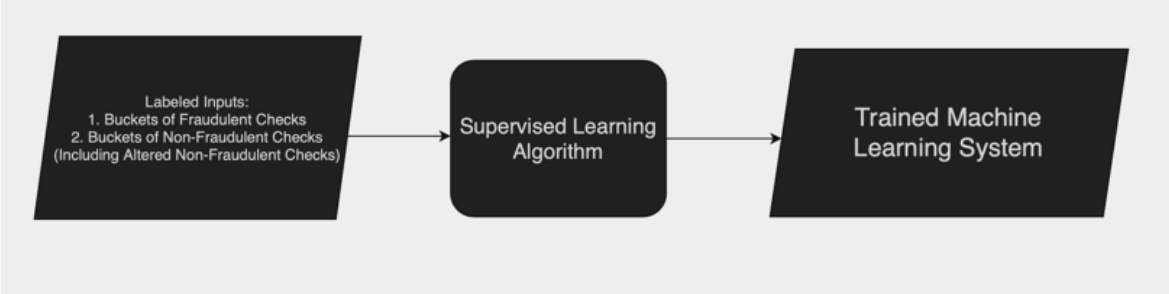
Design

- **Design Flow Chart**



Design

- **Training Flow Chart**



Design Constraints

- While determining if a check is fraudulent or not, we are assuming minor flaws will pass through our validation system in two forms:
  - False positive: flag a check as fraudulent while it is not.
  - False negative: flag a check as not fraudulent while it is.
- Therefore, we decided to set some agreed upon guide constraints:
  - The limit for false positive checks is below 5%.
  - The limit for false negative checks is below 1%.
- Additional constraints include:
  - Validating a check should be in less than 3 seconds.
  - The cost per transaction will be no more than 1 cent per check.

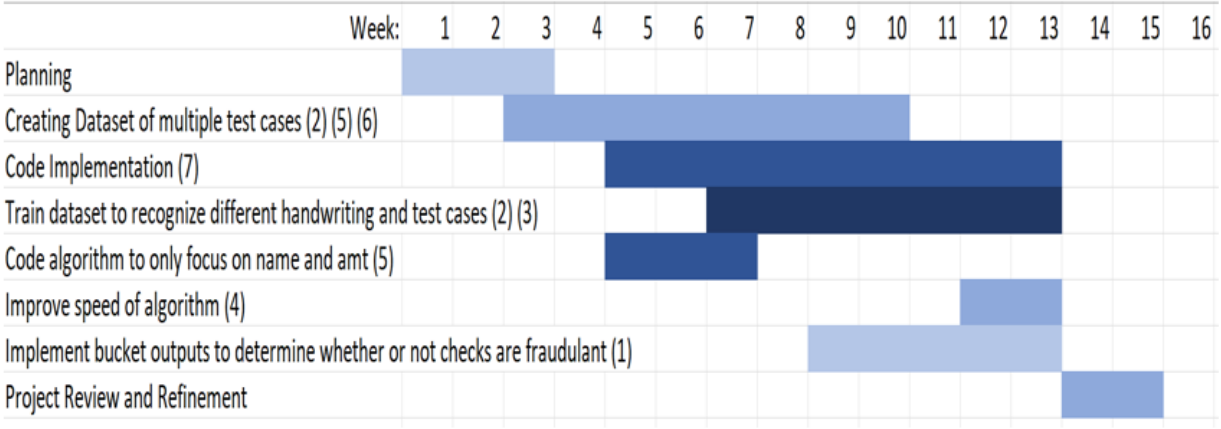
Applicable standards

- **Public safety**
  - Keep bank clients' data protected and not subject for any testing.
- **Code development**
  - Using Pytorch: a library in ML used to implement applications that simulate computer vision algorithms.
  - ISO/IEC 27001 information security management.
  - ISO/IEC 27002 Code of practice for information security controls.
  - ISO 25010 Testing and Quality Assurance standards.
  - IEEE Software Engineering Standards.
  - Machine Learning Ethics and Fairness Guidelines.

Risk analysis and mitigation

- Accuracy
  - Creating a third bucket of outputs that goes to manual checking.
  - Regular manual double checking.
- Dataset and false positives
  - Creating a diverse set of test cases and including more handwritings.
  - Regularly monitor data for potential issues and update data set.
- Time constraint
  - Worrying about the feature less, and the bulk of the algorithm more.
- False negative possibility
  - Creating a tuner to separate much larger checks.

Project plan



Final Presentation Slides

JP Morgan Fraudulent Check Detection

**Omar Alkhawaldeh, Alex Garcia,
Cassandra Hetrick, and
Mohamed Abdelmalek**

Department of Computer Science and Engineering
University of South Florida
Tampa, FL 33620

alkhawaldeh@usf.edu, alexgarcia@usf.edu,
chetrick@usf.edu, mabdelmalek@usf.edu

1

Acknowledgments

- Professor Ken Christensen
- Robert Andion
- Hari Sangaraju
- Arun Krish

2

Agenda

- Background
- Problem
- Requirements
- Demo
- Design
- Constraints
- Applicable standards
- Trade-offs made

3

Background

- The issue we face arises when customers notice unauthorized withdrawals of money from their account, indicative of fraudulent checks.
- **Current Process**
  - Done manually
  - Scammers improve over time
  - Difficult to put customer safety in the hands of customers
- **Forms of Fraud**
  - Abnormal spacing.
  - Crossed-out writing.
  - Dollar amount mismatch with database record.
  - Dollar amount not matching what is written on the amount line.
  - Bleached checks.
  - Serial Number Mismatch (on top and bottom of check).

4

Problem

- The problem addressed is: “Can we build an ML-based scanning system that can determine if a check is fraudulent within 3 seconds with 15% false positive rate and 1% false negative rate?”

5

Requirements

1. As a servicing agent, I want to be able to successfully flag fraudulent checks (in the cases of abnormal spacing, crossed out writing, and dollar amount mismatch with the database) so that I can reduce the amount of manually processed checks

2. As a servicing agent, I want to know that customers’ checks are checked for multiple types of fraud so that company and customer money is not lost.

3. As a service agent, I want to be able to scan a check from fraudulent activities in less time than the manual process so that I can justify using this scanning software.

6

Requirements

4. As a service agent I want to have a record of all the images used to train the algorithm so I can execute data correction if needed.

5. As a service agent, I want the scanning software to only focus on the name box and dollar amount box so that I can use only the necessary information.

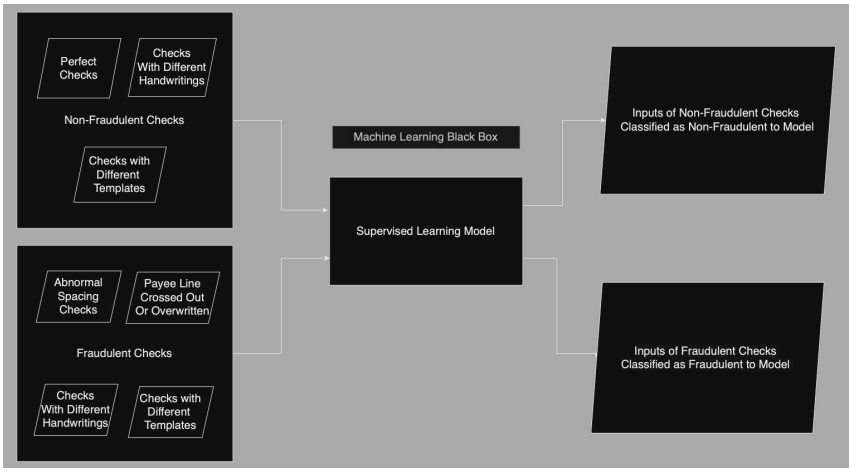
7

Demo

- **Demo agenda:**
  - Show our dataset.
  - Run the OCR algorithm which tackles the amount mismatch against database form of fraud.
  - Run the ML algorithm which tackles crossed-out names and abnormal spacing form of fraud.

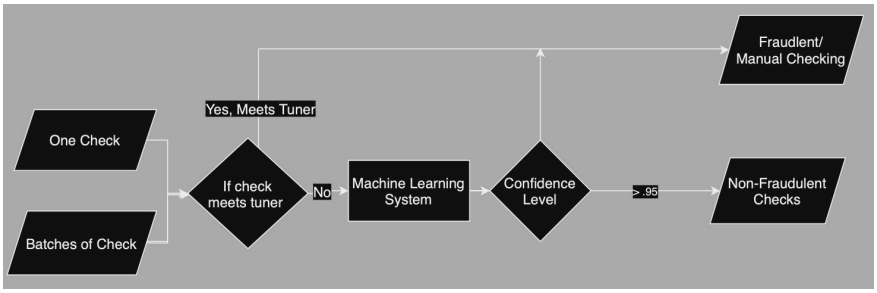
8

Flowchart Design



9

Testing Flowchart Design



10

Constraints

- **Three main constrains:**
  - **Time**
    - The project needs to be ready for presentation by December 1st .
    - Four months to develop a deliverable perfect project.
  - **Cost**
    - The project does not include a budget from JP Morgan & Chase.
    - Resources are limited to open-source code and free online tools.
  - **Financial records**
    - Self-created dataset of checks found online and filled in by group members.
    - No access to a database of checks provided by Chase

Applicable standards

- ISO 25010 Testing and Quality Assurance standards
  - Functional suitability
  - Usability
  - Performance
- IEEE 730-2014, Standard for Software Quality Assurance Processes.
  - Project Planning (Clause 6.1)
  - Verification and Validation (Clause 6.4)

Trade offs

- Extensibility vs. Functionality
  - Traded covering all cases of fraud for the functionality of 3 cases of fraud.
- Accuracy vs. Legal Consideration
  - Traded the accuracy of the algorithm for the safety of costumer data.

Press Release

Your Money is Safer with Chase!

(Tampa, Florida, December 1, 2023) – JP Morgan Chase & Co.

JP Morgan and Chase will be releasing an exciting new internal tool to its fraud detection team located in Tampa, Florida today, December 1st. This tool is a Machine Learning algorithm to detect potential fraudulent checks and flag them to be manually checked by company agents.

A team of four students from the University of South Florida has been working in correspondence with Chase for the past four months to develop this algorithm as part of a class project for the students.

This new product will be released initially to the Tampa branch of Chase. Chase will use and test the algorithm further to streamline the process of detecting fraud by ensuring that fewer "perfect checks" need to be manually checked. This will also reduce the possibility of human error.

This was implemented by the USF team using the Python programming language to create a Supervised Machine Learning Algorithm with aspects of Optical Character Recognition (OCR).

Chase is very excited not only to announce this tool, but for the prospects of it as well. This ability to process handwritten data could open many doors in the world of machine learning for the company.

JP Morgan Fraudulent Check Detection

Mohamed Abdelmalek, Omar Alkhawaldeh, Alex Garcia, Cassandra Hetrick

UNIVERSITY of SOUTH FLORIDA

Background

- The issue we face arises when customers notice unauthorized withdrawals of money from their account, indicative of fraudulent check activity
- The current process is done manually
- Forms of fraud
  - Abnormal Spacing
  - Crossed-out writing
  - Dollar amount mismatch with database record
  - Dollar amount not matching what is written on the amount line
  - Bleached checks
  - Serial number mismatch (on top and bottom of check)

* Forms written in red are the forms our project addresses

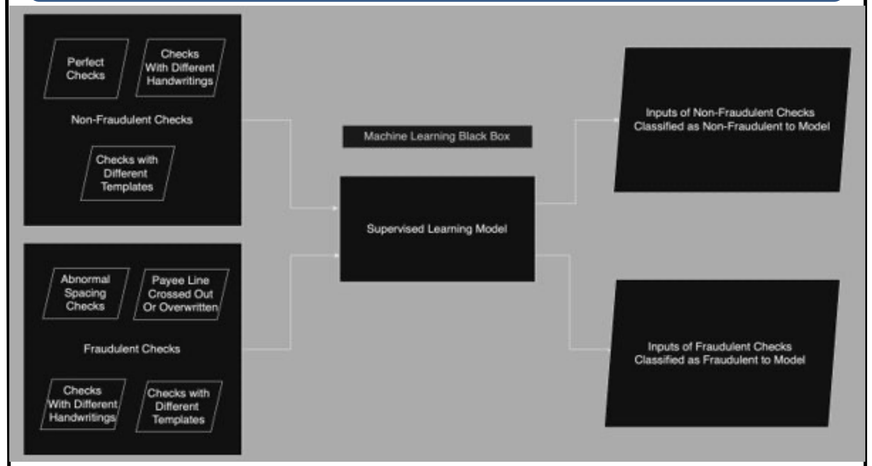
Problem

The problem addressed by our project is: “Can we build an ML-based scanning system that can determine if a check is fraudulent within 3 seconds with 15% false positive rate and 1% false negative rate?”

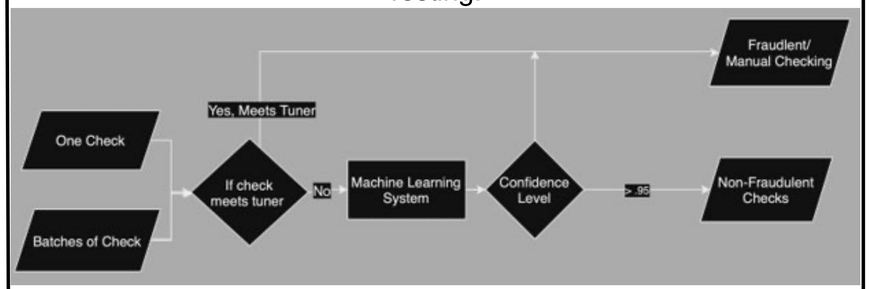
Requirements

1. As a servicing agent, I want to be able to successfully flag fraudulent checks (in the cases of abnormal spacing, crossed out writing, and dollar amount mismatch with the database) so that I can reduce the amount of manually processed checks.
2. As a servicing agent, I want to know that customers’ checks are checked for multiple types of fraud so that company and customer money is not lost.
3. As a service agent, I want to be able to scan a check from fraudulent activities in less time than the manual process so that I can justify using this scanning software.
4. As a service agent I want to have a record of all the images used to train the algorithm so I can execute data correction if needed.
5. As a service agent, I want the scanning software to only focus on the name box and dollar amount box so that I can use only the necessary information.

Design (Flowcharts)



Testing:



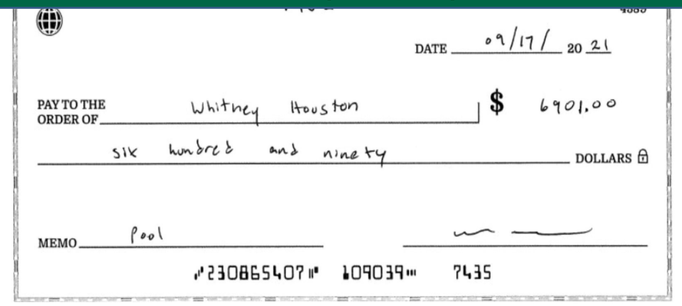
Constraints

- Time: 4 months to develop a deliverable project (due Dec1)
- Cost: No budget from Chase, so resources used must be free
- Financial Records: no access to Chase database, so checks had to be manually created

Applicable Standards

- ISO 25010 Testing and Quality Assurance Standards
  - Functional sustainability, usability, and performance
- IEEE 730-2014 Standard for Software Quality Assurance Processes
  - Clause 6.1: Project Planning
  - Clause 6.4: Verification and Validation

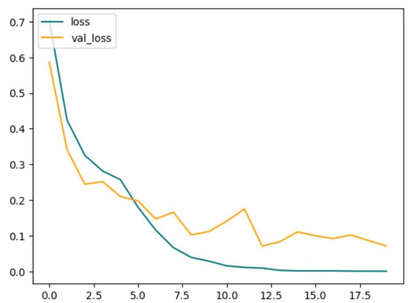
OCR Implementation (Output)



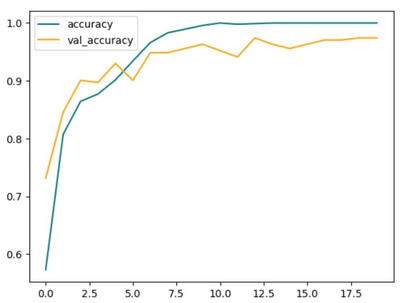

```
Amount number: scanned integer-> 6901

6901.00

The amounts don't match, please send to manual checking. Here is our record for this transaction:
[1902, '690', '', 'TRUE', 'Spreadsheet Mismatch', '', '', '', '', '', '', '']
```

ML Implementation (Output)

Loss



Accuracy




```
count 100 | TP 49 | TN 49 | FP 1 | FN 1
Percent TP: % 49.0
Percent TN: % 49.0
Percent FP: % 1.0
Percent FN: % 1.0
```

Acknowledgements

- Professor Ken Christenson
- Robert Andion (Chase Contact)
- Hari Sangaraju (Chase Contact)
- Arun Krish (Chase Contact)

Status Report for JP Morgan Chase #1 for Week #3

What were the goals for this two-week period?

1. Meet with product owner (Hari Sangaraju).
2. Schedule a regular basis for meeting.
3. Create Problem statement.
4. Learn about the check fraud detection process currently implemented at JP Morgan.
5. Learn about the desired check fraud detection process wanted at JP Morgan.
6. Learn about the tools and what languages are currently used by the bank's software developing team.

What goals were accomplished in this two-week period?

1- We were able to conduct a 30 min meeting with both Hari Sangaraju and Arun Krish.
2- We are scheduled to meet every Monday afternoon with Hari to discuss our project.
3- We were able to define a problem statement which will be later reviewed by the professor.
4- Learned a variety of test cases and scenarios of fraudulent, altered, counterfeit, and forged checks.
5- Learned the confusion matrix and the worst possible output in our scenario [false negative].
6- Obtained better understanding about the user story requirements through class.
7- What margin of error is acceptable at JP Morgan for detecting check fraud. The larger the amount of the check being deposited is, the less room there should be for error

Reflect critically on any goals not accomplished.

- Unaccomplished goal: Learn about the tools and what languages are currently used by the bank's software developing team.
- Reason: The duration time allocated for the meeting was not enough to go over technologies. However, Hari mentioned that he will provide us with more details in the next meeting.

What are the goals for next two weeks?

1. Complete the requirements document, by forming a revised problem statement and detailed user stories.
2. Begin creating a data set and familiarize ourselves with how to identify potential fraud.
3. Further discuss possible technologies to use with Hari and Arun.
4. Start researching possible algorithms used to solve similar problems in the past.

How many hours were spent on each goal noted above in the past two weeks?

Mohamed: 1.5 hours: Completed goals: Worked on rough draft of status report and did case study research.
Omar: 3 hours. Completed goals: set up meetings (30 min), attend company and team meetings (1 hour), Research (1 hour), and status report (30 min). Non-Completed goals: choosing technologies.
Alex: 2.5 hours Completed goals: researching confusion matrix, banking jargon, and check fraud, and how check transactions work (1 hour), attended meeting with Hari (30 min). learning basics of machine learning and the different associated algorithms for supervised machine learning (1 hour).
Cassie: 2.5 hours. Completed goals: Created drafts of Status Report (1 hour), met with Hari (30 min), researched similar projects on GitHub to find potential tech stacks, compatible with our project.

Status Report for JPM Chase 1 Week #5

What were the goals for this two-week period?

1. Complete the requirements document, by forming a revised problem statement and detailed user stories.
2. Begin creating a data set and familiarize ourselves with how to identify potential fraud.
3. Further discuss possible technologies to use with Hari and Arun.
4. Start researching possible algorithms used to solve similar problems in the past.

What goals were accomplished in this two-week period?

1. Complete the requirements document, by forming a revised problem statement and detailed user stories.
2. Created a dataset of 50 checks so far, checked out more suggested test cases, and started filling out the checks.
3. We agreed with Hari and his team to work with PyTorch as a start.
4. Start researching some good YouTube tutorials on PyTorch to be used to design our algorithm.
5. Decided on the coding platform to be used (Google Colab) for ease of sharing code.

Reflect critically on any goals not accomplished.

We were able to accomplish all the goals we set out to accomplish in the past two weeks. Way to go!

What are the goals for the next two weeks?

1. Come up with a working base for the code demo.
2. Upload the completed dataset of different types of fraud to Pytorch.
3. Design the Algorithm to test out at least one case.
4. Communicate with Hari and his team on our revised requirements to receive feedback.
5. Using the notes given by the professor on the requirement document, we plan to give another look and revise it.

How many hours were spent on each goal noted above in the past two weeks?

Cassandra: Completed goals: Created fake checks for the dataset (2 hours), attended meeting with Hari and team (30 min), worked on the requirements document (2.5 hours)

Alex: Completed goals: Researched about how PyTorch works and the basics of deep learning (2 hours), attended meeting with Hari (30 min), Worked on the requirements document (2.5 hours), installed Pytorch onto my personal computer and experimenting (30 min).

Mohamed: Researched about the use of Google Collab and watched PyTorch tutorials on machine learning (7 hours) Also worked on creating part of the data (invalid checks) (30 min)

Omar: Completed goals: worked on the requirement document (2.5 hours), attended meetings with JP morgan teams (30min). Research Google Collab and PyTorch (2 hours).

Status Report for JPM Chase 1 Week #9

What were the goals for this two-week period?

1. Come up with a working base for the code demo.
2. Upload the completed dataset of different types of fraud to Pytorch.
3. Design the Algorithm to test out at least one case.
4. Communicate with Hari and his team on our revised requirements to receive feedback.
5. Using the notes given by the professor on the requirement document, we plan to give another look and revise it.

What goals were accomplished in this two-week period?

1. Developed a base for the code that does work but needs to be improved upon for better accuracy.
2. Uploaded what we have thus far for the dataset but need to add more checks to the dataset
3. Hari missed our weekly meeting, but we emailed him and the team the Requirements Document and the team responded with their approval
4. We looked at the notes from the professor and were able to edit and perfect the Requirements Document
5. Created a flowchart of how our algorithm will work

Reflect critically on any goals not accomplished.

1. We were unable to design the Algorithm to effectively test out at least one case because we began with the most difficult case of detecting fraud by seeing if the number in the amount box matches what is written on the amount line, and we could not sort out how to get this to accurately print out what was written.

What are the goals for the next two weeks?

1. Split off and work on different components of the problem
    a. Two people will work on a supervised learning model to detect crossed out patterns on checks
    b. One person will work on making an algorithm that can detect what is handwritten accurately
    c. One person will further develop the dataset
2. Perform testing with new code.

How many hours were spent on each goal noted above in the past two weeks?

Cassandra: Completed goals: Worked on creating fraudulent checks for the database (2 hours). Worked on specifications document (3 hours). Met with the professor to discuss project (30 min). Researched possible past implementations of similar projects (4 hours). Developed code for supervised learning model based on findings (1 hour).

Alex: Completed goals:

Mohamed: Completed goals: Worked on specifications document (3 hours). Worked on doing research about different approaches to use ML on detecting handwritten signatures, reading text from scanned images (OCR) (5 hours). Studied image preprocessing and data augmentation in order to create a larger dataset (2 hours). Met with Professor Kenneth to discuss and get clear expectations about our project (30 min).

Omar: Completed goals: Completed goals: research OCR and other methods of reading checks (4hrs). Worked on the specification document (3hrs). Discuss the project with the professor (30 min). Developed a starting code for the project (3hrs).

Status Report for JP Morgan Chase 1 for Week # 11

What were the goals for this two-week period?

1. Split off and work on different components of the problem
    a. Two people will work on a supervised learning model to detect crossed out patterns on checks
    b. One person will work on making an algorithm that can detect what is handwritten accurately
    c. One person will further develop the dataset
2. Perform testing with new code.

What goals were accomplished in this two-week period?

1. We divided and conquered:
a. Accomplished the creation of the supervised learning model that helps detect the crossed-out patterns and output results into 3 buckets.
b. We achieved an algorithm that can compare amounts on the check vs. A spreadsheet with good accuracy.
c. The dataset reached 400 checks.

2. Tested both algorithms and got pretty decent results and demonstrated the results to the professor.

Reflect critically on any goals not accomplished.

We accomplished all of our goals for this two-week period. Go us!

What are the goals for next two weeks?

- We aim to increase our dataset from 400 checks to 1000 checks.
- We aim to continue to test our amount vs. Spreadsheet algorithm to decrease false positive rate.
- We aim to train the 1000 checks in our machine learning model to detect crossed out names better.
- We aim to start research implementation for a 3rd type of fraud.
- Develop ROI detection for supervised learning algorithm

How many hours were spent on each goal noted above in the past two weeks?

Cassandra: Completed goals: Created 100 checks (2 hours). Uploaded all checks in the right format to the database (2 hours). Research of OCR (5 hours). Writing code with Omar (14 hours). Demo preparation and participation (3 hours). Meeting with JPM Chase (30 min)
Omar: Completed goals: Planning and discussing with the team (4 hours). Created 100 checks (2 hours). Research OCR and other methods (6 hours). Writing code (14 hours). Demo preparation and participation (3 hours). Meeting with JPM & Chase (30 min).
Alex: Completed goals: created 150 checks but only scanned 100 (2.5 hours) Researched ResNet and CNN algorithms that best fit our issue (2 hours) worked on code that implemented our algorithm (10 hours) Demo preparation and participation (3 hours) met with JPM (30 min)
Mohamed: Completed goals: Completed 100 checks(2hours). Preprocessed checks images (scan, crop, resize) (4 hours). Studied and researched CNN algorithm to utilize it to serve our supervised model (3hours). Coding the part of the project to detect crossed-out names checks (4 hours). Meeting with JPM Chase (30 min). Demo preparation and participation (3 hours)

Status Report for JPM Chase 1 Week #13

What were the goals for this two-week period?

1. Increase our dataset from 400 checks to 1000 checks.
2. Continue to test our amount vs. Spreadsheet algorithm to decrease false positive rate.
3. Train the 1000 checks in our machine learning model to detect crossed out names better.
4. Start research implementation for a 3rd type of fraud.
5. Develop ROI detection for supervised learning algorithm

What goals were accomplished in this two-week period?

1. We have been able to train our newer larger data set in our machine learning model and it better detects crossed out names.
2. We were able to integrate detection of abnormal spacing (3rd type of fraud) into our ML algorithm.
3. We were able to implement ROI detection within the supervised learning algorithm

Reflect critically on any goals not accomplished.

1. We increased our dataset to 700
2. Our amount vs spreadsheet algorithm needs further refinement in order to work well with our larger dataset

What are the goals for the next two weeks?

1. Explore new ways to serial number our checks
2. Further grow our dataset to the thousands
3. Look into the 4th type of check fraud we may address (mismatch courtesy and number)
4. Train, test, and improve our current models

How many hours were spent on each goal noted above in the past two weeks?
Cassandra: Completed goals: Added 200 checks to the data set (4 hours). Completed Excel spreadsheet and cropped and uploaded/scanned checks (4 hours). Team collaboration (6 hours). Met with professor (1.5 hours). Met with JPMC team (30 min). Worked on code (2 hours).
Alex: Completed goals: Up to 250 checks in the data set and rescanned them into the system into excel sheet (6 hours), collaborated with group team session (2 hours), researched how to create shippable product to JP (1 hour), met with professor (1.5).
Mohamed: Completed goals: Worked on creating more checks my dataset is at 150 checks (3 hours). Preprocessed checks scanning/cropping/filling checks' data on Excel spreadsheet (6 hours). Worked collaboratively on a group team session on coding stage for our codes solution(3hours). Meet with professor to discuss our requirements and formation of our dataset (1.5 Hours). Meet with Robert, Andion JPMC representative (30 minutes).

Omar: Completed goals: Totaling about 150 checks in the dataset including scanning and inputting in the datasheet (5 hours). Worked with the team on coding (5 hours). Worked independently on code (5 hours). Further research to try to make our code better (4.5 hours). Emails and Meetings with JP (1 hour). Meeting with professor (1.5 hours).

New Knowledge

This project has been a fantastic learning experience for us in the world of Machine Learning. It all started with us doing some research online about Neural Networks and figuring out how to use them to make a Supervised Machine Learning Algorithm. We then moved on to using TensorFlow and PyTorch libraries in a GoogleColab Notebook. We didn't stop there – we also investigated how to change our images in the algorithm.

Once we got the hang of GoogleColab, we wanted our algorithm to read handwriting. That's when we found OCR, a cool way for computers to understand handwriting. We spent time researching and testing to make sure it worked smoothly in our project. But we wanted our algorithm to be even more precise, focusing on specific parts of a check. So, we dove into researching ROIs, which is like telling the algorithm to pay attention to important areas.

Then came a special request from Chase – they wanted a way to fine-tune our algorithm. We did some more online research and discovered how to implement a tuner. Allowing the user to set a limit on how big a check can be before someone must manually review it.

In a nutshell, this project took us on a journey from learning about Neural Networks to playing with libraries, adjusting our algorithm's vision, teaching it to read handwriting, focusing on specific areas, and fine-tuning its performance.