AI and Decision Support Systems for Crash Preventability PAR Processing

U.S. Department of Transportation
Federal Motor Carrier Safety Administration

December 2024

FOREWORD

The Crash Preventability Determination Program (CPDP) is employed by the Federal Motor Carrier Safety Administration (FMCSA) to provide registered motor carriers an opportunity to have crashes removed from their crash indicator Behavior Analysis and Safety Improvement Category (BASIC) percentile, if they provide compelling evidence that the crash is eligible for the CPDP and was not preventable. Conducting manual reviews of these requests places high demands on Federal and Federally contracted personnel resources. This report describes employable techniques to fully or partially automate machine-assisted reviews of police accident report (PAR) submissions. This includes descriptions of the specific needs to conduct the automation process as the system currently exists, including machine learning techniques such as computer vision-based document analysis, optical character recognition, and natural language processing. While this report is specifically tailored to the team responsible for the CPDP at FMCSA, other individuals, parties, or entities exploring automation methods may find these techniques and their deployment of interest.

NOTICE

This document is disseminated under the sponsorship of the U.S. Department of Transportation (USDOT) in the interest of information exchange. The U.S. Government assumes no liability for the use of the information contained in this document. The contents of this report reflect the views of the contractor, who is responsible for the accuracy of the data presented herein. The contents do not necessarily reflect the official policy of the USDOT. This report does not constitute a standard, specification, or regulation.

The U.S. Government does not endorse products or manufacturers named herein. Trademarks or manufacturers' names appear in this report only because they are considered essential to the objective of this report.

QUALITY ASSURANCE STATEMENT

The Federal Motor Carrier Safety Administration (FMCSA) provides high-quality information to serve Government, industry, and the public in a manner that promotes public understanding. Standards and policies are used to ensure and maximize the quality, objectivity, utility, and integrity of its information. FMCSA periodically reviews quality issues and adjusts its programs and processes to ensure continuous quality improvement.

Technical Report Documentation Page

1. Report No. **FMCSA-RRT-22-009**	2. Government Accession No.	3. Recipient's Catalog No.

4. Title and Subtitle
AI and Decision Support Systems for Crash Preventability PAR Processing

5. Report Date
December 2024

6. Performing Organization Code

7. Author(s)
Miller, Andrew; Datta, Debanjan; Sundharam, Vaibhov; Sarkar, Abhijit; Rooney, George; Lobb, Collin

8. Performing Organization Report No.

9. Performing Organization Name and Address
**Virginia Tech Transportation Institute
3500 Transportation Research Plaza
Blacksburg, VA 24061**

10. Work Unit No. (TRAIS)

11. Contract or Grant No.
693JJ420D000005 / 693JJ420F000057

12. Sponsoring Agency Name and Address
**U.S. Department of Transportation
Federal Motor Carrier Safety Administration
Office of Research, Advanced Technology Division
1200 New Jersey Ave. SE
Washington, DC 20590**

13. Type of Report and Period Covered
Final Report, September 2020–November 2021

14. Sponsoring Agency Code
FMCSA

15. Supplementary Notes
Contracting Officer's Representative: Brian Routhier

16. Abstract

Automation processes and related machine learning techniques are being implemented across most industries to increase production or performance output or to reduce repetitive human tasks. The incorporation of automation elements into the Crash Preventability Determination Program conducted by the Federal Motor Carrier Safety Administration would help alleviate overburdened staff through a human-in-the-loop model of automation implementation in which a decision support system is developed to identify data elements within police accident reports submitted to the program and provide useful output to the team members.

An investigation of automation techniques was conducted, and a demonstration of the most viable automation processes was performed. The automation pipeline includes retrieval of police accident reports and applicable information from the program website, machine reading of ingested files, and parsing data elements toward performing crash element determinations. A production-level system is expected to result in an approximately 40 to 50 percent reduction in total time spent by analysts in their determination efforts. Ultimately, a series of recommendations for future implementation steps is provided.

17. Key Words
Artificial Intelligence, Decision Support System, Crash BASIC

18. Distribution Statement
No restrictions

19. Security Classif. (of this report) **Unclassified**	20. Security Classif. (of this page) **Unclassified**	21. No. of Pages **65**	22. Price

Reproduction of completed page authorized.

TABLE OF CONTENTS

LIST OF FIGURES (AND FORMULAS)

LIST OF TABLES

LIST OF ACRONYMS, ABBREVIATIONS, AND SYMBOLS

Acronym	**Definition**
AI	artificial intelligence
API	application programming interface
AWS	Amazon Web Services
BASIC	Behavior Analysis and Safety Improvement Category
CMV	commercial motor vehicle
CPDP	Crash Preventability Determination Program
CSA	Compliance, Safety, Accountability (program)
CV	computer vision
CVAT	Computer Vision Annotation Tool
DL	deep learning
DSS	decision support system
FMCSA	Federal Motor Carrier Safety Administration
GUI	graphical user interface
JSON	JavaScript Object Notation
MCMIS	Motor Carrier Management Information System
ML	machine learning
NLP	natural language processing
OCR	optical character recognition
PAR	police accident report
PII	personally identifiable information
PSP	Pre-employment Screening Program
RDR	request for data review
SMS	Safety Measurement System
SVD	singular value decomposition

Acronym	**Definition**
TF-IDF	term frequency-inverse document frequency
USDOT	U.S. Department of Transportation
VTTI	Virginia Tech Transportation Institute

EXECUTIVE SUMMARY

PURPOSE

FMCSA's Crash Preventability Determination Program (CPDP) has utility in improving the accuracy of crash-related information regarding the driver's contribution to the crash, as well as improving the risk assessment of motor carriers within the Compliance, Safety, Accountability (CSA) enforcement program. As reported during the crash preventability demonstration program and the CPDP, it is cumbersome and time consuming for the Crash Analysts to review the various Police Accident Reports (PARs) and other data to complete a request for data review (RDR). This demonstrable need presented an opportunity to evaluate the role artificial intelligence (AI) and decision support systems (DSSs) could play to reduce the burden on the Crash Analysts and FMCSA reviewers.

The goals of this project were to evaluate the technical, economic, and operational feasibility of applying AI and other DSSs within the CPDP. The AI/DSSs would be applied to automate portions of the review and analysis of RDRs, namely, identifying relevant crash parameters by comparing police accident reports (PARs) to State violation codes, then using the extracted information to make eligibility and preventability recommendations to DataQs analysts.

PROCESS

This project sought to highlight the most feasible and cost-efficient ways to produce crash-related determinations for the CPDP, as well as to demonstrate the AI and DSS model and convey the associated costs of implementing full or partial AI/DSS models. The demonstration component of the project included the acquisition of all necessary information for the execution of the AI/DSS model using Texas State PARs. The model obtained targeted RDRs, processed all acquired PARs as PDFs to create a machine-readable format, then utilized customized logic to automate the eligibility and preventability determinations for each submitted RDR. The determinations and other relevant information were extracted to a summary report file for accessibility. Further, the process for creating determinations for a single PAR is estimated to take thirty seconds, though the model was designed to create determinations for a batch of selected PARs.

A practical application of future production-level process may include running the AI/DSS model weekly, creating summary report files and determinations for all eligible PARs. Depending on the extent of AI/DSS implementation, namely, the number of State PARs with customized parsing logic, a significant number of RDRs may have automated determinations with little effort on behalf of the enforcement team. The automated workflow process for obtaining these determinations is displayed in Figure 1; however, the interpretation logic used must be customized for each version of a PAR.

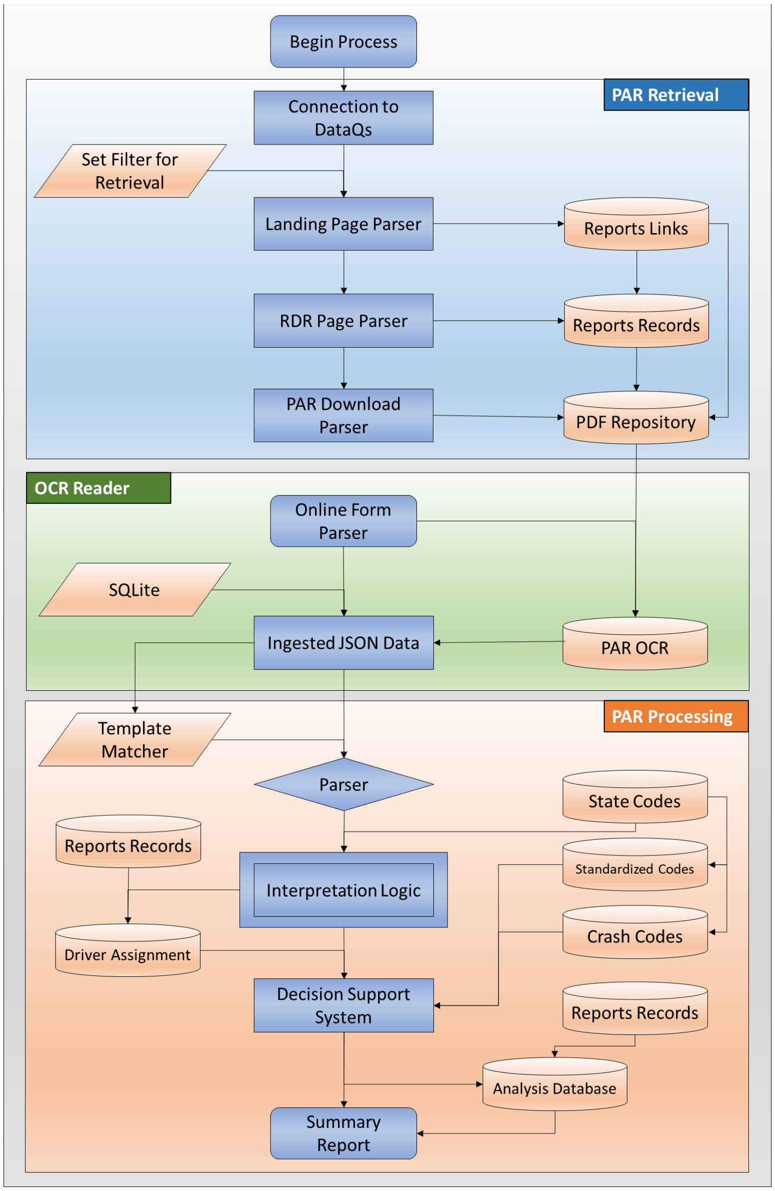


Figure 1. Diagram. Workflow for AI-based PAR analysis.

PROCESS WORKFLOW

The specific workflow of the AI/DSS model was established through conducting a literature review and trade-off analyses. The literature review was conducted on the available capabilities of various algorithms, tools, software, and models across both academic and industry sources. The trade-off analyses dictated which applied methodology would best fit the needs of FMCSA's enforcement team as well as assessing cost-to-implement. The steps to conduct the process were outlined as follows: (1) Environment Setup; (2) PAR Retrieval; (3) Document Parsing Prerequisites; (4) Document Parsing Process; (5) Narrative Analysis; (6) Crash Diagram Analysis; (7) Eligibility and Preventability Analysis; and (8) Summary Report Generation. These steps are described in Section 2 of the final report.

The executable process for implementing the current automation system is expected to perform the following steps:

1. Analyst sets a 1-week filter of open RDRs utilizing advanced search features and saves search to their profile.
2. Analyst establishes Python environment using preferred graphical user interface.
3. Analyst executes web parsing command to retrieve PARs from the FMCSA portal.
4. Analyst executes optical character recognition (OCR) parser on the Google Cloud Processing platform.
5. Analyst imports JSON output into the interpreter.
6. Analyst uploads summary reports to appropriate RDR and evaluates request based on report output to make final recommendation.

This process may be mitigated by the implementation of a production-level web application that is designed to perform each of these steps sequentially with minimal interaction with a user.

DEMONSTRATION RECOMMENDATIONS

While a learning-based system to read and assess different versions of PARs is both unwieldy and unreliable, a more nuanced creation consists of individual parsers for each version of the PAR in circulation. Each PAR provides a set of unique challenges that can be addressed using a continually built series of tools, codes, and modules that would ultimately hasten the development of each parser system.

With the development of the automation process for Texas, several additional options are available for future efforts. Features of other components may be identified and extracted using machine learning (ML) and computer vision (CV) techniques. These efforts may also be translated to the development of other PAR versions.

Throughout the exploration and demonstration of automation techniques, the research team has identified a series of process elements that resulted in a loss of data related to PARs or information contained in the PARs. These estimates are presented in Table 15 of the final report along with a proposed methodology for correcting the data loss through the automation system.

Ultimately, the research team recommends the following efforts for future iterations of the automation processing:

1. Expand the current automation system from Texas (approximately 11 percent of total RDRs) to the top 15 States (approximately 70 percent of total RDRs).
2. Expand the narrative using additional NLP techniques to utilize the unit assignment automation and crash type database, assist in determinations of eligibility and preventability, and highlight pertinent information for the DataQs analyst.
3. Build a repository of document excerpts that can be used for training purposes to identify key eligibility or preventability information.

4. Build a web application that can be utilized by DataQs analysts or other users without the need to interact with a terminal window or coding environment.
5. The proposed corrections could be implemented across the system process or as denoted for each specific version. This includes:
    a. Increasing accuracy by performing iterative investigation of incorrect values.
    b. Correcting web scraping issues to accommodate multiple files.
    c. Performing fuzzy matching or logic to ensure correct information is recorded.
    d. Performing CV OCR on selected excerpts from individual PAR versions that are not able to be read from Google's Document AI OCR processor.

Future coordinated efforts between State and Federal Government agencies may produce a database of digitized PARs, with data fields provided in a structured format. This would alleviate a significant burden on the AI/DSS process and allow for a simpler means for attaining automated determinations.

COST-BENEFIT ANALYSIS

A major goal of the project was to perform a cost-benefit analysis of the implementation of the AI/DSS model for all or most RDRs. Following the above recommendations, to attain a 75 percent efficacy rate among 80 percent of the submitted RDRs, customized solutions would have to be performed on the top 15 PAR versions by magnitude (i.e., TX, FL, CA, IN, PA, OH, GA, NC, IL, TN, AL, MI, MO, KY, NY). The efficacy rate constitutes the accuracy in which the model would produce eligibility and preventability determinations along with the summarized information, including the narrative, any crash diagrams, and the selected criteria that led to the creation of those determinations. Those summary reports with inaccurate information (i.e., the remaining 25 percent within the efficacy rate) would need more careful evaluation to make determinations for the RDR. The customized solution would include document reading, parsing, and interpretation as well as the DSSs as performed in the demonstration, but also include additional ML on extracted sections that are difficult to read by the OCR processor. Additional efforts to increase efficacy rate of determinations would include a full analysis on the extracted narratives, creating a token library of terminology for each PAR version while maintaining a library that exists across versions. Table 1 provides an estimated level of effort and associated costs for various entities to perform the analyses as demonstrated as well as the additional efforts described in the recommendations. The estimated labor and costs include the utilization of developed material throughout the duration of the project, including the web scraper for RDR information collection and PAR downloader, CV annotations for various States, and a compiled State code database used in the interpreters.

Table 1. Development estimations by organization entity for top 15 PAR versions to achieve 75 percent efficacy in eligibility and preventability determinations.

Entity	Interpreter Logic (hours)	Computer Vision on Extracted Data (hours)	Narrative Analysis (hours)	Quality Assurance (hours)	Total (hours)	Estimated Cost (fully encumbered)	Years to Positive Return (75% efficacy, fully automated)
VTTI	900	840	440	120	2,300	$230,000 (at $100/hr)	2.25
USDOT-IT	960	840	440	120	2,360	$188,800 (at $80/hr)	1.85
Private Sector	2,200	1,600	1,080	240	5,140	$819,200 (at $160/hr)	8.03

There are several considerations for the estimations identified in Table 1:

1. The estimated hours for development are to attain a 75 percent efficacy rate (i.e., accuracy of the determinations). The remaining 25 percent are likely unable to be parsed due to the limiting factors associated with OCR readers. These inaccuracies may be a result of not having enough information to make an informed decision, having incorrect information on the PARs, or are the result of processors not correctly identifying information. Despite an inability to make a determination within these instances, the summary reports may still be utilized by DataQs analysts in the PAR evaluation.
2. The estimates from private sector came from a rough order of magnitude of execution. The rough order of magnitude is targeted at -25 to +75 percent accuracy in costs.
3. The VTTI hourly rate includes labor, fringe, and indirect costs. A minimal fee is associated with Cloud OCR processing.
4. USDOT-IT hourly rates are estimated based on required position (e.g., Project Manager, Data Scientist, and Data Analyst).
5. An estimated number of 5,100 PARs would have automated determinations.
6. The estimated amount of time to complete one PAR by a DataQs analyst is 15 minutes. The implementation of the top 15 PAR versions at a 75 percent rate would decrease DataQs analysts' determination time by 1,275 hours annually if automated.
7. A human-in-the-loop model evaluating the produced summary report may decrease the reduction in total time spent by analysts in their determination efforts by roughly 20–33 percent (i.e., saving 854–1,020 hours annually).

SUMMARY

A customized solution for a subset of State PAR versions would require a significant effort by the automation developer but may result in an approximate 50 percent reduction in total time spent by DataQs analysts in their determination efforts. A cooperative effort between DataQs analysts, DataQs supervisors, and an automation development team may reduce the time further through mutual iterative development of each PAR version processor. This effort would involve the development team providing frequent updates on the status of the processor and having the DataQs team serve as subject matter experts to identify the most relevant information related to the determinations. The implementation of the AI/DSS model could ultimately alleviate the substantial burden on the enforcement team.

1. INTRODUCTION

One of the primary goals of the Federal Motor Carrier Safety Administration (FMCSA) is to prioritize its enforcement resources on the motor carriers that pose the greatest safety risks on U.S. roadways. FMCSA's Crash Preventability Determination Program (CPDP) began accepting Requests for Data Reviews (RDRs) in May 2020. In this program, FMCSA makes crash preventability determinations, allowing registered motor carriers an opportunity to have crashes removed from the calculation of their Crash Indicator Behavior Analysis and Safety Improvement Category (BASIC) percentile. Stakeholders have expressed concern that the use of all crashes in SMS, without an indication of preventability, may give an inaccurate impression about the risk posed by the company. Adjusted BASIC Crash Indicators can be used to evaluate if these preventability determinations improve the ability to identify the highest-risk motor carriers.

Under the CPDP, a motor carrier with a vehicle involved in one of 16 eligible crash types (see https://www.fmcsa.dot.gov/crash-preventability-determination-program) can submit an RDR to FMCSA's DataQs website (https://dataqs.fmcsa.dot.gov/). The RDR must include a police accident report (PAR) and/or other supporting documentation, such as insurance documents, pictures, or videos. The submitted documentation should demonstrate that the commercial motor vehicle (CMV) operator was involved in one of the eligible crash types, that the crash was not preventable, and that the crash in question should be removed from the calculation of the Crash Indicator BASIC measure and percentile in the Safety Measurement System (SMS) and noted in the Pre-employment Screening Program (PSP) as "not preventable." After submission to DataQs and confirmation of eligibility, FMCSA reviews the RDR and provides one of the following final determinations:

- "Reviewed – Not Preventable Crashes."
- "Reviewed – Preventable: FMCSA reviewed this crash and determined that it was Preventable."
- "Reviewed – Undecided: FMCSA reviewed this crash and could not make a preventability determination based on the evidence provided."

FMCSA DataQs analysts are responsible for the resolution of RDRs in the order the RDRs were submitted into the CPDP. An analyst first confirms the identity of the motor carrier and the record as listed in the Motor Carrier Management Information System (MCMIS) by reviewing the submitted PAR and MCMIS data to verify that the carrier is assigned to the crash record in question. Following confirmation of the match, an analyst reviews all documentation submitted by the carrier to determine if the crash fits within an eligible crash type. An analyst then reviews the information and subsequently decides the crash preventability determination.

Prior to the deployment of the CPDP, a demonstration of the program (Crash Preventability Demonstration Program) accepted RDRs during a 26-month period from June 1, 2017, to July 31, 2019. This demonstration program resulted in over 14,000 RDRs being submitted and reviewed and over 9,000 crash preventability determinations. During the demonstration program, the RDR process presented a significant time burden for DataQs analysts. During the rollout of the CPDP, which included eight additional eligible crash types, this burden continued and increased.

The growing capabilities of artificial intelligence (AI) and decision support systems (DSSs) could allow FMCSA to implement effective and efficient intelligence systems to alleviate a significant amount of time and effort on the part of the DataQs analysts. The tasks of reviewing PARs for eligibility and, if eligible, making crash preventability determinations, are often cumbersome due to the diversity of PARs. Further, they are often produced in a document format that is not easily searchable by a computer, significantly increasing the processing time. These factors necessitated the search for AI-based solutions for automated document analysis.

1.1 PROJECT OBJECTIVES

The objective of this project was to evaluate the technical, economic, and operational feasibility of applying AI and other DSSs in the CPDP. The AI/DSSs would be applied to automate portions of the review and analysis of RDRs, namely, identifying relevant crash parameters by comparing PARs to State violation codes, then using the extracted information to make eligibility and preventability recommendations to DataQs analysts. This report:

1. Demonstrates viable methodologies for implementing AI/DSS techniques.
2. Details the requirements and assumptions of technical steps.
3. Provides next steps for future implementation of DSSs.

This final report provides an overview of the project elements and details a comprehensive demonstration of the specific needs of AI techniques to analyze and review PAR/DataQs using:

- Computer Vision (CV)
- Document Analysis
- Optical Character Recognition (OCR) applications
- Natural Language Processing (NLP) applications and AI-based solutions incorporated into each of those techniques

1.2 EVALUATION AND DEMONSTRATION STEPS

The demonstration of crash eligibility and determination involves eight steps. Each of these steps aims toward a streamlined process that can efficiently incorporate potential AI techniques to handle the automation of crash eligibility and determination. The steps are as follows:

1. Environment Setup
2. PAR Retrieval
3. Document Parsing Prerequisites
4. Document Parsing Process

5. Narrative Analysis
6. Crash Diagram Analysis
7. Eligibility and Preventability Analysis
8. Summary Report Generation

Figure 2 illustrates the demonstrated process. Each step involves specific processing techniques that are useful to achieve a specific goal. The steps are described in Section 2.

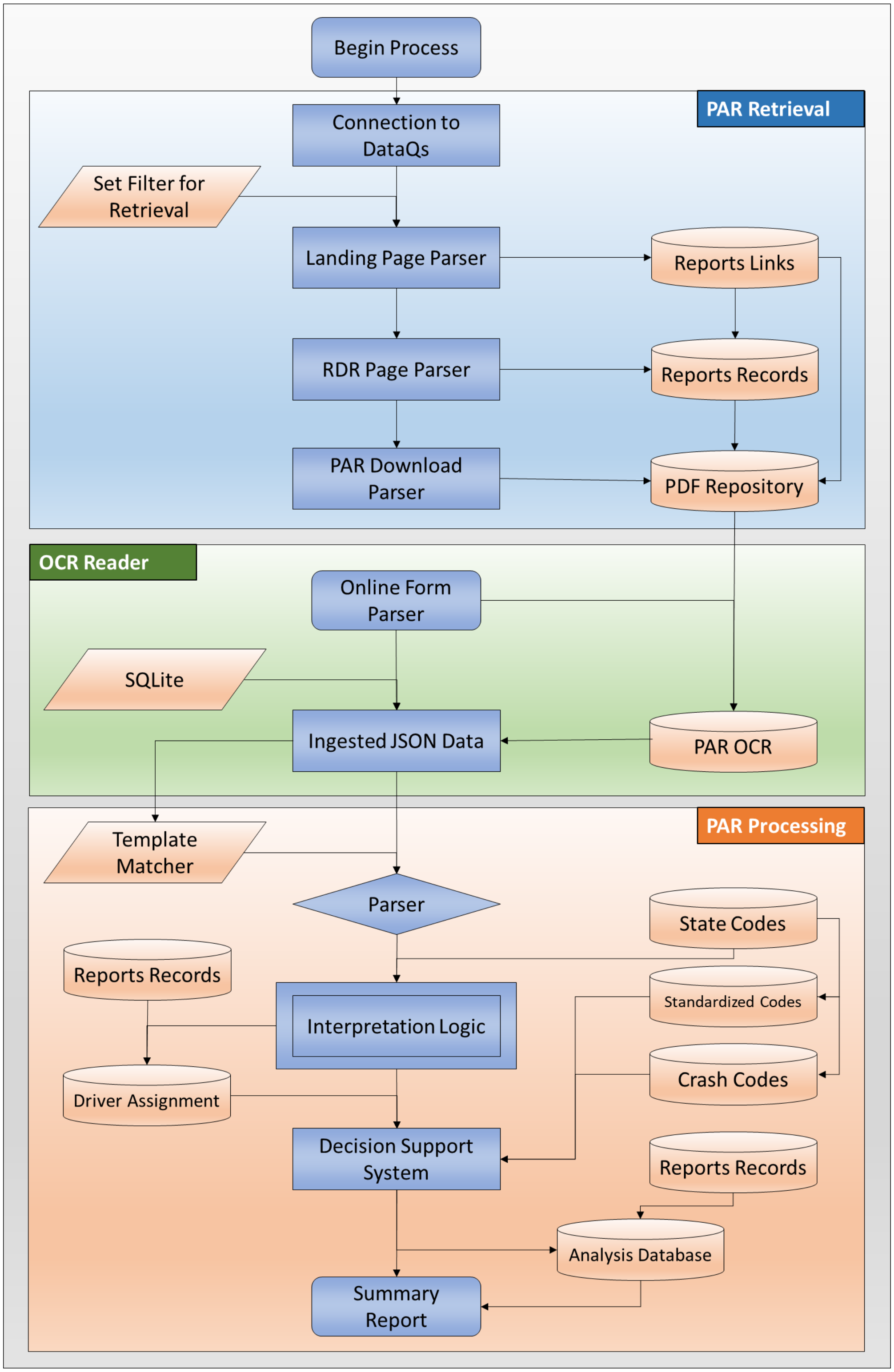


Figure 2. Diagram. Workflow for AI-based PAR analysis.

2. TECHNICAL STEPS

This section provides a step-by-step practical breakdown of the AI implementation. Each step details the execution of the task and considerations for implementation.

2.1 ENVIRONMENT SETUP

The system pipeline requires a customized environment to support the various functionalities implemented as part of the constituent modules. The current automation process is built in Python 3.8 due to (1) scalability and flexibility and (2) the accessibility of open-source libraries that support a wide range of functionalities. Additionally, Python 3.8 supports object-oriented design patterns that improve encapsulation and cohesiveness, and are also easy to read, maintain, and extend. The execution environment is set up to be an encapsulated, platform-agnostic environment that can be ported and easily set up and scaled. Anaconda is used to provide a virtual working environment.

Multiple Python packages are used to aid in the automation process, customization, and configurability. *YAML* (Yet Another Markup Language) files store extensible configurations and are supported by *PyYaml*. Multiprocessing is utilized at different stages using the built-in package *joblib*, which significantly improves throughput. For parsing XML-based annotations, *BeautifulSoup*, a versatile and powerful library, is utilized. For machine learning (ML) tasks, such as template page finder, *scikit-learn* is utilized. *Scikit-learn* is an open-source library that is widely adopted in the AI community. Numerical computations are aided by *NumPy*. The PAR retrieval process utilizes *Selenium* to scrape text from the DataQs website.

Much of the automation is encapsulated in the business and processing logic, which makes it transparent to the end user, improving usability. The overall design choice of the components of the environment was driven by the goals of creating a scalable, flexible, and maintainable software system.

2.1.1 Challenges and Future Steps

Although the operation of the process can be performed by any user with appropriate access, a production-level integration with current DataQs operations would provide a more approachable system solution. An option for system deployment to transform the project into a production-level service would be to migrate the code and database to a web application. This would provide analysts with several advantages. First, analysts would not have to create a virtual environment and install all the project dependencies manually, which can be time-consuming. Second, a web application would be easily assessed by all through a URL. Further, the implementation of the web application may include a graphical user interface (GUI) that can be used by analysts to download, parse, and produce PAR summary reports without having to go to the command line to run the program.

Alternatively, implementation of the system may be limited to specific individuals who provide reuploaded summary report documentation to each RDR submission. This would minimize DataQs involvement with the process while still providing necessary materials to contribute to accelerating the analysis processes. Further, both options may be implemented, with additional support provided to users to better track performed operations. For example, a production-level web application may provide information on the dates of the RDRs processed, States processed, or other filters. This would help prevent redundant PAR processing.

2.2 PAR RETRIEVAL ELEMENTS

After establishing the coding environment, the next critical step for automation requires connecting to the DataQs website, gathering relevant information regarding the RDR, and acquiring a PDF version of the PAR. The basic workflow is outlined in Figure 3, including databases created from the PAR retrieval process.

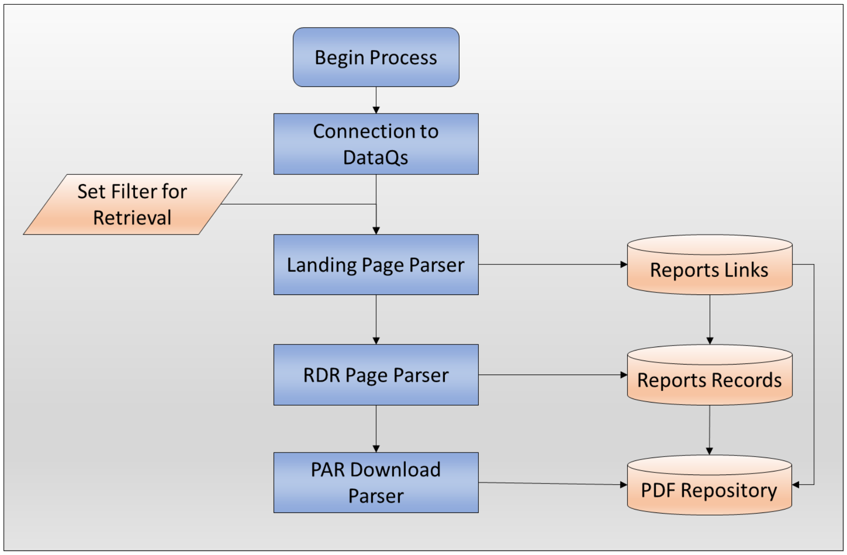


Figure 3. Diagram. Workflow for web parsing DataQs.

2.2.1 Submission Information Identification

The purpose of this step is to automate the acquisition and sorting of relevant information related to the RDR submission, download the associated PAR, and categorize data entries appropriately for use in the document parser and analysis. In general, the structure of the DataQs website dictates the format of the web parser. Further, the goals of the PAR eligibility and preventability analysis determine which information is scraped from the website.

The web parser produces a series of tables. These tables include information from the "My DataQs" webpage containing the filtered list of reviews requested and the subsequent details page for each RDR ID. The structure of the "My DataQs" webpage includes a filter function that dictates the data within an embedded master table in the webpage. The master table is structured to produce up to 100 results on each page, with the ability to change table elements without altering other elements on the webpage. Using this structure, the first parsed table includes a table of URL links to report details. The selected information is detailed in Table 2. The RDR report ID, crash report ID, and URL link to the details page are included. The "Key" column denotes whether the variable is used as a key to match with other databases. Due to the structure of the master table, creating a stored cache of URLs provides the most flexibility for parsing information from the website.

Table 2. Report links table of recorded variables and keys used to join tables.

Variable	Key	Description
id	No	Numeric designation of the entry within the web parsing tool.
timestamp	No	MM/DD/YYYY hh:mm:ss designation for when the entry was recorded.
report_id	Yes	The identification number of the RDR used for tracking by the DataQs system.
report_number	Yes	The PAR report number as provided by the requestor during the RDR submission process.
url	No	The landing URL page of the RDR containing detailed information and current status within the DataQs system.

Some information critical to automating the analysis is not obtainable from PAR uploads and must be obtained through the RDR submission. The submission information serves multiple purposes:

1. It provides a quick summary for filtering or manual review of the important information recorded across submissions.
2. It provides the automation algorithm with the USDOT number of the fleet involved in the crash to match with the values recorded within the PAR itself, allowing the algorithm to identify which unit number the fleet is associated with.
3. It records injuries and fatalities for use in the summary report.
4. It attains the submitted crash type to use in the automation algorithm during eligibility and preventability analysis.
5. It provides the current status of the open or closed RDR, which can be used for supervised learning during the analysis.
6. It allows DataQs analysts to match information to MCMIS from an aggregated source or allows future automation of that matching procedure.
7. It provides a link to the PAR, as it is attached in the details page of each RDR.

Table 3 provides the variables and descriptions of each of the elements included in the parsed table of RDR details.

Table 3. Report records table of recorded variables and keys used to join tables.

Variable	Key	Description
ID	No	Numeric designation of the entry within the web parsing tool.
report_number	Yes	The PAR report number as provided by the requestor during the RDR submission process.
state	Yes	The report State of the submitted PAR from the crash record.
date	No	MM/DD/YYYY record of the crash date as reported be the requestor.
carrier_name	No	The name of the carrier as submitted by the requestor.
us_dot_number	Yes	The USDOT number of the carrier as submitted by the requestor.

Variable	Key	Description
driver_name	No	The name of the driver submitted by the requestor as included in the crash record.
crash_type	No	The crash type submitted by the requestor chosen from the 18 available categories.
num_fatalities	No	The number of fatalities in the crash as submitted by the requestor.
num_injuries	No	The number of injuries in the crash as submitted by the requestor.
towaway	No	A yes or no flag of whether the vehicle involved in the crash was considered a towaway as submitted by the requestor.
vehicle_plate_number	No	The vehicle plate number associated with the requestor's vehicle.
par_status	No	The status of the RDR based on DataQs analysts' assessments.
par_url	Yes	The identified URL of the PAR associated with the RDR report ID.

2.2.2 PARs Download

The downloading of PARs is completed through parsing the URLs (par_url) within the report_records table (Table 3). Because PARs are uploaded as files and accessible on the system through a URL link (https://dataqs.fmcsa.dot.gov/MyDataQs/getFile.aspx?d=0000000), the files may be easily retrieved by anyone with appropriate access. The process identifies the uploaded document listed within the "Police Accident Report (PAR) Document" section and downloads the provided file. Using this process, the PARs are downloaded to a designated directory and renamed to a unique identifier consisting of a combination of the RDR ID and the crash report number, allowing ease of use for other automation processes.

2.2.3 Inputs, Outputs, and Outcomes

The process for executing the parser involves (1) parsing the report URLs, (2) using those URLs to record the specific RDR information and the URLs for the PARs, then (3) downloading the PARs and saving them to a set location. This process may be performed stepwise or concurrently. The PAR retrieval is performed using the Python files as described in Table 4.

Table 4. Executable Python files for the PAR retrieval process.

Python Files	Purpose
login_page.py	Sets logic for logging into the DataQs system.
landing_page.py	Sets the logic for parsing information on the landing page "My DataQs" including parsing through pages of the filtered table.
details_page.py	Sets the logic for parsing information on an individual RDR basis.
database.py	Sets the logic for enabling storage of entry records.
download.py	Sets the logic for downloading and naming PARs from the report records table.
main.py	Executes the parser with optional fields.

The DataQs website parser was executed on September 17, 2021, on closed RDRs and September 23, 2021, on open RDRs. The closed reports produced 12,331 data entries, and the open reports produced 2,156 files when filtering all RDRs submitted between May 1, 2020, and October 31, 2021, in which 1,000 files are listed. The breakdown of reports by State is detailed in Table 5, highlighting the total count of reports by States for each report designation. This breakdown is useful for prioritizing automation customization and algorithm development. The five most common States by RDR submission represent approximately 33 percent of all submissions by State. The top 20 States represent approximately 80 percent of all States. Creating the parsing logic for these States would allow for the bulk of submission evaluation to be automated.

Table 5. Recorded number of parsed reports by State.

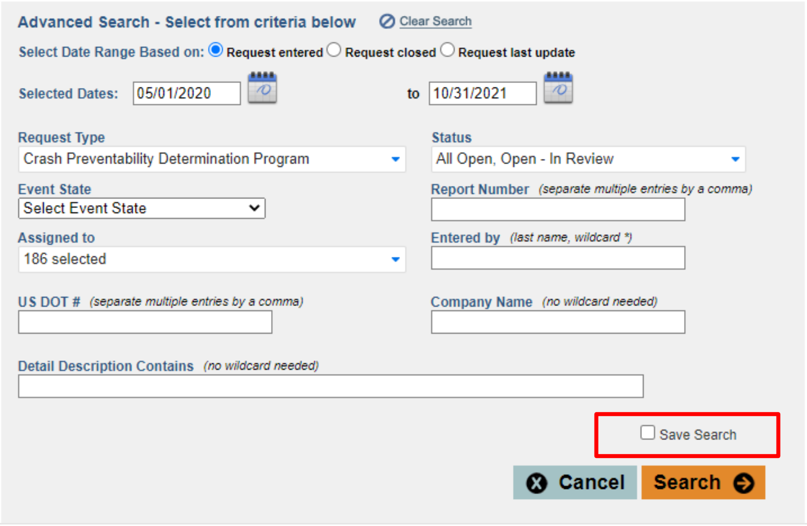
State	Closed Reports	Open Reports	Total Reports
TX	1,347	113	1,460 (11.0%)
FL	806	64	870 (6.5%)
CA	724	76	800 (6.0%)
IN	666	47	713 (5.3%)
PA	633	39	672 (5%)
OH	586	50	636 (4.8%)
GA	537	40	577 (4.3%)
NC	534	42	576 (4.3%)
IL	521	49	570 (4.3%)
TN	488	41	529 (4.0%)
AL	416	23	439 (3.3%)
MI	393	42	435 (3.3%)
MO	384	29	413 (3.1%)
KY	364	28	392 (2.9%)
NY	283	20	303 (2.3%)
LA	268	17	285 (2.1%)
SC	250	21	271 (2.0%)
AR	248	20	268 (2.0%)
VA	241	19	260 (2.0%)
OK	234	26	260 (2.0%)
WI	228	18	246 (1.8%)
AZ	179	17	196 (1.5%)
MS	178	15	193 (1.4%)
IA	154	4	158 (1.2%)
NJ	145	8	153 (1.1%)
MD	127	20	147 (1.1%)
KS	122	16	138 (1.0%)
CO	122	12	134 (1.0%)
WA	116	8	124 (0.9%)
MN	97	6	103 (0.8%)
WV	87	6	93 (0.7%)
WY	86	5	91 (0.7%)
NM	82	9	91 (0.7%)
CT	82	5	87 (0.7%)

State	Closed Reports	Open Reports	Total Reports
NE	80	5	85 (0.6%)
UT	76	8	84 (0.6%)
MA	72	2	74 (0.6%)
OR	61	6	67 (0.5%)
DE	55	4	59 (0.4%)
NV	49	3	52 (0.4%)
ID	47	4	51 (0.4%)
MT	47	3	50 (0.4%)
ME	37	1	38 (0.3%)
ND	24	0	24 (0.2%)
NH	18	5	23 (0.2%)
SD	20	2	22 (0.2%)
DC	7	0	7 (0.1%)
RI	7	0	7 (0.1%)
VT	3	1	4 (0%)
AK	0	1	1 (0%)
HI	0	0	0 (0%)
Total	12,331	1,000	13,331 (100%)

2.2.4 Customization Elements

Multiple customizations are available in the PAR retrieval process. These customizations are provided often as parameters to the automation execution or through the identification of relevant or desired data (e.g., filtering of RDRs).

1. The web parser function requires the appropriate access for logging in to the system. This may come through either a “DataQs Account” username and password or an “FMCSA Portal Account” username and password. This option can be set through altering automation code.
2. The login file containing ID and password should be saved in a secure location or may be entered before running the web parser.
3. The automation process runs exclusively on the current set of filters and does not request changes in the filter inside the executable process. Therefore, the data recorded and files downloaded are based on the filter as it is saved to the user’s account. This filter is saved to the profile of the user and persists across multiple devices. The filter used for open RDRs is displayed in Figure 4, with a red box highlighting the means for saving searches for utilization in the automation. Setting filters allows users to capture data relevant only to their current needs.



Advanced Search - Select from criteria below

Select Date Range Based on: ● Request entered ○ Request closed ○ Request last update

Selected Dates: 05/01/2020 to 10/31/2021

Request Type: Crash Preventability Determination Program

Status: All Open, Open - In Review

Event State: Select Event State

Report Number (separate multiple entries by a comma)

Assigned to: 186 selected

Entered by (last name, wildcard *)

US DOT # (separate multiple entries by a comma)

Company Name (no wildcard needed)

Detail Description Contains (no wildcard needed)

□ Save Search

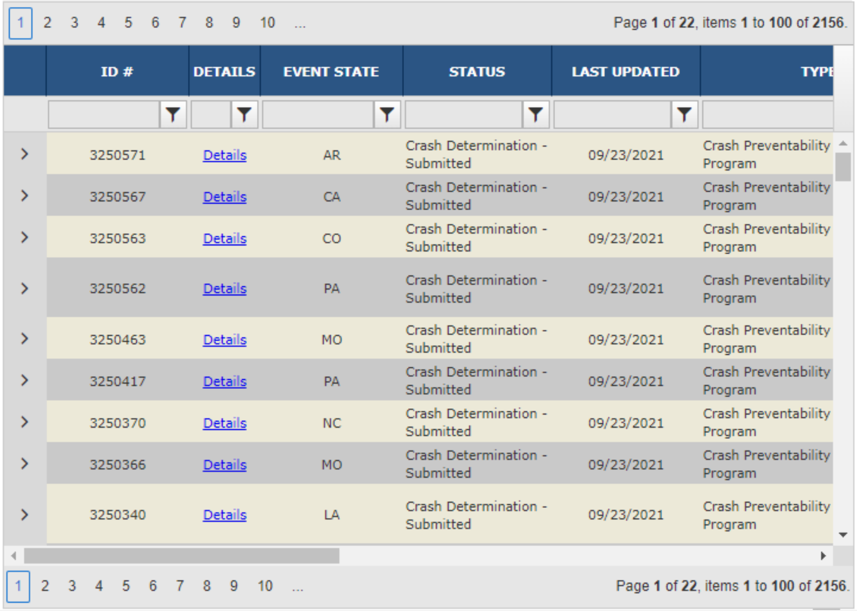
Figure 4. Screen capture. Excerpt from DataQs system of filtered table results.

4. The database storage can be designated to store records or files in certain directories and with designated names following a set logic. For the purposes of the evaluation, the research and development team chose to create "open" and "closed" sets. The difference is that the "closed" records contain a conclusion regarding the PAR preventability.
5. PARs can be downloaded using filters across any accessible variable in the tables. Common filters may be based on States or the timestamp of the initial records pull. Conversely, all PARs may be downloaded at once. Records and files may be set to append tables, overwrite tables, or create new tables.
6. The desired browser may be customized if up-to-date browser drivers are included in automation environment. Currently, the web parser operates using Google Chrome.

2.2.5 Challenges and Future Steps

Some noted challenges exist for the execution of the PAR retrieval process. First, within the CPDP, PARs are required and kept separate within the webpage from other supporting documents. However, some requestors may upload multiple .pdf files or other supporting documents in the wrong location. By nature, the parser cannot identify the file type until it is downloaded and thus may not pull the PAR as required. This may be remedied either by downloading manually or by recording the records for each file submitted and removing duplicates. The parser does currently handle the separation of .pdf files from other file types submitted (e.g., .jpg, .mov) and renames and stores them according to set parameters. The number of non-.pdf files is relatively small (approximately 2 percent) but noteworthy.

Second, the filtered results, presented in the webpage as a master table (Figure 5) highlight some considerations for the parser.



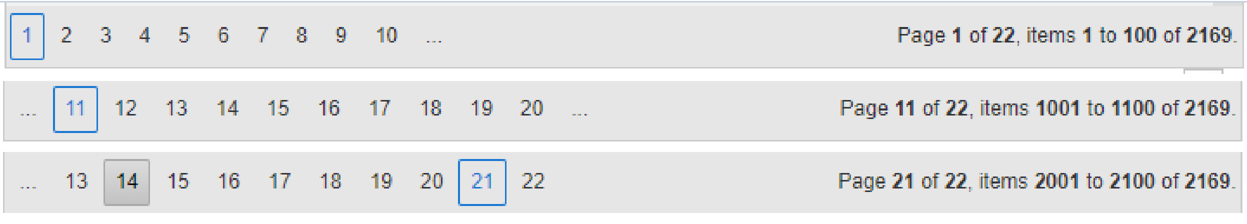
1 2 3 4 5 6 7 8 9 10 ... Page **1** of **22**, items **1** to **100** of **2156**.

	ID #	DETAILS	EVENT STATE	STATUS	LAST UPDATED	TYPE
>	3250571	Details	AR	Crash Determination - Submitted	09/23/2021	Crash Preventability Program
>	3250567	Details	CA	Crash Determination - Submitted	09/23/2021	Crash Preventability Program
>	3250563	Details	CO	Crash Determination - Submitted	09/23/2021	Crash Preventability Program
>	3250562	Details	PA	Crash Determination - Submitted	09/23/2021	Crash Preventability Program
>	3250463	Details	MO	Crash Determination - Submitted	09/23/2021	Crash Preventability Program
>	3250417	Details	PA	Crash Determination - Submitted	09/23/2021	Crash Preventability Program
>	3250370	Details	NC	Crash Determination - Submitted	09/23/2021	Crash Preventability Program
>	3250366	Details	MO	Crash Determination - Submitted	09/23/2021	Crash Preventability Program
>	3250340	Details	LA	Crash Determination - Submitted	09/23/2021	Crash Preventability Program

1 2 3 4 5 6 7 8 9 10 ... Page **1** of **22**, items **1** to **100** of **2156**.

Figure 5. Screen capture. Results of the open RDR filter within a master table.

The master table always shows 10 pages worth of results, where the ellipses ("…") following the designated 10 pages produces the next 10 pages of results (see Figure 6). When going to the final set of results, the 10 pages defaults to the next page but provides pages already parsed. This has been accounted for in the current version of web parsing.



1 2 3 4 5 6 7 8 9 10 ... Page **1** of **22**, items **1** to **100** of **2169**.

... 11 12 13 14 15 16 17 18 19 20 ... Page **11** of **22**, items **1001** to **1100** of **2169**.

... 13 14 15 16 17 18 19 20 21 22 Page **21** of **22**, items **2001** to **2100** of **2169**.

Figure 6. Screen capture. Demonstration of page changing within the master table.

Third, the filters included in the master table produce results for the current 100 elements on the page rather than the full set of data. This leads to any filtering to be done within the parser as needed (e.g., by State). This is currently being managed by the execution of the automation process as an options input to the command line.

Future steps may include further identifying appropriate PAR files through web scraping logic, or by customizing the report_records tables to produce multiple records, which are then parsed by the OCR reader and automation system. These would increase the number of viable summary reports that would be provided to DataQs analysts.

2.3 DOCUMENT PARSING PREREQUISITES

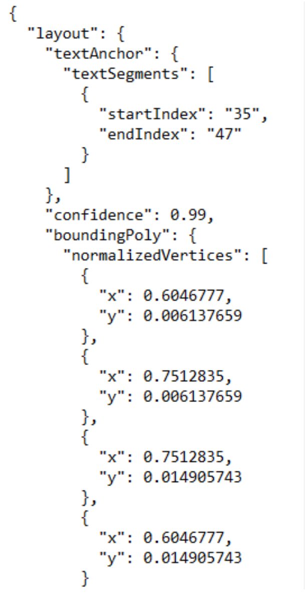
The various prerequisites for executing the document parsing process are detailed below.

2.3.1 Document OCR Process

With the PARs acquired through web parsing, the document parsing process begins by running each PAR through an OCR algorithm and processor. The goals of the OCR processor are to identify a character and find the pixel location of the character in the document. There are several cloud-based platforms that offer services to process forms and documents. These platforms include Amazon Web Services (AWS), the Google Vision application programming interface (API), and Microsoft Azure. These platforms provide a web-based interface and can be accessed via GUI or command line interface. These platforms can analyze any document to extract text. For example, Google's Document AI API provides a template-agnostic solution that can work with any form, including PARs, and can extract text, checkboxes, and tables.

These kinds of architectures are especially useful for cases where forms in different formats or versions need to be processed. Although each document parser may be able to read and decipher documents, additional work is necessary to ensure data validity. As all these platforms are built for further software development in mind, they result in structured output showing the exact location of the detected text in the document.

Throughout this project, Google's Document AI was utilized to provide a readily available document form parser and OCR, though both the AWS and Google AI OCR form parser were tested. Google API was chosen as their method seemed most appropriate for privacy protection of PARs. The output of the OCR processor is a JSON object format containing a string of text followed by a series of text blocks and segments associating the text to pixel-based (x, y) coordinates. Figure 7 displays an indexed segment from position 35 to 47 of the extracted text (e.g., "Law Enforcement") that is contained in the boundary box formed by the four (x, y) coordinate pixelized vectors. The JSON object format is input into the automation algorithm for field matching and DSS.

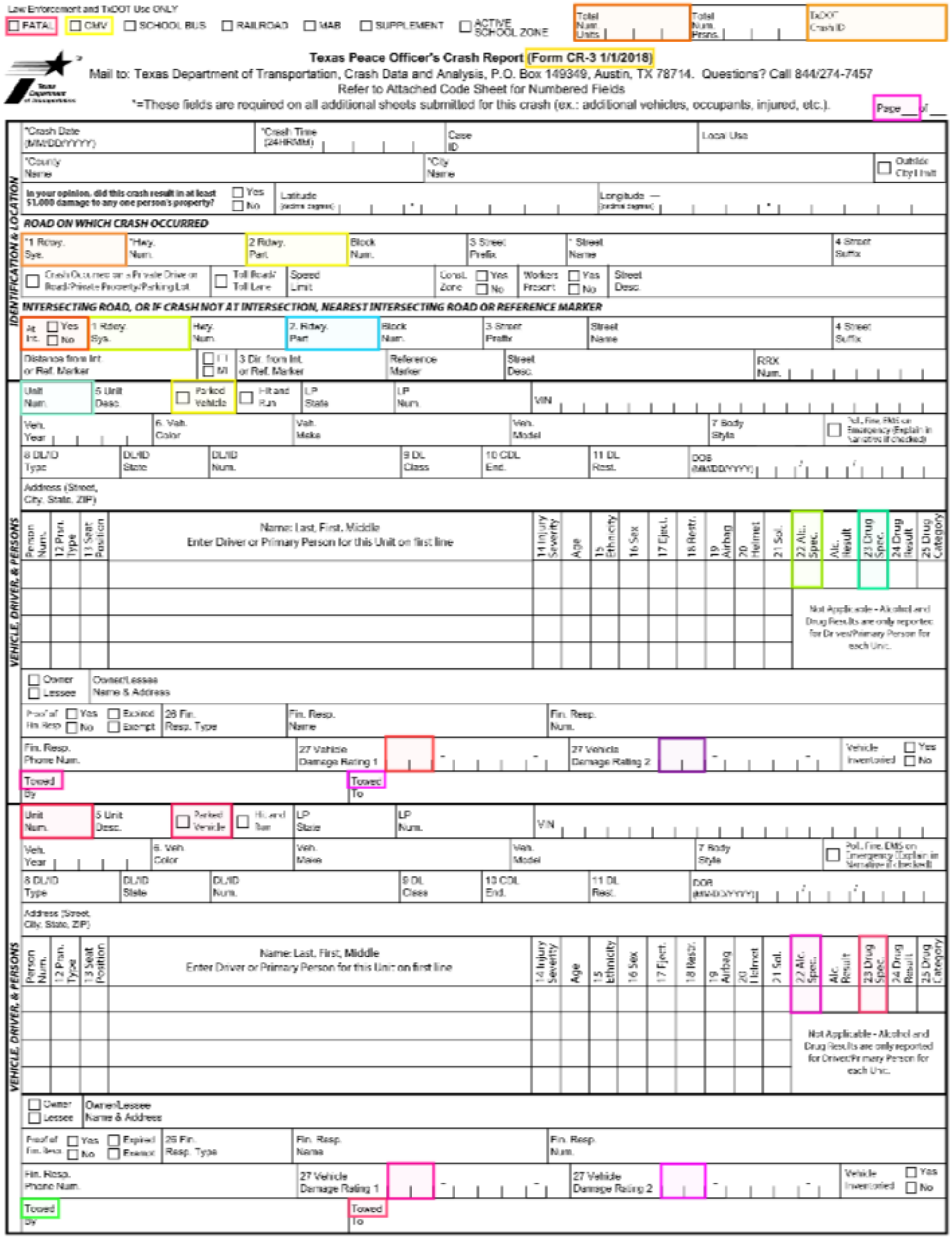


```
{
  "layout": {
    "textAnchor": {
      "textSegments": [
        {
          "startIndex": "35",
          "endIndex": "47"
        }
      ]
    },
    "confidence": 0.99,
    "boundingPoly": {
      "normalizedVertices": [
        {
          "x": 0.6046777,
          "y": 0.006137659
        },
        {
          "x": 0.7512835,
          "y": 0.006137659
        },
        {
          "x": 0.7512835,
          "y": 0.014905743
        },
        {
          "x": 0.6046777,
          "y": 0.014905743
        }
```

Figure 7. Screen capture. Extracted text segment and related boundary box coordinates.

2.3.2 Document Template

In addition to the JSON object format output produced by the OCR processor, several other inputs are required for automating and executing the document parsing process. First, an annotated template (see Figure 8 and Figure 9) is used to dictate the expected boundary boxes of tokens or keywords relevant to the analysis process. These boundary boxes consist of the (x, y) coordinates of the tokens.



Law Enforcement and TxDOT Use ONLY

□ FATAL □ CMV □ SCHOOL BUS □ RAILROAD □ MAB □ SUPPLEMENT □ ACTIVE SCHOOL ZONE | Total Num. Units | Total Num. Prsns. | TxDOT Crash ID

Texas Peace Officer's Crash Report (Form CR-3 1/1/2018)
Mail to: Texas Department of Transportation, Crash Data and Analysis, P.O. Box 149349, Austin, TX 78714. Questions? Call 844/274-7457
Refer to Attached Code Sheet for Numbered Fields
*=These fields are required on all additional sheets submitted for this crash (ex.: additional vehicles, occupants, injured, etc.).
Page __ of __

IDENTIFICATION & LOCATION

*Crash Date (MM/DD/YYYY) | *Crash Time (24HRMM) | Case ID | Local Use
*County Name | *City Name | □ Outside City Limit
In your opinion, did this crash result in at least $1,000 damage to any one person's property? □ Yes □ No | Latitude (decimal degrees) | Longitude — (decimal degrees)

ROAD ON WHICH CRASH OCCURRED

*1 Rdwy. Sys. | *Hwy. Num. | 2 Rdwy. Part | Block Num. | 3 Street Prefix | *Street Name | 4 Street Suffix
□ Crash Occurred on a Private Drive or Road/Private Property/Parking Lot | □ Toll Road/ Toll Lane | Speed Limit | Const. Zone □ Yes □ No | Workers Present □ Yes □ No | Street Desc.

INTERSECTING ROAD, OR IF CRASH NOT AT INTERSECTION, NEAREST INTERSECTING ROAD OR REFERENCE MARKER

At Int. □ Yes □ No | 1 Rdwy. Sys. | Hwy. Num. | 2. Rdwy. Part | Block Num. | 3 Street Prefix | Street Name | 4 Street Suffix
Distance from Int. or Ref. Marker | □ FT □ MI | 3 Dir. from Int. or Ref. Marker | Reference Marker | Street Desc. | RRX Num.

VEHICLE, DRIVER, & PERSONS

Unit Num. | 5 Unit Desc. | □ Parked Vehicle | □ Hit and Run | LP State | LP Num. | VIN
Veh. Year | 6. Veh. Color | Veh. Make | Veh. Model | 7 Body Style | □ Pol., Fire, EMS on Emergency (Explain in Narrative if checked)
8 DL/ID Type | DL/ID State | DL/ID Num. | 9 DL Class | 10 CDL End. | 11 DL Rest. | DOB (MM/DD/YYYY)
Address (Street, City, State, ZIP)

Person Num. | 12 Prsn. Type | 13 Seat Position | Name: Last, First, Middle Enter Driver or Primary Person for this Unit on first line | 14 Injury Severity | Age | 15 Ethnicity | 16 Sex | 17 Eject. | 18 Restr. | 19 Airbag | 20 Helmet | 21 Sol. | 22 Alc. Spec. | Alc. Result | 23 Drug Spec. | 24 Drug Result | 25 Drug Category

Not Applicable - Alcohol and Drug Results are only reported for Driver/Primary Person for each Unit.

□ Owner □ Lessee | Owner/Lessee Name & Address
Proof of Fin. Resp. □ Yes □ No □ Expired □ Exempt | 28 Fin. Resp. Type | Fin. Resp. Name | Fin. Resp. Num.
Fin. Resp. Phone Num. | 27 Vehicle Damage Rating 1 | 27 Vehicle Damage Rating 2 | Vehicle Inventoried □ Yes □ No
Towed By | Towed To

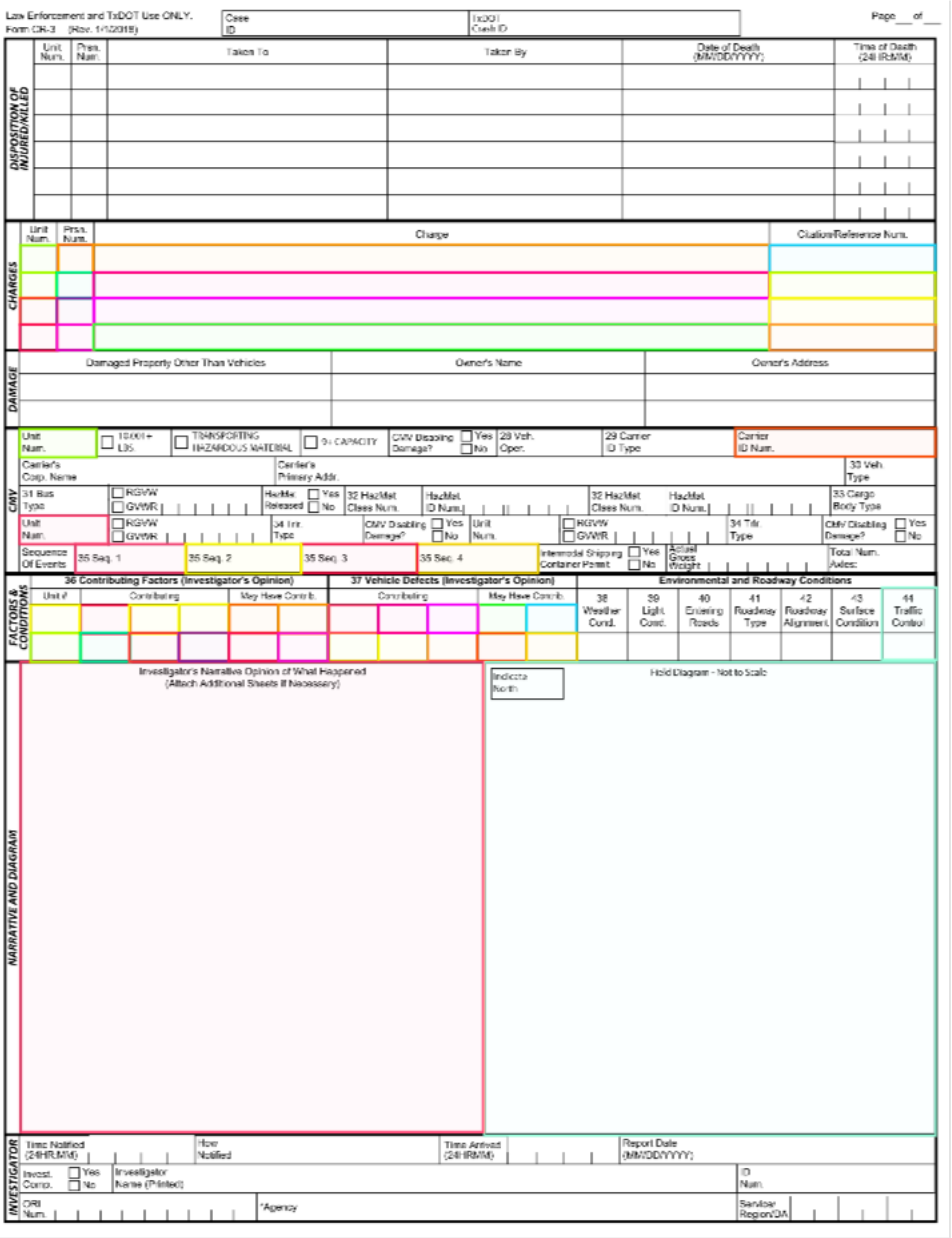
Unit Num. | 5 Unit Desc. | □ Parked Vehicle | □ Hit and Run | LP State | LP Num. | VIN
Veh. Year | 6. Veh. Color | Veh. Make | Veh. Model | 7 Body Style | □ Pol., Fire, EMS on Emergency (Explain in Narrative if checked)
8 DL/ID Type | DL/ID State | DL/ID Num. | 9 DL Class | 10 CDL End. | 11 DL Rest. | DOB (MM/DD/YYYY)
Address (Street, City, State, ZIP)

Person Num. | 12 Prsn. Type | 13 Seat Position | Name: Last, First, Middle Enter Driver or Primary Person for this Unit on first line | 14 Injury Severity | Age | 15 Ethnicity | 16 Sex | 17 Eject. | 18 Restr. | 19 Airbag | 20 Helmet | 21 Sol. | 22 Alc. Spec. | Alc. Result | 23 Drug Spec. | 24 Drug Result | 25 Drug Category

Not Applicable - Alcohol and Drug Results are only reported for Driver/Primary Person for each Unit.

□ Owner □ Lessee | Owner/Lessee Name & Address
Proof of Fin. Resp. □ Yes □ No □ Expired □ Exempt | 28 Fin. Resp. Type | Fin. Resp. Name | Fin. Resp. Num.
Fin. Resp. Phone Num. | 27 Vehicle Damage Rating 1 | 27 Vehicle Damage Rating 2 | Vehicle Inventoried □ Yes □ No
Towed By | Towed To

Figure 8. Screen capture. Annotated image of page 1 of the Texas CR-3 PAR template.



Law Enforcement and TxDOT Use ONLY. Form CR-3 (Rev. 1/1/2018)

Case ID | TxDOT Crash ID | Page ___ of ___

DISPOSITION OF INJURED/KILLED

Unit Num.	Prsn. Num.	Taken To	Taken By	Date of Death (MM/DD/YYYY)	Time of Death (24 HR:MM)

CHARGES

Unit Num.	Prsn. Num.	Charge	Citation/Reference Num.

DAMAGE

Damaged Property Other Than Vehicles	Owner's Name	Owner's Address

CMV

Unit Num. | □ 10,001+ LBS. | □ TRANSPORTING HAZARDOUS MATERIAL | □ 9+ CAPACITY | CMV Disabling Damage? □ Yes □ No | 28 Veh. Oper. | 29 Carrier ID Type | Carrier ID Num.

Carrier's Corp. Name | Carrier's Primary Addr. | 30 Veh. Type

31 Bus Type | □ RGVW □ GVWR | HazMat Released □ Yes □ No | 32 HazMat Class Num. | HazMat ID Num. | 32 HazMat Class Num. | HazMat ID Num. | 33 Cargo Body Type

Unit Num. | □ RGVW □ GVWR | 34 Trlr. Type | CMV Disabling Damage? □ Yes □ No | Unit Num. | □ RGVW □ GVWR | 34 Trlr. Type | CMV Disabling Damage? □ Yes □ No

Sequence Of Events | 35 Seq. 1 | 35 Seq. 2 | 35 Seq. 3 | 35 Seq. 4 | Intermodal Shipping Container Permit □ Yes □ No | Actual Gross Weight | Total Num. Axles:

FACTORS & CONDITIONS

36 Contributing Factors (Investigator's Opinion)						37 Vehicle Defects (Investigator's Opinion)					Environmental and Roadway Conditions						
Unit #	Contributing			May Have Contrib.		Contributing			May Have Contrib.		38 Weather Cond.	39 Light Cond.	40 Entering Roads	41 Roadway Type	42 Roadway Alignment	43 Surface Condition	44 Traffic Control

NARRATIVE AND DIAGRAM

Investigator's Narrative Opinion of What Happened (Attach Additional Sheets If Necessary)

Field Diagram - Not to Scale

Indicate North

INVESTIGATOR

Time Notified (24HR:MM) | How Notified | Time Arrived (24HR:MM) | Report Date (MM/DD/YYYY)

Invest. Comp. □ Yes □ No | Investigator Name (Printed) | ID Num.

ORI Num. | *Agency | Service Region/DA

Figure 9. Screen capture. Annotated image of page 2 of the Texas CR-3 PAR template.

Each token contains the boundary box and an annotation, or an identifier, that is unique to that token. For example, on the Texas CR-3 PAR form, there are three opportunities for units to be assigned as contributing factors of the crash. Although these fields share the same meaning and State codes, they are recorded separately to utilize the boundary boxes appropriately. The annotation labels used for Texas are displayed in Figure 10. These labels are unique to the Texas CR-3 PAR form, but have similar iterations across States, as the annotations are based on the RDR crash type and subsequent eligibility and preventability analysis.

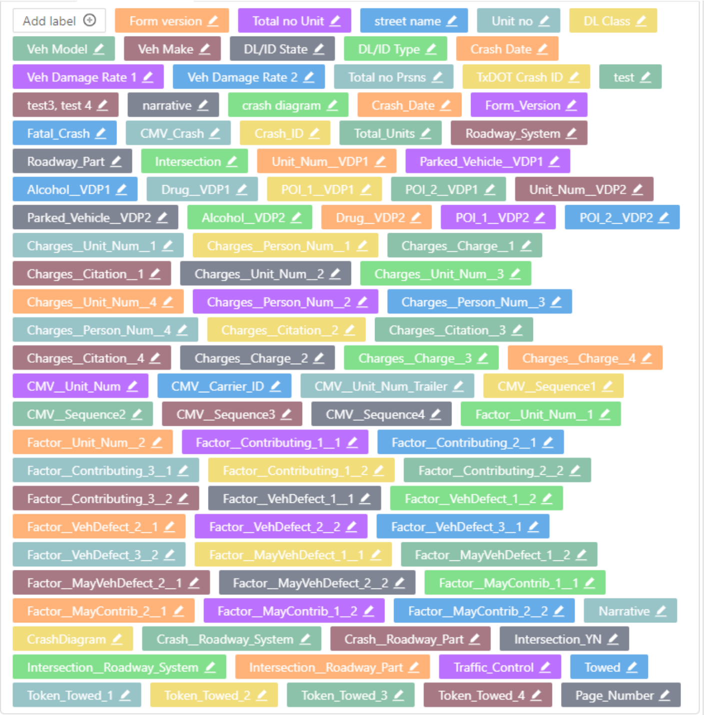
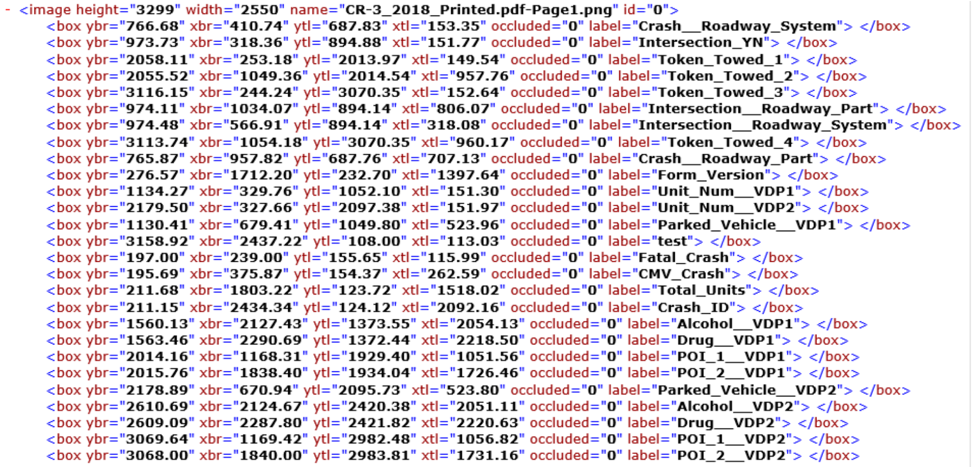


Figure 10. Screen capture. Annotation labels for the Texas CR-3 PAR template.

The annotations provide an .xml file containing the boundary box coordinates, as displayed in Figure 11. These coordinates are critical for matching the template annotations with those provided by the OCR processor.

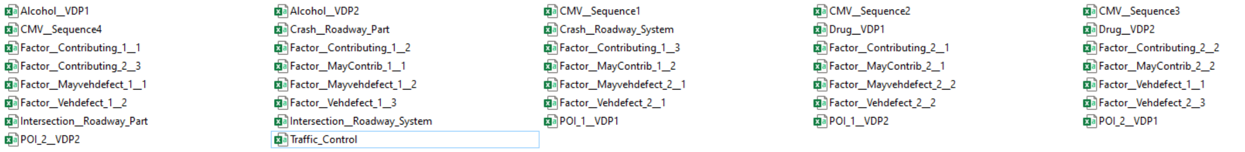


```
- <image height="3299" width="2550" name="CR-3_2018_Printed.pdf-Page1.png" id="0">
    <box ybr="766.68" xbr="410.74" ytl="687.83" xtl="153.35" occluded="0" label="Crash__Roadway_System"> </box>
    <box ybr="973.73" xbr="318.36" ytl="894.88" xtl="151.77" occluded="0" label="Intersection_YN"> </box>
    <box ybr="2058.11" xbr="253.18" ytl="2013.97" xtl="149.54" occluded="0" label="Token_Towed_1"> </box>
    <box ybr="2055.52" xbr="1049.36" ytl="2014.54" xtl="957.76" occluded="0" label="Token_Towed_2"> </box>
    <box ybr="3116.15" xbr="244.24" ytl="3070.35" xtl="152.64" occluded="0" label="Token_Towed_3"> </box>
    <box ybr="974.11" xbr="1034.07" ytl="894.14" xtl="806.07" occluded="0" label="Intersection__Roadway_Part"> </box>
    <box ybr="974.48" xbr="566.91" ytl="894.14" xtl="318.08" occluded="0" label="Intersection__Roadway_System"> </box>
    <box ybr="3113.74" xbr="1054.18" ytl="3070.35" xtl="960.17" occluded="0" label="Token_Towed_4"> </box>
    <box ybr="765.87" xbr="957.82" ytl="687.76" xtl="707.13" occluded="0" label="Crash__Roadway_Part"> </box>
    <box ybr="276.57" xbr="1712.20" ytl="232.70" xtl="1397.64" occluded="0" label="Form_Version"> </box>
    <box ybr="1134.27" xbr="329.76" ytl="1052.10" xtl="151.30" occluded="0" label="Unit_Num__VDP1"> </box>
    <box ybr="2179.50" xbr="327.66" ytl="2097.38" xtl="151.97" occluded="0" label="Unit_Num__VDP2"> </box>
    <box ybr="1130.41" xbr="679.41" ytl="1049.80" xtl="523.96" occluded="0" label="Parked_Vehicle__VDP1"> </box>
    <box ybr="3158.92" xbr="2437.22" ytl="108.00" xtl="113.03" occluded="0" label="test"> </box>
    <box ybr="197.00" xbr="239.00" ytl="155.65" xtl="115.99" occluded="0" label="Fatal_Crash"> </box>
    <box ybr="195.69" xbr="375.87" ytl="154.37" xtl="262.59" occluded="0" label="CMV_Crash"> </box>
    <box ybr="211.68" xbr="1803.22" ytl="123.72" xtl="1518.02" occluded="0" label="Total_Units"> </box>
    <box ybr="211.15" xbr="2434.34" ytl="124.12" xtl="2092.16" occluded="0" label="Crash_ID"> </box>
    <box ybr="1560.13" xbr="2127.43" ytl="1373.55" xtl="2054.13" occluded="0" label="Alcohol__VDP1"> </box>
    <box ybr="1563.46" xbr="2290.69" ytl="1372.44" xtl="2218.50" occluded="0" label="Drug__VDP1"> </box>
    <box ybr="2014.16" xbr="1168.31" ytl="1929.40" xtl="1051.56" occluded="0" label="POI_1__VDP1"> </box>
    <box ybr="2015.76" xbr="1838.40" ytl="1934.04" xtl="1726.46" occluded="0" label="POI_2__VDP1"> </box>
    <box ybr="2178.89" xbr="670.94" ytl="2095.73" xtl="523.80" occluded="0" label="Parked_Vehicle__VDP2"> </box>
    <box ybr="2610.69" xbr="2124.67" ytl="2420.38" xtl="2051.11" occluded="0" label="Alcohol__VDP2"> </box>
    <box ybr="2609.09" xbr="2287.80" ytl="2421.82" xtl="2220.63" occluded="0" label="Drug__VDP2"> </box>
    <box ybr="3069.64" xbr="1169.42" ytl="2982.48" xtl="1056.82" occluded="0" label="POI_1__VDP2"> </box>
    <box ybr="3068.00" xbr="1840.00" ytl="2983.81" xtl="1731.16" occluded="0" label="POI_2__VDP2"> </box>
```

Figure 11. Screen capture. Excerpt of boundary box coordinates for the Texas CR-3 PAR template.

2.3.3 State Code Database

A significant step in interpreting any results is to match the OCR output with the appropriate codes. These State codes contain information critical to the eligibility and preventability analysis. The State codes are stored as individual .csv files and named after the annotation labels (see Figure 12 for all Texas State codes) and contain only the keys and values attributed to that label (see Table 6 for examples). The .csv files are converted to SQLite database files and used for querying throughout the automation process.



```
Alcohol__VDP1                Alcohol__VDP2                  CMV__Sequence1             CMV__Sequence2              CMV__Sequence3
CMV__Sequence4               Crash__Roadway_Part            Crash__Roadway_System      Drug__VDP1                  Drug__VDP2
Factor__Contributing_1__1    Factor__Contributing_1__2      Factor__Contributing_1__3  Factor__Contributing_2__1   Factor__Contributing_2__2
Factor__Contributing_2__3    Factor__MayContrib_1__1        Factor__MayContrib_1__2    Factor__MayContrib_2__1     Factor__MayContrib_2__2
Factor__Mayvehdefect_1__1    Factor__Mayvehdefect_1__2      Factor__Mayvehdefect_2__1  Factor__Mayvehdefect_2__2   Factor__Vehdefect_1__1
Factor__Vehdefect_1__2       Factor__Vehdefect_1__3         Factor__Vehdefect_2__1     Factor__Vehdefect_2__2      Factor__Vehdefect_2__3
Intersection__Roadway_Part   Intersection__Roadway_System   POI_1__VDP1                POI_1__VDP2                 POI_2__VDP1
POI_2__VDP2                  Traffic_Control
```

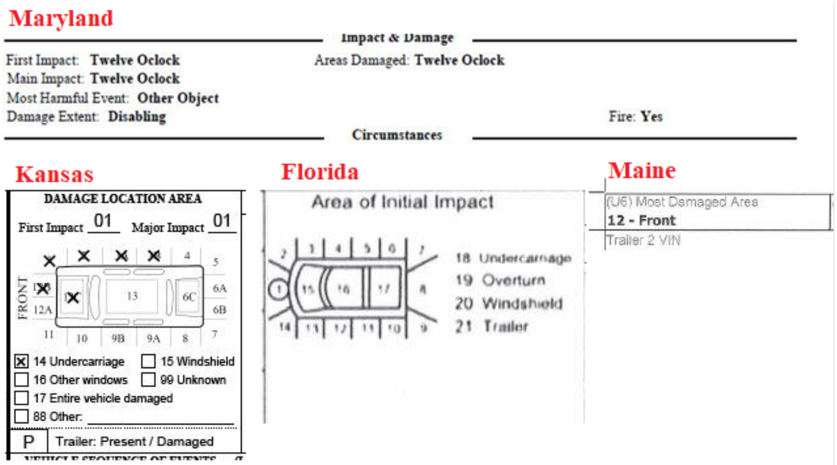
Figure 12. Screen capture. State code .csv files for the Texas CR-3 PAR template.

Table 6. "Traffic_Control" annotation keys and values.

Key	Value
2	Inoperative (Explain in Narrative)
3	Officer
4	Flagman
5	Signal Light
6	Flashing Red Light
7	Flashing Yellow Light
8	Stop Sign
9	Yield Sign
10	Warning Sign
11	Center Stripe/Divider
12	No Passing Zone
13	RR Gate/Signal
15	Crosswalk
16	Bike Lane
17	Marked Lanes
18	Signal Light With Red Light Running Camera
96	None
98	Other (Explain in Narrative)

2.3.4 Standardized State Code Database

While the State codes provide meaning to the alphanumeric labels used throughout the PARs, the codes must be interpreted further to match those values utilized in the crash type database. This process allows for ease in the maintenance of the databases, as any changes to the inherent codes can be made to the standardized State code database rather than to the automation process. Figure 13 provides an example of four States utilizing different codes to represent point of impact. The four OCR outputs for a crash that would occur on the front of a vehicle are "Twelve Oclock," "12A, 12B," "1," and "12 – Front." These labels all have the same meaning but must be standardized before being used in the DSS.



Maryland

Impact & Damage

First Impact: Twelve Oclock
Main Impact: Twelve Oclock
Most Harmful Event: Other Object
Damage Extent: Disabling

Areas Damaged: Twelve Oclock

Fire: Yes

Circumstances

Kansas

DAMAGE LOCATION AREA

First Impact 01 Major Impact 01

☒ 14 Undercarriage ☐ 15 Windshield
☐ 16 Other windows ☐ 99 Unknown
☐ 17 Entire vehicle damaged
☐ 88 Other:

P Trailer: Present / Damaged

Florida

Area of Initial Impact

18 Undercarriage
19 Overturn
20 Windshield
21 Trailer

Maine

(U6) Most Damaged Area
12 - Front
Trailer 2 VIN

Figure 13. Screen capture. Four States' point-of-impact diagram examples and associated codes.

2.3.5 Crash Type Database

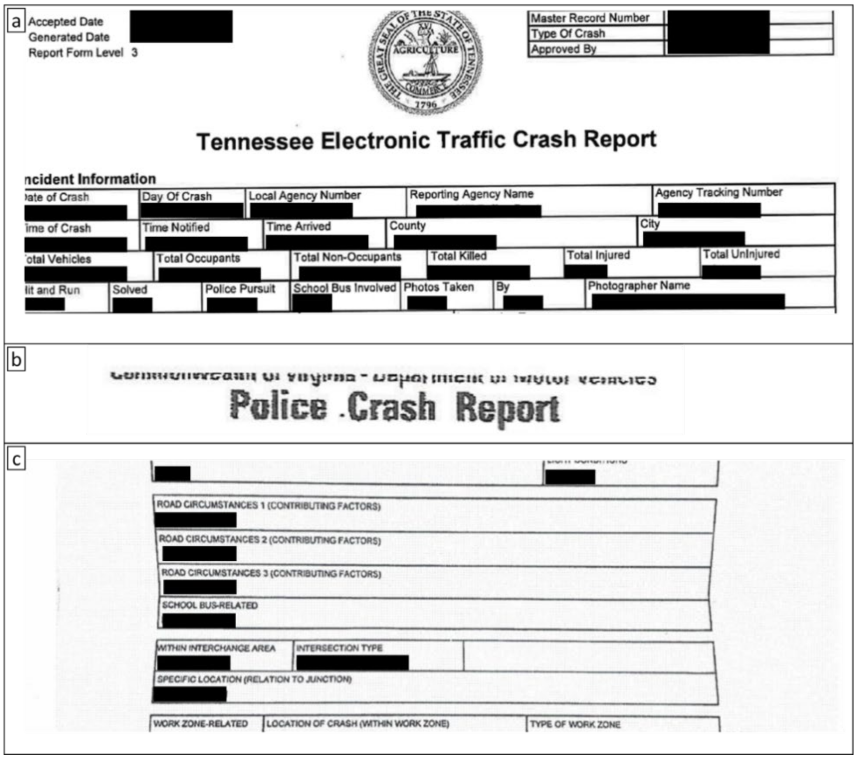
The crash type database incorporates all submittable crash types and contains the values to match to the standardized State codes to use in the crash eligibility and preventability analysis. This database contains criteria as defined by the Model Minimum Uniform Crash Criteria guidelines, with additional criteria resulting from States exceeding the minimum criteria or conveying additional information relevant to analyses.

Multiple codes may be recognized within one crash type data entry. For example, in a scenario where the requestor's driver was struck by another motorist who was driving while distracted, contributing factors may be listed that relate to distraction. But there may be additional factors related to other driver's behaviors, such as speeding.

The crash type database is set up as a relational database to allow for proper organization of entries as structured data. The database used in analysis is matched with the RDR ID and submitter's crash type as recorded from the DataQs website, obtained from the web parsing process. Other keys are identified using standardized codes.

2.3.6 Challenges and Future Steps

The orientation and quality of each PAR dictates how accurate the OCR reader output is for extracted information. Though the JSON output is standardized, there are elements that do not get recorded appropriately. Scanning quality issues can include as stray markings, faded printing, or text cut off from improperly orienting the pages while scanning (see Figure 14). This could also include radial distortion, a particularly challenging problem for aligning documents and detecting fields. While these specific issues may present no impact on the effectiveness or efficiency of a manual review, the OCR reader may produce erroneous results depending on the severity of the issues.



a

Accepted Date [redacted]
Generated Date
Report Form Level 3

Master Record Number	[redacted]
Type Of Crash	
Approved By	

Tennessee Electronic Traffic Crash Report

ncident Information

ate of Crash [redacted]	Day Of Crash [redacted]	Local Agency Number [redacted]	Reporting Agency Name [redacted]	Agency Tracking Number [redacted]		
ime of Crash [redacted]	Time Notified [redacted]	Time Arrived [redacted]	County [redacted]	City [redacted]		
otal Vehicles [redacted]	Total Occupants [redacted]	Total Non-Occupants [redacted]	Total Killed [redacted]	Total Injured [redacted]	Total UnInjured [redacted]	
lit and Run [redacted]	Solved [redacted]	Police Pursuit [redacted]	School Bus Involved [redacted]	Photos Taken [redacted]	By [redacted]	Photographer Name [redacted]

b

Commonwealth of Virginia - Department of Motor Vehicles
Police .Crash Report

c

[redacted] [redacted]

ROAD CIRCUMSTANCES 1 (CONTRIBUTING FACTORS)
[redacted]

ROAD CIRCUMSTANCES 2 (CONTRIBUTING FACTORS)
[redacted]

ROAD CIRCUMSTANCES 3 (CONTRIBUTING FACTORS)
[redacted]

SCHOOL BUS-RELATED
[redacted]

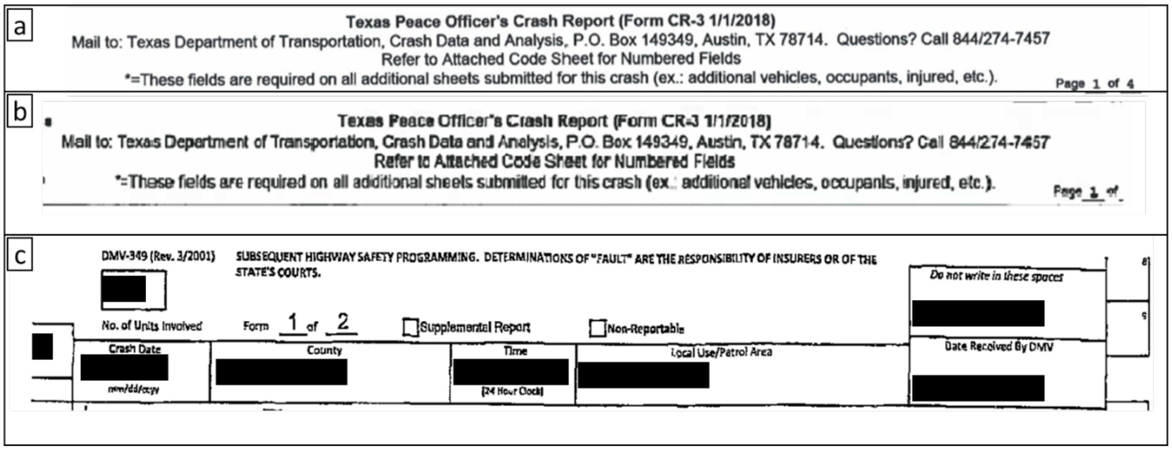
WITHIN INTERCHANGE AREA [redacted]	INTERSECTION TYPE [redacted]	

SPECIFIC LOCATION (RELATION TO JUNCTION)
[redacted]

WORK ZONE-RELATED	LOCATION OF CRASH (WITHIN WORK ZONE)	TYPE OF WORK ZONE

Figure 14. Screen capture. Excerpts from PARs demonstrating scanning quality issues related to paper orientation.

Another significant issue arises from the quality of the scanned image and if any information is lost or muddled through the scanning and uploading process. Figure 15 demonstrates a high-quality scan in item "a" and a lower quality scan of the same text in item "b." Printing and scanning in low quality will result in fewer dots per inch, which may cause quality issues during OCR of text. Severely affected PARs may have additional errors while identifying the version during the structured data matching.



a

Texas Peace Officer's Crash Report (Form CR-3 1/1/2018)
Mail to: Texas Department of Transportation, Crash Data and Analysis, P.O. Box 149349, Austin, TX 78714. Questions? Call 844/274-7457
Refer to Attached Code Sheet for Numbered Fields
*=These fields are required on all additional sheets submitted for this crash (ex.: additional vehicles, occupants, injured, etc.).
Page 1 of 4

b

Texas Peace Officer's Crash Report (Form CR-3 1/1/2018)
Mail to: Texas Department of Transportation, Crash Data and Analysis, P.O. Box 149349, Austin, TX 78714. Questions? Call 844/274-7457
Refer to Attached Code Sheet for Numbered Fields
*=These fields are required on all additional sheets submitted for this crash (ex.: additional vehicles, occupants, injured, etc.).
Page 1 of

c

DMV-349 (Rev. 3/2001) SUBSEQUENT HIGHWAY SAFETY PROGRAMMING. DETERMINATIONS OF "FAULT" ARE THE RESPONSIBILITY OF INSURERS OR OF THE STATE'S COURTS.

Do not write in these spaces
[redacted]

[redacted]
No. of Units Involved Form 1 of 2 □Supplemental Report □Non-Reportable

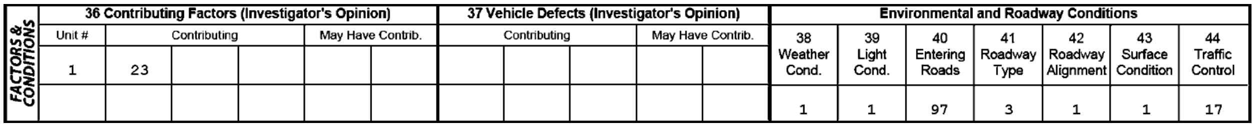
[redacted]

Crash Date [redacted] mm/dd/ccyy	County [redacted]	Time [redacted] (24 Hour Clock)	Local Use/Patrol Area [redacted]	Date Received By DMV [redacted]

Figure 15. Screen capture. Excerpts from PARs displaying degradation of text quality present across PARs.

Although OCR is advancing rapidly and improvements are included in the applied Google Document AI, it is not free from limitations. In this context, OCR works well for plain text standardized in size compared to text with various orientations, sizes, and line segmentation. Also, the background color, orientation, and skew of a document can alter the output quality of the OCR engine. Due to variabilities in the uploaded PARs, additional implementation options for eligibility and preventability analysis may improve accuracy.

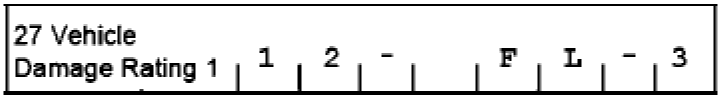
Future efforts could add specific cropped images from the PAR that are most relevant for the determination of eligibility and preventability. Using cropped images, a few strategies are available. First, introducing a humans-in-the-loop model can help make the system more robust to unforeseen variabilities. For example, in the Texas PAR, the "Factors & Conditions" section (Figure 16) holds much of the relevant information to perform the analysis. This section can be extracted and included in a summary report to ensure that all relevant fields are readily available to analysts for their decision-making process. Further, allowing the analyst to manually enter or correct information in either the PDF or a web application would provide a human checkpoint for critical information. In the case of corrected information, the analyst would either enter values as they appear on the PAR or choose from a dropdown. Entered State codes would be cross-referenced with the standardized State code database to populate the field, providing analysts with the matching codes to avoid spending effort to search for codes.



FACTORS & CONDITIONS	36 Contributing Factors (Investigator's Opinion)						37 Vehicle Defects (Investigator's Opinion)					Environmental and Roadway Conditions						
	Unit #	Contributing			May Have Contrib.		Contributing			May Have Contrib.		38 Weather Cond.	39 Light Cond.	40 Entering Roads	41 Roadway Type	42 Roadway Alignment	43 Surface Condition	44 Traffic Control
	1	23																
												1	1	97	3	1	1	17

Figure 16. Screen capture. Factors & Conditions excerpt from a Texas PAR form.

A second option available for improving the accuracy of information would be to extract the targeted cropped images, then execute additional OCR engines to extract the targeted information. This step would be critical for components in which the Google Document AI engine is systematically unable be able to identify critical information. The most expected application of this would be to the point-of-impact diagrams (see the Florida diagram in Figure 13), though other extractions may be identified during iterations of the parsing process to improve accuracy. Examples identified from the Texas PAR include the contributing factors (shown in Figure 16), as well as the point-of-impact force in Figure 17, which has spaced-out numbers separated by a hash mark. An example of the OCR engine utilized is detailed in the Crash Diagram Analysis Process section below.



27 Vehicle Damage Rating 1	1	2	-		F	L	-	3

Figure 17. Screen capture. Point-of-impact extraction from sampled Texas PAR form.

One other identified limitation is from Google's Document AI OCR processor. There is a five-page limit of processed documents for those documents sent directly to Google's OCR processor. This limit is increased to 100 pages if the PDF is stored on Google's storage before executing the OCR processor. Due to the nature of the personally identifiable information (PII) enclosed in the PAR, the research team elected to not store PARs on Google storage. Two solutions exist:

(1) store the PARs on Google's storage with the permission of all appropriate parties accounting for the security of the PII included, or (2) modify the script to perform OCR processing to run through all PDFs after splitting the PARs into five-page increments, stitching them together on the backend. Depending on the version, this five-page limit may have significant impact on the recorded information.

2.4 DOCUMENT PARSING PROCESS

To ingest and parse the information from the PAR, a meaning (semantic interpretation) must be assigned to each relevant data element extracted from the PAR. In this case, a <field: value> pair must be assigned for each field in the document. A key-value format was chosen to aid in extensibility and interpretability. Based on the structured data matching and vectorization, fields and values are stored in temporary database tables and used for inferencing and analyses. After the document is ingested, it is compared to the appropriate State code or crash type database. The basic workflow for ingestion and the DSS is outlined in Figure 18, including the prerequisite databases as well as those created from the ingestion process.

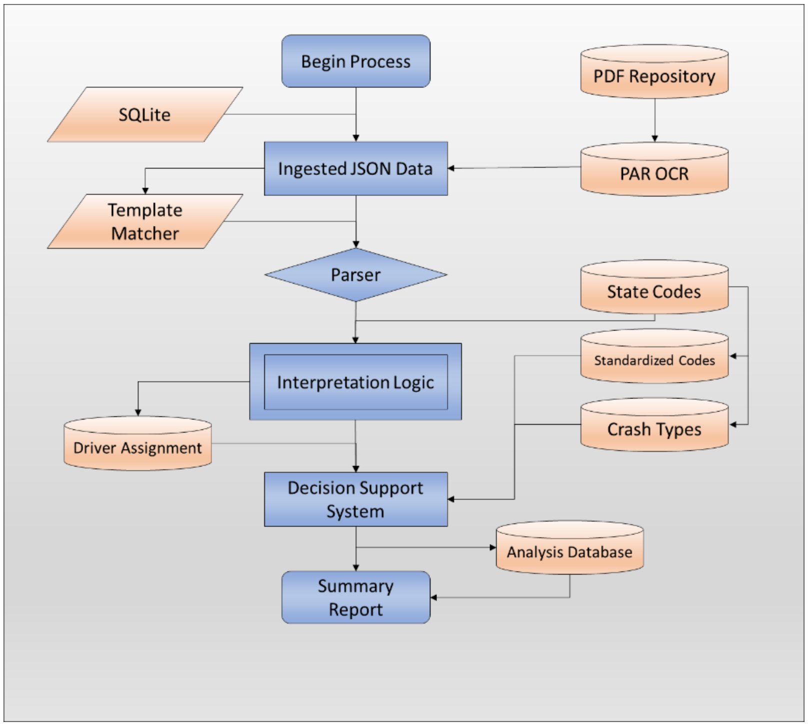


Figure 18. Diagram. Workflow for ingestion and parsing automation process.

The automation algorithm performs a series of steps using the OCR parsed data and prerequisite databases. This process is represented by the "Interpretation Logic" step as defined in the workflow diagram.

2.4.1 PAR Pre-processing and Standardization

The document data obtained from the input source often contain extra cover pages or incorrectly ordered pages. It is thus important to disambiguate which page of the template is being processed before any further action is performed. A featurization and protype-based approach is utilized to build an ML model for identification of pages. Utilizing an empty form, OCR is performed to obtain the word tokens. Then the information is converted to a vector space model, treating each page as a document and tokens as terms. Term frequency-inverse document frequency (TF-IDF) and singular value decomposition (SVD) are applied on the obtained features to perform dimensionality reduction. Using a 1-nearest neighbor prototypical proximity-based approach, the page identity is assigned to the pages of the incoming document, which is in JSON object format. Annotations from CVAT in XML format are utilized to obtain the X and Y coordinate values of bounding boxes of the reference template image.

Using this template matcher, the first page in the document is identified corresponding to the first page in the template. This allows users to utilize known landmark (anchor) tokens in the input document page that correspond to known annotated landmark tokens (for Texas, that word is "Towed"). These are empirically chosen tokens, based on their uniqueness in the document. Using superposition of the input document coordinates over the annotated template coordinates, the required translation of the origin along X and Y coordinates of the input coordinate system is compared to the reference. It also allows users to calculate the scaling required to align their sizes. The page image size is used for the reference and input images to normalize the X and Y values to aid in computations. Some heuristic simplifications with limited assumptions of orthogonality of the axes are performed along with scaling and translations applicable to the entire document.

This process is demonstrated in Figure 19, in which the two red boxes represent the anchor words. Coordinates of bounding boxes on the PAR are compared with those on the template, and the PAR is rotated or scaled appropriately. This standardization has limitations and is not able to handle excessive rotations that affect the initial OCR of the PAR (e.g., uploading an upside-down PAR).

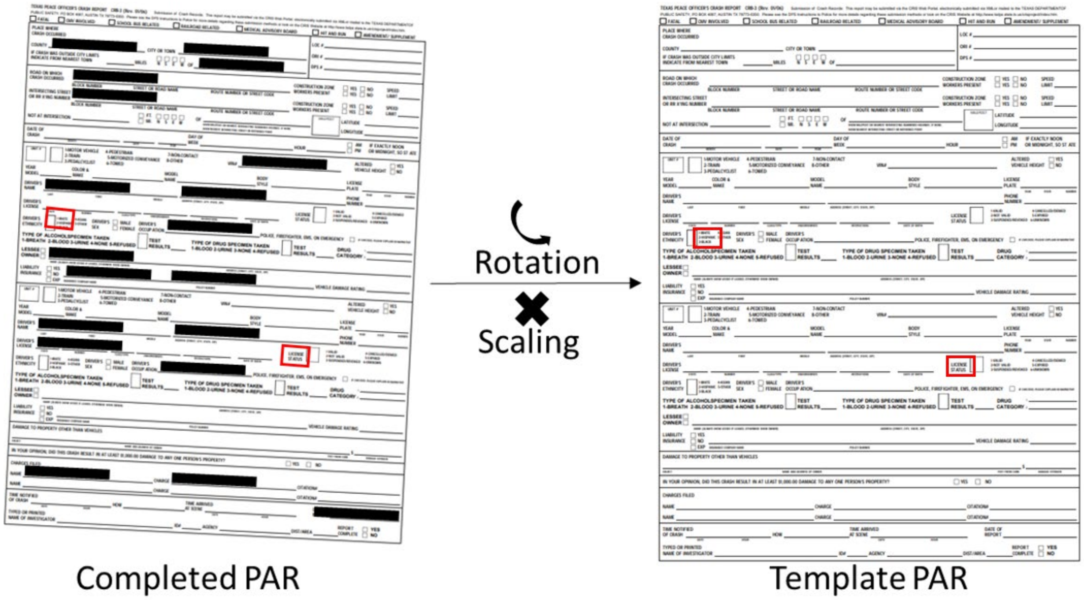


Figure 19. Screen capture. Standardization using anchors to rotate or scale.

2.4.2 Record Entry

After standardizing the PAR, the overlap is recorded and input into the DSS as dictated by a two-stage technique to evaluate if the boundary boxes are aligned with the appropriate, expected responses. If the coordinates provide an expected answer using a regular expression-based match, that entry is recorded. If not, a nearest neighbor token match is found from the centroid of the annotated box and compared to the five nearest centroids of bounding boxes on the PAR. If no expected answers are found within a reasonable set threshold, no entry is recorded. Customized logic is incorporated to handle special elements such as single and multiple checkboxes, along with options to detect checkbox values along different orientations.

Certain elements may assist in the process. If there are multiple candidates, the first or best match is recorded, based on location or match to expected fields. It may also be important to avoid using the nearest neighbor for fields that may be text entry but often empty (e.g., in Texas form CR-3, the “Charges” field may be filled in by the officer using non-standardized responses but is most often left empty). Further, the process may utilize tokens that are part of the template (i.e., known values) to act as identifiers for desired entries to record. Table 7 displays the annotation label “Key” and the associated text from the template.

Table 7. Known value tokens within bounding boxes in Texas Form CR-3.

Key	Associated Text
Fatal_Crash	Fatal
CMV_Crash	CMV
Total_Units	Total Num. Units
Crash_ID	TxDOT Crash ID
Form_Version	(Form CR-3 1/1/2018)
Crash__Roadway_System	1 Rdwy. Sys.
Crash__Roadway_Part	2 Rdwy. Part
IntersectionYN	At Int. Yes No
Intersection__Roadway_System	1 Rdwy. Sys.
Intersection__Roadway_Part	2 Rdwy. Part
Unit_Num__VDP1	Unit Num.
Parked_Vehicle__VDP1	Parked Vehicle
Alcohol__VDP1	22 Alc. Spec.
Drug__VDP1	23 Drug Spec.
POI_1__VDP1	27 Vehicle Damage Rating 1
POI_2__VDP1	27 Vehicle Damage Rating 2
Unit_Num__VDP2	Unit Num.
Parked_Vehicle__VDP2	Parked Vehicle
Alcohol__VDP2	22 Alc. Spec.
Drug__VDP2	23 Drug Spec.
POI_1__VDP2	27 Vehicle Damage Rating 1
POI_2__VDP2	27 Vehicle Damage Rating 2
CMV__Unit_Num	Unit Num.

Key	Associated Text
CMV__Unit_Num_Trailer	Unit Num.
CMV__Sequence1	35 Seq. 1
CMV__Sequence2	35 Seq. 2
CMV__Sequence3	35 Seq. 3
CMV__Sequence4	35 Seq. 4
Traffic_Control	44 Traffic Control
Narrative	Investigator's Narrative Opinion of What Happened (Attach Additional Sheets if Necessary)
Crash_Diagram	Indicate North Field Diagram - Not to Scale

2.4.3 Inputs, Outputs, and Outcomes

The process for executing the document parsing process as described above is performed using the Python files described in Table 8.

Table 8. Executable Python files for the document parsing process.

Python Files	Purpose
ingest_TX_statecodes.py	Acquire and index the Texas State codes
ingestion_v1.py	Acquire and index the crash type database
template_matcher.py	Assess the submitted PAR version
PAR_parser_v1.py	Perform parsing using the matched template and OCR-output JSON file
executor.py	Executes the document parser with optional fields

The direct output of the PAR_parser_v1 program produces the identified data records as they are recorded and presented on the processed PAR. Table 9 presents the results from the parsed and matched PAR. Certain elements were not accurately recorded, and further training or coding is necessary. For example, the Form_Version label produced “, -3 CR Austin,” which is the direct output of the JSON object format. If necessary, alterations may be made to correct the issue or to train using multiple PARs.

Table 9. Example of a parsed Texas PAR using annotation labels.

Label	Value	Page
CMV__Sequence2	0	1
Crash__Roadway_Part	1	0
Unit_Num__VDP1	1	0
Factor__Unit_Num__1	1	1
Factor__Unit_Num__2	1	1
Unit_Num__VDP2	2	0
POI_1__VDP2	12	0
POI_1__VDP1	2	0
CMV__Unit_Num	2	1
CMV__Unit_Num_Trailer	3	1
POI_2__VDP1	11	0

Label	Value	Page
Traffic_Control	11	1
CMV__Sequence1	13	1
Drug__VDP1	24	0
CMV__Sequence3	37	1
Factor__Contributing_1__1	60	1
Factor__Contributing_1__2	60	1
Drug__VDP2	96	0
Alcohol__VDP1	96	0
Form_Version	, - 3 CR Austin	0
Crash_ID	/ 18058392.1 2021011541	0
Crash__Roadway_System	IH	0
Narrative	[Narrative]	1
Intersection_YN	No	0
Intersection__Roadway_System	OR	0
Parked_Vehicle__VDP2	FALSE	0
Fatal_Crash	FALSE	0
CMV_Crash	TRUE	0
Parked_Vehicle__VDP1	TRUE	0

2.4.4 Customization Elements

Several customizations are available in the PAR retrieval process. First, a crucial component of the analysis is to match the appropriate vehicle or unit number with the appropriate target of the RDR submission. Utilizing information obtained from the DataQs website, the appropriate vehicle and person must be correctly designated. This designation influences the structure of any future created tables and is also a prerequisite for eligibility and preventability analysis.

Second, the point-of-impact clock diagrams are represented in numerous ways across PAR versions. While many diagrams use alphanumeric representations, certain versions utilize some other visualization, such as bubbling or location marking. Depending on the complexity of the marking type, the correct document coordinates must be segmented and recorded, and other extraction methods may be required. Further, there are various levels of specificity across versions, including fewer or more than the 12 expected clock locations. Additionally, some PARs may have specific point-of-impact diagrams for CMVs, which are utilized separately.

Third, the OCR text extractor excels at identifying checkboxes and recording these in the JSON object format. However, for PAR fields with multiple checkboxes, further deconstruction must be data mined to determine the appropriate true/false values. Utilizing (x, y) coordinates of the individual markings provides a quick way to estimate any horizontal or vertical differences in coordinates. Additionally, care must be used to ensure that stray document artifacts do not create false positives for checkboxes.

2.4.5 Challenges and Future Steps

The output of OCR is dependent on the scanned image. This can potentially impact downstream processing quality and dramatically impact OCR readability. The most common issue is poor image quality, which impacts the accuracy of the ingested data. Specific issues relate to non-standard text. For example, checkboxes are not reliably picked up by the OCR processor and, in some cases, there are multiple variations of checkboxes, such as single or multiple tick marks, along with different horizontal or vertical alignments.

Another issue is that OCR has nonuniform output across the different inputs, which often cannot be attributed to a specific root cause. Word or token orders are not preserved. Additionally, multiple non-alphabet symbols are sometimes not read correctly. Further, the delimiters in the inherent format split tokens into subparts which do not conform to majority patterns and require specific extraction logic. Reconstruction of phrases or co-occurring content is sometimes difficult to achieve. Further, OCR does not read in co-located tokens together.

Across many of these challenges, customized routines are utilized. These routines may include performing a nearest-neighbor search for tokens in instances where there is no initial value recorded by the OCR parser. Any identified values may then be evaluated for appropriateness based on the stored database values of the specific label identified in the template annotations. Continual improvements can be made based on identified challenges and the appropriate solutions (e.g., cropped image OCR for illegible items).

2.5 NARRATIVE ANALYSIS PROCESS

Landmarking is used based on each template to determine the location of the narrative. The narrative text is extracted during the automation process. Feature-rich and open-source libraries, such as *StanfordNLP*, are utilized and contain some degree of customization. Other open-source, stable, and standard libraries such as *NLTK*, *Spacy*, and *TextAcy* are used to process text. Feature embedding and extraction is done using *gensim*, *gloVe*, or *fastText*. Certain PARs will have additional narrative information recorded in areas not identified on the template, such as an extra or duplicated page allowing for additional information to be recorded.

An initial NLP process was created to analyze a set of narratives to extract useful keywords, phrases, or semantic values within the narrative to identify matching data elements with crash eligibility and preventability. Additional semantic values can be assigned to common language found within future narrative iteration. This semantic coding may provide more refined contextual information related to the crash, and subsequent characteristics may provide additional insight into the narrative analysis. There is a scope to explore the interoperability of the ML models to perform further analysis in terms of the model usability.

A series of tokens and phrases were created that consist of the common contributing factors and event natures that relate to eligibility criteria. Keywords consist of elements directly related to the eligible crash types (e.g., debris, animal, intoxicated) or factors related to crash parameters (e.g., failed to control speed, failed to stay in lane). Further, the narrative field is parsed from the PDF and is provided on the summary report for quick review by DataQs analysts.

2.5.1 Challenges and Future Steps

Current efforts have concentrated on extracting information from the tabular data and have shown that they are independently capable of answering questions about preventability and eligibility, but extended efforts involving parsing the narrative can be equally useful for a similar goal. Manual review of PAR narratives has shown that most narratives include a relatively standardized description of the crash along with key words related to the State code. For example, the narrative includes terms utilized in State codes such as "failed to control speed," "took a wide right turn," and "fatigued and fell asleep." The narrative also includes information on the contact types, typically referencing the initial point of impact. Though most of this information is available in the PAR as tabular entry, the narrative can be used to either conform with the tabular entry or add more information that is not available in the table, either due to OCR error or manual error from the law enforcement agent. Narrative parsing will be useful in bolstering the decision of the automated process.

Future efforts to integrate the narrative into the automation process would be to assign key words or behavioral attributes to each unit or vehicle involved in the crash. For example, consider the following narrative: "Unit 1 was traveling NW in the XXX road. Unit 2 was legally parked and unoccupied on the side of the roadway. Unit 1 failed to control speed striking the Unit 2's BL with Unit 1's FR." In this narrative:

- Unit 1 corresponds to three descriptions/attributes: traveling NW, failed to control speed, striking Unit 2.
- Unit 2 corresponds to two descriptions/attributes: legally parked, unoccupied.
- The interaction is described as "Unit 2's BL with Unit 1's FR."

These pieces of information may be used to aid to the decision of preventability or eligibility by evaluating the attributes, or sentiment for each of the units. Ideally, the NLP algorithm may help assign these attributes to each of the units and facilitate the determinations. Similarly, the interaction between the vehicles or their initial point of contact may be identified from the narratives as well. A specified tool for NLP called "dependency parsing" may be useful in developing such relationship between units and their attributes. The dependency parsing divides a sentence by its parts of speech and creates a relational hierarchical structure. Figure 20 shows an example of such parsing showing the relations between each word; in this case, it shows that Unit 1 failed, and the nature of failure is to control the speed. The structure helps to identify how one word is related to another.

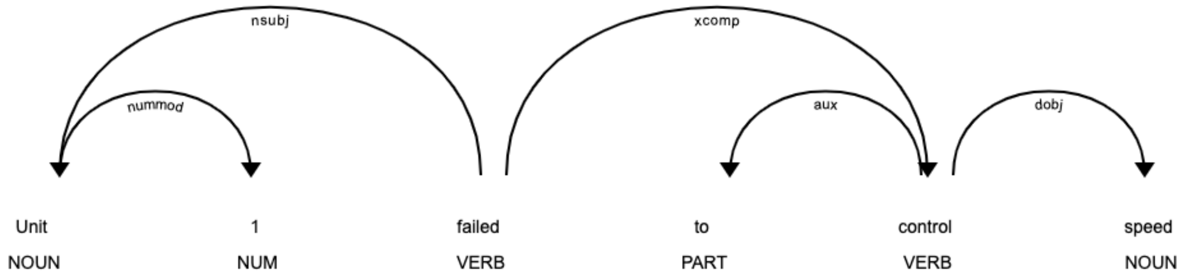


Figure 20. Diagram. Example of dependency parsing showing relation between each word and its semantic meaning.

Specific considerations for the implementation of a narrative process include:

- PAR narratives are generally homogeneous in terms of the use of keywords for each State, though the keywords may vary slightly between States. These variations may result in altered keywords based on version templates. One example is the term used to represent the vehicle, as it is typically portrayed as “unit,” “vehicle,” “U,” or “V.”
- Major variations exist across certain elements, specifically the narrative style, formatting, and grammatical use of terms. Special concentration may be needed to overcome such variations or grammatical errors.
- The keywords and language in general are unique and domain specific to crash events and driving scenarios. Existing NLP algorithms are often not trained to prioritize on such keywords and are designed for more general usage. Therefore, to build a successful narrative parsing system, a targeted vocabulary for the crash narrative must be created. This may include words that may often appear together; such as “control” and “speed,” “signal” and “intersection,” or “right/left” and “turn.” Multiple narratives must be evaluated to create these models.

2.6 CRASH DIAGRAM ANALYSIS PROCESS

Analyzing crash diagrams represents a complex CV task that is exacerbated by the rather free-form and highly variable nature of diagrams anticipated (i.e., substantial variability in diagrams can be seen even within a single crash report version from the same State). For example, a relatively straightforward sample Texas crash diagram of a CMV being struck in the rear is shown in Figure 21.

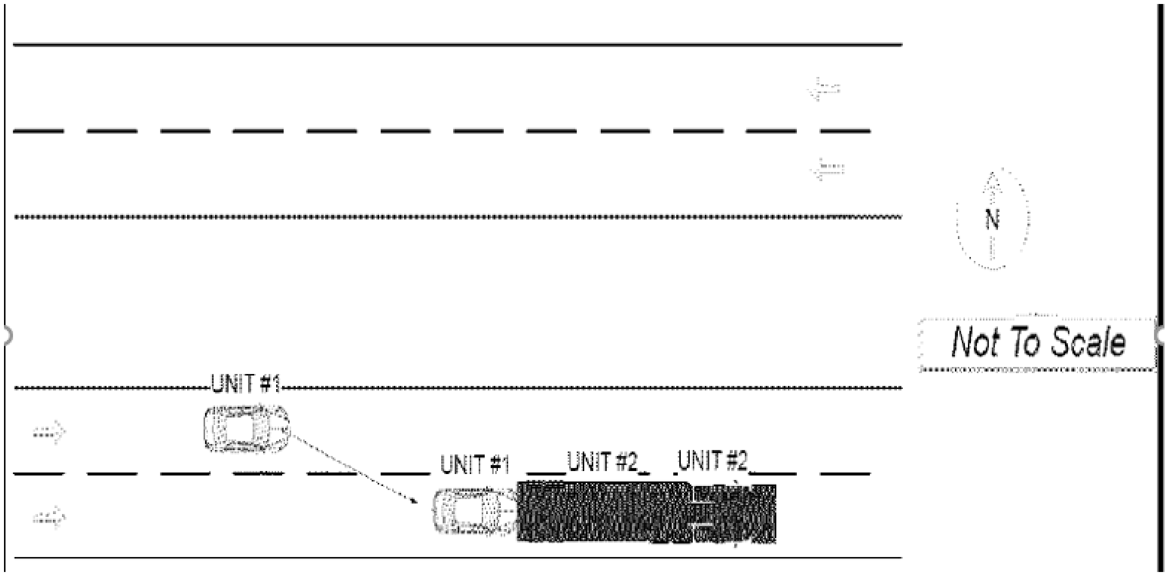


Figure 21. Screen capture. Texas crash diagram showing Unit #1 striking Unit #2 in the rear.

2.6.1 Technical Overview

To approach this problem, several entities (vehicles, highways, etc.) must be detected using object detection techniques (returning bounding boxes) using CV or ML/deep learning (DL) methods, and then the relationship between them can be established. In Figure 21, several elements can be distinguished:

1. **Compass**. A basic compass showing that the top of the page points to "North" is provided.
2. **Vehicles**. Two vehicles labeled "Unit 1" and "Unit 2." It is not known *a priori* that the CMV is Unit #2, although it can be inferred from its relative size or imported from the PAR ingestion and field matching.
3. **Vehicle Movement Over Time**. In this image, "Unit 1" is shown twice, representing movement of the vehicle. The system must be able to detect the same vehicle uniquely and use any provided arrows to determine its position over time. It cannot always be assumed that the direction of traffic and the direction of the vehicles are the same. Another more complex example showing the same vehicle ("Unit 1") nine different times is shown in Figure 22.

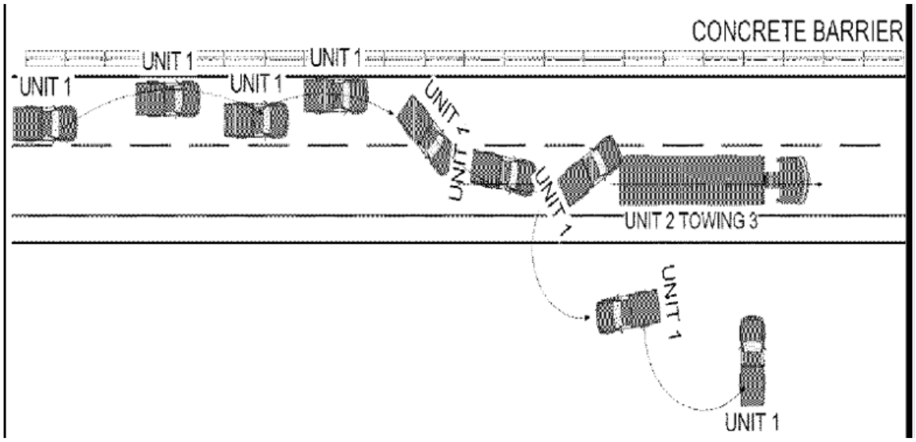


Figure 22. Screen capture. Texas crash diagram showing Unit #1 striking Unit #2 in the rear with complex travel.

4. **Highway**. A divided highway with two traffic lanes traveling in opposite directions and separated by a divider are shown. Only one is relevant in this example. Each direction of traffic has two lanes separated by a dashed line.
5. **Direction of Traffic**. The direction of traffic is shown with arrows. For the accident, both vehicles were traveling eastward. Making this inference requires correctly processing the arrows with respect to the compass.

After these components are properly detected using CV or ML/DL, the AI system can decide on a particular sub-question using that diagram. This decision can be made using defined rules or calculations or by predictive inference using ML/DL methods. The answers to these sub-questions act as data sources for the corresponding features in the eligibility and preventability AI models. Table 10 provides a non-exhaustive list of several of these sub-questions for rear-end crashes.

Table 10. Crash eligibility sub-questions and approaches for answering.

Crash Eligibility Sub-questions	Desired Answers	Approach
Determine the final location of the CMV at the time of the crash	A single bounding box	Rules or ML/DL inference
Determine the final location of the non-CMV at the time of the crash	A single bounding box	Rules or ML/DL inference
Determine the direction of the flow of traffic?	A number (0–360 degrees)	
Determine if both vehicles were traveling in the same direction and/or lane	Yes or No	
Determine if the non-CMV was traveling *behind* the CMV prior to the crash	Yes or No	
Based on the crash diagram, determine the angle of impact on the CMV	A number (0–360 degrees)	Geometric calculation based on flow of traffic

Crash Eligibility Sub-questions	Desired Answers	Approach
Determine the number of vehicles that were involved in the accident	Integer ≥ 1	OCR with fuzzy string matching

Models and systems that provide answers to these questions are dependent on the quality of the diagrams, the resolution of scanning, stray marks present, and inconsistent presence of required elements within an image. In addition, the free-form nature of the diagrams, combined with the limited number of available crash diagrams when accounting for the cross between PAR versions and crash types, makes the potential accuracy of these models difficult to evaluate. Some of these sub-questions are nested and all will feed into future predictions for preventable crashes.

2.6.2 Entity Detection using Object Detection Methodologies

Due to this small sample size, as with other earlier models, the benefits of transfer learning result in supervised DL-based algorithms for object detection or image segmentation being the best candidates. The models are composed of a core model that is trained for classification of images into specific classes (e.g., dog versus car versus table) using large open-source datasets. This model creates generalized features that are then used as inputs to models for object detection or image segmentation.

2.6.3 Use Cases and Examples

2.6.3.1 Use Case #1: What is the final location of the non-CMV at the time of the crash?

Initial testing would be done using OCR text locations to determine if simple distance between string boxes could be used to find locations. If this did not prove to provide acceptable levels of accuracy, a training set of data would be compiled of roughly 100 images, labeling the bounding boxes for each vehicle along with distinct classes (e.g., CMV versus non-CMV). Available training time and computational resources would be factored into the decision on speed versus accuracy in selection of both the core model and the detection algorithm to use. Swapping the channels to distance-based methods versus trimming the first convolutional layer would be tested to evaluate best performance, and finally the overall accuracy of both object classification and boundary box location would be jointly computed using a holdout dataset.

DL models require a training, validation, and testing dataset split so the availability and quality of training data may not be sufficient to train and reasonably evaluate the models even with transfer learning. Variability of images, defects, skews, stray marks, and resolution discrepancies would all be factors that could contribute to modeling errors and mislabeled features.

2.6.3.2 Use Case #2: Was the non-CMV traveling behind the CMV prior to the crash?

Utilizing the results of the sub-questions on vehicle location at impact time, direction of traffic, and travel direction, a rules-based system could be built to determine the answer to the question. First, check if vehicles were traveling the same direction and, if not, the answer is no. Otherwise, using the direction of traffic, normalize the image via transformation so that all CMVs are traveling the same way (e.g., left to right), then determine if the bounding box at impact of the non-CMV was to the left of the CMV's bounding box. If so, then the answer is yes, and otherwise no.

The process logic with four correct pieces of information is straightforward to create, but inaccurate predictions in the models need to be accounted for. If the input models cannot get to acceptable accuracy for rules-based systems, this question becomes significantly more challenging to answer. For example, this model likely cannot perform any better than the system used to determine the flow direction of traffic and the one used to determine whether vehicles were traveling in the same direction.

2.6.4 Inputs, Outputs, and Outcomes

Current utilization of the crash diagram includes outputting a .jpg image of the diagram for use in the generated summary report. Landmarking based on State template determines the location of the crash diagram, which is then extracted as a .jpg and included in the .xlsx to .pdf translation.

2.6.5 Challenges and Future Steps

For the summary report diagram extraction, some crash diagrams may continue to additional pages, have an enlarged or blown-out diagram, have legends, or contain other pieces of information that may be segmented separately from the typical vehicle(s) and roadway information. These cases are not currently handled and occur infrequently enough that training a model or providing template instructions is not feasible.

For the crash diagram analysis, certain features can be generated and may provide incremental improvement to eligibility and preventability model performance. However, the development of these feature generation models is a highly involved process. In addition, as with any model, the error associated with predictions and using these errors to then train models on preventability could propagate that error throughout the AI system. As such, extensive testing and validation procedures would be required for each sub-question to produce models with acceptable accuracy given that their results feed into future predictions.

2.7 ELIGIBILITY AND PREVENTABILITY ANALYSIS

The eligibility and preventability analysis involves linking the inferenced fields to the prerequisite crash type database, which stores information regarding the crash parameters of the eligible crash types. These parameters include all relevant information able to be extracted from a PAR to be matched with common decisions made by DataQs analysts. This relies heavily on the accuracy of the document parsing and subsequent match with the eligibility database.

During the analyses, the crash must match the eligible crash type and the fleet driver must be shown to have had no means of preventing the crash from occurring. This investigation examines all involved drivers to make that argument. Depending on the crash type, the driver behaviors are segmented and matched against the crash type database to confirm that the PAR did not record anything that would determine the crash was preventable by the fleet driver. This includes violations, contributing factors, driver behaviors, and other relevant information to make the conclusion. Table 11 highlights the process for determining crash eligibility and preventability using extracted database elements.

Table 11. Database matching during crash analysis.

Label	Record	State Code	Standardized State Code	Crash Type
CMV__Sequence2	0	N/A	N/A	N/A
Crash__Roadway_Part	1	Main/Proper Lane	Main	N/A
Unit_Num__VDP1	1	1	1	1
Factor__Unit_Num__1	1	1	1	1
Factor__Unit_Num__2	1	1	1	1
Unit_Num__VDP2	2	2	2	2
POI_1__VDP2	12	12	12:00	12:00
POI_1__VDP1	2	2	2	2
CMV__Unit_Num	2	2	2	2
CMV__Unit_Num_Trailer	3	3	3	3
POI_2__VDP1	11	11	11:00	11:00
Traffic_Control	11	Center Stripe/Divider	Divided Lanes	Divided Lanes
CMV__Sequence1	13	Collision Involving Motor Vehicle in Transport	Collision Involving Motor Vehicle in Transport	Collision Involving Motor Vehicle in Transport
Factor__Contributing_1__1	60	Unsafe Speed	Unsafe Speed	Unsafe Speed
Factor__Contributing_1__2	60	Unsafe Speed	Unsafe Speed	Unsafe Speed
Drug__VDP2	96	None	None	None
Alcohol__VDP1	96	None	None	None
Form_Version	, - 3 CR Austin	Form CR-3	N/A	N/A
Crash_ID	/ 18058392.12021011541	N/A	N/A	N/A
Crash__Roadway_System	IH	Interstate	Interstate	Interstate
Narrative	[Narrative]	N/A	N/A	N/A
Intersection_YN	No	No	No	No
Intersection__Roadway_System	OR	CR	County Road	N/A
Parked_Vehicle__VDP2	FALSE	FALSE	FALSE	False
Fatal_Crash	FALSE	FALSE	FALSE	N/A
CMV_Crash	TRUE	TRUE	TRUE	N/A
Parked_Vehicle__VDP1	TRUE	TRUE	TRUE	TRUE

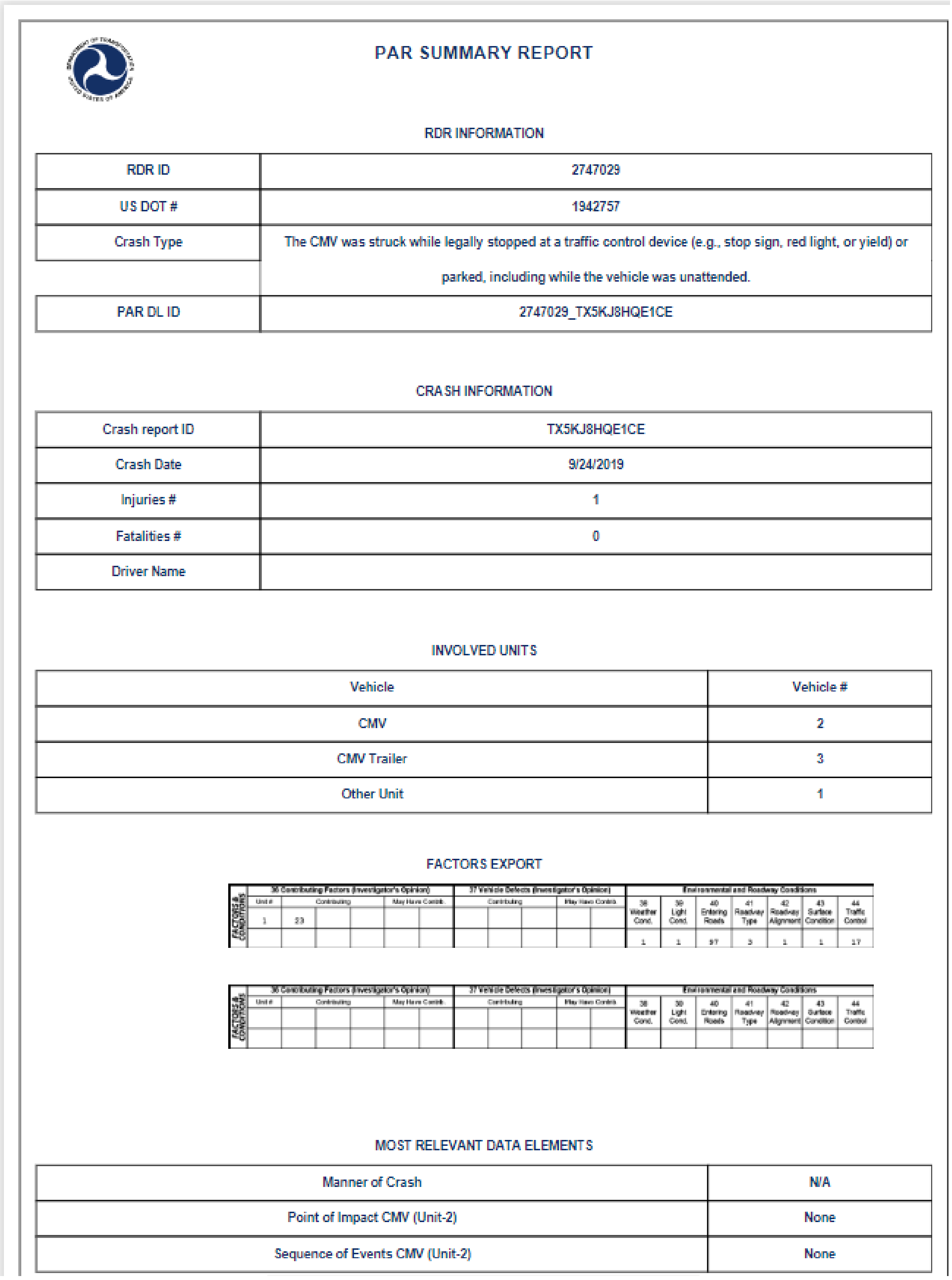
These matches may have dynamic weights based on standardized codes and crash information. For example, the eligible crash type “CMV struck in the rear” may rely heavily on the point-of-impact diagram; however, certain PAR versions may have “rear end” listed as the crash or event type that may supplant the weight of the point-of-impact diagram. This weighting process, which involves traditional ML, is often helpful to understand the pattern of the data and realize the relations of the dependent and independent variables. Finding the preventability or eligibility can be formulated as a classification problem. Then feature engineering can be performed to evaluate dimension reduction to create an ML-based classifier.

2.7.1 Challenges and Future Steps

The current implementation of the eligibility and preventability analysis includes a series of decisions based on a set of flagged elements and data matching. A standardized crash type database created for these analyses provides sufficient information regarding crash or driver parameters that can be compared to the information on the PAR diagram. Eligibility criteria are typically related to crash parameters such as event nature, point of impacts, and vehicle parked status, while preventability criteria are typically related to driver-specific parameters, such as driver contributing factors, alcohol or drug involvement, or unsafe driving behavior. These criteria are flagged by subject matter experts and used by the parser interpreter to identify where mismatches or nonaligned data elements occur. Thus, if the parser does not identify any flagged elements, it makes a positive determination that the RDR matches the eligible crash type or the crash was not preventable by the requestor's CMV. This rules-based system can incorporate most of the information on a PAR if it is integrated into the automation process.

2.8 SUMMARY REPORT GENERATION

The collective summary of information acquired throughout the document parsing, AI algorithm, and analysis is provided in a summary report. Information extracted for the summary report includes PAR retrieval data, extracted matched data elements, eligibility and determination outputs, and any information extracted directly from the PAR itself, such as the narrative and crash diagram. Figure 23 and Figure 24 display the summary report output as recorded in .xlsx and saved as .pdf. The fields are populated by queries across all populated databases. The summary report is produced for each PAR submitted in the automation process.



PAR SUMMARY REPORT

RDR INFORMATION

RDR ID	2747029
US DOT #	1942757
Crash Type	The CMV was struck while legally stopped at a traffic control device (e.g., stop sign, red light, or yield) or parked, including while the vehicle was unattended.
PAR DL ID	2747029_TX5KJ8HQE1CE

CRASH INFORMATION

Crash report ID	TX5KJ8HQE1CE
Crash Date	9/24/2019
Injuries #	1
Fatalities #	0
Driver Name	

INVOLVED UNITS

Vehicle	Vehicle #
CMV	2
CMV Trailer	3
Other Unit	1

FACTORS EXPORT

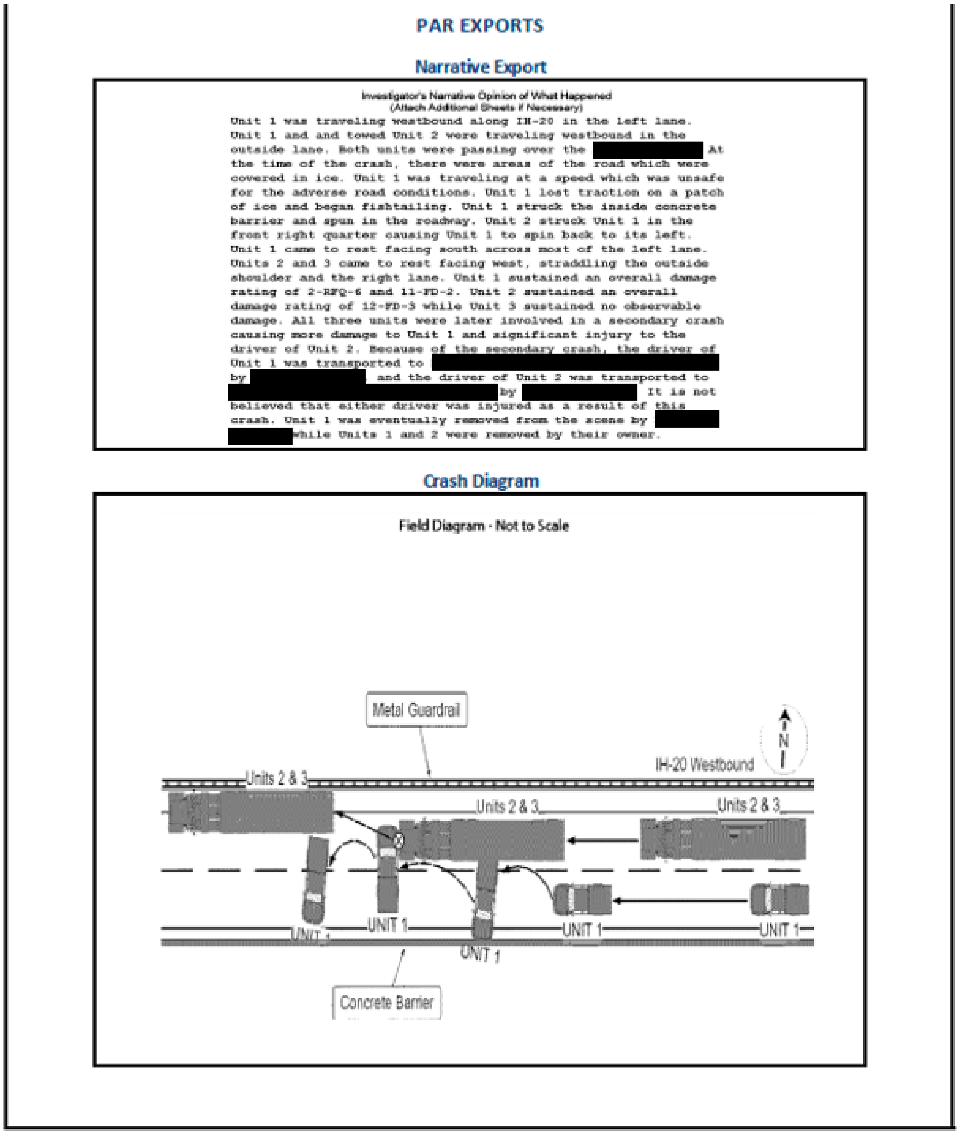
FACTORS & CONDITIONS	36 Contributing Factors (Investigator's Opinion)						37 Vehicle Defects (Investigator's Opinion)					Environmental and Roadway Conditions						
	Unit #	Contributing			May Have Contrib.		Contributing			May Have Contrib.		38 Weather Cond.	39 Light Cond.	40 Entering Roads	41 Roadway Type	42 Roadway Alignment	43 Surface Condition	44 Traffic Control
	1	23																
												1	1	97	3	1	1	17

FACTORS & CONDITIONS	36 Contributing Factors (Investigator's Opinion)						37 Vehicle Defects (Investigator's Opinion)					Environmental and Roadway Conditions						
	Unit #	Contributing			May Have Contrib.		Contributing			May Have Contrib.		38 Weather Cond.	39 Light Cond.	40 Entering Roads	41 Roadway Type	42 Roadway Alignment	43 Surface Condition	44 Traffic Control

MOST RELEVANT DATA ELEMENTS

Manner of Crash	N/A
Point of Impact CMV (Unit-2)	None
Sequence of Events CMV (Unit-2)	None

Figure 23. Screen capture. First page of a generated summary report.



PAR EXPORTS

Narrative Export

Investigator's Narrative Opinion of What Happened
(Attach Additional Sheets if Necessary)

Unit 1 was traveling westbound along IH-20 in the left lane. Unit 1 and and towed Unit 2 were traveling westbound in the outside lane. Both units were passing over the [redacted] At the time of the crash, there were areas of the road which were covered in ice. Unit 1 was traveling at a speed which was unsafe for the adverse road conditions. Unit 1 lost traction on a patch of ice and began fishtailing. Unit 1 struck the inside concrete barrier and spun in the roadway. Unit 2 struck Unit 1 in the front right quarter causing Unit 1 to spin back to its left. Unit 1 came to rest facing south across most of the left lane. Units 2 and 3 came to rest facing west, straddling the outside shoulder and the right lane. Unit 1 sustained an overall damage rating of 2-RFQ-6 and 11-FD-2. Unit 2 sustained an overall damage rating of 12-FD-3 while Unit 3 sustained no observable damage. All three units were later involved in a secondary crash causing more damage to Unit 1 and significant injury to the driver of Unit 2. Because of the secondary crash, the driver of Unit 1 was transported to [redacted] by [redacted], and the driver of Unit 2 was transported to [redacted] by [redacted] It is not believed that either driver was injured as a result of this crash. Unit 1 was eventually removed from the scene by [redacted] while Units 1 and 2 were removed by their owner.

Crash Diagram

Field Diagram - Not to Scale

Figure 24. Screen capture. Second page of a generated summary report.

2.8.1 Challenges and Future Steps

The summary reports include the relevant information that was identified for eligibility and preventability criteria. These criteria are standardized for crash types (e.g., 6:00 CMV impact and 12:00 other vehicle impact for a "CMV struck in the rear-end crash") and include State- or version-specific information that may also contribute to the eligibility or preventability analysis (e.g., Florida PAR contains explicit information about rear-end crashes). However, more detailed information may be available for inclusion depending on the data contained in the specific version of the PAR.

Ultimately, the information displayed in the summary report should be dictated by DataQs analysts to best assist them with their responsibilities in affirming eligibility and determining crash preventability.

3. COST-BENEFIT ANALYSIS

A cost-benefit analysis was performed during the project to highlight the amount of effort it would take to develop similar automation methods for other States' versions of the PAR.

The primary element of the automation algorithm, namely, the parser and logic interpreter, as identified in Figure 2, takes elements from the output JSON and parses, interprets, and matches these data with subsequent analytic components to create a decision and produce a summary report output.

Table 12 outlines the deliverables that would be produced in an extension of the project to other PARs, along with an expected timeframe for their development. These represent the bulk of the implementation of the DSS remaining in the process of executing a comprehensive, whole system process across all versions. Existing products developed by the research team are labeled "N/A" and can be provided to FMCSA.

Table 12. Development estimations in hours by version.

#	State	Estimated Annotation Development (hours)	Estimated Parser Development (hours)
1	TX	N/A	N/A
2	FL	N/A	80.00
3	CA	N/A	66.67
4	IN	N/A	64.33
5	PA	N/A	62.12
6	OH	N/A	60.01
7	GA	N/A	58.01
8	NC	N/A	56.11
9	IL	N/A	54.30
10	TN	N/A	52.59
11	AL	N/A	50.96
12	MI	N/A	49.41
13	MO	N/A	47.94
14	KY	N/A	46.54
15	NY	N/A	45.22
16	LA	3.00	43.96
17	SC	2.75	42.76
18	AR	2.60	41.62
19	VA	2.52	40.54
20	OK	2.47	39.51
21	WI	2.44	38.54
22	AZ	2.42	37.61
23	MS	2.41	36.73
24	IA	2.41	35.89

#	State	Estimated Annotation Development (hours)	Estimated Parser Development (hours)
25	NJ	2.40	35.10
26	MD	2.40	34.34
27	KS	2.40	33.63
28	CO	2.40	32.94
29	WA	2.40	32.30
30	MN	2.40	31.68
31	WV	2.40	31.10
32	WY	2.40	30.54
33	NM	2.40	30.02
34	CT	2.40	29.52
35	NE	2.40	29.04
36	UT	2.40	28.59
37	MA	2.40	28.16
38	OR	2.40	27.75
39	DE	2.40	27.36
40	NV	2.40	26.99
41	ID	2.40	26.65
42	MT	2.40	26.31
43	ME	2.40	26.00
44	ND	2.40	25.70
45	NH	2.40	25.41
46	SD	2.40	25.14
47	DC	2.40	24.88
48	RI	2.40	24.64
49	VT	2.40	24.41
50	AK	2.40	24.19
51	HI	2.40	23.98
Total	-	90.24	1,917.74

Table 13 displays the estimated development hours for each version based on the estimated number of monthly RDR submissions and the development times associated with those versions. This coefficient represents an inverse relationship between the number of net hours saved and the implementation time. For example, developing Florida to 75 percent accuracy (that is, reading the majority of PARs at 75 percent efficiency) would allow for 39 of the estimated 52 monthly PAR submissions to be processed, interpreted, and detailed in a summary report for a DataQs analyst's review. If additional PARs are submitted for Florida beyond the 52, the number of processed PARs will also increase.

Table 13. Implementation by estimated monthly submissions and total development hours.

#	State	Estimated Monthly Submissions	Expected Parsed Documents	Estimated Development Time (hours)
1	TX	88	66	N/A
2	FL	52	39	80
3	CA	48	36	66.67
4	IN	43	32	64.33
5	PA	40	30	62.12
6	OH	38	29	60.01
7	GA	35	26	58.01
8	NC	35	26	56.11
9	IL	34	26	54.3
10	TN	32	24	52.59
11	AL	26	20	50.96
12	MI	26	20	49.41
13	MO	25	19	47.94
14	KY	24	18	46.54
15	NY	18	14	45.22
16	LA	17	13	46.96
17	SC	16	12	45.51
18	AR	16	12	44.22
19	VA	16	12	43.06
20	OK	16	12	41.98
21	WI	15	11	40.98
22	AZ	12	9	40.03
23	MS	12	9	39.14
24	IA	10	8	38.3
25	NJ	9	7	37.5
26	MD	9	7	36.74
27	KS	8	6	36.03
28	CO	8	6	35.34
29	WA	7	5	34.7
30	MN	6	5	34.08
31	WV	6	5	33.5
32	WY	5	4	32.94
33	NM	5	4	32.42
34	CT	5	4	31.92
35	NE	5	4	31.44
36	UT	5	4	30.99
37	MA	4	3	30.56

#	State	Estimated Monthly Submissions	Expected Parsed Documents	Estimated Development Time (hours)
38	OR	4	3	30.15
39	DE	4	3	29.76
40	NV	3	2	29.39
41	ID	3	2	29.05
42	MT	3	2	28.71
43	ME	2	2	28.4
44	ND	1	1	28.1
45	NH	1	1	27.81
46	SD	1	1	27.54
47	DC	0	0	27.28
48	RI	0	0	27.04
49	VT	0	0	26.81
50	AK	0	0	26.59
51	HI	0	0	26.38
Total	-	798	604	2,005.56

In order to attain a 75 percent efficacy rate among 80 percent of the submitted RDRs, customized solutions would have to be performed on the top 15 PAR versions by magnitude (i.e., TX, FL, CA, IN, PA, OH, GA, NC, IL, TN, AL, MI, MO, KY, NY). The efficacy rate constitutes the accuracy in which the model would produce eligibility and preventability determinations along with the summarized information, including the narrative, any crash diagrams, and the selected criteria that led to the creation of those determinations. Those summary reports with inaccurate information (i.e., the remaining 25 percent within the efficacy rate) would need more careful evaluation to make determinations for the RDR. The customized solution would include document reading, parsing, and interpretation as well as the DSSs as performed in the demonstration, but also include additional ML on extracted sections that are difficult to read by the OCR processor. Additional efforts to increase efficacy rate of determinations would include a full analysis on the extracted narratives, creating a token library of terminology for each PAR version while maintaining a library that exists across versions. Table 14 dictates an estimated level of effort and associated costs to federal regulators for various entities to perform the analyses as demonstrated as well as the additional efforts described in the recommendations in the following section. The estimated labor and costs include the utilization of developed material throughout the duration of the project, including the web scraper for RDR information collection and PAR downloader, CV annotations for various States, and a compiled State code database used in the interpreters.

Collectively, the customized solutions across the top 15 of the submitted PARs versions would produce an approximate 50 percent reduction in total time spent by DataQs analysts in their determination efforts. This reduction in time is the result of 80 percent of the submitted RDRs being automatically processed to generate a crash summary report highlighting the eligibility and preventability determinations made by the DSS, the criteria utilized in making those determinations, the extracted narrative, and any crash diagram. Further reductions in time may occur as processes for completing review of submitted RDRs evolve in tandem with the DSS.

Table 14. Development estimations by organization entity for top 15 PAR versions to achieve 75 percent efficacy in eligibility and preventability determinations.

Entity	Interpreter Logic (hours)	CV on Extracted Data (hours)	Narrative Analysis (hours)	Quality Assurance (hours)	Total (hours)	Estimated Cost (fully encumbered)
VTTI	900	840	440	120	2,300	$230,000 (at $100/hr)
USDOT-IT	960	840	440	120	2,360	$188,800 (at $80/hr)
Private Sector	2,200	1,600	1,080	240	5,140	$819,200 (at $160/hr)

There are a few considerations for the associated elements identified in Table 13 and Table 14:

1. The estimated hours for development are to attain a 75 percent efficacy rate (i.e., accuracy of the determinations). The remaining 25 percent are likely unable to be parsed due to the limiting factors associated with OCR readers. These inaccuracies may be a result of not having enough information to make an informed decision, having incorrect information on the PARs, or are the result of processors not correctly identifying information. Despite an inability to make a determination within these instances, the summary reports may still be utilized by DataQs analysts in the PAR evaluation.
2. The development hours are expected to decrease over time due to process familiarity and reusability of modular code. However, each version may contain unique challenges that must be handled appropriately.
3. There are approximately 800 RDR submissions per month, which are stratified by version estimates.
4. Development includes testing and iteratively refining the processed PARs to capture appropriate information and reduce the inappropriate information captured.
5. Incorporation of the narrative analysis may supplement missing data elements or complement processed data elements but requires additional logic to implement.
6. DataQs analysts with the environment established are capable of executing all processes.
7. The estimates from private sector came from a rough order of magnitude of execution. The rough order of magnitude is targeted at -25 to +75 percent accuracy in costs.
8. VTTI hourly rate includes labor, fringe, and indirect costs. A minimal fee is associated with Cloud OCR processing.
9. USDOT-IT hourly rates are estimated based on required position (Project Manager, Data Scientist, and Data Analyst).

10. An estimated number of 5,100 PARs would have automated determinations.
11. The estimated amount of time to complete one PAR by a DataQs analyst is 15 minutes. The implementation of the top 15 PAR versions at a 75 percent rate would decrease DataQs analysts' determination time by 1,275 hours annually if automated.
12. A human-in-the-loop model evaluating the produced summary report may decrease the reduction by roughly 20–33 percent (i.e., saving 854–1,020 hours annually).

3.1 EXECUTABLE PROCESS

The executable process for implementing the current automation system is anticipated to perform the following steps:

1. Analyst sets a 1-week filter of open RDRs utilizing advanced search features and saves the search to their profile.
    a. The filter is expected to contain new RDRs, though some RDRs may not have attached PARs until the official submission.
    b. Analyst may first perform rudimentary checks on the appropriateness of the PAR submission, ensuring that the PAR is attached and all pages are available.
2. Analyst establishes Python environment using preferred GUI.
    a. A one-time setup includes creating a local file for DataQs website login information and establishing certain directories for storing databases.
3. Analyst executes web parsing command using options: get, parse, download.
4. Analyst executes OCR parser on the Google Cloud Processing platform.
5. Analyst imports output JSON into the interpreter.
    a. Summary reports are generated from the interpreter based on available State templates and processors developed.
6. Analyst uploads summary reports to appropriate RDR and evaluates on report output to make final recommendation.

3.2 PARSER MAINTENANCE ELEMENTS

Several software or process components for the processor must be maintained:

1. Any time a PAR is updated by a State, elements must be verified before implementing in that State. These changes may take as little as 1 hour to complete if changes are minimal, but up to a full process execution (e.g., 40 hours) if the version is completely reworked. The data elements to verify include:
    a. Annotation labels are accurate and boundary boxes are correctly placed.
    b. State codes are accurately updated.

    c. Interpreter logic is updated to reflect any changes to OCR output or boundary box shifts.

    d. Designated text may be introduced to distinguish between forms within State based on adoption rates by locale. RDRs submitted may continue to contain both forms even after full adoption due to submission dates.

2. Any changes to the structure of the DataQs website table will severely impact the efficacy of the web scraper.

    a. If changes are made, verifying or correcting the parsed information from the web scraper can be performed to fix the location of information included in the records tables.

3. Major changes to the OCR reader used (e.g., Google Document AI) will alter the structure of the JSON file outputs. These changes may increase or decrease fidelity of the output based on readability or other factors of the document. Minor or incremental improvements are expected to be made by Google Cloud Platform but should have minimal impact on the PAR processing task.

4. Gradual improvements may be made to increase the accuracy and confidence of interpreted data, including:

    a. Increasing accuracy of parsed data from the JSON file.

    b. Providing additional tokens within NLP of narratives.

    c. Improving readability of submitted PAR.

5. Monthly costs are minimal and are limited to the Google Cloud Platform.

[This page intentionally left blank.]

4. RECOMMENDATIONS

Although the project sought to explore the capabilities of AI and DSSs to support or supersede human involvement in the manual reduction of DataQs elements, a demonstrable solution exists in which a DataQs analyst may utilize the produced summary report while evaluating eligibility and preventability criteria to reduce their time spent making determinations. This human-in-the-loop model ensures that determinations are decided by the DataQs analyst, but the process is accelerated by using information gathered and provided by the automation system.

While a learning-based system to read and assess different versions of PARs is both unwieldy and unreliable, a more nuanced creation of individual parsers for each version of the PAR in circulation can fit the human-in-the-loop model as described. Each PAR provides a set of unique challenges that can be addressed using a continually built series of tools, codes, and modules that would ultimately hasten the development of each parser system.

With the development of the automation process for Texas, several additional options are available for future efforts; namely, narratives, crash diagrams, and other token words have been created in bulk, which may be utilized throughout other processes. The creation of Texas n-gram models can be performed through NLP of the extracted narratives. Features of other components may be identified and extracted using ML and CV techniques. These efforts may also be translated to the development of other PAR versions.

Throughout the exploration and demonstration of automation techniques, the research team has identified a series of process elements that resulted in a loss of data related to PARs or information contained in the PARs. These estimates are presented in Table 15 along with a proposed methodology for correcting the data loss through the automation system.

Table 15. List of process elements and the associated estimated percentage loss of PARs.

Process Element	Estimated Percentage Data Loss	Proposed Correction	Estimated Percentage of Data Loss Recovered
PAR Retrieval: Wrong filetype submitted	1%	DataQs analyst can ensure RDR contains appropriate PAR filetype before or after running web scraper and identifying issue RDR IDs.	100%
PAR Retrieval: Wrong file pulled	5%	Alter code in the web scraping to pull all files, then parse codes to identify PAR amidst other files.	90%
OCR Reader: Five-page limit of OCR processor	10–40% PARs affected	Provide Google storage with the PARs or slice and stitch PARs that exceed five pages in length.	100%

Process Element	Estimated Percentage Data Loss	Proposed Correction	Estimated Percentage of Data Loss Recovered
Document Parser: Unrecognized template match	20%	Certain documents do not match any annotated labels and do not get parsed by the OCR processor. Ensuring that the correct documentation is provided or that any extreme rotations or substitutions are accounted for between the parser and the web scraper will reduce data loss.	85%
Document Parser: Unmatched USDOT Number	1%	Investigation revealed a somewhat common mismatch between requestor USDOT number and the USDOT number provided on the form. This may be attributed to police typos or to poor OCR. Fuzzy matching may assist in improving performance.	25%
Document Parser: Unread or misread data	20–30% of PAR's content	A series of efforts can increase the accuracy and confidence of parsed elements through additional logic or OCR processes. These elements are highly unstandardized and are addressed by an iterative process.	60%

Ultimately, the research team recommends the following efforts for future iterations of the automation processing:

1. Expand the current automation system from Texas (approximately 11 percent of total RDRs) to the top 15 States (approximately 70 percent of total RDRs).
2. Expand the narrative using additional NLP techniques to utilize the unit assignment automation and crash type database, assist in determinations of eligibility and preventability, and highlight pertinent information for the DataQs analyst.
3. Build a repository of document excerpts that can be used for training purposes to identify key eligibility or preventability information.
4. Build a web application that can be utilized by DataQs analysts or other users without the need to interact with a terminal window or coding environment.
5. The proposed corrections outlined in Table 15 could be implemented across the system process or as denoted for each specific version. This includes:
    a. Increasing accuracy by performing iterative investigation of incorrect values.
    b. Correcting web scraping issues to accommodate multiple files.
    c. Performing fuzzy matching or logic to ensure correct information is recorded.
    d. Performing CV OCR on selected excerpts from individual PAR versions that are not able to be read from Google's Document AI OCR processor.

Future coordinated efforts between State and Federal Government agencies may produce a database of machine-readable PARs, with data fields provided in a structured format. This would alleviate a significant burden on the AI/DSS process and allow for a simpler means for attaining automated determinations.

This customized solution requires significant effort by the automation developer but is expected to result in an approximate 50 percent reduction in total time spent by DataQs analysts in their determination efforts. A cooperative effort between DataQs analysts, DataQs supervisors, and an automation development team may reduce the time further through mutual iterative development of each PAR version processor. This effort would involve the development team providing frequent updates on the status of the processor and having the DataQs team serve as subject matter experts to identify the most relevant information related to the determinations.