Data pipeline for GTFS transit arrival and departure information

Teck Kean Chin
The University of Sydney

Tingsen (Tim) Xian (tingsen.xian@sydney.edu.au)
The University of Sydney

Benji Marks
The University of Sydney

John D. Nelson
The University of Sydney

Emily Moylan
The University of Sydney

Research Article

Keywords: Big Data, Open Data, General Transit Feed Specification (GTFS), GTFS-S, GTFS-R, Public Transport, Transport Performance

Posted Date: September 23rd, 2022

DOI: https://doi.org/10.21203/rs.3.rs-2090922/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License
Data pipeline for GTFS transit arrival and departure information

Teck Kean Chin\textsuperscript{1}, Tingsen (Tim) Xian\textsuperscript{1*}, Benjy Marks\textsuperscript{1}, John D. Nelson\textsuperscript{2} and Emily Moylan\textsuperscript{1}

\textsuperscript{1}School of Civil Engineering, The University of Sydney, 2006, New South Wales, Australia.
\textsuperscript{2}Institute of Transport and Logistics Studies, The University of Sydney, 2006, New South Wales, Australia.

*Corresponding author(s). E-mail(s): tingsen.xian@sydney.edu.au;

Abstract

Cities generate large volumes of data daily through digital services and smart city applications, these include Public Transport Authorities which generate big data as part of their daily operations, such as vehicle positions, counts of passengers and user travel patterns. The General Transit Feed Specification (GTFS) is a data format that allows public transport data to be consumed by a wide variety of software applications. This paper presents a data pipeline developed to manipulate the GTFS feeds into a general and flexible dataset of realtime transit arrivals. There are three barriers to widespread access to the information addressed by creating a one-size-fits-all data pipeline for realtime operations from GTFS. First, the protocol buffer format is not human readable and requires processing before use in most transport applications. Secondly, the general specification does vary place-to-place and the conditionally required and optional fields are inconsistent between locations. Thirdly, the raw data may contain errors including missing stop sequence or a reverse direction bus being detected in the bus stop area. The pipeline is constructed of set of data cleaning and transformation steps to address these challenges. The paper briefly presents a potential use cases of the processed data to illustrate its relevance to researchers and practitioners.

Keywords: Big Data, Open Data, General Transit Feed Specification (GTFS), GTFS-S, GTFS-R, Public Transport, Transport Performance
1 Introduction

Cities generate large volumes of data daily through digital services and smart city applications such as sensors, video cameras, traffic management systems, smart meters, vehicles, mobile phones and Internet of Things devices. The volume of this data is growing exponentially over time (Zinieris, 2022). Smart City investments are sometimes rationalised by the sense of inherent value of the data they generate without comprehensively specifying the plans for using that data. Meanwhile, the amount of data generated by smart city applications is growing exponentially. A relatively small number of organisations use the data, and most organisations use less than half of the data they have collected (Barrett, 2018).

Public Transport Authorities (PTAs) also generate big data as part of their daily operations, such as vehicle positions, counts of passengers and user travel patterns (gtfs.org, 2021a). This data is being used by these authorities for service delivery and planning, to improve network performance and safety, and to optimise operational costs and resources. There is an opportunity to harness this data for the development of applications that support data-driven decision-making, enable proactive customer engagement and improve customer experience, thus making cities more efficient and liveable. In the development of these data uses, there is an awareness of the importance of data privacy and the need to build in data protection and ensure that data uses are aligned with customer expectations (Cottrill, 2021).

The General Transit Feed Specification (GTFS) is a data format that allows PTA data across all modes to be consumed by a wide variety of software applications. Thousands of public transport providers worldwide use the GTFS data format to openly share their data (gtfs.org, 2021a). This common sharing format allows for the private development of applications (apps) such as those that monitor realtime arrivals, track public transport vehicles, support Mobility as a Service (MaaS), sell commercial advertisements and or share customised messaging.

There are two distinct components to the specification. The GTFS static format contains planned operational information such as routes, stops, trips and schedules. In contrast, the GTFS realtime format contains live transit data about forecasted trip updates, vehicle positions and network status. Due to the large volume of data produced, the realtime data exchange format is based on Protocol Buffers, a language- and platform-neutral mechanism for serialising structured data.

A key type of information contained in the GTFS realtime feed is the actual arrival of every transit vehicle at every planned stop. The feed contains redundant, duplicate and incorrect data because the data is generated on the fly. In order to extract the actual arrival information, the protocol buffers
Data pipeline for GTFS transit arrival and departure information

need to be processed and filtered. Since the forecasted arrival and departure times are updated continuously, we assume that the most recent update conveys the best estimation of arrival and departure time.

This paper presents a data pipeline developed to manipulate the GTFS realtime trip updates feeds into a general and flexible dataset of realtime bus arrivals. The pipeline, written in Python, has been tested on six (6) metropolitan areas in Australia and New Zealand (ANZ). The pipeline is used to generate 25 months of real departure and arrival information for Sydney, Australia for use by researchers and practitioners. Finally, this paper explores a potential use cases of the processed data.

1.1 Objectives
The objectives of this work are to:

• present a data pipeline that extracts (without human supervision) the GTFS trip updates data for generic consumption across locations and applications
• warehouse a long baseline example of the processed datasets that can be used to prototype and explore transit performance analysis;
• illustrate a relevant application of the pipeline’s output

Section 2 discusses the development of the data pipeline for processing the Static and Realtime feeds using Python. It also reports the validation of the pipeline on six cities in Australia and New Zealand (ANZ). Sections 3 and 4 explore a potential use case of the processed datasets and provide suggestions for future research respectively. Finally, Section 5 contains information on data availability.

2 GTFS Data Pipeline Development

2.1 GTFS Data Pipeline Process
There are three barriers to widespread access to the information addressed by creating a one-size-fits-all data pipeline for realtime operations from GTFS. First, the protocol buffer format is not human readable and requires processing before use in most transport applications. Secondly, the general specification does vary place-to-place and the conditionally required and optional fields are inconsistent between locations. Thirdly, the raw data may contain errors including missing stop sequence or a reverse direction bus being detected in the bus stop area. The pipeline is constructed of set of data cleaning and transformation steps to address these challenges as detailed below.

Figure 1 illustrates an overview of the GTFS data pipeline. The pipeline has been developed using Python 3.9. Each step discussed in the figure represents a modular function in order to provide greater flexibility to operate each step.
in the data pipeline independently. Guides for using the pipeline scripts are provided on the Python Scripts Repository.

![Fig. 1 GTFS Data Pipeline Overview](image)

The input to this pipeline are protocol buffers detailing the trip updates. A trip update includes a forecasted arrival and departure time for every active vehicle at every planned stop, and we equate the last published forecast to the actual arrival or departure time. The protocol buffers are obtained through the relevant PTA’s Application Programming Interface (API), which needs to be queried on small intervals (1-5min) to ensure accurate forecasts of the arrival and departure time. Further information on what is contained in the specification can be found on:

- GTFS Static, which contains schedule, fare, and geographic transit information: [https://gtfs.org/reference/static](https://gtfs.org/reference/static)
- GTFS Realtime, which provides realtime information about forecasted arrival times, vehicle locations, and service disruptions (station closures, lines not operating, important delays, etc.). GTFS Realtime v1: [https://gtfs.org/reference/realtime/v1/](https://gtfs.org/reference/realtime/v1/) GTFS Realtime v2: [https://gtfs.org/reference/realtime/v2/](https://gtfs.org/reference/realtime/v2/)

The data structure of the processed datasets and definition for the additional fields resulted from the data pipeline are discussed below.

### 2.1.1 Version of the input data

As can be seen in Table 1, most of the testing locales are currently using version 1 of GTFS Realtime. For this reason, the study has developed the data pipeline using GTFS Realtime v1 datasets, and it is primarily tested on the bus datasets acquired through Transport for New South Wales [Transport for NSW (2021)](https://www.transport.nsw.gov.au). Some of the data issues identified during the data transformation and cleaning steps are expected to be addressed in GTFS Realtime v2, which is discussed further in Section 4.

### GTFS Datasets Ingestion

The GTFS Realtime datasets are ingested from relevant PTAs’ APIs and saved as the Raw Protocol Buffer Data. The realtime Protocol Buffers .PB files are compressed using gzip file format .GZ during the data ingestion process.
Table 1 Summary of GTFS Datasets Tested

<table>
<thead>
<tr>
<th>Location</th>
<th>Transport Authority</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Greater Sydney, NSW</td>
<td>Transport for New South Wales</td>
<td>v1 / v2</td>
</tr>
<tr>
<td>Greater Brisbane, QLD</td>
<td>TransLink</td>
<td>v2</td>
</tr>
<tr>
<td>Greater Adelaide, SA</td>
<td>Adelaide Metro</td>
<td>v1</td>
</tr>
<tr>
<td>Canberra, ACT</td>
<td>Transport Canberra</td>
<td>v2</td>
</tr>
<tr>
<td>Auckland, NZ</td>
<td>Auckland Transport</td>
<td>v1</td>
</tr>
<tr>
<td>Christchurch/Canterbury, NZ</td>
<td>Environment Canterbury (ECan) / Metro</td>
<td>v1</td>
</tr>
</tbody>
</table>

1 At time of writing, TfNSW is transitioning the GTFS Realtime datasets from v1 to v2.

A directory of the feeds used for testing the pipeline is available at https://github.com/TeckKean/GTFS-Feeds-Directory which tabulates information about the version of the GTFS feeds, API URLs and other associated information. The first step of the data pipeline uses the data in the feeds directory to select the appropriate specifics to process. The database currently contains information the six (6) locations tested in the study. It is anticipated that the list will grow in the future with contributions from the GTFS data pipeline users.

From Protocol Buffers to Human-Readable Comma Separated Values

GTFS Realtime is a data format for communicating public transport’s real-time information. The GTFS Realtime data exchange format is based on Protocol Buffers, a language- and platform-neutral mechanism for serialising structured data (Google Developers, 2021b). The data structure is defined in a .proto file, which then is used to generate source code to easily read and write the structured data from and to a variety of data streams, using a variety of languages (e.g. Python). The standard hierarchy of elements and their type definitions are specified in the gtfs-realtime.proto file (Google Developers, 2021a). The only exception is when the GTFS Realtime datasets have extended data which is specific to a PTA. An Extension Registry is being maintained by Google Transit (2021). The following two (2) datasets tested in this study have extended data, which require .proto files provided by relevant PTAs:

- TfNSW protocol file uses extension id 1007 that includes TfnsVehiceldescriptor and CarriageDescriptor.
- Adelaide Metro’s protocol file uses extension id 1999 for wheelchair accessibility information.

The .proto file is used to generate the necessary libraries in the programming language of choice, Python in this case. These libraries provide the classes and functions needed for constructing GTFS Realtime data model objects and serialising them as binary data, or in the reverse direction, parsing binary data into data model objects.

The pre-generated GTFS Realtime language binding for Python, gtfs_realtime_pb2.py, from MobilityData (2021), is published as a package.
gtfs-realtime-bindings is used in the GTFS data pipeline to convert the gzipped protocol buffers to gzipped .CSV files providing standard GTFS Realtime data. The CSV files generated from the current GTFS data pipeline exclude extended data. PTA-specific .proto file can be used if extended data is required. Compared to the compressed protobufs, the CSVs require substantial storage space but are human readable and compatible with many common software.

Key observations from converting the .PB.GZ to .CSV.GZ files for the six (6) study locations are:

- Optional fields provided in the datasets are generally inconsistent among the study locations.
- The organisation of the geographical and mode aspects of the data in a feed vary by PTA. For example, the Trip Update feed from Transport for NSW covers the entire New South Wales by public transport mode (bus, lightrail, ferry, etc), whereas the Trip Update feed from TransLink in Queensland covers all public transport modes by region.
- The realtime feeds for Canberra combined both Trip Update and Vehicle Position datasets. Service Alert dataset is not available from Canberra. In addition, only light rail GTFS realtime feed available in Canberra at the time of writing. Realtime bus information is available in Service Interface for Real Time Information (SIRI) standard.

These findings underscore the value of the pipeline as there is significant commonality between modes and locations to justify reuse of the code used to create a single dataset for a given mode and location.

Data Wrangling

The quality of the input data is essential for the performance of the pipeline. There are a number of GTFS feeds validation tools available for checking the data quality before publishing (gtfs.org, 2021b). Currently there are 52 types of errors and nine (9) types of warnings encountered in the GTFS Realtime feeds as listed in the GTFS Realtime Validator developed by the Center for Urban Transportation Research (2020). This pipeline used GTFS feeds published by PTAs and assumed all feeds contain valid data.

To ensure the quality of the data pipeline outputs, data quality prescreening is performed as part of the Data Wrangling/Transformation process.

3 Example application for transit signal priority validation

Various bus priority measures have been implemented globally with the goal of improving traveller experience and operational performance. One implementation is transit signal priority where late and full buses are given extended
green time at lights to get back on schedule. In order to quantify and validate the benefits of TfNSW’s Public Transport Information and Priority System (PTIPS), the final product of the data pipeline can be used to validate the performance of the priority request system. Since the benefit arises from small time savings over a large number of vehicles and passengers, we need high resolution, vast data in order to understand the value of the priority. Figure 2 illustrates the validation process including the addition of traffic signal locations and traffic volume data to the final products of the original data pipeline. The Python scripts for this pipeline can be found on https://github.com/SCALUT/GTFS-Data-Pipeline-TfNSW-Bus.

Fig. 2 GTFS Data Pipeline for Transit Signal Priority Request Validation

4 Suggestions for Future Research

As PTAs undertake the shift towards GTFS Realtime V2, it will be necessary to refine the current GTFS data pipeline to take into account the improvements.

The current GTFS data pipeline could be expanded to include other data standards, such as Service Interface for Real Time Information (SIRI), an XML protocol to allow distributed computers to exchange real-time information about public transport services and vehicles (European Committee for Standardisation, 2021).

The pipeline has been primarily developed using Sydney’s bus data, and tested against six cities in Australasia. The pipeline could be tested on a wide range of feeds to ensure it works elsewhere in the world. This process could result in the creation of additional datasets to share within the data warehouse.

The development of the pipeline relied heavily on the researchers’ understanding of the engineering and operations of the public transport system. Data
cleaning and processing was based on relevant insights into why the data could be recording in misleading ways (for example, a vehicle being detected in the geofence of a stop on the opposite side of the road on an out-and-back shaped bus route). Future development could take advantage of machine learning-powered anomaly detection techniques to improve data quality without relying on domain expertise in public transit operations.

The current pipeline fuses planned and realtime information about the public transport system. Future applications may share common data requirements outside of this such as dynamic and spatial data about the weather, roadworks or traffic volumes. These merges could be incorporated into the pipeline.

Some issues identified in the GTFS Realtime v1 datasets as discussed in the GTFS data pipeline process above are expected to be resolved in the GTFS Realtime v2. Most of these issues are caused by the optional fields in GTFS Realtime v1. For instance, only seven (7) data fields were required out of the 63 data fields. The optional fields allowed some critical information to be missing but still reported in the data feed, which affects data quality (Barbeau, 2017).

5 Data Availability Statement

5.1 Data Repository

The study processed a passively-collected GTFS Realtime big dataset obtained through the TfNSW Open Data Hub’s APIs from Transport for NSW (2021), representing 25 months of bus performance. The dataset generated by the pipeline are available on The University of Sydney’s Sydney eScholarship Repository (SES), https://doi.org/10.25910/1pfb-4z05.

5.2 Python Scripts Repository

The Python scripts of the GTFS data pipeline are available on GitHub (https://github.com/SCALUT). The scripts were prepared using Python 3.9 and have been tested to work on Windows, Linux and Mac operating systems.

5.3 Data and Scripts Licensing

Both the datasets and Python scripts in reported this paper are made available under open licences that permit reuse.

Acknowledgements

This research is partly funded by iMOVE CRC and supported by the Cooperative Research Centres program, an Australian Government initiative. The authors acknowledge the Sydney Informatics Hub and The University of Sydney’s high performance computing cluster, Artemis, for providing the high
performance computing resources that have contributed to the research results reported within this paper.

**Disclosure and Declaration Statements**

The authors are not aware of any potential conflicts of interest.

**References**


Figures

Figure 1
GTFS Data Pipeline Overview

Figure 2
GTFS Data Pipeline for Transit Signal Priority Request Validation