

---

# SUPPLEMENTARY MATERIAL - SC2EGSET: STARCRAFT II ESPORT REPLAY AND GAME-STATE DATASET

---

Redacted Authors

## 1 Motivation

This document is based on Datasheets for Datasets framework [1] and contains the appendices to the main article introducing SC2EGSet dataset.

### **For what purpose was the dataset created? Was there a specific task in mind?**

We have created this dataset to open StarCraft II to the broader scientific community. The goal of this dataset is to attract more research in esports. Especially in order to create, and verify new training methods with technological help against the existing and publicly available data. As of the date of dataset publication there were no publicly available pre-processed esports StarCraft II datasets available.

### **Who created the dataset (e.g., which team, research group) and on behalf of which entity (e.g., company, institution, organization)?**

The dataset was created by (REDACTED) and a team of independent collaborators without institutionalized help.

### **Who funded the creation of the dataset?**

This dataset was self-funded.

### **Any other comments?**

Not applicable.

## 2 Composition

### **What do the instances that comprise the dataset represent (e.g., documents, photos, people, countries)? Are there multiple types of instances (e.g., movies, users, and ratings; people and interactions between them; nodes and edges)? Please provide a description.**

Every directory represents a single tournament. Within the directories, you will find 4 different types of files: extracted data from viable games in zip format, main log of the program that was used for data extraction, secondary log that contains a list of paths to processed replays alongside the list of paths to replays that were not processed due to technical errors, and summary JSON file containing information that can be used for data distribution verification.

Raw data that was used to generate the dataset (SC2ReSet) is also available in a different repository (REDACTED).

### **How many instances are there in total (of each type, if appropriate)?**

The first version of the dataset consists of 55 tournaments and 17930 replays.

**Does the dataset contain all possible instances or is it a sample (not necessarily random) of instances from a larger set? If the dataset is a sample, then what is the larger set? Is the sample representative of the larger set (e.g., geographic coverage)? If so, please describe how this representativeness was validated/verified. If it is not representative of the larger set, please describe why not (e.g., to cover a more diverse range of instances, because instances were withheld or unavailable).**

This dataset consists of all publicly available StarCraft II tournaments that are classified as "Premiere" or "Major" according to Liquipedia, starting from 2016 IEM Taipei until 2022 IEM Katowice tournaments. Therefore, it is not a sample but a collection of all the available tournament replays (SC2ReSet) that were published by the tournament organizers or administrators. Additionally, all of the replays were pre-processed through our toolset (REDACTED; REDACTED; REDACTED). This resulted in the final dataset (SC2EGSet). Only the replays that our toolset failed to process are not present in the final dataset.

**What data does each instance consist of? "Raw" data (e.g., unprocessed text or images) or features?**

Each archived zip instance consists of multiple pre-processed replays that were saved into a JSON data format.

**Is there a label or target associated with each instance?**

No additional labeling of the data was performed. The simplest label that could be used is "victory" or "defeat" for any inspected replay of a tournament game.

**Is any information missing from individual instances? If so, please provide a description, explaining why this information is missing (e.g., because it was unavailable). This does not include intentionally removed information, but might include, e.g., redacted text.**

Not applicable.

**Are relationships between individual instances made explicit (e.g., users' movie ratings, social network links)?**

No relationships between individual instances were made explicit. The dataset contains tournament data at its different stages. Due to this, separate instances can contain related information about tournament progress.

**Are there recommended data splits (e.g., training, development/validation, testing)? If so, please provide a description of these splits, explaining the rationale behind them.**

In our API, we expose routines that randomly assign data into three splits. Test split consists of  $N * 1/6$  entries, validation split consists of  $N * 1/10$ , and train split consists of  $N * 11/15$

**Are there any errors, sources of noise, or redundancies in the dataset?**

Explicit errors within the dataset can be verified by accessing the log files. Any implicit errors were not investigated.

**Is the dataset self-contained, or does it link to or otherwise rely on external resources (e.g., websites, tweets, other datasets)? If it links to or relies on external resources, a) are there guarantees that they will exist, and remain constant, over time; b) are there official archival versions of the complete dataset (i.e., including the external resources as they existed at the time the dataset was created); c) are there any restrictions (e.g., licenses, fees) associated with any of the external resources that might apply to a future user? Please provide descriptions of all external resources and any restrictions associated with them, as well as links or other access points, as appropriate.**

This dataset is self-contained.

**Does the dataset contain data that might be considered confidential (e.g., data that is protected by legal privilege or by doctor/patient confidentiality, data that includes the content of individuals' non-public communications)?**

no - Described dataset consists of esports tournament games that were published by the tournament organizers and administrators, therefore it was public before any information was inferred.

**Does the dataset contain data that, if viewed directly, might be offensive, insulting, threatening, or might otherwise cause anxiety?**

Not applicable.

**Does the dataset relate to people?**

Yes.

**Does the dataset identify any subpopulations (e.g., by age, gender)? If so, please describe how these subpopulations are identified and provide a description of their respective distributions within the dataset.**

Not applicable.

**Is it possible to identify individuals (i.e., one or more natural persons), either directly or indirectly (i.e., in combination with other data) from the dataset?**

Yes, the dataset consists of information relating to public esports players, who are identifiable by their nicknames or by the stage of the tournament.

**Does the dataset contain data that might be considered sensitive in any way (e.g., data that reveals racial or ethnic origins, sexual orientations, religious beliefs, political opinions or union memberships, or locations; financial or health data; biometric or genetic data; forms of government identification, such as social security numbers; criminal history)?**

Not applicable.

### 3 Collection process

**How was the data associated with each instance acquired? Was the data directly observable (e.g., raw text, movie ratings), reported by subjects (e.g., survey responses), or indirectly inferred/derived from other data (e.g., part-of-speech tags, model-based guesses for age or language)? If data was reported by subjects or indirectly inferred/derived from other data, was the data validated/verified? If so, please describe how.**

The initial raw data was collected through online searches of StarCraft II "replaypacks". After that, processing was conducted to extract the information. The data was verified in pre-processing stages by comparing and contrasting information found in redundant data structures (REDACTED).

**What mechanisms or procedures were used to collect the data (e.g., hardware apparatus or sensor, manual human curation, software program, software API)? How were these mechanisms or procedures validated?**

The process of data collection was conducted manually by downloading the "replaypacks".

**If the dataset is a sample from a larger set, what was the sampling strategy (e.g., deterministic, probabilistic with specific sampling probabilities)?**

Not applicable.

**Who was involved in the data collection process (e.g., students, crowdworkers, contractors) and how were they compensated (e.g., how much were crowdworkers paid)?**

The data collection process was done in full by (REDACTED).

**Over what timeframe was the data collected? Does this timeframe match the creation timeframe of the data associated with the instances (e.g., recent crawl of old news articles)?\*\* If not, please describe the timeframe in which the data associated with the instances was created.**

The data collection was started in 2021.08, the timeframe of the data itself is since the first tournament played using StarCraft II: Legacy of The Void.

**Were any ethical review processes conducted (e.g., by an institutional review board)? If so, please provide a description of these review processes, including the outcomes, as well as a link or other access point to any supporting documentation.**

No.

**Does the dataset relate to people?**

Yes.

**Did you collect the data from the individuals in question directly, or obtain it via third parties or other sources (e.g., websites)?**

The dataset was collected from: Liquipedia, SpawningTool, tournament organizer's websites, and from tournament administrator publicly provided repositories.

**Were the individuals in question notified about the data collection? If so, please describe (or show with screenshots or other information) how notice was provided, and provide a link or other access point to, or otherwise reproduce, the exact language of the notification itself.**

Not applicable. The data was pre-collected by the tournament organizers.

**Did the individuals in question consent to the collection and use of their data? If so, please describe (or show with screenshots or other information) how consent was requested and provided, and provide a link or other access point to, or otherwise reproduce, the exact language to which the individuals consented.**

Not applicable.

**If consent was obtained, were the consenting individuals provided with a mechanism to revoke their consent in the future or for certain uses? If so, please provide a description, as well as a link or other access point to the mechanism (if appropriate).**

Not applicable.

**Has an analysis of the potential impact of the dataset and its use on data subjects (e.g., a data protection impact analysis) been conducted? If so, please provide a description of this analysis, including the outcomes, as well as a link or other access point to any supporting documentation.**

Not applicable.

## **4 Preprocessing/Cleaning/Labeling**

**Was any preprocessing/cleaning/labeling of the data done (e.g., discretization or bucketing, tokenization, part-of-speech tagging, SIFT feature extraction, removal of instances, processing of missing values)?**

Yes, during the preparation of the dataset raw files in SC2Replay format were ingested, restructured and outputted. No direct labelling was done. For details, refer to the open-sourced software used for extraction ([REDACTED](#))

**Was the "raw" data saved in addition to the preprocessed/cleaned/labeled data (e.g., to support unanticipated future uses)? If so, please provide a link or other access point to the "raw" data.**

The raw data is publicly available online on Liquipedia and other aforementioned sources.

**Is the software used to preprocess/clean/label the instances available?\*\*\* If so, please provide a link or other access point.**

Yes, the software used to preprocess the instances that are within this dataset is available ([REDACTED](#); [REDACTED](#); [REDACTED](#)).

## 5 Uses

### Has the dataset been used for any tasks already?

At the time of publication the dataset have not been used for any tasks.

### Is there a repository that links to any or all papers or systems that use the dataset? If so, please provide a link or other access point.

No.

### What (other) tasks could the dataset be used for?

This dataset could be used for exploring sports analytics, sequence modeling, classification tasks, dimensionality reduction tasks, clustering of different playstyles, and possibly more.

### Is there anything about the composition of the dataset or the way it was collected and preprocessed/cleaned/labeled that might impact future uses? For example, is there anything that a future user might need to know to avoid uses that could result in unfair treatment of individuals or groups (e.g., stereotyping, quality of service issues) or other undesirable harms (e.g., financial harms, legal risks) If so, please provide a description. Is there anything a future user could do to mitigate these undesirable harms?

Not applicable.

## 6 Distribution

### Will the dataset be distributed to third parties outside of the entity (e.g., company, institution, organization) on behalf of which the dataset was created?

Yes, the dataset will be publicly available on Zenodo.

### How will the dataset will be distributed (e.g., tarball on website, API, GitHub)?\*\* Does the dataset have a digital object identifier (DOI)?

The dataset will be distributed via Zenodo ([REDACTED](#)).

### When will the dataset be distributed?

The dataset will be distributed publicly by end of September 2022.

### Will the dataset be distributed under a copyright or other intellectual property (IP) license, and/or under applicable terms of use (ToU)? If so, please describe this license and/or ToU, and provide a link or other access point to, or otherwise reproduce, any relevant licensing terms or ToU, as well as any fees associated with these restrictions.

#### 6.1 Blizzard StarCraft II AI and Machine Learning License

##### BLIZZARD® STARCRAFT® II AI AND MACHINE LEARNING LICENSE

##### IMPORTANT NOTICE:

YOU SHOULD CAREFULLY READ THIS AGREEMENT (THE “AGREEMENT”) BEFORE INSTALLING OR USING BLIZZARD’S (“BLIZZARD”) STARCRAFT II AI AND MACHINE LEARNING SOFTWARE AND ENVIRONMENT (THE “SOFTWARE”). IF YOU DO NOT AGREE WITH ALL OF THE TERMS OF THIS AGREEMENT, YOU MAY NOT INSTALL OR OTHERWISE ACCESS THE SOFTWARE.

Subject to the terms of this Agreement, your use of the Software is governed by Blizzard’s End User License Agreement (“EULA”), which is incorporated by reference herein and is available for review here. (<http://us.blizzard.com/en-us/company/legal/eula.html>) Please carefully review the EULA and this Agreement prior to installing or using the Software. IF YOU DO NOT AGREE TO THE TERMS OF THE EULA AND THIS AGREEMENT, YOU ARE NOT PERMITTED TO INSTALL, COPY, OR USE THE SOFTWARE.

1. Use Of The Software.

- A. AI Testing And Machine Learning Use Only: Subject to your compliance with this Agreement, Blizzard grants you a limited, revocable, non-sublicensable license to use the Software for purposes of AI testing, machine learning, and related research only.
- B. Blizzard Account Not Required: Notwithstanding the requirements of Section 1.A of the EULA, creation of a Blizzard Account is not required in order to use the Software. Legal entities other than natural persons are authorized to use the Software. However, other than as specifically excepted in this Agreement, the remaining provisions and requirements of the EULA are controlling.
- C. EULA Exceptions: The terms of Blizzard's EULA govern your use of the Software, subject to the following narrow exceptions:
  - i. Derivative Works: Section 1.C.i of the EULA shall not be read to prohibit the authorized use of the Software or data generated or collected from such use. However, no portion of this Agreement shall give you the right to create, distribute, or otherwise exploit unauthorized derivative works of the Software.
  - ii. Automation: The provisions of Section 1.C.ii of the EULA prohibiting the use of automation processes or software do not apply to use of the Software.
  - iii. Commercial Use: The provisions of Section 1.C.iii of the EULA govern your use of the Software, except that you are authorized to use and exploit data derived from using the Software in connection with AI and machine learning programs for personal or internal use, despite that such use of the data may ultimately be for a commercial purpose. You may not otherwise use or exploit the Software for any commercial purpose.
  - iv. Data Mining: The provisions of Section 1.C.iv of the EULA shall not prohibit the authorized use of the Software or data generated or collected from such use.
  - v. Matchmaking: The provisions of Section 1.C.vi of the EULA shall not prohibit the authorized use of the Software or data generated or collected from such use.

## 2. Ownership.

- A. The provisions of Section 2 of the EULA apply in full force to the Software (including generated by or collected through the authorized use of the Software).

**Have any third parties imposed IP-based or other restrictions on the data associated with the instances? If so, please describe these restrictions, and provide a link or other access point to, or otherwise reproduce, any relevant licensing terms, as well as any fees associated with these restrictions.**

The data itself may be used for research purposes, although the IP is owned by Blizzard Entertainment. Please reference the end user license agreement (EULA), and AI and ML License for details. Blizzard StarCraft II AI and Machine Learning License is available in subsection A.1.

**Do any export controls or other regulatory restrictions apply to the dataset or to individual instances? If so, please describe these restrictions, and provide a link or other access point to, or otherwise reproduce, any supporting documentation.**

No.

## 7 Maintenance

**Who is supporting/hosting/maintaining the dataset?**

The dataset is supported and maintained by the main author (REDACTED) the data is hosted on Zenodo.

**How can the owner/curator/manager of the dataset be contacted (e.g., email address)?**

The corresponding author can be contacted by using the following email: (REDACTED)

**Is there an erratum? If so, please provide a link or other access point.**

Not applicable.

**Will the dataset be updated (e.g., to correct labeling errors, add new instances, delete instances)? If so, please describe how often, by whom, and how updates will be communicated to users (e.g., mailing list, GitHub)?**

Yes, the dataset be updated as soon as new replaypacks are available under the conditions that the main author and contributors have time to perform all of the processing and validation required.

**If the dataset relates to people, are there applicable limits on the retention of the data associated with the instances (e.g., were individuals in question told that their data would be retained for a fixed period of time and then deleted)? If so, please describe these limits and explain how they will be enforced.**

No, as far as the authors are concerned, there are no limits on retention of the data.

**Will older versions of the dataset continue to be supported/hosted/maintained? If so, please describe how. If not, please describe how its obsolescence will be communicated to users.**

Yes. If someone wishes to contribute to this dataset, please contact the authors.

## A Appendix

### A.1 Blizzard StarCraft II AI and Machine Learning License

BLIZZARD® STARCRAFT® II AI AND MACHINE LEARNING LICENSE

IMPORTANT NOTICE:

YOU SHOULD CAREFULLY READ THIS AGREEMENT (THE “AGREEMENT”) BEFORE INSTALLING OR USING BLIZZARD’S (“BLIZZARD”) STARCRAFT II AI AND MACHINE LEARNING SOFTWARE AND ENVIRONMENT (THE “SOFTWARE”). IF YOU DO NOT AGREE WITH ALL OF THE TERMS OF THIS AGREEMENT, YOU MAY NOT INSTALL OR OTHERWISE ACCESS THE SOFTWARE.

Subject to the terms of this Agreement, your use of the Software is governed by Blizzard’s End User License Agreement (“EULA”), which is incorporated by reference herein and is available for review here. (<http://us.blizzard.com/en-us/company/legal/eula.html>) Please carefully review the EULA and this Agreement prior to installing or using the Software. IF YOU DO NOT AGREE TO THE TERMS OF THE EULA AND THIS AGREEMENT, YOU ARE NOT PERMITTED TO INSTALL, COPY, OR USE THE SOFTWARE.

#### 1. Use Of The Software.

- A. AI Testing And Machine Learning Use Only: Subject to your compliance with this Agreement, Blizzard grants you a limited, revocable, non-sublicensable license to use the Software for purposes of AI testing, machine learning, and related research only.
- B. Blizzard Account Not Required: Notwithstanding the requirements of Section 1.A of the EULA, creation of a Blizzard Account is not required in order to use the Software. Legal entities other than natural persons are authorized to use the Software. However, other than as specifically excepted in this Agreement, the remaining provisions and requirements of the EULA are controlling.
- C. EULA Exceptions: The terms of Blizzard’s EULA govern your use of the Software, subject to the following narrow exceptions:
  - i. Derivative Works: Section 1.C.i of the EULA shall not be read to prohibit the authorized use of the Software or data generated or collected from such use. However, no portion of this Agreement shall give you the right to create, distribute, or otherwise exploit unauthorized derivative works of the Software.
  - ii. Automation: The provisions of Section 1.C.ii of the EULA prohibiting the use of automation processes or software do not apply to use of the Software.
  - iii. Commercial Use: The provisions of Section 1.C.iii of the EULA govern your use of the Software, except that you are authorized to use and exploit data derived from using the Software in connection with AI and machine learning programs for personal or internal use, despite that such use of the data may ultimately be for a commercial purpose. You may not otherwise use or exploit the Software for any commercial purpose.
  - iv. Data Mining: The provisions of Section 1.C.iv of the EULA shall not prohibit the authorized use of the Software or data generated or collected from such use.

- v. Matchmaking: The provisions of Section 1.C.vi of the EULA shall not prohibit the authorized use of the Software or data generated or collected from such use.

## 2. Ownership.

- A. The provisions of Section 2 of the EULA apply in full force to the Software (including generated by or collected through the authorized use of the Software).

### A.2 Datasheets for datasets

Datasheets for datasets [1] are defined and available as a part of the original pre-processed dataset publication (REDACTED).

### A.3 Additional Visualizations

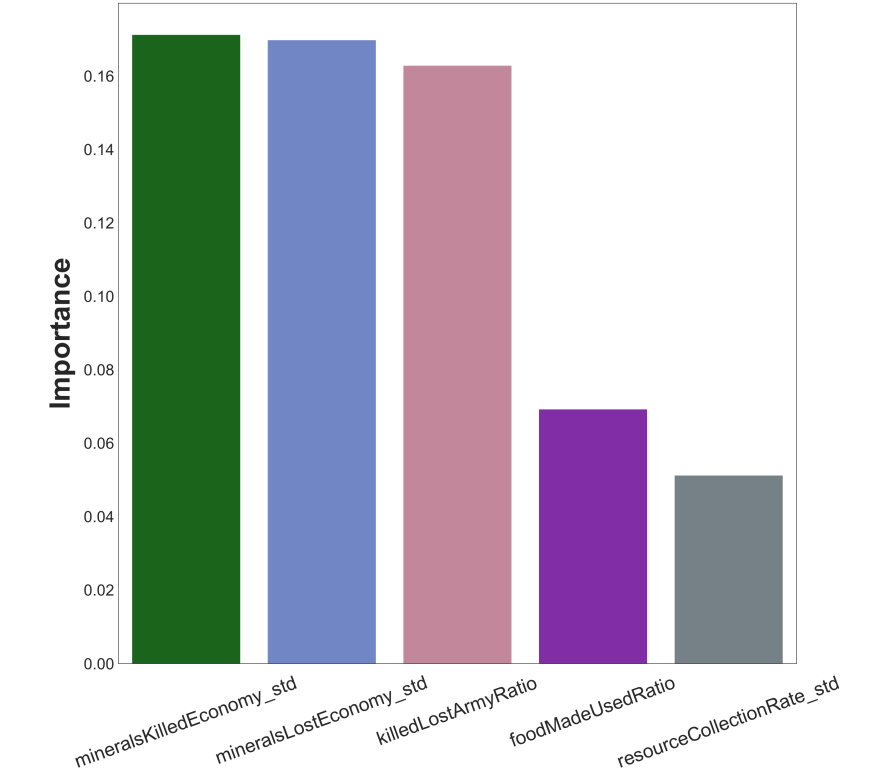


Figure 1: Percentages of feature importances based on XGBoost fit to all data.

### A.4 Pipeline Visualizations

Due to the relative complexity of our infrastructure, we include additional visualizations of the processing pipeline for all potential users on Figures 2 - 4. The dataset pre-processing is shown on Figure 2, highlighting the use of a set of tools named SC2DatasetPreparator (REDACTED). The dataset processing is shown on Figure 3, highlighting the use of SC2MapLocaleExtractor (REDACTED) to acquire the english map names, SC2InfoExtractorGo (REDACTED) to extract the data, and SC2DatasetPreparator (REDACTED) to collect the final .zip archives. The dataset post-processing and experiments are briefly visualized on Figure 4 and highlight the use of PyTorch [2], and Lightning [3].



## Dataset Pre-processing

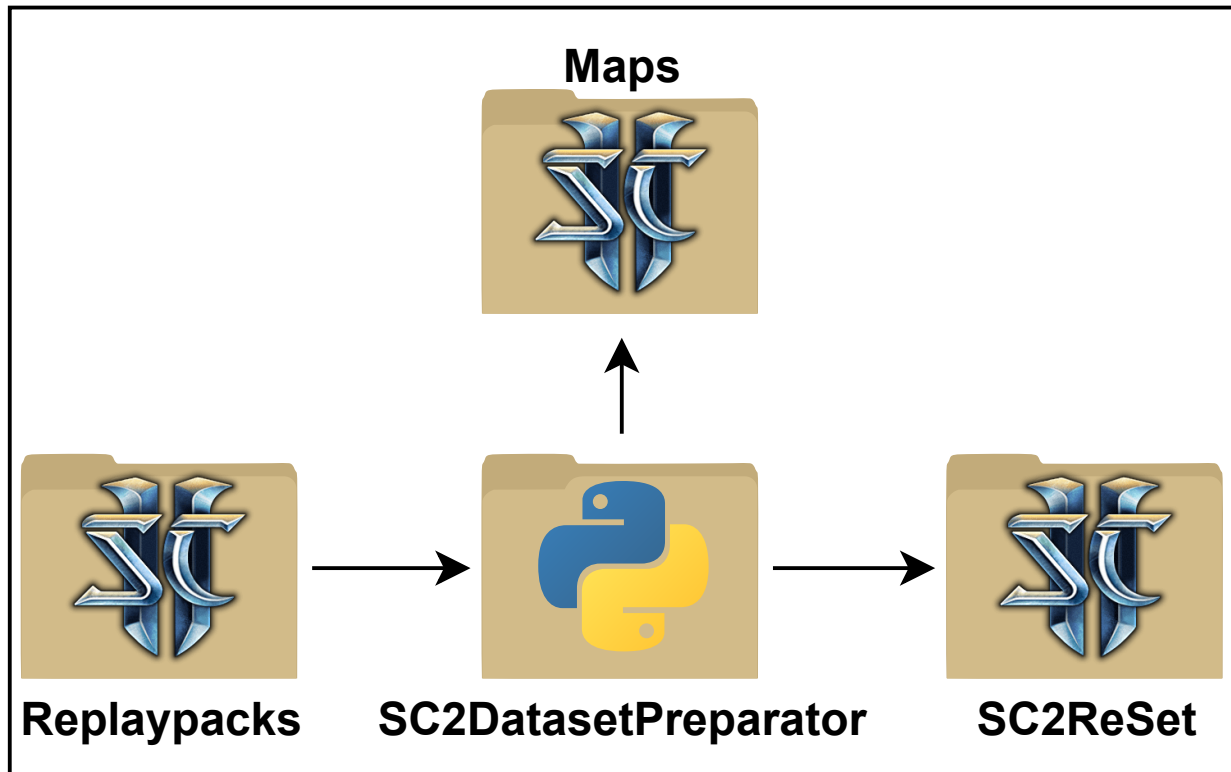


Figure 2: Pre-processing steps of our pipeline that result in SC2ReSet (REDACTED) . We are using a custom data processing toolset including SC2DatasetPreparator (REDACTED) .

## Dataset Processing

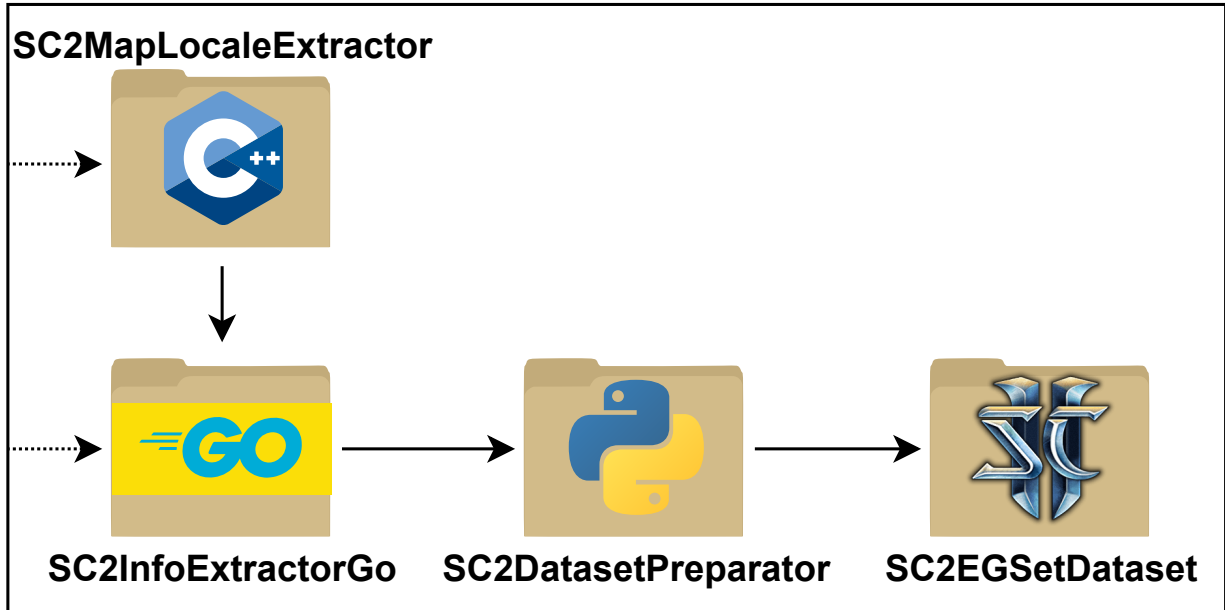


Figure 3: Processing steps of our pipeline that result in SC2EGSetDataset (REDACTED) . We are using a custom data processing toolset including SC2DatasetPreparator (REDACTED), SC2MapLocaleExtractor (REDACTED) , and SC2InfoExtractorGo (REDACTED) .

## Experiment Workflow

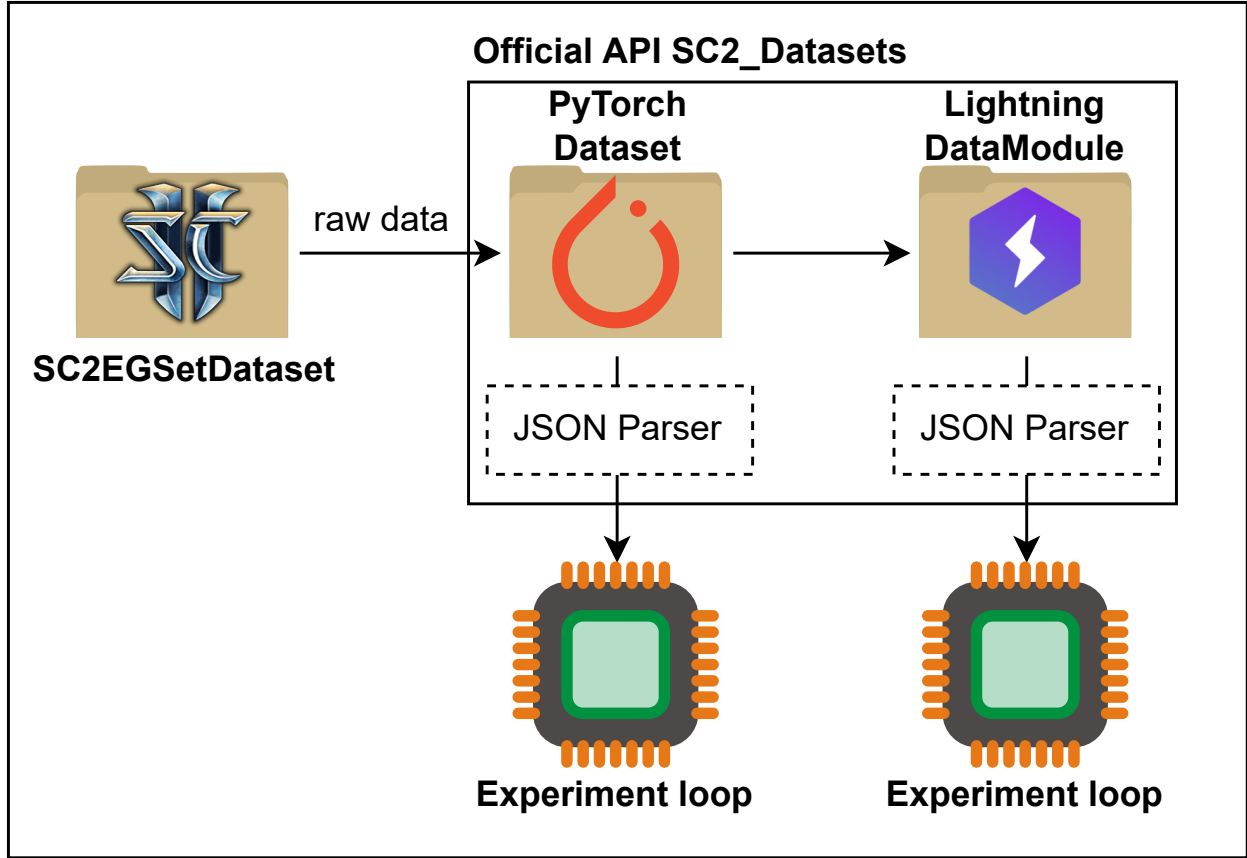


Figure 4: Using the SC2EGSetDataset (REDACTED) with the officially provided API (REDACTED) to conduct experiments.

### A.5 Dataset Usage Examples

There are various ways to use our dataset; one way includes using the custom PyTorch dataset class. Due to the page limit we were unable to visualize all of the potential uses of our infrastructure in the main text; Listing 1 showcases the most basic use of the Lightning custom DataModule class that we implemented for our dataset. For further information please refer to the official documentation.

```

from sc2_datasets.lightning.sc2_egset_datamodule import (
    SC2EGSetDataModule
)
from sc2_datasets.available_replaypacks import (
    EXAMPLE_SYNTHETIC_REPLAYPACKS
)

if __name__ == "__main__":
    # Initialize the datamodule:
    sc2_egset_datamodule = SC2EGSetDataModule(
        unpack_dir="./unpack_dir_path",
        download_dir="./download_dir_path",
        download=True,
        replaypacks=EXAMPLE_SYNTHETIC_REPLAYPACKS,
    )

    # Initializing the PyTorch dataset within the DataModule class:
    sc2_egset_datamodule.prepare_data()
    # Obtaining the splits for training, testing, and validation:
    sc2_egset_datamodule.setup()

```

Listing 1: Example use of the SC2EGSetDataModule with Lightning using a synthetic replaypack prepared for testing.

It is important to note that our classes by default return a custom SC2ReplayData class which is a serialization of the initial pre-processed JSON files. To construct a custom tensor required for further modeling, users should use the exposed keyword argument “transform”, which should be a function that transforms the default SC2ReplayData into some custom tensor required for further modeling.

## A.6 Dataset Structure Examples

We include human-readable examples of various fields showcase on Listings 2-23; these are a part of the SC2EGSet dataset JSON files. Users should refer to the respective parts of the official documentation for more information and a list of all of the available fields. Access to these can be used to define parsers in other programming languages.

### A.6.1 Top level fields

```

{
    ...
    "header": {
        "elapsedGameLoops": 7855,
        "version": "3.4.0.44401"
    },
    ...
}

```

Listing 2: Example header field containing a JSON object.

```

{
  ...
  "initData": {
    "gameDescription": {
      "gameOptions": {
        "advancedSharedControl": false,
        "amm": false,
        "battleNet": true,
        "clientDebugFlags": 265,
        "competitive": false,
        "cooperative": false,
        "fog": 0,
        "heroDuplicatesAllowed": true,
        "lockTeams": true,
        "noVictoryOrDefeat": false,
        "observers": 0,
        "practice": false,
        "randomRaces": false,
        "teamsTogether": false,
        "userDifficulty": 0
      },
      "gameSpeed": "Faster",
      "isBlizzardMap": true,
      "mapAuthorName": "5-S2-1-1",
      "mapFileSyncChecksum": 360400735,
      "mapSizeX": 152,
      "mapSizeY": 152,
      "maxPlayers": 2
    },
    ...
  }
}

```

Listing 3: Example initData field containing a JSON object with nested information.

```

{
  ...
  "details": {
    "gameSpeed": "Faster",
    "isBlizzardMap": true,
    "timeUTC": "2016-07-29T04:50:12.5655603Z"
  },
  ...
}

```

Listing 4: Example details field containing a JSON object.

```

{
  ...
  "metadata": {
    "baseBuild": "",
    "dataBuild": "",
    "gameVersion": "",
    "mapName": "Galactic Process LE"
  },
  ...
}

```

Listing 5: Example metadata field containing a JSON object.

```
{  
  ...  
  "metadata": {  
    "baseBuild": "",  
    "dataBuild": "",  
    "gameVersion": "",  
    "mapName": "Galactic Process LE"  
  },  
  ...  
}
```

Listing 6: Example metadata field containing a JSON object.

```

{
  ...
  "ToonPlayerDescMap": {
    "5-S2-1-7361539": {
      "nickname": "somePlayerNickname",
      "playerID": 2,
      "userID": 5,
      "SQ": 105,
      "supplyCappedPercent": 4,
      "startDir": 1,
      "startLocX": 127,
      "startLocY": 131,
      "race": "Zerg",
      "selectedRace": "",
      "APM": 0,
      "MMR": 0,
      "result": "Win",
      "region": "China",
      "realm": "China",
      "highestLeague": "Unknown",
      "isInClan": false,
      "clanTag": "",
      "handicap": 100,
      "color": {
        "a": 255,
        "b": 0,
        "g": 66,
        "r": 255
      }
    },
    "5-S2-1-7361634": {
      "nickname": "AnotherPlayerNickname",
      "playerID": 1,
      "userID": 1,
      "SQ": 115,
      "supplyCappedPercent": 7,
      "startDir": 7,
      "startLocX": 24,
      "startLocY": 20,
      "race": "Zerg",
      "selectedRace": "",
      "APM": 0,
      "MMR": 0,
      "result": "Loss",
      "region": "China",
      "realm": "China",
      "highestLeague": "Unknown",
      "isInClan": false,
      "clanTag": "",
      "handicap": 100,
      "color": {
        "a": 255,
        "b": 180,
        "g": 20,
        "r": 30
      }
    }
  }
  ...
}

```

Listing 7: Example ToonPlayerDescMap field containing a JSON object mapping player statistics to unique toon id.

## A.6.2 Game Events

All of the game events that were recorded by the game engine are available in one of the fields named `"gameEvents"`, all of the events that are available are presented in listings below.

```
[
  ...
  {
    "baseBuildNum": 44401,
    "buildNum": 44401,
    "cameraFollow": false,
    "debugPauseEnabled": false,
    "developmentCheatsEnabled": false,
    "evtTypeName": "UserOptions",
    "gameFullyDownloaded": true,
    "hotkeyProfile": "\u003ccustom\u003e",
    "id": 7,
    "isMapToMapTransition": false,
    "loop": 0,
    "multiplayerCheatsEnabled": false,
    "platformMac": false,
    "syncChecksummingEnabled": false,
    "testCheatsEnabled": false,
    "useGalaxyAsserts": false,
    "userid": {
      "userId": 0
    },
    "versionFlags": 0
  },
  ...
]
```

Listing 8: Example UserOptions game event JSON object.

```
[
  ...
  {
    "distance": null,
    "evtTypeName": "CameraUpdate",
    "follow": false,
    "id": 49,
    "loop": 2,
    "pitch": null,
    "reason": null,
    "target": {
      "x": 0.7109375,
      "y": 0.5469970703125
    },
    "userid": {
      "userId": 6
    },
    "yaw": null
  },
  ...
]
```

Listing 9: Example CameraUpdate game event JSON object.



```
[
  ...
  {
    "controlGroupId": 10,
    "delta": {
      "addSubgroups": [
        {
          "count": 1,
          "intraSubgroupPriority": 1,
          "subgroupPriority": 32,
          "unitLink": 108
        }
      ],
      "addUnitTags": [
        56885249
      ],
      "removeMask": {
        "None": null
      },
      "subgroupIndex": 0
    },
    "evtTypeName": "SelectionDelta",
    "id": 28,
    "loop": 12,
    "userid": {
      "userId": 5
    }
  },
  ...
]
```

Listing 10: Example SelectionDelta game event JSON object.

```
[
  ...
  {
    "abil": {
      "abilCmdData": null,
      "abilCmdIndex": 0,
      "abilLink": 188
    },
    "cmdFlags": 256,
    "data": {
      "None": null
    },
    "evtTypeName": "Cmd",
    "id": 27,
    "loop": 15,
    "otherUnit": null,
    "sequence": 1,
    "unitGroup": null,
    "userid": {
      "userId": 5
    }
  },
  ...
]
```

Listing 11: Example Cmd game event JSON object.

```
[
  ...
  {
    "evtTypeName": "CmdUpdateTargetUnit",
    "id": 105,
    "loop": 37,
    "target": {
      "snapshotControlPlayerId": 0,
      "snapshotPoint": {
        "x": 64.5,
        "y": 68.75,
        "z": 5.994140625
      },
      "snapshotUnitLink": 369,
      "snapshotUpkeepPlayerId": 0,
      "tag": 2883585,
      "targetUnitFlags": 111,
      "timer": 0
    },
    "userid": {
      "userId": 5
    }
  },
  ...
]
```

Listing 12: Example CmdUpdateTargetUnit game event JSON object.

```
[
  ...
  {
    "evtTypeName": "CommandManagerState",
    "id": 103,
    "loop": 37,
    "sequence": 3,
    "state": 1,
    "userid": {
      "userId": 5
    }
  },
  ...
]
```

Listing 13: Example CommandManagerState game event JSON object.

```
[
  ...
  {
    "controlGroupIndex": 1,
    "controlGroupUpdate": 2,
    "evtTypeName": "ControlGroupUpdate",
    "id": 29,
    "loop": 1639,
    "mask": {
      "None": null
    },
    "userid": {
      "userId": 1
    }
  },
  ...
]
```

Listing 14: Example ControlGroupUpdate game event JSON object.

```
[
  ...
  {
    "evtTypeName": "CmdUpdateTargetPoint",
    "id": 104,
    "loop": 2965,
    "target": {
      "x": 19.133056640625,
      "y": 26.369140625,
      "z": 5.73388671875
    },
    "userid": {
      "userId": 5
    }
  },
  ...
]
```

Listing 15: Example CmdUpdateTargetPoint game event JSON object.

```
[
  ...
  {
    "evtTypeName": "GameUserLeave",
    "id": 101,
    "leaveReason": 0,
    "loop": 7845,
    "userid": {
      "userId": 5
    }
  },
  ...
]
```

Listing 16: Example CmdUpdateTargetPoint game event JSON object.

### A.6.3 Tracker Events

All of the game events that were recorded by the game engine are available in one of the fields named `"trackerEvents"`, all of the events that are available are presented in listings below.

```
[
  ...
  {
    "evtTypeName": "PlayerSetup",
    "id": 9,
    "loop": 0,
    "playerId": 1,
    "slotId": 0,
    "type": 1,
    "userId": 1
  },
  ...
]
```

Listing 17: Example PlayerSetup tracker event JSON object.

```
[
  ...
  {
    "evtTypeName": "PlayerStats",
    "id": 0,
    "loop": 1,
    "playerId": 1,
    "stats": {
      "scoreValueFoodMade": 57344,
      "scoreValueFoodUsed": 49152,
      "scoreValueMineralsCollectionRate": 0,
      "scoreValueMineralsCurrent": 50,
      "scoreValueMineralsFriendlyFireArmy": 0,
      "scoreValueMineralsFriendlyFireEconomy": 0,
      "scoreValueMineralsFriendlyFireTechnology": 0,
      "scoreValueMineralsKilledArmy": 0,
      "scoreValueMineralsKilledEconomy": 0,
      "scoreValueMineralsKilledTechnology": 0,
      "scoreValueMineralsLostArmy": 0,
      "scoreValueMineralsLostEconomy": 0,
      "scoreValueMineralsLostTechnology": 0,
      "scoreValueMineralsUsedActiveForces": 0,
      "scoreValueMineralsUsedCurrentArmy": 0,
      "scoreValueMineralsUsedCurrentEconomy": 1050,
      "scoreValueMineralsUsedCurrentTechnology": 0,
      "scoreValueMineralsUsedInProgressArmy": 0,
      "scoreValueMineralsUsedInProgressEconomy": 0,
      "scoreValueMineralsUsedInProgressTechnology": 0,
      "scoreValueVespeneCollectionRate": 0,
      "scoreValueVespeneCurrent": 0,
      "scoreValueVespeneFriendlyFireArmy": 0,
      "scoreValueVespeneFriendlyFireEconomy": 0,
      "scoreValueVespeneFriendlyFireTechnology": 0,
      "scoreValueVespeneKilledArmy": 0,
      "scoreValueVespeneKilledEconomy": 0,
      "scoreValueVespeneKilledTechnology": 0,
      "scoreValueVespeneLostArmy": 0,
      "scoreValueVespeneLostEconomy": 0,
      "scoreValueVespeneLostTechnology": 0,
      "scoreValueVespeneUsedActiveForces": 0,
      "scoreValueVespeneUsedCurrentArmy": 0,
      "scoreValueVespeneUsedCurrentEconomy": 0,
      "scoreValueVespeneUsedCurrentTechnology": 0,
      "scoreValueVespeneUsedInProgressArmy": 0,
      "scoreValueVespeneUsedInProgressEconomy": 0,
      "scoreValueVespeneUsedInProgressTechnology": 0,
      "scoreValueWorkersActiveCount": 12
    }
  },
  ...
]
```

Listing 18: Example PlayerStats tracker event JSON object.

```
[
  ...
  {
    "evtTypeName": "UnitTypeChange",
    "id": 4,
    "loop": 15,
    "unitTagIndex": 218,
    "unitTagRecycle": 1,
    "unitTypeName": "Egg"
  },
  ...
]
```

Listing 19: Example UnitTypeChange tracker event JSON object.

```
[
  ...
  {
    "controlPlayerId": 1,
    "evtTypeName": "UnitBorn",
    "id": 1,
    "loop": 652,
    "unitTagIndex": 238,
    "unitTagRecycle": 1,
    "unitTypeName": "Drone",
    "upkeepPlayerId": 1,
    "x": 23,
    "y": 17
  },
  ...
]
```

Listing 20: Example UnitBorn tracker event JSON object.

```
[
  ...
  {
    "evtTypeName": "UnitTypeChange",
    "id": 4,
    "loop": 652,
    "unitTagIndex": 203,
    "unitTagRecycle": 1,
    "unitTypeName": "Larva"
  },
  ...
]
```

Listing 21: Example UnitTypeChange tracker event JSON object.

```
[
  ...
  {
    "evtTypeName": "UnitDied",
    "id": 2,
    "killerPlayerId": null,
    "killerUnitTagIndex": null,
    "killerUnitTagRecycle": null,
    "loop": 652,
    "unitTagIndex": 203,
    "unitTagRecycle": 1,
    "x": 23,
    "y": 17
  },
  ...
]
```

Listing 22: Example UnitDied tracker event JSON object.

```
[
  ...
  {
    "evtTypeName": "UnitPositions",
    "firstUnitIndex": 265,
    "id": 8,
    "items": [
      0,
      50,
      27,
      14,
      49,
      26,
      3,
      49,
      28,
      9,
      48,
      28,
      18,
      50,
      26,
      11,
      50,
      27,
      21,
      55,
      23
    ],
    "loop": 6000
  },
  ...
]
```

Listing 23: Example UnitPositions tracker event JSON object.

## References

- [1] T. Gebru, J. Morgenstern, B. Vecchione, J. W. Vaughan, H. Wallach, H. Daumé, and K. Crawford, “Datasheets for Datasets,” 2018. [Online]. Available: <https://arxiv.org/abs/1803.09010>
- [2] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, “PyTorch: An Imperative Style, High-Performance Deep Learning Library,”

- in *Advances in Neural Information Processing Systems* 32, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds. Curran Associates, Inc., 2019, pp. 8024–8035. [Online]. Available: <http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf>
- [3] W. Falcon and The PyTorch Lightning team, “PyTorch Lightning,” 3 2019. [Online]. Available: <https://github.com/PyTorchLightning/pytorch-lightning>