

消防局員を対象とした、救急ビッグデータ 分析・活用プラットフォームの開発に関する研究

神戸大学大学院システム情報学研究科 中村匡秀 masa-n@cs.kobe-u.ac.jp

令和3年9月10日

概要

本稿は令和2年度より行った消防局員を対象とした、「救急ビッグデータ分析・活用プラットフォームに関する研究」の報告書である。本研究は消防局員が利用することを前提とした救急ビッグデータを分析し、実際の業務に利活用を進めるためのアプリケーションプラットフォームの構築が目的である。本研究では特に我々が救急ビッグデータと定義している、救急車・消防車の出動記録を元とした分析に焦点を当て、地図上へ救急車の動態を可視化するシステムADR (Ambulance Dispatch Reviewer)・年間、月間を通しての出動状況を可視化し、過去の出動状況の視覚的振り返りが行えるシステムASR (Ambulance Service Reviewer)・消防署や消防署に配備されている車両を仮想的に配置し、その結果駆け付け時間がどのように変化するかをシミュレート可能なシステムFD-CAST (Fire Department Configuration Analysis and Simulation Tool)、過去の出動記録を元に直近の熱中症患者の発生数を予測するシステムHSP (HeatStroke-Prediction)の4つのシステムを開発した。本報告書ではこれら4つのシステム開発について報告を行う。

研究体制

本研究は下記の通り1名の研究代表者と、共同研究者として神戸市消防局より6名、研究協力者として4名が参画している。

研究代表者

- 神戸大学大学院システム情報学研究科 准教授 中村匡秀

共同研究者

- 神戸市消防局 総務部総務課企画係 松田圭太・加百正人
- 神戸市消防局 警防部救急課救急係 藤田浩一・福田直人
- 神戸市消防局 警防部警防課計画係 山口盛司・村上大介
(所属は令和元年申請時のものである)

研究協力者

- 高知工科大学データ&イノベーション学教室 准教授 佐伯幸郎
- キー・ポイント株式会社 矢吹直也
- 神戸大学大学院システム情報学研究科 博士課程前期 中井哲也
- 神戸大学大学院システム情報学研究科 博士課程前期 松場建都

1 救急出動記録の時空間情報への変換による救急車の動態可視化

1.1 概要

自治体に所属する救急隊は、24時間365日、市民の救急要請に応じて、現場に駆け付け、病院に搬送する業務を担っている。近年、我が国の超高齢化に伴い、救急隊の出動要請は年々増加している。一方で、自治体の財政を考えると、救急隊の大幅な拡充は見込めない。そこで、増え続ける出動要請に対し、限られた救急隊のリソースをいかに効率よく割り当てるかが救急行政を考える上での重要な命題であり、そのためには、日々の救急要請や出動の記録を分析し、時間によって変化する需要や出動の実態を、きちんと把握する必要がある。

最近の消防局では最新のICTシステムを用い、消防や救急の様々なデータを記録・管理する取り組みが始まっている。例えば、神戸市消防局では消防救急情報システムが導入されており、司令部および救急隊が実施する各種のイベントがリアルタイムに記録されている。具体的には、各出動に対して、119番通報があった時刻(覚知時刻)や、救急車が出動した時刻(出場時刻)、現場の住所や、搬送先病院、患者の疾病の詳細等が記録されている。これらのデータに基づくと、各救急車がいつ、どこで、どのような状態(これを動態と呼ぶ)にあったかを詳細に振り返ることができる。

蓄積されていく膨大なデータを、消防局職員自らが分析するには相当のデータ分析スキルが必要である。したがって、救急車の過去の動態を詳細に理解、分析するためには多大な学習コストと時間が必要となり、現状の業務の合間に、こうした分析を行うことは非常に困難である。そのため、技術を持たない消防局職員でも簡単に振り返ることができるシステムが必要となる。

そこで、本研究では、データ分析の専門知識を持たない消防局職員でも、手軽に過去の救急車動態を把握できるようにすることを目的とする。この目的の実現のために、Ambulance Dispatch Reviewer(ADR)を開発する。ADRは、蓄積されたデータをもとに、過去の任意の時刻での救急車のおおよその位置と動態を算出し、地図上に分かりやすく可視化するシステムである。ADRの実装を行うために、以下の2つの技術的課題に取り組む。

- **P1:** 任意の時刻における救急車の位置と動態の計算
- **P2:** 救急車の時間、空間両方を考慮し、様々な条件で可視化する手法の開発

これらの課題を解決するために、以下の2つのアプローチを図る。

- **A1:** 救急出動記録の時空間情報への変換

情報システムが記録した各種イベントの発生時刻の間を内挿し、毎分の救急車の動態を表す時系列データを生成する。また、救急車が所属する消防署、駆け付け現場の住所、病院の住所等の緯度経度情報を統合し、毎分の救急車のおおよその位置を表す時系列データを生成する。これにより、離散的な出動記録データが連続的な時空間情報へ変換される。

- **A2:** 時空間情報の可視化

APIで取得した時空間情報を地図上に可視化する。変換した時空間データを指定された時刻で取り出し、地図上に可視化する。救急車のおおよその位置を緯度経度に基づいて配置し、その時刻での動態に合わせて色付けされたアイコンで表示する。また、駆け付けの現場や病院も表示し、出動の状況を振り返る助けとする。

本研究では、これらの2つのアプローチに基づき、ADRを実装した。また、共同研究先である神戸市消防局の7年間、約60万件の出動データをシステムに入力し、実際の出動動態を可視化した。開発したADRは、現在消防局職員に利用してもらい、高度なデータ分析の知識を必要とせずに過去の動態を容易に振り返ることができることを確認している。

1.2 準備

1.2.1 救急出動の現状

図1に、神戸市消防局の過去数年間の救急出動件数と搬送者数の推移を示す。神戸市消防局の発表によると、平成29年の救急出動件数は83081件で、前年に比べて2222件(2.7%)増加し、搬送者数についても70217人で1620人(2.4%)増加している[14]。また、図2に全国での現場到着所要時間及び病院収容所要時間の推移を示す。総務省の発表によると、平成28年の現場到着所要時間は全国平均で8.5分で、平成8年と比べて2.5分増加している[15]。

搬送者の内訳を年齢別でみたものが図3である。高齢者の割合は60%にもおよび、搬送者数の増加に大きく影響している。

1.2.2 救急出動記録

各救急出動について、日時、場所等を記録したもののことを指す。今回の研究で使用するデータは、神戸市消防局が記録しているデータの中から、研究に必要な項目を抜き出したものであり、本来のデータには傷病者についての情報や、現場につい

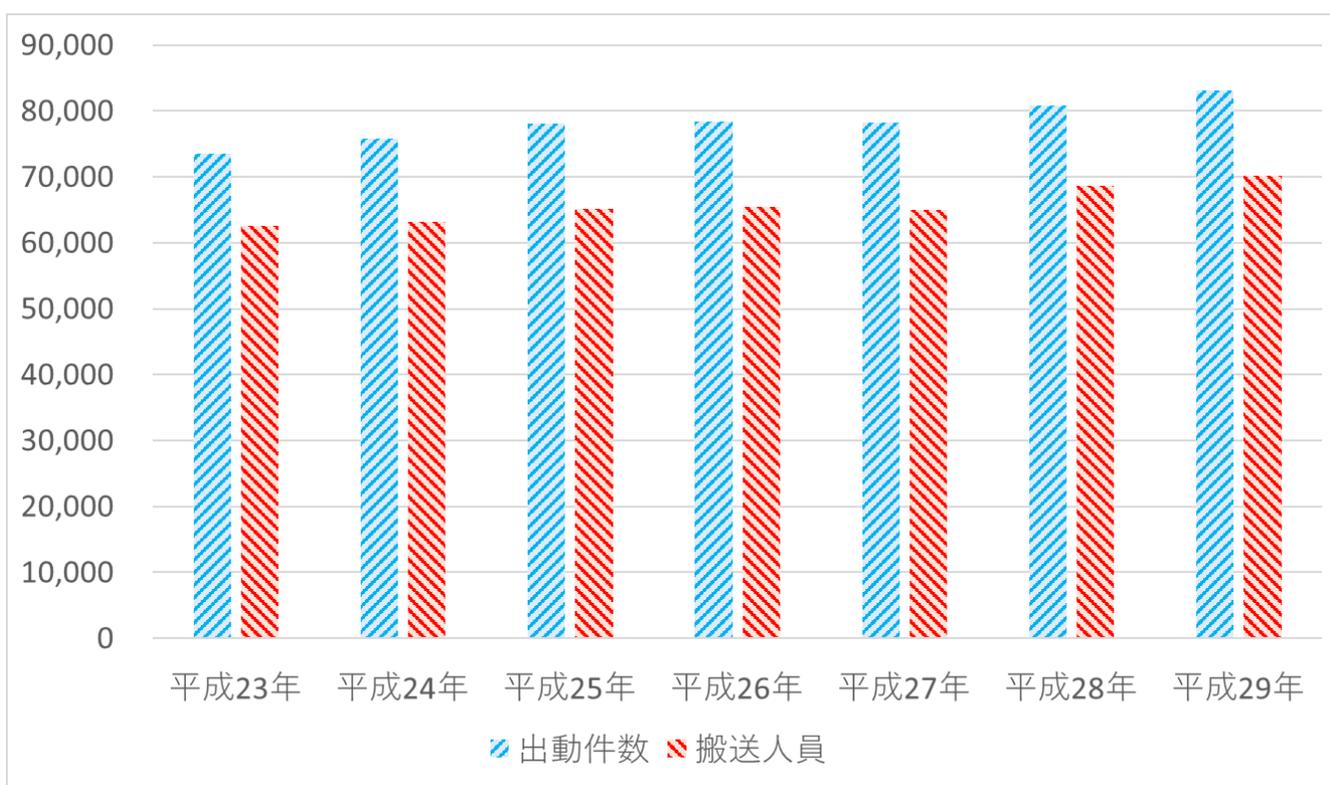


図1 救急出動件数と搬送人員の推移

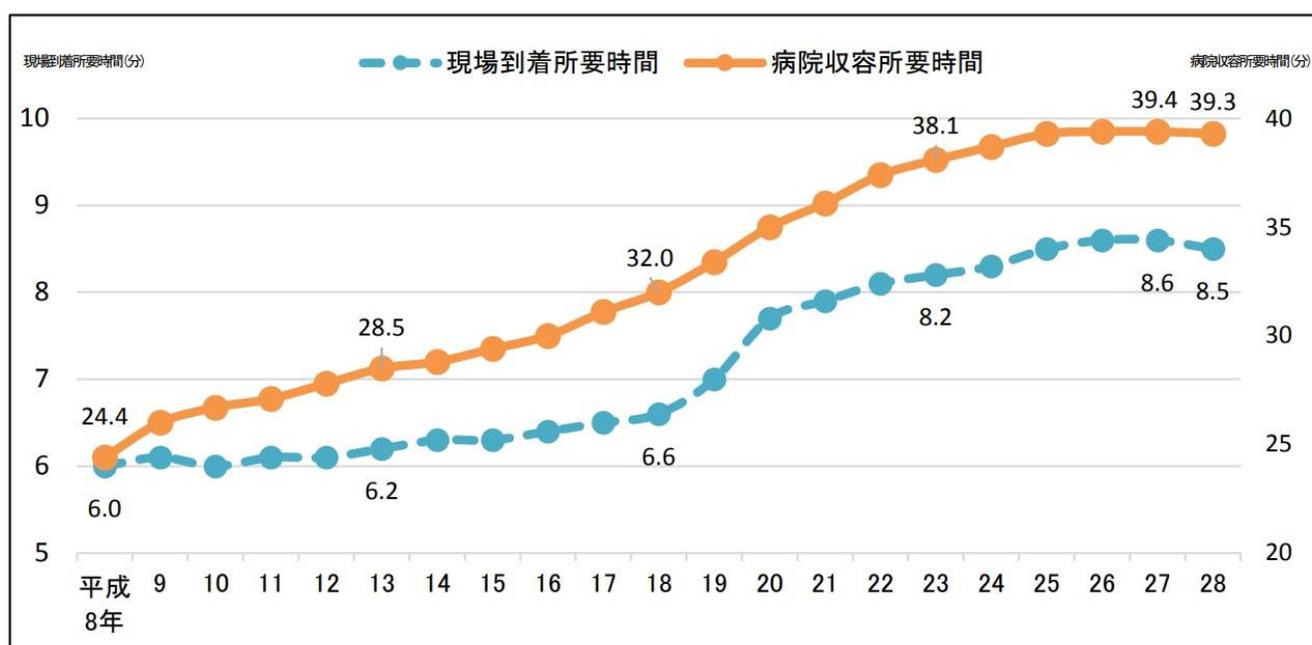


図2 平均駆けつけ時間、および平均搬送時間

てのより詳しい情報などが含まれている。なお、傷病者が軽傷であった場合等、病院への搬送を行わない（これを不搬送という）こともある。この場合は、上記のうち現発時刻から医師引渡時刻までが存在せず、現場を出発した時刻が病発時刻となる。

1.2.3 救急車の動態

各救急車が、現在どのような状態にあるのかを表すものである。表1に、動態の種類を示す。実際の現場では、これらの動態は出動できるかどうかの判断等に使用されている。これらの動態が切り替わるのは、各地点へ到着、もしくは出発した瞬間である。

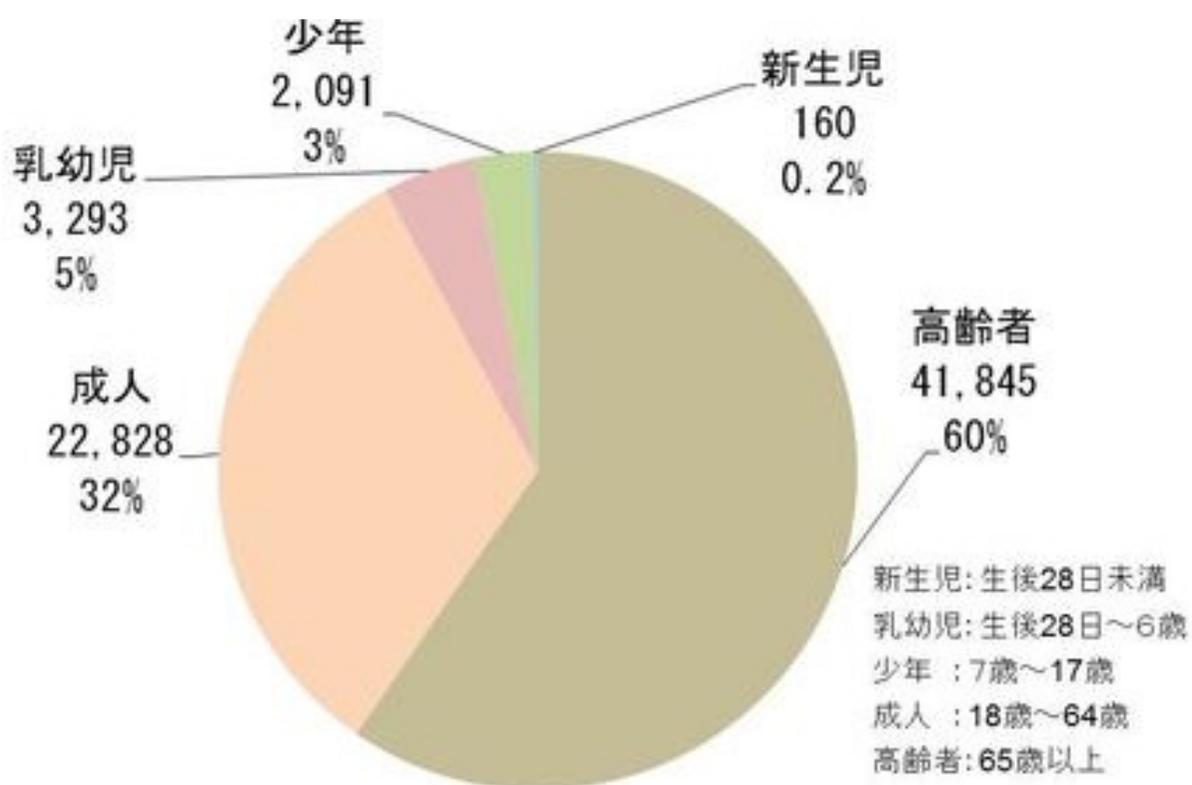


図3 救急搬送された人々の年齢

表1 救急車の動態

動態	内容
帰署	消防署に待機している状態
出動	通報を受け、現場へと向かう途中の状態
現着	現場に到着し、処置を行っている状態
現発	現場を出発し、病院へと搬送している状態
病着	病院へ到着し、傷病者を引き渡している状態
引揚	引き渡しを終え、消防署へ戻る途中の状態

1.2.4 消防局の要望

現在、神戸市消防局では大まかに分けると以下のような要望がある。

1. 過去の出動の様子をわかりやすく見たい
2. 異常な出動があった場合、その原因を把握したい
3. 最適な出動戦略を考えたい

この研究では、特に1に焦点を当てている。1をより詳しく分けると、以下のようになる。

1. 地図上で、ある時刻の救急隊の出動状況を見る
2. その時刻の駆けつけ時間や出動隊数など、異常値の発見につながるデータを見る
3. 地域ごとに分けた出動状況を見ることで、需要と供給が釣り合っているか見る

今回は、これらの要望を解決する。

1.3 提案システム

本研究では、救急出動記録を時空間情報へと変換し、それを地図上に可視化するシステムを提案する。本節では研究の目的と提案手法について述べる。

1.3.1 研究の目的と全体アーキテクチャ

本研究の目的は、データ分析の専門知識を持たない消防局職員でも、手軽に過去の救急車動態を把握できるようにすることである。これを達成するために、蓄積された膨大な出動記録データをもとに、各救急車がいつ、どこで、どのような動態にあったかを地図上に可視化するシステムの開発を行う。

図4に提案システムのシステムアーキテクチャ図を示す。まず1で、救急出動記録の各種イベントの発生時刻の記録から、毎分の位置と状態を計算し、連続したデータを作成する。

次に2では、1で作成したデータを様々な条件で検索するためのインターフェース群を開発する。これにより、救急車の位置情報や状態のデータを、日時や出動番号を条件に検索することが可能になる。

最後に3では、2で作成したインターフェースを用いて取得した時空間情報を地図上に可視化する。救急車や現場、病院の位置をアイコンで示し、時間を進めることでアニメーション化できる。

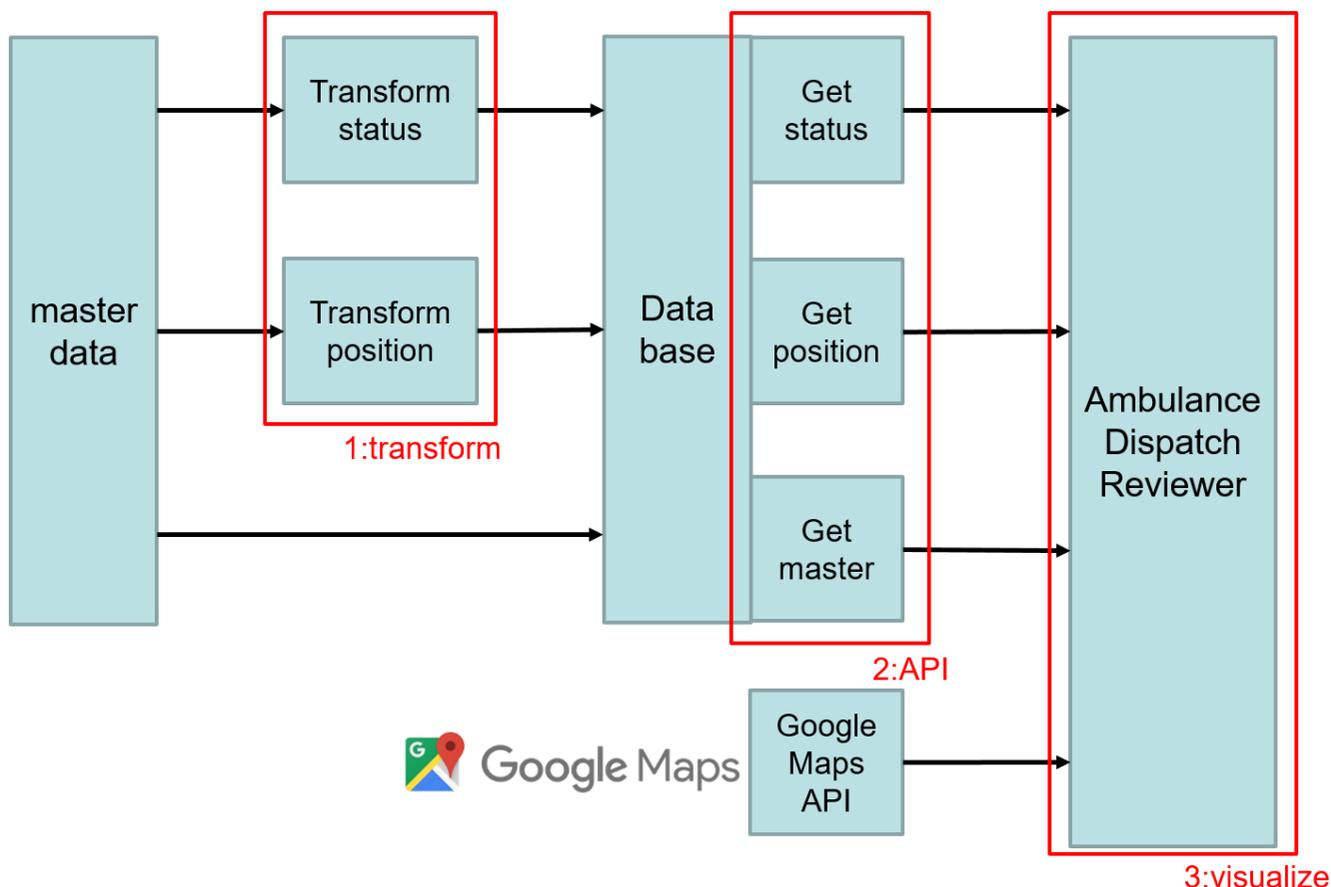


図4 システムアーキテクチャ

1.3.2 救急出動記録の時空間情報への変換

消防局が記録しているデータは、各出動案件についての覚知時刻や出場時刻といった、イベントの発生したタイミングのみの情報である。これをアニメーション化するためには、その間の時間にどこにいたのか、何をしていたのかという情報が必要である。本節ではどのようにして連続したデータへと変換するかについて決定する。

- 動態の計算：まず、各動態が変化するのは、表2のとおりである。
ここで問題となるのは、不搬送の場合である。不搬送の場合、現発時刻と病着時刻は存在しない。そこで今回、不搬送の場合は現場=病院とみなし、現発時刻と病着時刻は病発時刻と同じとし、病院座標も現場座標と同じとした。
この動態の変化のタイミングの間は、変化前の動態が続いているはずである。よって、帰署→出動の前は帰署、出動→現着の前は出動というようにする。
- 位置の計算：座標については、実際に通った道についての情報が記録されていないという問題がある。そこで、救急車は消防署から現場、現場から病院、病院から消防署を直線状に、一定速度で移動したと仮定する。具体的には、消防署、現場の座標をそれぞれ P_f , P_s , 現着時刻, 出場時刻, 現在時刻をそれぞれ T_a , T_d , T_c とすると、消防署から現場へ移動中

表 2 動態変化の瞬間

動態の変化	タイミング
帰署→出動	出場時刻
出動→現着	現着時刻
現着→搬送	現発時刻
搬送→病着	病着時刻
病着→引揚	病発時刻
引揚→帰署	帰署時刻

の座標であれば以下の計算式となる。

$$(P_f) + \frac{(P_s) - (P_f)}{(T_a) - (T_d)} * \{(T_c) - (T_d)\}$$

これらを 1 分間隔で計算を行い、5 年分の各時刻の位置情報と動態が記録されたデータを作成する。

1.3.3 時空間情報に問い合わせる API の開発

1.3.2 で作成したデータに、外部からアクセスするために、データを様々な条件、例えば時刻等で検索できる API を配備する。これらの API を web サービスとしてデプロイし、HTTP を介してデータの受信を行う。これらの API を用いることで、ある時刻の救急車全ての位置情報と動態のデータを取得することができる。以下に作成する API の詳細を示す。

- getMasterByNumbers(numbers)
出動番号を複数指定することで、その出動番号の救急出動記録を取得する。
- getMasterByDateTime(datetime)
日時を指定することで、その日時に出勤中のデータ全てを取得する。
- getStatus(datetime)
日時を指定することで、その日時の全救急隊の動態を取得する。
- getPosition(datetime)
日時を指定することで、その日時の全救急隊の位置を取得する。

getStatus と getPosition は、1.3.2 で作成したデータを取得するために必要となり、getMasterByNumbers と getMasterByDatetime は可視化の際に情報を追加するために必要となる。

1.3.4 時空間情報の可視化

最後に、1.3.3 で作成した API を利用して取得したデータを可視化する。今回は、座標と時間を同時に可視化する方法として、地図上への表示を行う。まず、ユーザーが日時を指定すると、その時刻のデータを取得する。次に、取得した動態からアイコンの色を決定し、取得した座標にそのアイコンを表示する。図 5 に、使用するアイコンを示す。消防局での実際の基準に基づき、灰色のアイコンが帰署、黄緑色のアイコンが引揚、赤色のアイコンがその他の動態（出勤中の状態）を表す。また、駆けつけ現場を電話のアイコン、搬送先を病院のアイコンで表す。一定時間ごとにその 1 分後のデータを取得し、アイコンを更新することで、救急車の出勤の様子をアニメーション化することが可能となる。

また、1.2.4 で述べた要望に応じて、統計情報も計算し表示する。表 3 に統計情報に含まれる内容を示す。ここで、駆けつけ時間とは覚知時刻から現着時刻までの間、搬送時間とは現発時刻から病着時刻までの間、出勤時間とは出場時刻から帰署時刻までの間とする。また、駆けつけ時間が一定を超える部隊を表示する理由は、一定を超える場合は通常時に比べて著しい遅延が発生していると考えられているからであり、神戸市の場合この基準を約 13 分としている。

さらに、地域ごとの出勤部隊数と、出勤中の部隊の駆けつけ時間の平均も計算し、表示する。これにより、どの地域での出勤が多いのか、駆けつけに必要な時間が長いのはどの地域か、確認することができる。以下に、表示までの手順を示す。

1. 日時を指定する
2. 指定した日時の位置データを取得する
3. 指定した日時の動態データを取得する
4. 3 で取得した動態データのうち、帰署以外のものの出動番号を取り出してリストにする

5. 4で作成したリストの出動番号の救急出動記録を取得する
6. 5で取得したデータから現場の情報と病院の情報を取り出す
7. 救急車のマーカーを作成し、動態に応じて色分けする
8. 現場と病院のマーカーを作成し、表示する
9. 統計情報を計算し、表示する



図5 救急車のアイコン

表3 統計情報

項目	内容
出動台数	出動中の部隊の数
平均駆けつけ時間	出動中の部隊の駆けつけ時間の平均
最大駆けつけ時間	出動中の部隊の駆けつけ時間の最大
平均搬送時間	出動中の部隊の搬送時間の平均
最大搬送時間	出動中の部隊の搬送時間の最大
平均出動時間	出動中の部隊の出動時間の平均
最大出動時間	出動中の部隊の出動時間の最大
駆けつけが一定を超える台数	

1.4 プロトタイプ実装

本研究では 1.3 で述べた提案手法に基づき実装を行った。具体的には、救急出動記録を 1 分毎の救急車の位置情報と動態情報へと変換した。さらに、これらのデータを取得するための API を配備し、取得したデータを用いて地図上への可視化を行った。以下の小節では実装の詳細について述べる。

1.4.1 実装方法

まず、1.3.2 で述べた方法で、救急出動記録を時空間情報へと変換する言語は python を使用した。

次に作成したデータをデータベースに保存した。このデータベースは MySQL を利用した。

さらに、データベースに保存したデータを取得するために、1.3.3 で述べた API を配備した。これらの API は php で実装を行った。

最後に、取得したデータを地図上へ可視化するアプリケーションの開発を行った。表示する地図は Google Maps API[6] を利用した。また、表示システムは HTML5 と Javascript で実装し、Apache をサーバーとして公開している。

1.4.2 神戸市消防局のデータの可視化

神戸市消防局での 2013 年から 2020 年のデータおよそ 60 万件を用いて、可視化を行った。図 6 に全体の画面を示す。以下では、この画面の詳細について述べる。

- 項目選択：図 6 のうち、表示する項目を選択する部分を拡大したものを図 7 に示す。まず、どの救急隊を表示したいかチェックボックスで指定する。次に、表示したい日時を入力し、Go ボタンを押すことでその瞬間の状況を表示、Play ボタンを押すことで再生することができる。また、再生中に Pause ボタンを押すことで停止したり、Rewind ボタンを押すことで巻き戻すことも可能となっている。さらに、更新間隔、進める時間のバーを調節することで、アニメーションの速度を変更することもできる。
- 地図表示：図 6 のうち、出動の様子を表す部分を拡大したものを図 8 に示す。各救急隊のアイコンをクリックすることで、救急隊名、出動番号、動態を確認することができる。また、病院や駆けつけ現場のアイコンをクリックすることで、病院名とどの救急隊が向かっているかを確認することができる。さらに、i アイコンをクリックすることで、地域ごとの統計情報を表示することができる。このグラフについては後述する。
- 統計情報：図 6 のうち、統計情報を表示する部分を拡大したものを図 9 に示す。ここで表示される統計情報は、1.3.4 で述べた項目となっている。

さらに、前述した地域別の統計情報の表示を拡大したものを図 10 に示す。左のグラフが出動隊数、右のグラフが駆けつけ時間を表している。最大値はそれぞれ総部隊数とその地域の全出動データにおける平均駆けつけ時間の 2 倍になっており、この 50% を超えると黄色、75% を超えると赤色となる。

1.4.3 開発環境

本研究の開発環境は以下のとおりである。

- 開発言語: python, Javascript
- データベース: MySQL 5.7.23
- Web サーバー: Apache 2.4.34
- API: PHP 7.0.27
- 可視化システム: HTML5, Google Maps API

1.5 評価

実装したシステムを利用し、神戸市消防局の 2013 年から 2020 年のデータを用いて可視化を行った。

1.5.1 実行時間

まず、1.4.1 で行った変換を事前に行うことの効果があるかどうかの確認として、救急出動記録をそのまま利用して可視化するシステムを作成し、動作速度の比較を行った。結果、ある時刻の表示を 1 回行うまでにかかる時間は、事前に変換を行わな

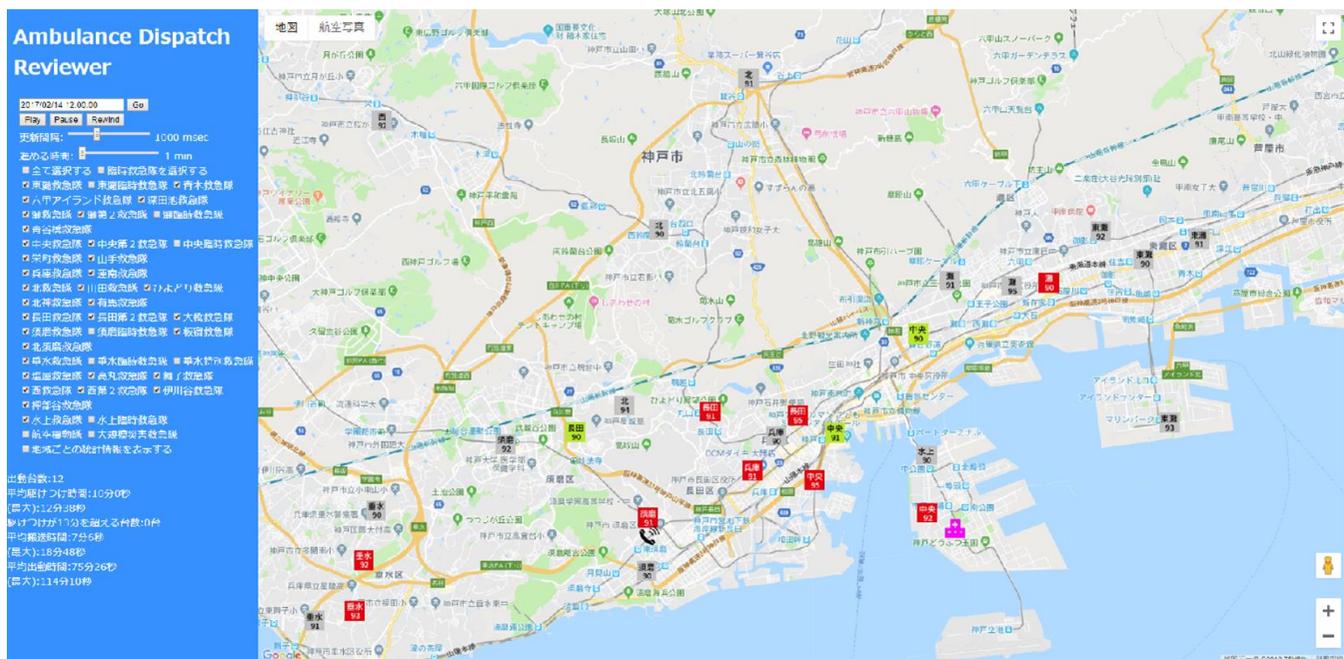


図6 Ambulance Dispatch Reviewer(ADR)



図7 選択メニュー

い場合は約 800ms だったのに対し、変換を行うことで約 20ms まで短縮することができた。この違いは、データを取得するためにかかる時間が大幅に短縮されたことによるものと判明した。

1.5.2 消防局職員による評価

また、1.4.2 で行った可視化の効果の確認として、消防局職員の方数名に利用していただいた。以下に、得られた評価を挙げる。

- C1: アプリにより、救急車の出動状況が可視化されたことは、非常に意義がある
- C2: 特に、区域ごとの救急車の出動状況が刻一刻と変わっていく状況を見ることができる機能は、非常に有効である
- C3: 今まで現場の感覚的なものでしかなかったことが、データをもとにここまで可視化できた点は非常に評価できる
- C4: 今後に向けて、過去だけでなく現在の状況も可視化したい
- C5: 状況を見るだけでなく、状況の原因を探る必要がある



図8 地図

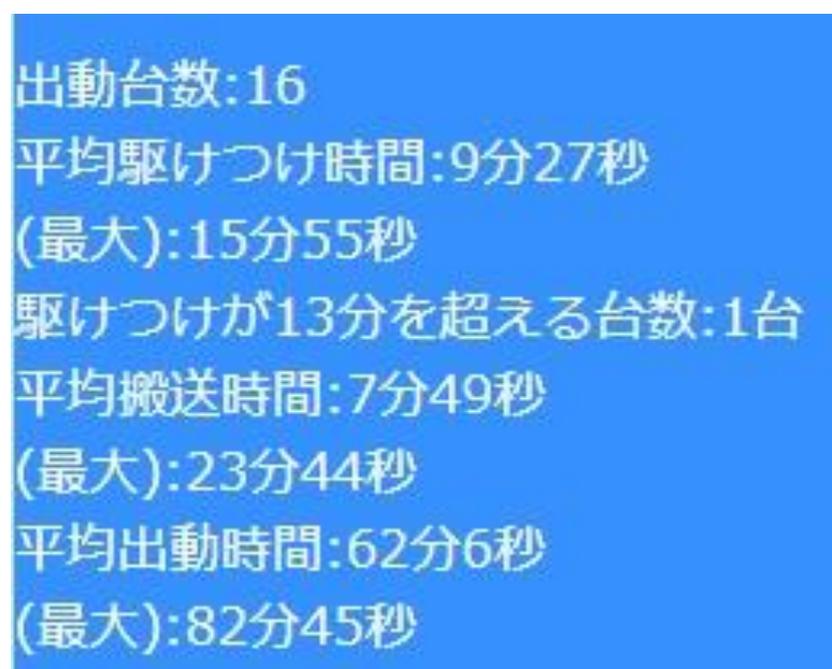
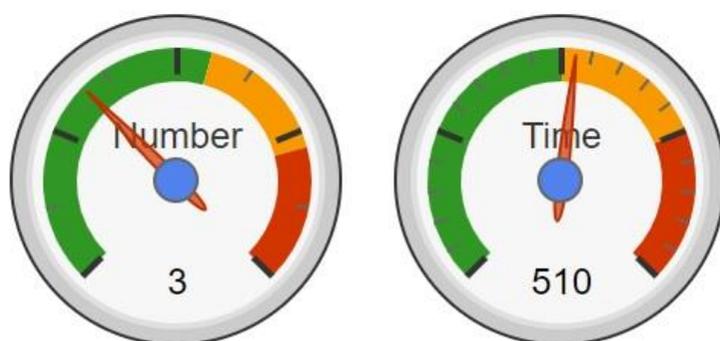


図9 全体の統計

1.5.3 考察

1.5.2 得られた評価を元に、考察を述べる。C1 から、本システムにより救急車の出動状況が誰にでも見られる形で可視化することができたと考えられる。また C2 から、各区域の状況を個別に確認でき、また時間経過により状況が変化する様子も確認

東市街地



平均駆けつけ時間:8分30秒

図 10 地域別の統計

できるようになったと考えられる。さらに C3 から、これまで、出動の逼迫状態は現場の人間の経験に基づく感覚でしか分からなかったが、可視化を行ったことで誰にでも分かる形で示せるようになったと考えられる。しかし C4 を達成するには、現在消防局で利用されている消防救急情報システムとの連携が必要である。このことから、現状この課題を解決する事は困難となっている。また、C5 で挙げられたように、本システムでは、出動状況を見ることはできるが、その状況が起きた原因が何かまでは見ることができない。しかし、その状況になった日時・場所を参考にすることで、起きたイベントや気候などを調べ、原因を判断できると考えられる。

以上より、本システムは状況の原因までは確認できないが、過去の救急車の出動状況を誰にでも分かりやすい形で可視化できたと言える。

2 様々な時間粒度で救急活動を振り返るアプリケーション

2.1 概要

ADR を用いることで、振り返りの可視化は実現できたが、消防局での試用とそのアンケート結果から実際の振り返りでの利用について、以下の点に対する指摘があった。

P1: いつ問題が起きたのかを探ることができない

P2: 問題が発生日時が、異常なのか確認することができない

P3: 救急隊全体でのひっ迫状況が分かりにくい

これらの指摘事項に対し、消防局と検討を行った結果以下 4 点の要求として整理された。

R1: 一年のうちいつ問題が起きたか知りたい

R2: 現状が他の日時、場所とどう違うか知りたい

R3: 一日のうちいつ部隊が動いたのか俯瞰したい

R4: R1~R3 で絞った日時を ADR で振り返りたい

そこで本研究では、年間、月間を通しての出動状況を可視化する Ambulance Service Reviewer(ASR) を開発する。ASR では以下 4 つのアプローチに基づき、出動が多い日、時間など、特徴的な日時を把握できるよう、各日時の出動状況を取得し、グラフとして表示する。

A1: 様々な時間粒度で出動を可視化

A2: 異なる年、地域と比較

A3: 同時刻での出動隊の状況を可視化

A4: ADR との連携

A1 では、3 つの時間粒度で出動を可視化する。最初に年間を通しての日単位のデータを表示するもの、次に月間での分単位のデータを表示するもの、最後に日間での分単位のデータを表示するものとなっている。これにより、まず年間を通した中から特異な時期を見つけられる。次にその時期の中からより特徴的な日を見つけ、さらに細かく見ることで、いつがどれだけ出動が多かったのか把握することが可能となる。

次に A2 では、複数の年、複数の地域のデータを同時に重ねて表示することができるようにする。ユーザーが見たい地域、もしくは見たい年を複数同時に選択することで、選択したものを示すグラフを重ねて表示する機能となる。これにより、その年の中で出動が多い日、少ない日が、例年と比較するとどうなのか、ということ調べることができる。また、全体でみると出動が多い場合でも、地域別でみると多い地域と少ない地域を調べることができる。

さらに A3 では、各時間ごとに出勤中の部隊を一目で確認することができるようにするため、各隊の 1 日の出勤をタイムラインで表示する。これを見ることで、特定の隊が頻繁に出勤している時間帯、および、多くの隊が出勤している時間帯を把握することができる。

最後に、A4 では ADR との連携を行う。A1~A3 を利用してどの日時を見るべきか発見したのち、その日時の様子を詳細に見る、ADR に接続する機能を付ける。

2.2 提案手法

ASR は、救急車の過去の出動数等をグラフに可視化するシステムである。図 11 に ASR の処理フロー図を示す。まず、救急出動記録から各日時の出動隊数を計算し、データベースに保存する。これを分単位で行った後、そのデータを利用して日単位でのデータを作成する。日単位のデータには、出動隊数×出動時間を計算したもの、その日の出動数の最大数を計算したもの、最小数を計算したのものがある。これを API を用いて取得し、web 上に横軸を時間、縦軸を出動数としたグラフ化して表示する。

次節以降では、2.1 で述べた各アプローチについて説明する。

2.2.1 A1: 様々な時間粒度で可視化

一年のうちいつ問題が起きたかを把握するための機能として、様々な時間粒度での可視化を行う。ASR のグラフの表示方法は三段階あり、最初に年間を通しての日単位のデータを表示するもの、次に月間での分単位のデータを表示するもの、最後に日間での分単位のデータを表示するものとなっている。これにより、まず年間を通した中から特異な時期を見つけられる。次

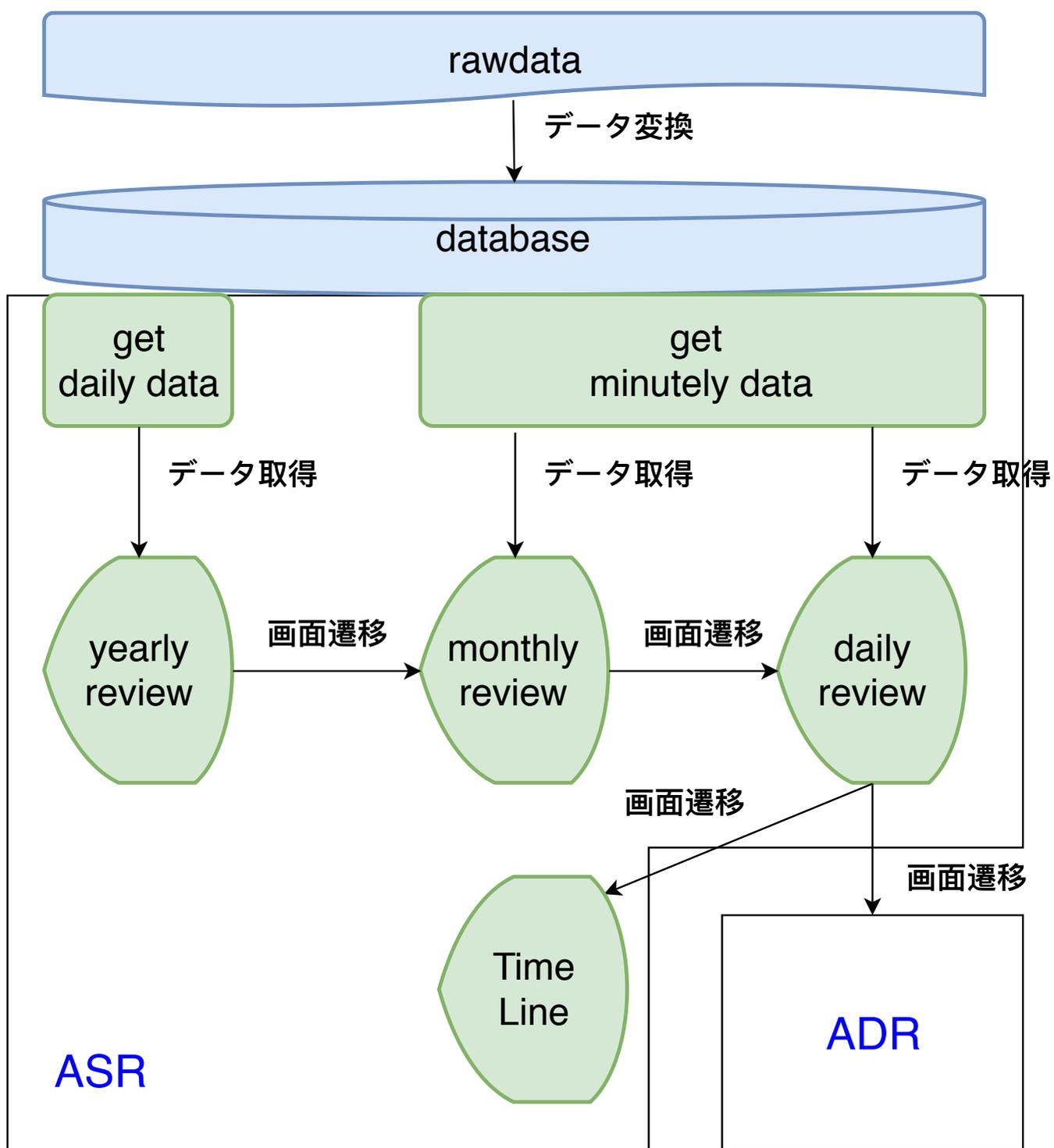


図 11 ASR の処理フロー図

にその時期の中からより特徴的な日を見つけ、さらに細かく見ることで、いつがどれだけ出動が多かったのか把握することが可能となる。

まず、年間表示の画面について説明する。図 12 に、画面のイメージを示す。年間表示画面では、指定した年の 1 年分のデータをグラフ化して表示する。初期状態では、最新の年のデータが表示されているが、ユーザは見たい年のデータを選択する。そして、どの地域の情報を見たいかを選択し、画面を更新する。この選択機能については 2.2.2 で詳しく述べる。ここでいう地域とは、表 4 に示すとおりである。表示されるグラフは、3 か月ごとに分けられており、1~3 月、4~6 月、7~9 月、10~12 月の 4 つが表示される。見たい項目に応じて、出動数の合計や、出動隊数の最大値、最小値などを表示することができる。もし、気になった日時があれば、その点を選択することでその日から 1 か月間のデータ表示画面に移動する。

次に、月間表示の画面について説明する。図 13 に、画面のイメージを示す。月間表示画面では、年間表示画面で選択した日から 1 か月分のデータをグラフ化して表示する。ここでも、年や地域を選択することで変更することができる。この画面で表示されるグラフは、1 週間ごとに分けられており、1~7 日、8~14 日、15~21 日、22~28 日、29 日~月末の 5 つのグラフが

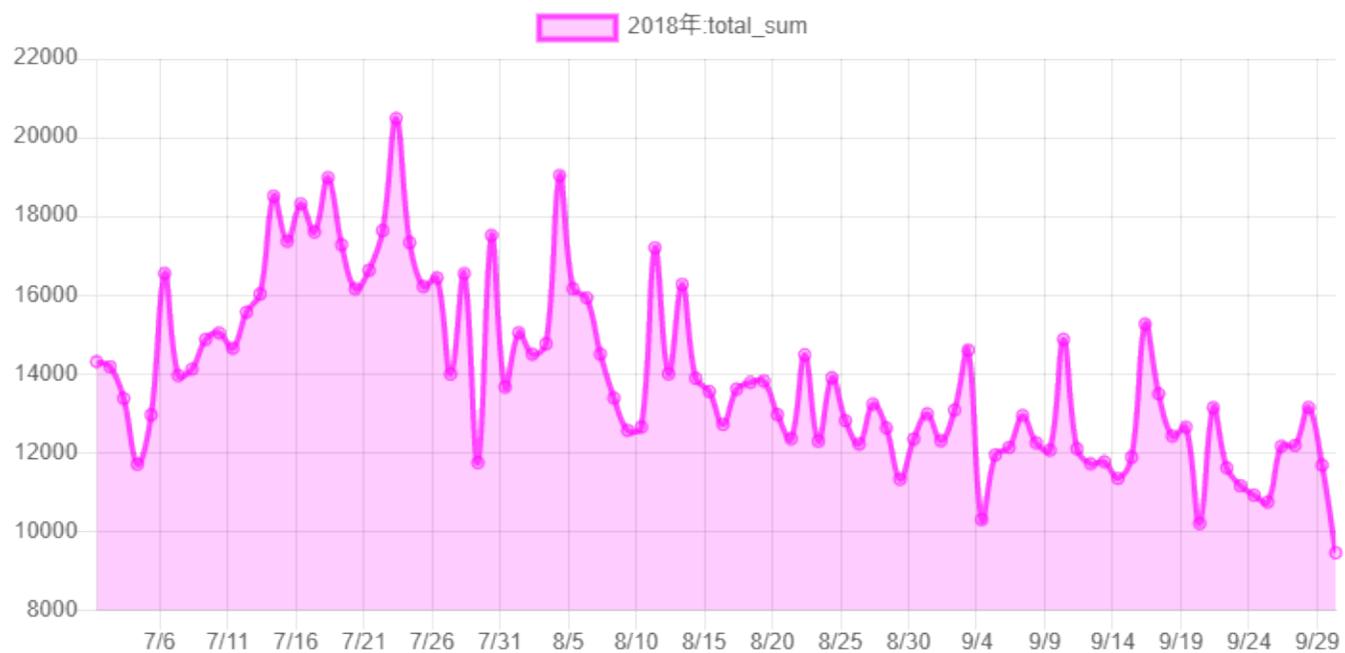


図 12 年間表示

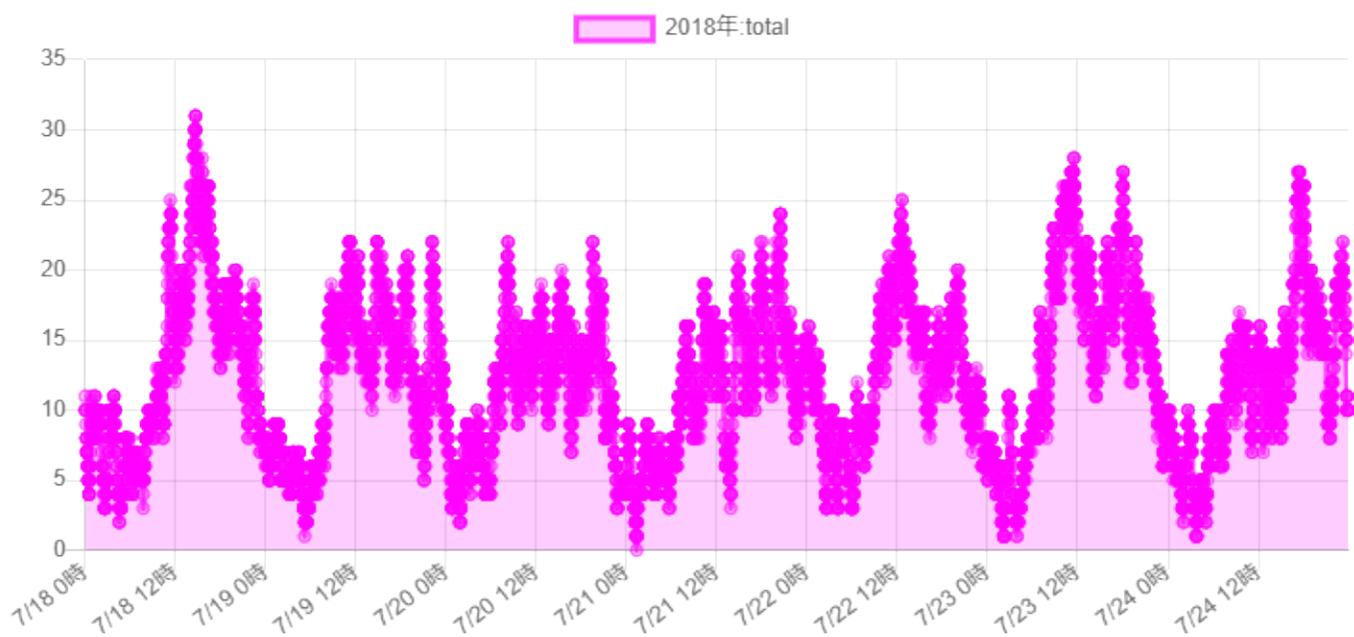


図 13 月間表示

表 4 神戸市の地域分け

地域名	区
市街地東	東灘区, 灘区, 中央区
市街地中央	中央区, 兵庫区, 長田区
市街地西	須磨区, 垂水区
北	北区
西	西区

表示される。この画面で気になった日時を選択すると、その日時を含む 1 日分のデータ表示画面に移動する。

最後に、日間表示の画面について説明する。図 14 に、画面のイメージを示す。日間表示画面では、月間表示画面で選択した

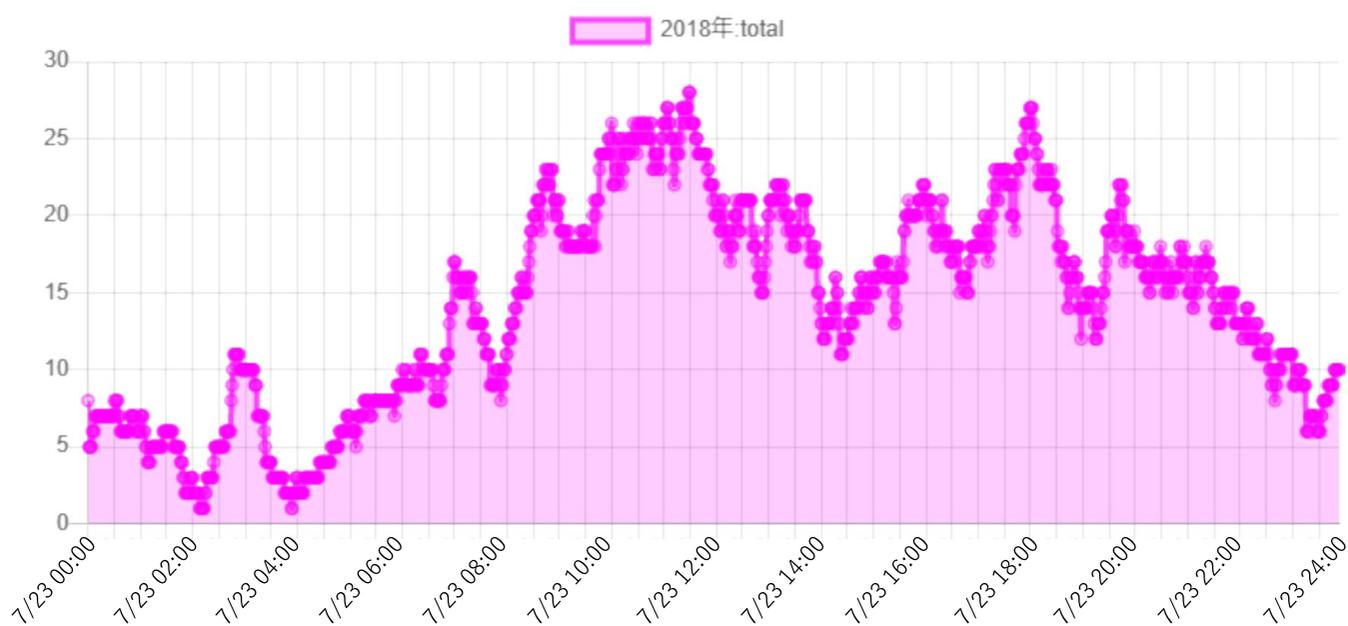


図 14 日間表示

2018年7月18日 (水)

(凡例) ■ 駆け付け, ■ 現場処置, ■ 病院搬送, ■ 病院引渡, ■ 引き揚げ

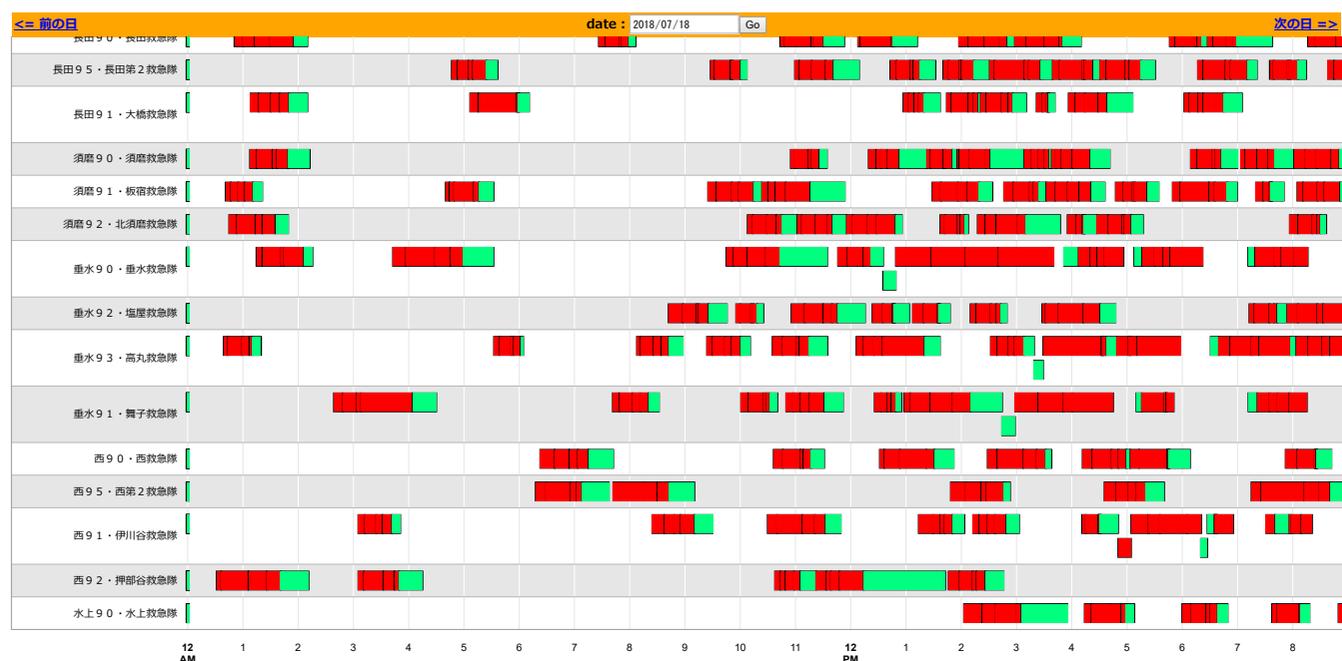


図 15 タイムラインを表示する画面

日 1 日分のデータをグラフ化して表示する。この画面では、一つのグラフに 1 日分のデータが表示される。

2.2.2 A2: 異なる年, 地域と比較

現状が他の日時, 地域とどう違うのかを把握するための機能として, 複数の年, 複数の地域のデータを同時に重ねて表示することができるようにする。

ユーザーが見たい地域, もしくは見たい年を複数同時に選択することで, 選択したものを示すグラフを重ねて表示する機能となる。これにより, その年の中で出勤が多い日, 少ない日, 例年と比較するとどうなのか, ということ調べることができる。また, 全体でみると出勤が多い場合でも, 地域別でみると多い地域と少ない地域を調べることができる。

2.2.3 A3: 同時刻での出動隊の様子を可視化

各時間ごとに出勤中の部隊を一目で確認することができるようにするため、各隊の1日の出勤をタイムラインで表示した。図15は、その画面の様子を示したものである。2.2.1で述べた日間表示の画面から、その日の様子を表すタイムラインへと移動することができる。

この画面では、横軸を時間として、その部隊が出動している時間を赤色で、出勤を終えて消防署に戻っている時間を黄緑色で、帰署中の時間を無色で表示する。この色分けは現場到着や病院到着などでより細分化したものと切り替えることができ、特定の動態に時間がかかったものを見分けることができる。

この画面では、表示したい部隊を選択しておくことで、表示を絞ることができる。

これを見ることで、特定の隊が頻繁に出動している時間帯、および、多くの隊が出動している時間帯を把握することができる。

2.2.4 A4: ADR との連携

この章で述べた機能によりどの日時を見るべきか発見したのち、その日時の様子を詳細に見るために、ADRに接続する機能を付ける。2.2.1で述べた日間表示の画面で、詳細を見たい日時の点を選択することで、その日時を表示するADRに移動することができる。

2.3 プロトタイプ実装

本研究では、2.2で述べた提案手法に基づき実装を行った。以下では実装の詳細について述べる。

2.3.1 実装

まず、救急出勤記録から、各日時の地域ごとの出勤隊数と合計出勤隊数を計算し、データベースに保存した。この操作はpythonを使用して行い、データベースはMySQLを利用した。

さらに、データベースに保存したデータをシステムで利用するために、指定した日時のデータを取得するAPIを作成した。このAPIはphpで実装を行った。

最後に、取得したデータからグラフを作成して表示するWEBアプリケーションの開発を行った。このアプリケーションはHTML5とJavascriptで実装を行い、グラフの表示にはChart.js[1]を利用した。このシステムは、Apacheをサーバーとして公開している。

2.3.2 使用データ

今回、神戸市消防局が記録している2013年～2020年の救急出勤記録を使用した。このデータはおよそ60万件あり、出勤時刻、出勤先住所、傷病者の詳細などが記録されている。今回の研究では表5の項目を利用した。

表5 使用したデータ項目

項目名	内容
出勤隊名	出勤した部隊名
出場日時	部隊が出動した日時
病発日時	部隊が搬送を終えて病院を出発した日時
帰署日時	部隊が消防署に戻った日時

2.4 評価

今回の提案に対する評価として、各要望を達成できたかどうかを評価する。

2.4.1 R1: 一年のうちいつ問題が起きたか知りたい

一年のうちいつ問題が起きたかを確認するため、今回は2018年で神戸市全体の出動が最も多かった日を分析する。まず、年間表示で2018年、全地域を選択し表示したものが図12である。これを見ると、7月23日が多かったことが分かる。さらに、7月18日から月間表示に移動したものが図13である。7月23日の最大同時出動数はその時期の他の日とあまり変わらないが、一日を通して出動が多かったことがわかる。そして、7月23日の日間表示に移動したものが図14である。この日は午前8時頃から午後2時頃にかけてと、午後3時半頃から午後6時半頃の長時間にわたって出動が多い状態が続いていることが分かった。

なお、この日に出動が多かった理由は主に熱中症患者の搬送が多かったためだと考えられている。
このように、出動が多かった日時を発見することができた。

2.4.2 R2: 現状が他の日時、場所とどう違うか知りたい

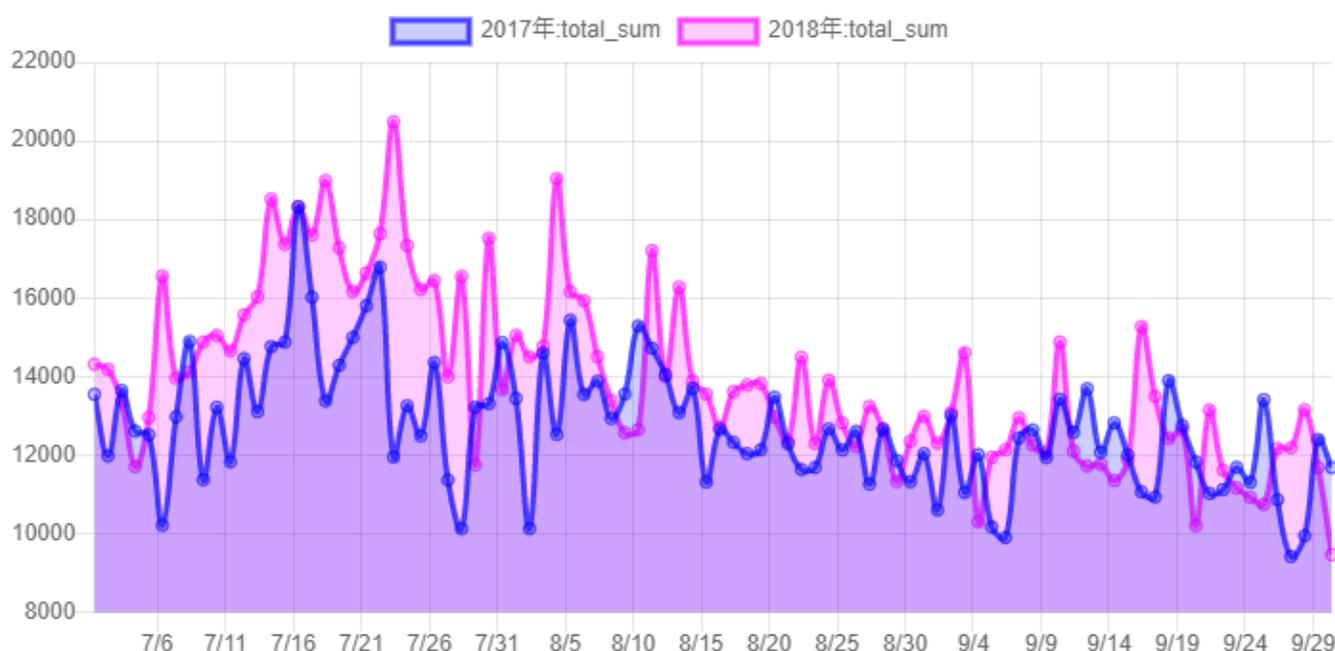


図16 年度ごとの出動の差の確認

現状が他の日時や場所とどう違うかを確認するため、今回は同じ時期の年度ごとの差と、特定の日時の地域ごとの差を分析する。

まず、年度ごとの差の分析を行う。図16は、年間表示の画面で2017年と2018年の7月から9月を表示したものである。青のグラフが2017年、桃のグラフが2018年を示す。この二年を比較すると、7月後半から8月前半にかけては9月と比較して出動が多くなりやすいことがわかるが、中でも2018年7月23日は突出して多いことがわかる。今回は二年のみでの比較を行ったが、実際には六年間で比較することも可能となっている。

次に、地域ごとの差の分析を行う。図17は、日間表示の画面で、2018年02月05日を表示したものである。赤のグラフが市街地東、緑のグラフが市街地西、青のグラフが西を示し、桃のグラフは全体の出動を示す。14時30分頃から17時頃にかけては、全体の出動数は15隊以上と多いのに対し、市街地東は最大7隊中3隊以下とあまり出動していないことが分かった。また、17時から18時にかけては、全体はほとんど15隊以上出動しているのに対し、西の出動数は0隊と地域差があることが分かった。

このように、現状が他の日時や場所とどう違うか確認することができた。

2.4.3 R3: 一日のうちいつ部隊が動いたのか俯瞰したい

R1で気になった日の一日の出動の様子を俯瞰できるかどうかの確認として、今回は2018年で最も最大出動隊数が多かった7月18日（31隊）の様子を見た。図15はタイムライン画面で7月18日の様子を確認したものである。この日は全体として出動している時間が長かった日だが、押部谷救急隊など一部の救急隊は比較的短いことがわかる。また、この日の14時から

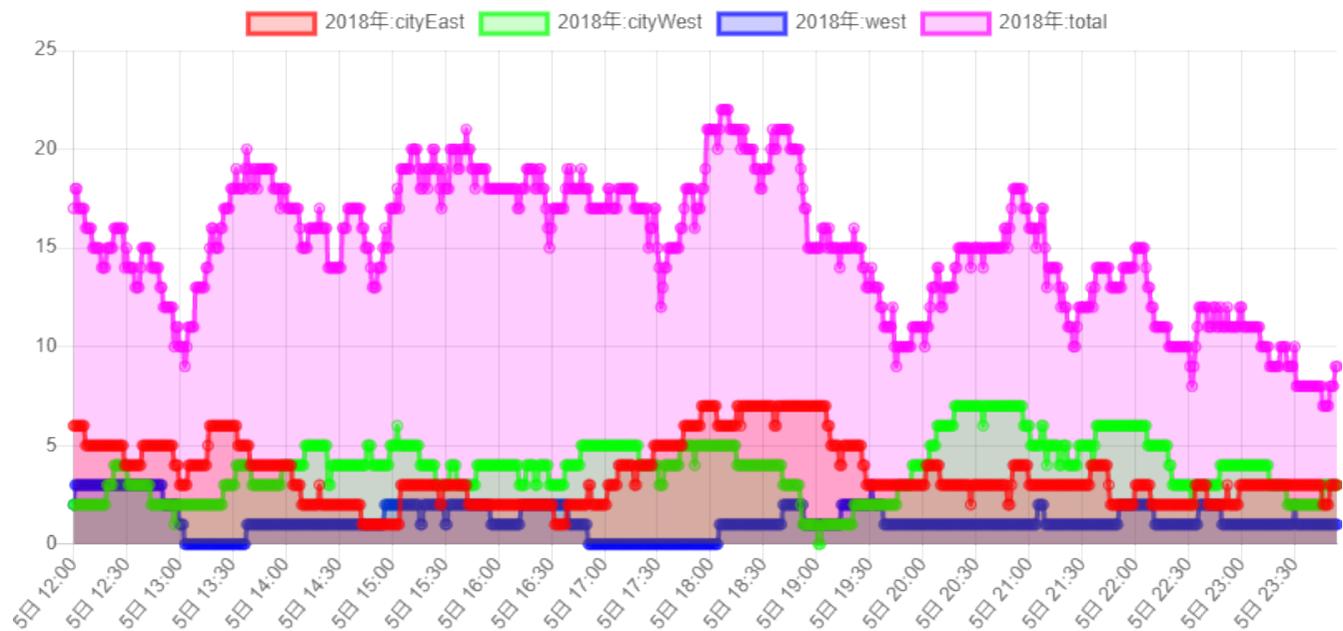


図 17 地域ごとの出動の差の確認

16 時頃は出動が密集しており，中でも長田第 2 救急隊や須磨救急隊は立て続けに出動していたことが分かった。このように，一日のうちいつ部隊が動いたのか確認することができた。

2.4.4 R4: R1~R3 で絞った日時を ADR で振り返りたい

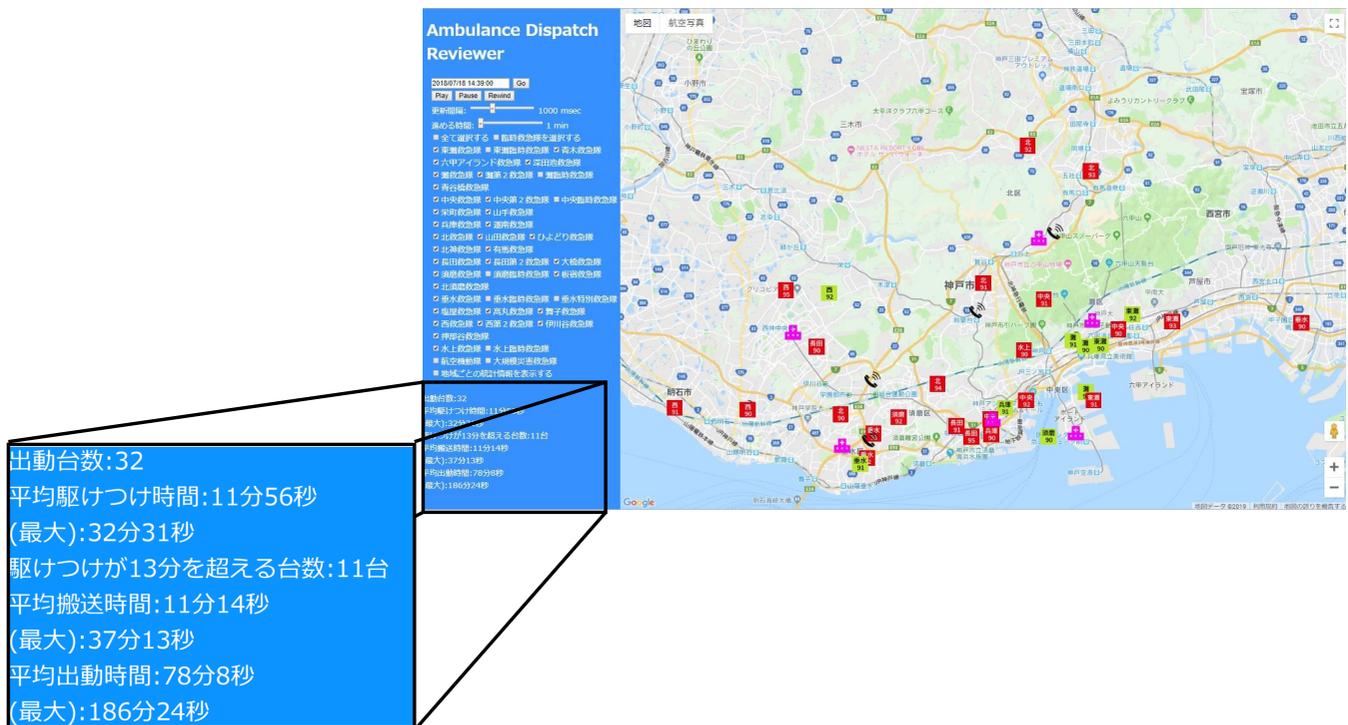


図 18 ADR での振り返り

R1~R3 で絞った日時を実際に ADR で見てみる。今回は，R3 で確認した 2018 年 7 月 18 日の様子を ADR で表示した。この日の中で最も出動が増えたのは 14 時 40 分で，この時刻はすべての救急隊が出動していることが確認できた。また，出動数が一定を超えると駆けつけ時間がのびることが分かっているが，実際に駆けつけ時間が 13 分以上と長くなっている隊が 11 隊と多く，危険な状態であることが確認できた。

2.4.5 消防局職員による評価

最後に、作成したシステムを神戸市消防局職員の方による実利用での評価を行った。その結果、良かった点としては、以下の点が挙げられた。

C1: 複数年分を重ねて見せられる点がわかりやすい

C2: 救急出動の逼迫状況を視覚的にとらえられ、注目すべきポイント（時期、時間帯）を見つけるのに有益である

C3: 年間表示、月間表示、日間表示と入っていけるのは詳細を見ていくにあたって使いやすい

C4: 日間表示からタイムラインへ遷移できるのも便利

このことから、2.1 で述べた4つの要求については概ね達成できたと考えられる。

しかし一方で、改善すべき点としては以下の点が挙げられた。

C5: 出動した隊数だけでなく、残っている隊数や割合を表示できると、逼迫の発生状況をより把握しやすい

C6: 年間表示で、総出動時間に加え、総出動件数でも見たい

C7: 見栄えをもう少し整えたい

2.4.6 考察

2.4.5 で得られた評価を元に、考察を述べる。まず C1 から、違う年の同じ月日を容易に比較でき、例年と比べ出動が多くなっている年を見つけやすくなったと考えられる。次に C2 から、出動が多くなる時期はいつか、何時ごろに増えるのかといった情報を読み取ることができるようになったと考えられる。また C3 から、いつのデータに着目すべきか目安をつけてから詳細を確認できるようになったと考えられる。さらに C4 から、出動が多い時にどの部隊が忙しくなっているのか把握できることが有効だと考えられる。

改善点として挙げられた点について、C5 は全体の部隊数を利用することで計算可能だと考えられる。また、総出動件数についても救急出動データを集計することで作成可能だと考えられる。C7 については配色を工夫したり、フレームワークを利用することで改善できると考えられる。

以上より、本システムは出動が多い日時を発見するときに非常に有用であること、また、データ項目を増やすことでより多様な観点で分析できるようになると言える。

3 消防局の構成を分析・シミュレーションするツールの研究開発

近年、日本における多くの消防局では消防や救急の出動記録を ICT システムにより収集・記録している。この出動記録は、救急・消防の需要予測やリソース運用の最適化等のデータ駆動アプローチに活用することが期待されている。例として、消防防災科学センターでは、消防需要の分布を可視化し、現在の充足状況の把握を行っている [12]。

我々が共同研究を行っている神戸市消防局でも同様の取り組みを開始している。この取り組みの目的は、消防局の構成（すなわち、市内の消防署の数、位置および各署の消防車の編成）が、市内の各町丁目からの要請をどれだけ満たせるかを把握することである。より具体的には、以下の問いに関する答えを知りたいと考えている。

問い Q1: 「ある町丁目 B で火事が起きた時、指定された消防車の部隊セット F は、許容時間 T 以内に、B へ到着できるか？」もし Q1 の答えが否であるならば、よりよい構成を知るために以下の問題の答えも必要となる。

問い Q2: 「消防局の構成を変更したとき、Q1 の答えはどのように変化するか？」構成の変更とは、ある消防署の消防車の編成の変更、消防署の場所の移動、新しい消防署の建設を含む。

現在、一部の消防局職員によって、これらの問いに答える取り組みが試行錯誤的に行われているが、エクセルや GIS[11] 等の汎用ツールを用いた分析にとどまっている。神戸市には 28 の消防署と 3,326 の町丁目があり、その組み合わせは 93,128 通りにおよぶ。さらに消防局の構成を変えると組み合わせは爆発的に増えるため、手動による分析は困難を極めている。

本研究の目的は、一般の消防局員がデータマイニングの専門知識を必要とせず、問い Q1 や Q2 のための分析を行えるようにすることである。目的を達成するため、本稿では消防局構成分析・シミュレーションツール **FD-CAST (Fire Department Configuration Analysis and Simulation Tool)** の提案・開発を行う。FD-CAST は、それぞれ Q1, Q2 に対応する以下の要求 R1, R2 に基づいて開発される。

要求 R1 (駆けつけの可視化): 全町丁目のそれぞれ B において、B に最も早く到着できる消防車を、車種ごとにリストすること。リストの各要素は、駆けつけた消防車の ID と、消防署からの距離、到着までの所要時間（駆けつけ時間）を含むこと。このリストは Q1 の分析を容易化するように可視化すること。

要求 R2 (シミュレーション): ツールの全てのユーザはそれぞれ消防局の構成を自由に変更できること。可能な変更は、消防署内の消防車の追加、移動、削除であること。同様に、消防署の追加や移動、削除も可能であること。変更後の新たな構成において、P1 と同様に可視化を行えること。

これらの要求を満たすため、我々は FD-CAST を以下の段階に基づいて開発する。

Step 1 (データの準備): FD-CAST に必要なデータを準備する。消防局の構成データ、神戸市内の町丁目の位置、町丁目と消防署の間の道なりの距離などを含む。

Step 2 (駆けつけの可視化): ある町丁目 B で火災が起こった際、各タイプの消防車両がどこからどのぐらいの時間で駆けつけるかを計算し、結果を可視化する*¹。

Step 3 (構成の更新とシミュレーション): それぞれの消防局員が独立して構成を変更し、可視化を再実行できるような機能を実装する。

本研究では、提案する FD-CAST のプロトタイプを Web アプリケーションとして実装し、ケーススタディを行った。具体的には、神戸市消防局の放水隊の全町丁目に対する駆けつけのおおよその現状把握と、放水車の再配置による改善策のシミュレーションを行った。また、消防署を新設した際の効果を、駆けつけの観点から分析した。さらに、FD-CAST を実際に神戸市消防局での業務に使用してもらい、フィードバックをいただいた。その結果、FD-CAST が問い Q1, Q2 に答えるための有用なツールであることが示された。

3.1 準備

3.1.1 消防局におけるデータ

近年、各自治体における消防局では、最新の ICT システムが導入され、消防・救急業務の効率化が図られている。こうした ICT システムで記録・管理されるデータの代表的なものとして、出動データ、署所データ、車両データ等がある。各種出動データは、消防局における救急や消防の出動の詳細を記録したもので、データ項目には出動日時、災害種別、出動車両番号、現着日時等が含まれる。災害種別は、火災、救急、救助、水災、警戒等、対応した災害の種類・任務を表す。署所データとは、消防署についてのデータで、消防局管轄の消防署の名称や住所が含まれる。車両データは、消防局が所持している消防・救急車両のデータで、車両番号、配属消防署、車種、兼務車両といったデータが含まれる。兼務車両とは、複数任務を兼務する同一小

*¹ 駆けつけの距離や時間は、地理情報と車両の平均速度から計算によって求めるもので、実際の出動の実績データに一致するものではない

隊によって運用される車両群を指す。したがって、いずれかの車両が出動中は、他の兼務車両は出動できない。

3.1.2 消防局の構成問題

自治体の消防局が、地域内のどこに何か所の消防署を設置し、それぞれの消防署に車両を何台配置するかを決定する問題を、本稿では消防局の構成問題と呼ぶ。構成問題は、自治体の地域において火災が起きた時、迅速かつ効率的な消火活動が可能かどうか直結する。したがって、現状や将来の状況を見すえて、よりよい消防局の構成を決定することが求められている。

神戸市の場合、9つの区に28の消防署が設置されており、合計117の消防車、救急車が日常的に運用されている。各区に配置されている消防署の数と、車種別の消防車の数を表6に示す。表の各行は、それぞれの区に存在する消防署と消防車の数を表している。()内は同時に出勤可能な台数を示しており、兼務車両の制約によるものである。自治体の財政を鑑みると、部隊や車両を増やすことは容易ではない。限られたリソースの範囲で効率のよい構成をいかに求めるかが命題となっている。

表6 神戸市の消防署数および車種別消防車数

区	消防署	指揮車	救助車	放水車	救急車	はしご車	計
東灘	4	1	1	9(6)	4	2(1)	17
灘	2	1	1	3(2)	3	0	8
中央	3	1	2	5(3)	4	2(1)	14
兵庫	2	1	1	3(2)	2	0	7
長田	2	1	1	3(2)	3	2(1)	10
須磨	3	1	1	5(3)	3	0	10
垂水	4	1	1	7(5)	4	2(1)	15
西	3	1	1	6(3)	4	1	13
北	5	2	2	11(6)	5	3(2)	23
計	28	10	11	52	32	12	117

3.1.3 データに基づく構成の評価・改善

地域で火災が発生した際、火災の種類や規模に応じて、適切な消防車両群が速やかに現場に駆けつけ消火に当たる必要がある。例えば、一つの(理想的な)基準として「一般建物火災の場合、発生から8分以内に放水車が4台到着すること」というものが存在する。当然ながら、火災発生場所や消防局の構成に依存して、こうした基準は充足可能な場合とそうでない場合がある。したがって、消防局にとって「任意の町丁目に対して、指定された車両のセットが何分で到着できるか」を把握することは最大の関心事であり、これは3で述べた問いQ1である。

神戸市消防局では、様々な基準を設定して、消防局の構成をデータに基づいて評価し、改善する取り組みを行っている。しかしながら、現状ではエクセルやGIS(地理情報システム)等、汎用ツールを用いてデータ分析を行っており、現在の構成を評価するだけでも膨大な作業時間がかかっている。さらに、様々な構成をシミュレーションで試して評価することは現状不可能であり、これは3で述べた問いQ2に対応している。

3.2 FD-CAST: Fire Department Configuration Analysis and Simulation Tool

前節で述べた課題を解決すべく、本研究では消防局構成分析・シミュレーションツール **FD-CAST (Fire Department Configuration Analysis and Simulation Tool)** を開発する。FD-CASTは、3で述べた要求R1(駆けつけの可視化)およびR2(シミュレーション)に基づいて開発される。

3.2.1 システムアーキテクチャ

FD-CASTのシステムアーキテクチャを図19に示す。FD-CASTは、クライアント・サーバ形式のWebアプリケーションとして開発する。図の左側は、クライアント側の機能を表している。ユーザは、Webブラウザを用いて、駆けつけの可視化を行ったり、構成を変更したりすることができる。複数のユーザが独立して構成を変更したりシミュレーションしたりできる

ように、ユーザ管理の機能も備えている。図の右側は、サーバ側の機能およびデータを表す。以降の節で詳細を説明する。

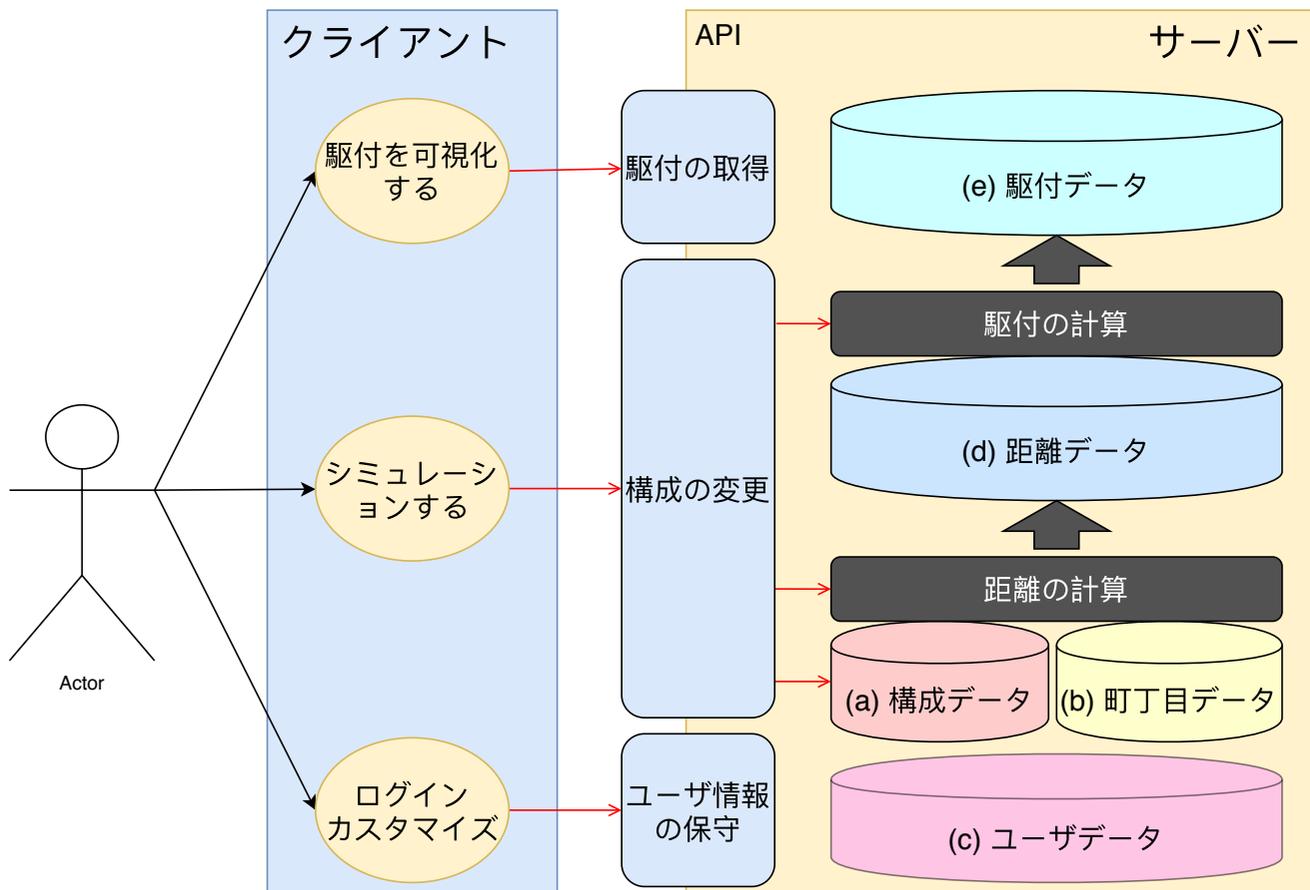


図 19 FD-CAST のシステムアーキテクチャ

3.2.2 データの準備

FD-CAST の実現に必要なデータを準備する。図 19 に示す通り、FD-CAST は、3 種類のマスタデータ ((a) 構成データ, (b) 町丁目データ, (c) ユーザデータ) と、2 種類の導出データ ((d) 距離データ, (e) 駆付データ) を利用する。図 20 に FD-CAST のデータモデルを定義する ER 図を示す。この ER 図は文献 [16] で示された記法に基づいている。四角はエンティティ、 $+-\in$ は親子関係、 $+-\dots$ は参照関係を表す。各エンティティのデータ項目の下には、インスタンスを例示している。

(a) 構成データ：消防局の構成を定義・管理するデータであり、消防署、車両、車種、割当の 4 つのエンティティで構成される。消防署は、消防局管轄の各消防署の名前、管轄区、住所を保持する。車両は、消防局管轄の各車両の ID、所属する消防署、車種、兼務車両の情報を保持する。車種は車両のタイプ (クラス) を定義するもので、用途に応じた車種の定義、および、複数の車種をまとめた車種群をグループとして定義する。割当は、どの火災種別・任務分類にどの車種 (群) を何台出動させるべきかを割り当てる情報を保持する。消防署と車両については、ユーザの ID を主キーに含めることで、FD-CAST のユーザ毎に自由に追加・変更できるようにしている。今回の構成データは、神戸市消防局から提供されたデータに基づき作成している。

(b) 町丁目データ：自治体が管理するすべての町丁目の住所とその中心点の緯度・経度を保持する町丁目エンティティで構成される。神戸市の場合、9 つの区に 3,326 の町丁目が存在し、それぞれの 9 桁の住所コード (towncode) で識別される。今回の町丁目データは、神戸市のオープンデータカタログから神戸市全市の町丁目リスト [13] を取得し、その緯度・経度を Yahoo! ジオコード API [9] によって求めて作成した。

(c) ユーザデータ：FD-CAST のユーザの情報を管理するデータであり、ユーザ、色の 2 つのエンティティで構成される。ユーザは、ユーザの ID、名前、パスワードを保持する。色は、ユーザ定義の可視化用カラースケールを定義する。ユーザデータは FD-CAST のユーザ登録機能で作成される。

(d) 距離データ：消防署から町丁目までの距離を管理する距離エンティティで構成される。各インスタンスは、1 つの消防署と 1 つの町丁目の距離 (m) とおおよその所要時間 (秒) を保持する。距離の値は、消防署の住所と町丁目の座標から、地理情報に基づいて導出する。所要時間は、距離を車両の平均速度で割ることで算出する。今回の距離データは、Google Direction API [5] を用いて、神戸市の 28 消防署 \times 3,326 町丁目の全ペアに対して道なりの経路を求め、その経路長を距離として作成した。また、車両の平均速度は FD-CAST のユーザが自由に設定できるものとし、デフォルト値を時速 30km としている。

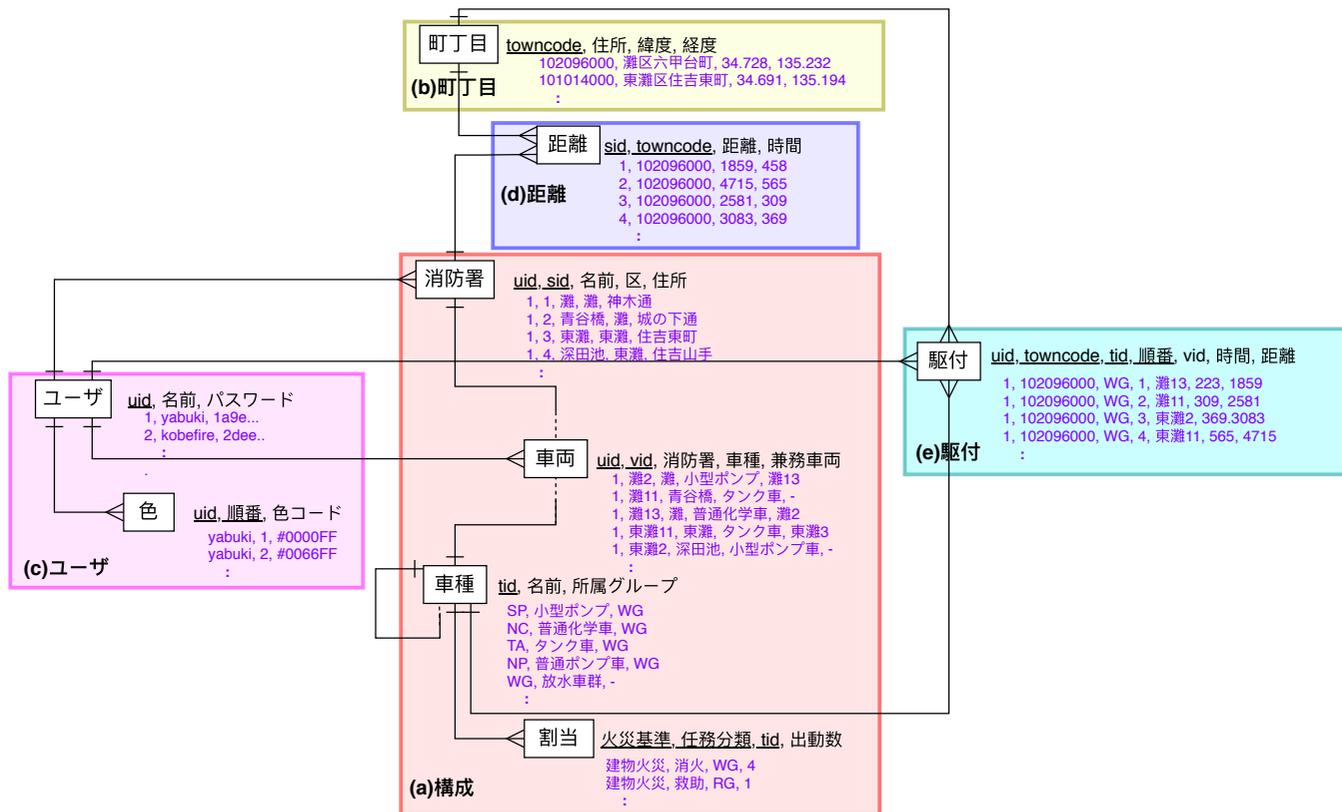


図 20 FD-CAST のデータモデル

(e) 駆付データ：各町丁目にとどの消防車両が何分で駆けつけられるかを保持する駆付エンティティで構成される。主キーは [ユーザ ID, 町丁目コード, 車種, 順番] で構成されており、ユーザごとに独立した駆付けの情報を管理すること、および、各町丁目に着する車両を車種別・到着順で管理することを想定している。各インスタンスの属性は、駆けつける車両の ID とその所要時間、消防署からの距離を保持する。駆付データは、次節で述べる手順によって、FD-CAST が計算によって作成する。

3.2.3 駆けつけの可視化

各町丁目 B で火災が発生した際、それぞれのタイプの消防車両について、どの消防車両がどこからどのぐらいの時間で駆けつけるかを、以下の手順によって計算する。以降では、図 20 のエンティティを [名前] で略記する。

- 1) 指定された火災種別・任務に必要な車種 τ とその出動数 n を、[割当] から取得する。
- 2) τ に属するすべての車種 $\tau_1, \tau_2, \dots, \tau_k$ を [車種] から取得する。
- 3) 車種が $\tau_1, \tau_2, \dots, \tau_k$ である全車両 v_1, v_2, \dots, v_s を [車両] から求める。
- 4) 各 v_i が所属する消防署 s_i と現在の町丁目 B をキーとして [距離] を参照し、 s_i から B の距離 δ_i と所要時間 t_i を求める。
- 5) v_1, v_2, \dots, v_s を所要時間で昇順にソートし、先頭 n 台の車両 $v_{f1}, v_{f2}, \dots, v_{fn}$ を町丁目 B の車種 τ に関する駆けつけ車両として選出する。このとき、兼務車両が含まれていればその車両を除外し、次の順番の車両を繰り上げて選出する。
- 6) $1 \leq j \leq n$ について、 $(B, \tau, j, v_{fj}, \delta_{fj}, t_{fj})$ を [駆付] に登録する。

例として、図 20 のインスタンスを参照しながら、神戸市灘区六甲台町で建物火災の消火任務に必要な駆けつけ情報を計算する。[割当] から、放水車群 (WG) が 4 台必要になることがわかる。[車種] から放水車群に含まれるのは、小型ポンプ車、普通化学車、タンク車、普通ポンプ車の 4 種類である。[車両] から、灘 2、灘 13 等の車両がリストアップされる。[消防署] と [距離] から各車両の所属消防署と灘区六甲台町の距離、および、所要時間を求め、昇順にソートする。なお、灘 2 は灘 13 の兼務車両なので、除外される。その結果、灘 13、灘 11、東灘 2、東灘 17 の 4 台が選出された。

全町丁目に対する駆付データが上記の手順によって作成されると、それらを表形式に一覧して可視化する。ここで、表の各行は町丁目を、各列はその町丁目への車種別の駆けつけ情報を表示する。表の各セルには、車両 ID と、消防署から町丁目までの距離・所要時間を表示する。

また、所要時間に応じてセルを色分けすることで、データの理解を助ける。デフォルトでは、所要時間が短いほど青く、長いほど赤く表示されるようになっている。このカラースケールは、ユーザ好みの配色にカスタマイズ可能であり、設定値は色エンティティで管理される。また、駆付けデータは、各列の属性でソートしたり、住所や駆けつけ時間の値で検索・絞り込みを行

うことで、細かい分析ができるようになっている。

さらに、FigMap4SC ([8] 参照) と連携することで、駆付けデータを地図上に可視化することもできる。駆付けデータの各町丁目の住所とセルに割り当てられている色コードのペアを FigMap4SC の Web-API に渡すことで、現在の構成から計算された駆付け情報を階級区分図上に可視化できる。これにより、地理的にどの町丁目への駆付けに時間がかかるか地図上で一目で確認できる。

3.2.4 構成の変更とシミュレーション

要求 R2 に対応するために、FD-CAST は各ユーザが独立に構成データを変更することを許可する。具体的には、図 20 の構成データのうち、[消防署] および [車両] のデータを追加、更新、削除することを許す。車両データについては、車両を出動不能の設定にすることも可能である。これにより、地域内に新しい消防署を設置したり、ある消防署が管轄する車両を別の消防署へ移動したり、様々な消防署の構成を試すことができる。なお、[車種] や [割当] は全国の消防局で共通的に使われるものであるから、現在のところこれらの変更は許可していない。

ユーザが構成データを変更したら、FD-CAST は再び距離の計算、および、駆付の計算を実行し、新しい構成データに基づく距離データ、駆付データを導出する (図 19 参照)。ユーザは再び 3.2.3 の駆付けの可視化によって、更新されたデータを閲覧・確認できる。

上記のような構成の変更、可視化のプロセスを繰り返し実行することで、ユーザは様々な消防局の構成をシミュレーションし、評価することができる。

3.2.5 ユーザのログイン、カスタマイズ

FD-CAST は複数のユーザが独立して構成情報の分析・シミュレーションを行えるように、ユーザ情報を管理する。ユーザは、ログイン ID とパスワードを FD-CAST に登録することで、新しいユーザアカウントを作成できる。

新しいユーザが作成されると、FD-CAST はそのユーザ専用の [消防署]、[車両]、[距離]、[駆付け] テーブルを作成し、デフォルトのデータをロードする。また、[色] テーブルを作成し、デフォルトのカラースケールも作成する。各ユーザ専用のテーブルを作ることで、ユーザは他のユーザに影響することなく、構成情報の変更・シミュレーションや可視化のカスタマイズを行える。

3.3 実装

3.2 で述べたデータモデルおよび機能に基づいて、FD-CAST のプロトタイプを実装した。

3.3.1 利用した技術

プロトタイプの実装に用いた技術を表 7 に示す。FD-CAST は Web アプリケーションとして実装されており、クライアント側には HTML5 と JavaScript、サーバ側には PHP を利用してプログラミングした。距離データの生成には Python を用いた。データベースは MySQL を、Web Server には Apache 2.4.37 を、ライブラリは Google Maps API[6] と Directions API[5]、PHPOffice[2] を利用した。

表 7 実装に用いた技術

クライアント	HTML5, JavaScript
サーバ	PHP5, Python (データ生成)
データベース	MySQL
Web サーバ	Apache 2.4.37
ライブラリ	Google Maps API, Direction API, PHPOffice

3.3.2 駆付けの可視化画面

図 21 に、駆付けの可視化画面を示す。3.2.3 で述べた通り、表の各行は町丁目を表し、各列はその町丁目への車種別の駆付け情報を表示している。例として、表の 1 行 5 列目をみると、神戸市中央区三ノ宮町 1 丁目に 1 番目に到着できる放水車

は「中央 13」であり、1,137m の距離を 138 秒で到着する計算であることがわかる。

表の各セルは、各車両の到着時間に応じて色分けされており、デフォルトでは速いほど青く、遅いほど赤く表示される。左上のプルダウンリストで、車両の平均速度を変更することができる。また、各列の値でデータをソートするボタンや、検索のためのボタンがついている。

また、図 22 に、地図形式での表示例を示す。この図は神戸市の全域で放水車が最速で駆けつけできる時間を視覚化したものであり、表形式での可視化と同じ色で、駆けつけの所要時間が短い地域を青く、長い地域を赤く表示している。

最速駆けつけ一覧

検索 車両配置の変更 KVMで表示 救急車の駆けつけ一覧 ログアウト

駆けつけ一覧 車両の速度 30km/h 地図変更

地域	統括指揮車	指揮隊群	救助工作車	救助車群	放水車群1	放水車群2	放水車群3	放水車群4	救急車	はしご車群	地図
神戸市中央区三宮町 1 丁目	ID: 中央 7 5 距離: 1157m 時間: 138秒	ID: 兵庫 7 0 距離: 3521m 時間: 422秒	ID: 神清 3 0 距離: 1157m 時間: 138秒	ID: 兵庫 3 3 距離: 3521m 時間: 422秒	ID: 中央 1 3 距離: 1157m 時間: 138秒	ID: 中央 5 距離: 1157m 時間: 138秒	ID: 中央 4 距離: 1948m 時間: 221秒	ID: 中央 2 距離: 1848m 時間: 221秒	ID: 中央 9 0 距離: 1157m 時間: 138秒	ID: 中央 1 8 距離: 1157m 時間: 138秒	地図
神戸市中央区三宮町 2 丁目	ID: 中央 7 5 距離: 1332m 時間: 159秒	ID: 兵庫 7 0 距離: 3335m 時間: 400秒	ID: 神清 3 0 距離: 1332m 時間: 159秒	ID: 兵庫 3 3 距離: 3335m 時間: 400秒	ID: 中央 1 3 距離: 1332m 時間: 159秒	ID: 中央 5 距離: 1332m 時間: 159秒	ID: 中央 4 距離: 1550m 時間: 186秒	ID: 中央 2 距離: 1550m 時間: 186秒	ID: 中央 9 0 距離: 1332m 時間: 159秒	ID: 中央 1 8 距離: 1332m 時間: 159秒	地図
神戸市中央区三宮町 3 丁目	ID: 中央 7 5 距離: 1203m 時間: 144秒	ID: 兵庫 7 0 距離: 3206m 時間: 384秒	ID: 神清 3 0 距離: 1203m 時間: 144秒	ID: 兵庫 3 3 距離: 3206m 時間: 384秒	ID: 中央 1 3 距離: 1203m 時間: 144秒	ID: 中央 5 距離: 1203m 時間: 144秒	ID: 中央 3 距離: 1665m 時間: 199秒	ID: 中央 4 距離: 1929m 時間: 231秒	ID: 中央 9 0 距離: 1203m 時間: 144秒	ID: 中央 1 8 距離: 1203m 時間: 144秒	地図
神戸市中央区上開井通 1 丁目	ID: 中央 7 5 距離: 2518m 時間: 302秒	ID: 兵庫 7 0 距離: 3206m 時間: 384秒	ID: 神清 3 0 距離: 2034m 時間: 244秒	ID: 神清 3 0 距離: 2518m 時間: 302秒	ID: 灘 1 1 距離: 1467m 時間: 176秒	ID: 灘 3 距離: 1467m 時間: 176秒	ID: 灘 2 距離: 2034m 時間: 244秒	ID: 灘 5 距離: 2034m 時間: 244秒	ID: 灘 9 1 距離: 1467m 時間: 176秒	ID: 中央 1 8 距離: 2518m 時間: 302秒	地図
神戸市中央区上開井通 2 丁目	ID: 中央 7 5 距離: 2376m 時間: 285秒	ID: 灘 7 0 距離: 2421m 時間: 290秒	ID: 神清 3 0 距離: 2376m 時間: 285秒	ID: 灘 3 0 距離: 2421m 時間: 290秒	ID: 灘 1 1 距離: 1489m 時間: 178秒	ID: 灘 3 距離: 1489m 時間: 178秒	ID: 中央 5 距離: 2376m 時間: 285秒	ID: 中央 1 3 距離: 2376m 時間: 285秒	ID: 灘 9 1 距離: 1489m 時間: 178秒	ID: 中央 1 8 距離: 2376m 時間: 285秒	地図
神戸市中央区上開井通 3 丁目	ID: 中央 7 5 距離: 2276m 時間: 273秒	ID: 灘 7 0 距離: 2421m 時間: 278秒	ID: 神清 3 0 距離: 2276m 時間: 273秒	ID: 灘 3 0 距離: 2321m 時間: 278秒	ID: 灘 1 1 距離: 1388m 時間: 166秒	ID: 灘 3 距離: 1388m 時間: 166秒	ID: 中央 5 距離: 2276m 時間: 273秒	ID: 中央 1 3 距離: 2276m 時間: 273秒	ID: 灘 9 1 距離: 1388m 時間: 166秒	ID: 中央 1 8 距離: 2276m 時間: 273秒	地図
神戸市中央区上開井通 4 丁目	ID: 中央 7 5 距離: 2411m 時間: 289秒	ID: 灘 7 0 距離: 2580m 時間: 309秒	ID: 神清 3 0 距離: 2411m 時間: 289秒	ID: 灘 3 0 距離: 2590m 時間: 309秒	ID: 灘 1 1 距離: 1320m 時間: 158秒	ID: 灘 3 距離: 1320m 時間: 158秒	ID: 中央 5 距離: 2411m 時間: 289秒	ID: 中央 1 3 距離: 2411m 時間: 289秒	ID: 灘 9 1 距離: 1320m 時間: 158秒	ID: 中央 1 8 距離: 2411m 時間: 289秒	地図
神戸市中央区上開井通 5 丁目	ID: 中央 7 5	ID: 灘 7 0	ID: 神清 3 0	ID: 灘 3 0	ID: 灘 1 1	ID: 灘 3	ID: 中央 5	ID: 中央 1 3	ID: 灘 9 1	ID: 中央 1 8	地図

図 21 駆けつけの可視化画面

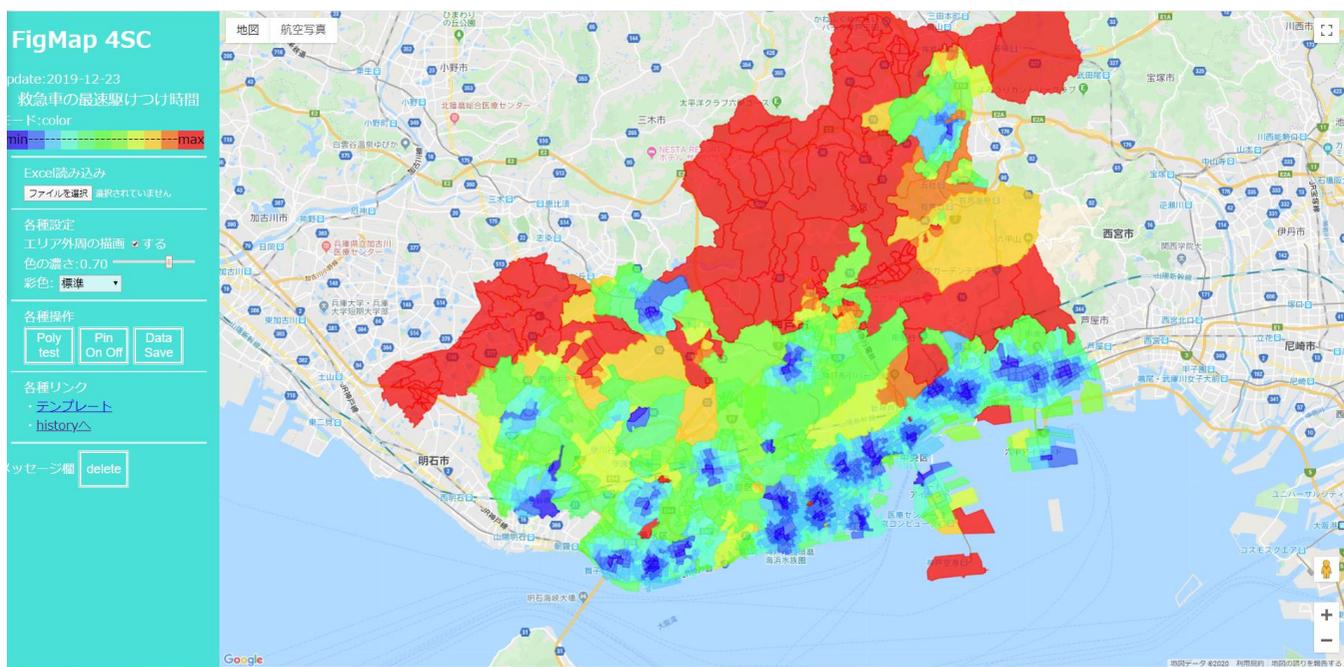


図 22 FigMap4SC との連携による地図上への駆けつけの可視化

3.3.3 構成の変更画面

図 23 に、車両構成の変更画面を示す。図 20 で定義したように、各消防車両には車両番号 (ID)、配置消防署、車種、兼務車両があり、一覧画面ではこれらが表示される。「編集」ボタンを押すことで編集画面へと遷移し、これらの情報のうち、車両番号以外の情報を変更できる。変更された情報は、データベースへと保存される。また、「追加」ボタンから新しい車両を追加したり、「不能化」ボタンから、その車両を出動不可の状態に変更したりすることもできる。特定の車両を探したい場合、左上

の「検索」ボタンを押すことで検索画面に遷移し、特定条件を満たす車両のみを表示して探すこともできる。これらの機能により、実際の現場では試せない仮想的な消防署の構成をシミュレーションすることが可能となる。

消防車の配置設定

検索 追加 計算実行 計算せずに普通駆けつけ一覧に戻る 配置の初期化

臨時車両一覧 消防署一覧 ログアウト

全130件中 1-20件を表示

1 2 3 4 3 次へ 検索

区	車両番号	配置消防署	車両種別1	車両種別2	車両種別3	業務車両1	業務車両2	出動時間	編集	状態変更	削除
東灘	東灘1 0	東灘消防署	大型水槽車			東灘2 7	東灘1 7	00:00:00~23:59:59	編集	不能化(解除)	削除
東灘	東灘1 1	東灘消防署	タンク車	除染活動隊		東灘3		00:00:00~23:59:59	編集	不能化(解除)	削除
東灘	東灘1 2 0	東灘消防署	ホース延長車					00:00:00~23:59:59	編集	不能化(解除)	削除
東灘	東灘1 3	青木出張所	普通化学車			東灘2 1		00:00:00~23:59:59	編集	不能化(解除)	削除
東灘	東灘1 4	六甲アイランド出張所	普通化学車			東灘4		00:00:00~23:59:59	編集	不能化(解除)	削除
東灘	東灘1 7	東灘消防署	直伸はしご車			東灘2 7	東灘1 0	00:00:00~23:59:59	編集	不能化(解除)	削除
東灘	東灘2	深田池出張所	小型ポンプ車					00:00:00~23:59:59	編集	不能化(解除)	削除
東灘	東灘2 1	青木出張所	ポンプ特災車	小型ポンプ車		東灘1 3		00:00:00~23:59:59	編集	不能化(解除)	削除
東灘	東灘2 7	東灘消防署	小はしご車			東灘1 7	東灘1 0	00:00:00~23:59:59	編集	不能化(解除)	削除
東灘	東灘3	東灘消防署	小型ポンプ車	除染活動隊	トリアポンプ隊	東灘1 1		00:00:00~23:59:59	編集	不能化(解除)	削除
東灘	東灘3 3	東灘消防署	ポンプ救助車	普通ポンプ車				00:00:00~23:59:59	編集	不能化(解除)	削除
東灘	東灘4	六甲アイランド出張所	小型ポンプ車			東灘1 4		00:00:00~23:59:59	編集	不能化(解除)	削除
東灘	東灘7 5	東灘消防署	統括指揮車					00:00:00~23:59:59	編集	不能化(解除)	削除

図 23 車両構成の変更画面

3.3.4 カラースケールの設定画面

FD-CAST では、表形式で駆けつけ情報を可視化する際、駆けつけ時間に応じた色分けを行う。この色は、0 秒~600 秒まで 60 秒単位の各区分、および、600 秒以上の範囲にそれぞれ異なる 11 色を割り当てるカラースケールで定義されている。デフォルトのカラースケールを図 24 に示す。この画面において、ユーザは各区分の色を自分の好きな色に変更し、可視化結果の見映えを変えることができる。例えば、480 秒以内を無色、480 秒以上を赤色にすることで、8 分以上の地域のみを彩色する等、注目する基準に合わせてカラースケールを定義することで、分析に適した可視化手法を設定できる。

color1	color2	color3	color4	color5	color6	color7	color8	color9	color10	color11
0~60	60~120	120~180	180~240	240~300	300~360	360~420	420~480	480~540	540~600	600~660

図 24 カラースケールの設定画面

3.4 ケーススタディ

FD-CAST の有効性を確認するため、現実的な事例を題材としたケーススタディを行った。

3.4.1 ケース 1: 放水車の駆けつけの現状と再配置

放水車の迅速な駆けつけは火災の延焼をできるだけ抑えるために重要な要素となる (3.1.3 参照)。

そこで FD-CAST を用いて、神戸市の各町丁目への放水車がどのぐらいの時間で到着可能かを可視化する。図 25 は、FD-CAST が計算した「各町丁目に 2 台目の放水車が到着する所要時間」を、FigMap4SC を用いて地図上に表示したものである。消防署が多く配置されている中央南部の市街地は寒色で彩色されており、この地域には 2 台の放水車がすぐに到着することがわかる。一方、消防署から距離がある北部や西部の地域は暖色で彩色されており、到着には時間がかかることがわかる。

次に、市街地の消防署の放水車 1 台を、北部の消防署へと移動させる変更を行い、再び駆けつけの可視化を行った。図 26 は、構成変更後の放水隊 2 台の到着時間を表示したものである。図中の × 印と ○ 印は、それぞれ市街地の消防署と北部の消防署の位置を示している。図 25 と図 26 を見比べると、この構成の変更によって、市街地の駆けつけ時間をあまり犠牲にすることなく、北部の一部の地域の駆けつけ時間を大幅に短縮できたことがわかる。

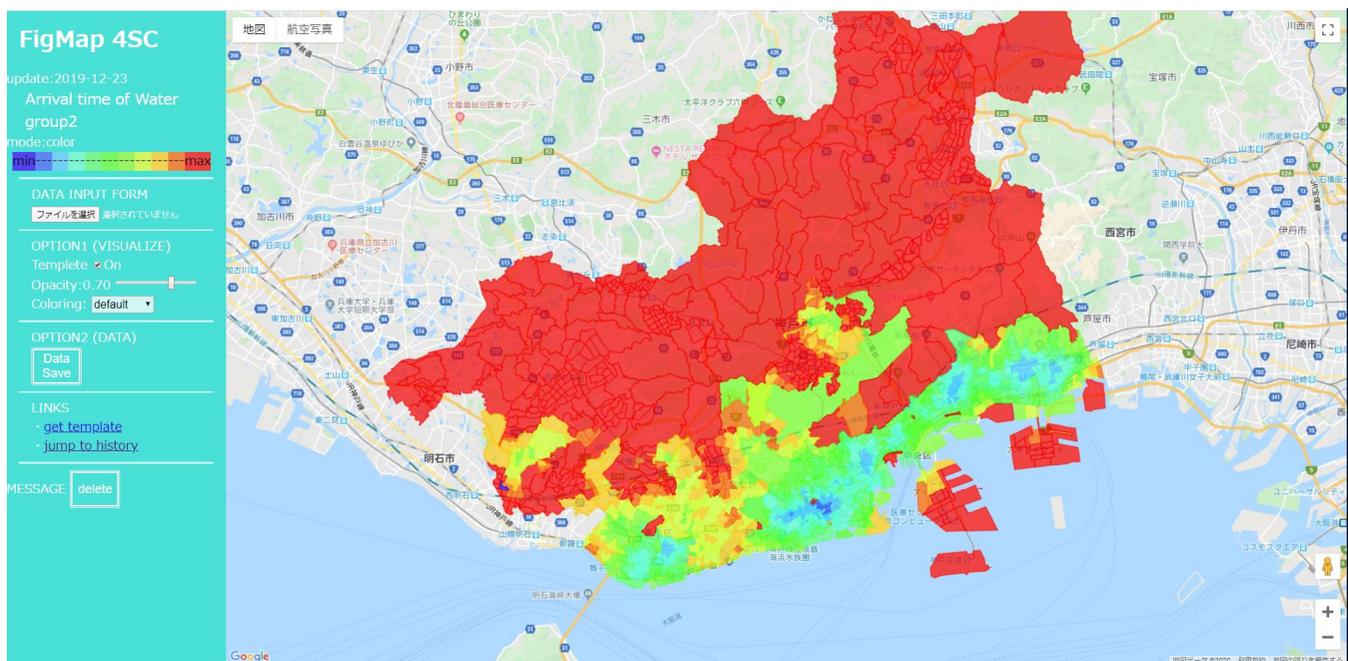


図 25 放水車 2 台の駆けつけ所要時間

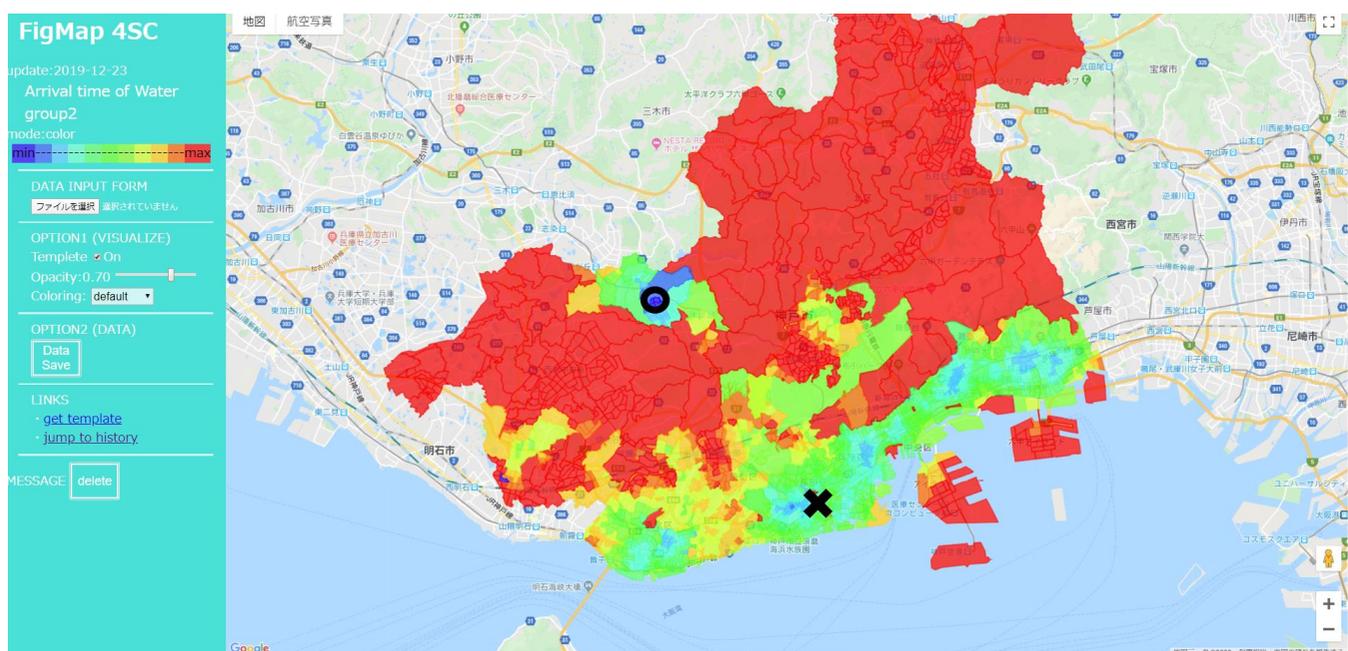


図 26 放水車 2 台の駆けつけ所要時間 (構成変更後)

3.4.2 ケース 2: 消防署の新設

現在、神戸市消防局では、西区に新たな消防署を新設することを予定している。そこで、建設予定地に新しい消防署ができた場合の放水隊 2 台の到着時間についてシミュレーションを行った。具体的には、神戸市西部の当該住所に新しく消防署を作成し、そこに放水隊を 1 台追加する構成の変更を行った。図 27 は、新しい消防署を追加した場合の放水隊 2 台の到着時間を可視化したものである。図中の丸印は、新設した消防署の位置を示している。図 25 と比較すると、設置した消防署の周囲は、到着時間が若干短くなったことがわかる。

3.4.3 実務者からのフィードバック

開発した FD-CAST のプロトタイプは、現在、神戸市消防局で実際に利用されている。職員の方々が業務で使う中で、その利点や改善点に関するフィードバックをいただいている。フィードバックの一部を以下に紹介する。

- 各町丁目への駆けつけを、手軽に確認・可視化できて便利

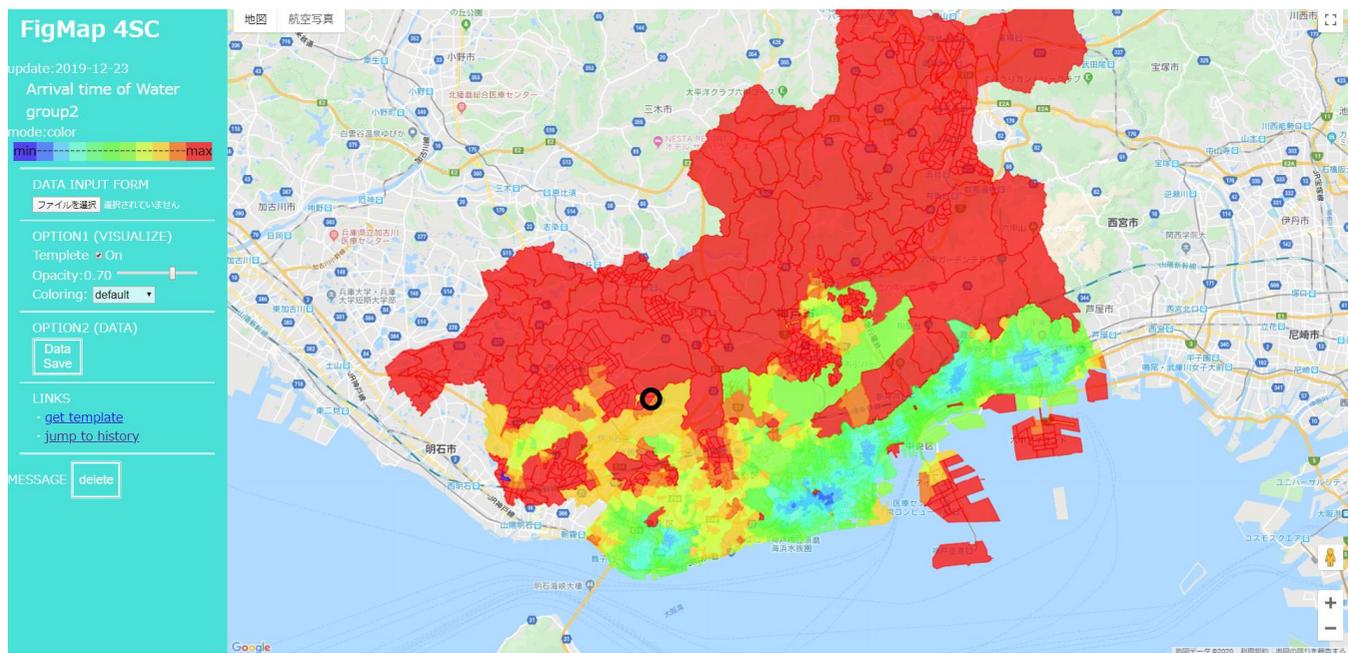


図 27 放水車 2 台の駆けつけ所要時間 (消防所新設後)

- 実際の現場では容易には試せない車両編成や消防署の新設・変更に伴う影響を、シミュレーションできるのが良い
- 一部の地域で、駆けつけの時間が実績と大きくずれているのはなぜか？
- 最適な構成を計算によって自動的に求められないか？

1 番目と 2 番目のフィードバックは、実際の現場において FD-CAST が消防局の構成問題の分析・検討に貢献できていることを表している。3 番目のコメントは、距離の計算に利用した Google Direction API が算出した経路が、実際に消防車が通る経路と大きく異なったことに起因している。API のオプションを見直すなどして、改善したい。4 番目のコメントは、最適化問題を解くことと等価であり、計算を行うことは可能であると考えられる。

4 熱中症予測に関するアプリケーションの研究開発

4.1 概要

近年、救急搬送数が増加傾向にあり、救急医療のひっ迫が指摘されている。特に、慢性的なひっ迫ではなく、特定の条件下においてのみ急激に増加するような傷病への対応は、常に受け入れ態勢を整えることは難しく、救急施策として考慮することは困難である。

本研究では、このような一時的な救急医療をひっ迫する要因として、近年その発生増加が大きな問題となっている熱中症に着目する。熱中症に関する対策として神戸市では、熱中症が集中する地域において、臨時救急隊の増隊などを実施している。一方で「いつ」「どこに」増隊するかなどは、過去の経験則から行っているため、実際に臨時救急隊の能力を最大限発揮できているか、その効果を十分に活用できているかなどは今後の課題として挙げられている。

さらなる熱中症に対する救急医療の取り組みとして、熱中症搬送者数の予測が有効であると考えられる。もし将来の熱中症搬送者数の予測を実現出来れば、救急医療としてはその予測に合わせた臨時救急隊の配置を行える。しかし、臨時隊の増隊といった救急施策を準備するためには、少なくとも4~7日の準備期間を要する。そのため、向こう1週間程度の中期的な予測を行う必要があり、その予測精度は分かっていない。

そこで本研究では、神戸市消防局が持つ救急ビッグデータを細粒度に分析し、以下のR1, R2のリサーチ・クエスチョンを設定する。

- **RQ1:** 中期の熱中症搬送者数の予測は可能か？
- **RQ2:** 神戸市において、RQ1での提案手法は有効か？

RQ1へのキーアイデアとして、週間天気予報に含まれる項目のみを活用し、熱中症搬送者数の予測モデルの構築を行う。利用可能な項目として、予測日の最高気温、最低気温、天気がある。これに加え、予測したい地域における、熱中症が発生しうる気温の閾値を加えることで、その地域性を反映させる。

本研究では、予測の仮定を説明可能にするため、部分的最小二乗回帰 (Partial Least Squares Regression, PLS) という手法を用いて、重回帰予測モデルを構築する。加えて、予測値に対して、雨が降った日の予測は0.85倍するヒューリスティックを適用し、救急隊員による経験則を反映させる。

提案手法を元に、神戸市のデータを例としてモデルを構築する。利用するデータとして、気象庁より、神戸市における、2013年から2019年の各年6月から9月の各日(全854日)の過去の気象データを取得する。加えて、神戸市消防局が持つ同期間の救急ビッグデータより、各日の熱中症搬送者数を取得する。以上のデータを用いて、予測モデルを構築した。

構築したモデルに対して、神戸市における2020年6月から9月に対して発表された実際の週間天気予報データを入力し、同日の熱中症に対する救急出動回数を予測しヒューリスティックを適用したところ、7日後の熱中症搬送者数を平均誤差2.79人で予測できた。

RQ2では、熱中症搬送者数の予測値を表示するWebアプリケーション *HSP (HeatStroke-Prediction)* のプロトタイプ開発を行う。HSPは、現在発表されている週間天気予報とRQ1で開発した予測モデルを用いて、今日を含めた1週間後までの各日の熱中症搬送者数に予測値を計算、表示を行う。この時、週間天気予報の取得や説明変数の計算は、アプリケーション内部で行うため、ユーザは特別な操作せず、アプリケーションにアクセスするだけで予測値を確認することができる。また、予測値の値域を同時に表示することで、週間天気予報より暑くなった場合や、寒くなった場合を検討することに繋げる。

提案したHSPを、HTML5, JavaScript及びPythonで実装する。本アプリケーションの有効性を確認するため、神戸市の救急搬送業務に携わる救急隊員に毎日確認して頂き、アプリケーションや予測値に対するフィードバックを頂いた。その結果、予測結果の見やすさには問題なく、予測値も経験則とよく似ているという意見を頂いた。加えて、中期の熱中症搬送者数の予測により、ピンポイントの熱中症予防施策を検討できるという意見も頂き、本アプリケーションの有効性が示された。

しかし一方で、現場では熱中症者の増加を想定した準備を行っており、予測に合わせた変更や増強といった施策は今回は行わなかったという意見も頂いた。

4.2 準備

4.2.1 熱中症

熱中症は「体温が上昇して重要な臓器が高温にさらされたりすることにより発症する障害の総称」と定義されている [10]。多くの場合、体内の熱産生が熱放散を上回ることで、体の体温が上昇し熱中症が発症する。熱中症を引き起こす要因として、気温や湿度といった環境条件、高齢や体調不良といった身体条件、激しい運動や長時間の屋外作業といった行動条件がある。例

例えば環境条件に着目すると、高温多湿の日、日差しが強い日、風が弱い日、急に暑くなった日は、熱中症が発生しやすい。また、高齢者の熱中症において、連続3日間の気温が熱中症発生に影響しており、暑い日が続くことで熱中症発生の可能性が向上することが分かっている [7]。

熱中症の主な症状として、めまいや立ちくらみ等の軽症から、けいれんや意識障害等の重症まで多岐にわたる。重度の熱中症（熱射病と呼ばれる）では、死亡率が高く、後遺症が残ることも少なくない。近年では、COVID-19(新型コロナウイルス)対策として、身体的距離を保ち、マスクの着用が推奨される「新しい生活様式」が実施されている。しかし、マスクを着用することで熱がこもり、体温調節が難しくなることで、熱中症を発症する事案が発生している。

このように熱中症は、生命にかかわる身近な傷病ではある。しかし、こまめな水分補給、空調機器を利用した室温の管理、外出時の日傘の利用等の予防法を実践することで、重症化を回避する、また熱中症の発症を防ぐことが可能である。そのため、国や地方自治体が熱中症予防に関する啓蒙活動を行うことは、熱中症搬送者数をの減少に効果的である。

4.2.2 消防局が持つ救急ビッグデータ

我々が共同研究を行っている神戸市消防局では、スマートシティを目指す取り組みの一環として、救急出動に関するデータを ICT システムにより収集・記録している。

救急出動に関するデータには、各救急出動に対して、出動隊や車両のデータ、入電や現場への駆けつけなどの時刻データ、傷病者の年齢や傷病に関するデータ、搬送先の病院のデータ等が記録されている。これらの膨大なデータが蓄積されたビッグデータ（以降、救急ビッグデータと呼ぶ）を分析・活用することで、これまで定量的な評価が困難であった、救急需要を満たす救急リソースの再配置、救急需要を見越した出動戦略の策定、などの救急事業の改善が実現可能である。

4.2.3 気象データ

表 8 2020 年 8 月における、過去の気象データの例

日	降水量 (mm)	気温 (°C)			湿度 (%)		風速 (m/s)		天気概要	
		平均	最高	最低	平均	最小	平均	最大	昼 (6~18 時)	夜 (18 時~6 時)
1	—	28.4	31.7	24.6	68	53	2.4	4.9	晴後一時曇	曇時々晴
2	0.0	28.6	32.1	26.3	73	57	3.1	6.5	曇一時雨後晴、雷を伴う	晴時々曇

表 9 気象庁による、週間天気予報の例

2021 年 07 月 23 日 17 時 神戸地方气象台 発表									
日付	23 日 (金)	24 日 (土)	25 日 (日)	26 日 (月)	27 日 (火)	28 日 (水)	29 日 (木)	30 日 (金)	
兵庫県天気	晴時々曇	晴後時々曇	晴時々曇	晴時々曇	晴時々曇	晴時々曇	曇時々晴	曇時々晴	
降水確率 (%)	-/-/20	0/0/10/10	10	10	20	10	20	20	
神戸 気温 (°C)	最高	-	33	33 (31~34)	33 (31~35)	32 (31~34)	32 (30~34)	32 (29~34)	33 (30~35)
	最低	-	26	26 (25~27)	26 (24~27)	26 (24~27)	26 (23~27)	26 (24~27)	26 (24~27)

気象庁では、1872 年から現在に至るまでの過去の気象データ [3] を記録、蓄積している。気象データとは、特定の日や時間帯における、気温、湿度、降水量、天気などを指す。近年では、1 日の気象データだけでなく、1 時間ごと、10 分ごとの気象データも記録、蓄積している。表 8 は、2020 年 8 月における日ごとの神戸市の気象データの一部を表している。気温や湿度などの気象データの一部には、その日における最高値や最低値という気象統計値が表される。このようなデータはオープンデータとして一般に公開され、イノベーションの契機に活用することが期待されている。

また、発表日翌日から 7 日先までの天気、気温などを予報する週間天気予報 [4] を発表している。表 9 は、2021 年 07 月 23 日 17 時に発表された、兵庫県の週間天気予報を表している。特定の日付 d における天気 $W(d)$ 、降水確率 $RP(d)$ 、最高気温 $T_{max}(d)$ 、最低気温 $T_{min}(d)$ が、列ごとに表される。気温の行に注目すると、予報日の 2 日後以降の気温には、気温の予測範囲が表示されている。本稿では、日付 d の最高気温の予測範囲の下端を $LT_{max}(d)$ 、最高気温の予測範囲の上端を $UT_{max}(d)$ 、最低気温の予測範囲を下端 $LT_{min}(d)$ 、最低気温の予測範囲の上端を $UT_{min}(d)$ とする。

4.3 RQ1: 中期の熱中症搬送者数の予測は可能か？

4.3.1 特徴開発

熱中症搬送者数を予測するために、その数に強く影響を与える要因、すなわち説明変数が必要である。本稿では、過去の気象データ及び週間天気予報の両方に存在する項目の内、熱中症の発生に強く起因する要素を説明変数とする。4.2.3 節より、説明変数として利用可能な項目は、「天気 (天気概要)・最高気温・最低気温」である。

ここで、気温の特徴をより際立たせるため、新たな特徴量を定義する。

$$SqT_{max}(d) = K_{\alpha} \times \{T_{max}(d) - \alpha\}^2$$

$$SqT_{min}(d) = K_{\beta} \times \{T_{min}(d) - \beta\}^2$$

$$K_{\alpha} = \begin{cases} 1 & (T_{max}(d) \geq \alpha) \\ -1 & (T_{max}(d) < \alpha) \end{cases}$$

$$K_{\beta} = \begin{cases} 1 & (T_{min}(d) \geq \beta) \\ -1 & (T_{min}(d) < \beta) \end{cases}$$

α : 熱中症が発生する $T_{max}(d)$ の閾値

β : 熱中症が発生する $T_{min}(d)$ の閾値

上式における α 及び β は、予測モデルの対象となる地域の実績に基づいて自由に決定出来る。ここでは、神戸市を例に α 及び β を決定する。

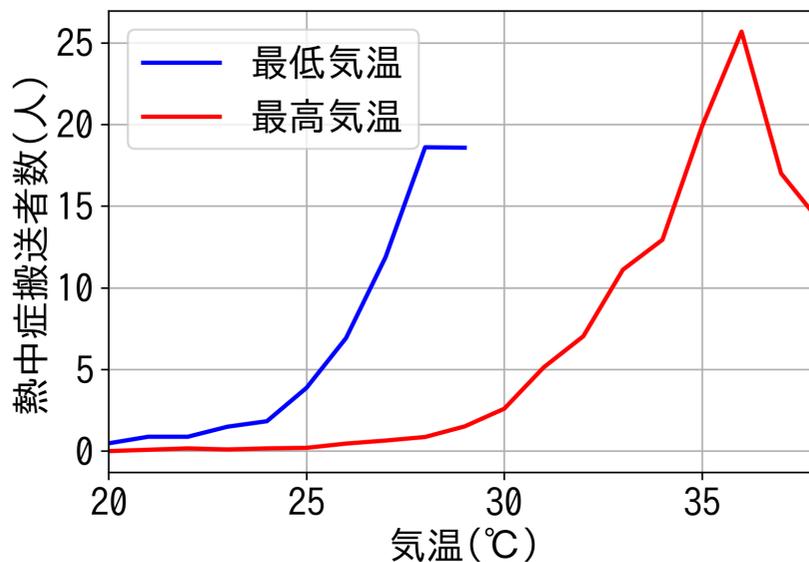


図 28 特定の気温における 1 日の平均熱中症搬送者数

図 28 は、神戸市における、最低気温及び最高気温と、1 日の平均熱中症搬送者数の関係を表している。これより、神戸市では、最低気温が 22 °C、最高気温が 25 °C を超えると、徐々に熱中症が発生すると考えられる。よって、 $\alpha = 25, \beta = 22$ と設定する。また、最高気温及び最低気温がそれぞれ α, β の値を超えると、熱中症搬送者数が急激に増加することが分かる。この影響を強く反映させるため、 $SqT_{max}(d), SqT_{min}(d)$ に 2 乗式を採用している。一方、最高気温及び最低気温がそれぞれ α, β の値を下回った場合、熱中症搬送者が殆ど出ないと考えられる。そのため、 K_{α}, K_{β} が負の値になりうる条件を与え、その影響を反映させている。

次に、天気の特徴量を変数にすることを考える。天気概要について、表 8 にある通り、「晴後一時曇」や「曇一時雨後晴、雷を伴う」といった複雑な表現が混じり、その種類は 20 以上に及ぶ。これらの表現は正確であるものの、予測精度に悪影響を及

ばす可能性や、週間天気予報との突合が難しいという問題がある。そのため、天気を単純な表現に変換する必要があると考える。また、熱中症には日差しの強さが影響していることから、「晴かどうか」にのみ着目し、以下のように特徴量を定義する。

$$IS_{am}(d) = \begin{cases} 1 & (\text{午前の天気が晴である}) \\ 0 & (\text{午前の天気が晴でない}) \end{cases}$$

$$IS_{pm}(d) = \begin{cases} 1 & (\text{午後の天気が晴である}) \\ 0 & (\text{午後の天気が晴でない}) \end{cases}$$

ここで、天気概要から $IS_{am}(d), IS_{pm}(d)$ に変換する方法を説明する。天気概要の1文字目に着目し、これを午前の天気とする。次に、天気概要に「後」という言葉が存在するか確認する。この「後」という言葉は、天気が時間によって変化するという意味を持っている。「後」が含まれていた場合、その次の文字を午後の天気とする。含まれていない場合は、午前の天気で午後の天気を同じとして扱う。

例として、2020年8月1日の天気である「曇一時雨後晴、雷を伴う」を変数に変換する。1文字目は「曇」であるため、午前の天気は「曇」とする。次に、「後」が含まれているため、その次の文字「晴」より、午後の天気は「晴」とする。これを変数の条件に当てはめると、 $IS_{am}(2020/08/01) = 0, IS_{pm}(2020/08/01) = 1$ となる。

これらの特徴量から、4.2.1節の熱中症の発生要因を元に、以下の6変数を説明変数として選択する。なお、説明変数の予測日の日付を d 、予測日より n 日前の日付を $d - n$ とする。

- $SqT_{max}(d)$
- $SqT_{min}(d)$
- $SqT_{max}(d - 1)$
- $SqT_{max}(d - 2)$
- $IS_{am}(d)$
- $IS_{pm}(d)$

4.3.2 重回帰モデルによる予測モデル構築

機械学習を用いたモデル構築では、予測の元となる説明変数のほかに、予測したい値である目的変数が必要となる。本稿の目的は、熱中症搬送者数を予測することであり、目的変数は任意の日の熱中症搬送者数となる。本章では、神戸市のデータを例として、神戸市における1日の熱中症搬送者数を予測する。

説明変数は、気象庁ホームページにおいて2013年から2019年の各年6月から9月における各日(全854日)の気象データをcsvファイルとして取得し、Pythonで4.3.1節の特徴量に加工する。また、得られた特徴量を正規化し、標準正規分布に従うよう、値を変換する。目的変数である各日の熱中症搬送者数は、データベースに保存されている神戸市消防局が持つ救急ビッグデータから、同日の熱中症に対する救急出動回数を取得する。

予測モデルを作成するために、Pythonのオープンソース機械学習ライブラリである *scikit-learn* を利用する。学習のためのアルゴリズムとして、部分的最小二乗回帰 (Partial Least Squares Regression, PLS) を用いる。PLSは重回帰分析の1つであり、説明変数間の相関が高い場合においても、回帰式を導くことが出来るという特徴がある。今回のように、天気と最高気温のといった、相関の強い説明変数を使うと、回帰係数が不安定になる場合が多い。PLSを用いることで、この現象を解消することが出来る。

また、さらなる精度向上のため、ヒューリスティックを導入する。ヒューリスティックとは、経験則や直感から答えを導き出すことである。本稿では、天気に関する変数において、熱中症リスクを下げる可能性が高い「雨」という要素を排除している。そのため、消防局員への聞き取りと、試行錯誤的なパラメータ調整から、「雨の日は、出力される予測値を0.85倍する」というヒューリスティックを導入する。

以上から、取得した全854日の説明変数及び目的変数を入力し、PLSによる重回帰予測モデルの構築を行った。

4.3.3 予測精度の評価

構築したモデルに対して、神戸市の気象データおよび週間天気予報のデータを入力し、ヒューリスティックを適用することで、熱中症搬送者数の予測値を取得する。本予測対象である熱中症搬送者数は、0以上の整数でのみ表現されることから、出力された予測値に対して、「小数点以下を四捨五入する」「負の値が出力された場合、予測値を0とする」の2つの処理を行う。その後、実際の熱中症搬送者数と比較することで、平均絶対誤差 (MAE) と二乗平均平方根誤差 (RMSE) を計算し、小数点第3位を四捨五入することでモデルの精度を確認する。

気象データとして入力するデータは、神戸市における 2020 年 6 月から 9 月までの全 122 日の気象データである。週間天気予報として入力するデータは、同一日時の天気を予報したデータを用いる。この週間天気予報データは、神戸地方気象台より、気象庁保有の情報を提供して頂いた。また、週間天気予報データから熱中症搬送者数を予測する場合、中期の予測評価のため、予報発表日の 4 日後から 7 日後のデータを用いる。

表 10 熱中症搬送者数の予測精度

元データ	MAE	RMSE
気象データ	2.45	3.41
4 日後予報	2.89	4.23
5 日後予報	2.93	4.30
6 日後予報	2.98	4.29
7 日後予報	2.79	3.87

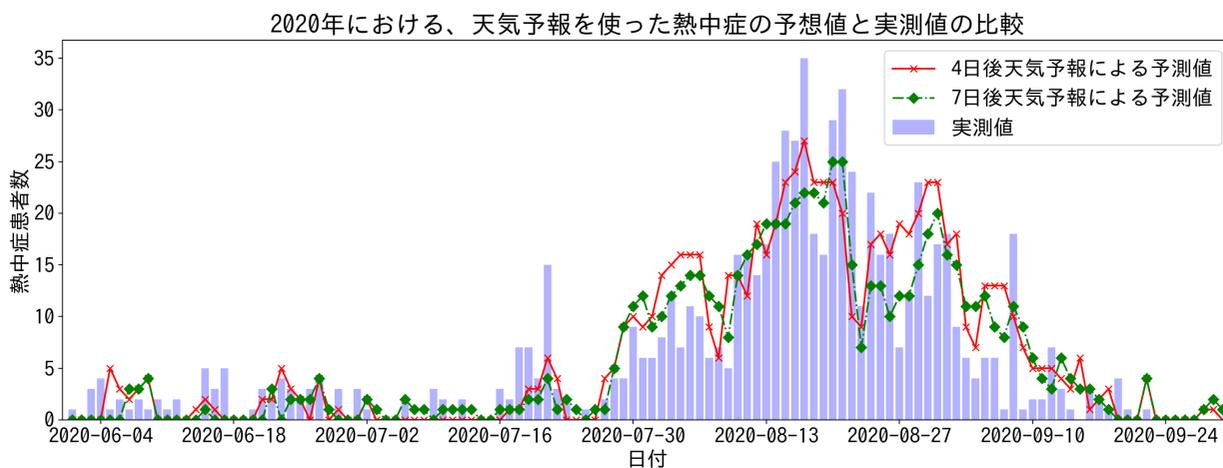


図 29 週間天気予報を用いた、2020 年における熱中症搬送者数の予測値

表 10 は、気象データを用いた場合と、特定日後の天気予報を活用した予測の誤差を評価する指標を表している。また、図 29 は、4 日後と 7 日後の天気予報を利用した、熱中症搬送者数の予測値と実測値を表している。これから、実際の気象データを用いた予測より、週間天気予報を活用した予測の方が、予測精度が悪いことが分かる。しかしながら、4, 5, 6, 7 日後予報の平均誤差は 2.90 と、大きな違いはなかった。

また、7 日後予報を用いた予測は、他の日より精度が高いことが分かる。これは、天気予報誤差が影響している。8 月 27 日を例として挙げると、当日の気温の実測値が高く、経験則からも熱中症発症のリスクが高いと考えられるが、実際熱中症者数は少なかった。そのため、気象データを用いた予測では、誤差が大きく出ている日であった。一方で 7 日後の天気予報では、気温の実測値より約 2℃ほど低く予報していたため、他より誤差が小さくなったと考えられる。そのため、天気予報の誤差により、予測精度にブレが生じる可能性が見られた。

以上より、中期の熱中症搬送者数の予測はある程度可能であると考えられる。

4.4 RQ2: 神戸市において、RQ1 での提案手法は有効か？

本節では、構築した熱中症搬送者数の予測モデルを活用できる Web アプリケーション *HSP (HeatStroke-Prediction)* を提案する。

4.4.1 アプリケーションが持つ機能

HSP は、以下の 4 つの機能を持つ。

- A1: 熱中症搬送者数の予測値推移の表示：ユーザが HSP にアクセスした時、現在公開されている気象庁の週間天気予報から予測した熱中症搬送者数の値を、直近 1 週間の値を線グラフで表示する。また、天気予報の誤差を考慮した、予測値の値

域を同様に表示する。

A2: 天気と予測値の表示：現在公開されている週間天気予報と、同日の熱中症搬送者数の予測値を表示する。

A3: 自由入力による予測：ユーザが自由に気温、天気などを入力し、熱中症搬送者数の予測を行う。予測に必要な説明変数の計算はサービス内部で行うため、ユーザは実際の気象データを入力するのみでよい。

A4: 特定日時における予測の表示：ユーザが入力した日時における週間天気予報データを用いて、A1 及び A2 の機能を実行する。

A1 及び A2 によってユーザは、簡単に中期の熱中症搬送者数予測を見ることが可能になる。A3 によって、今後の気象が変化した場合のことを、自由に分析を行うことが可能になる。A4 によって、過去の予測の振り返りや精度の評価をユーザ自身が行うことが可能になる。

4.4.2 予測値の計算

熱中症搬送者数の予測は、アプリケーションにアクセスした時点で発表されている週間天気予報を、4.3.3 と同様の方法で計算する。週間天気予報は、気象庁が発表するデータをサーバで取得し、予め予測値を計算し、週間天気予報と共にデータベースに保存しておく。これにより、アプリケーションにアクセスした時、待ち時間なくデータを表示できる。

また、週間天気予報には、予報日の 2 日後以降の気温の予測範囲が含まれる。気温の予測範囲を活用して、予測値の値域を計算する。熱中症搬送者数の予測値の上端を計算する場合、現時点の予報より暑くなったと仮定し、 $UT_{max}(d)$ 及び $UT_{min}(d)$ を用いて説明変数を計算する。逆に予測値の下端を計算する場合、現時点の予報より涼しくなったと仮定し、 $LT_{max}(d)$ 及び $LT_{min}(d)$ を用いて説明変数を計算する。

4.4.3 利用した技術

HSP のプロトタイプは Web アプリケーションとして実装されている。クライアント側には HTML5 と JavaScript を利用している。サーバ側には、Spring boot という Java フレームワークを利用し、今後の機能拡張に備えている。週間天気予報データの取得、予測モデルの活用には Python を用いた。データベースは MongoDB を、Web サーバには Apache 2.4.37 を、グラフ表示ライブラリは Chart.js を利用した。

表 11 実装に用いた技術

クライアント	HTML5, JavaScript
サーバ	Java, Spring boot, Python
データベース	MongoDB
Web サーバ	Apache 2.4.37
ライブラリ	Chart.js

4.4.4 実際の画面

図 30 に、アプリケーションの画面の上部を示す。図の上側には、週間天気予報から計算した、神戸市における 1 週間の熱中症搬送者数の予測が表示されている。週間天気予報を発表した日を含む 7 日後までの、各日の予測値の推移を、赤の実線で表している。加えて、4.4.2 における予測値の値域を、赤の点線及び赤の背景色で表している。

図の下側には、週間天気予報と予測値を表形式で表している。2 日後以降の予測値には、値域の具体値を表記している。ここで掲載しているデータは、アクセス日時時点で気象庁ホームページから得られるデータである。

また、図 31 に、アプリケーションの画面の下部を示す。図の上側には、ユーザが自由に気象情報を入力して、熱中症搬送者数を予測するフォームが表示されている。このフォームに、気温や天気などの項目を入力し予想ボタンを押すことで、特徴量をサービスで計算し、構築した予測モデルを用いて予測値を得ることができる。また、4.3.2 で導入したヒューリスティックについて、雨フラグという形で、適用するか否かを選択することができる。

図の下側には、ユーザが年月日を入力できるフォームが表示されている。ここに年月日を入力し更新ボタンを押すことで、入力された年月日に発表された週間天気予報をデータベースより取得する。そして、熱中症搬送者数の予測、およびその時点での気温などを表示する。ただし、データベースに週間天気予報が保存されていない年月日を入力した場合は、エラーを返す。

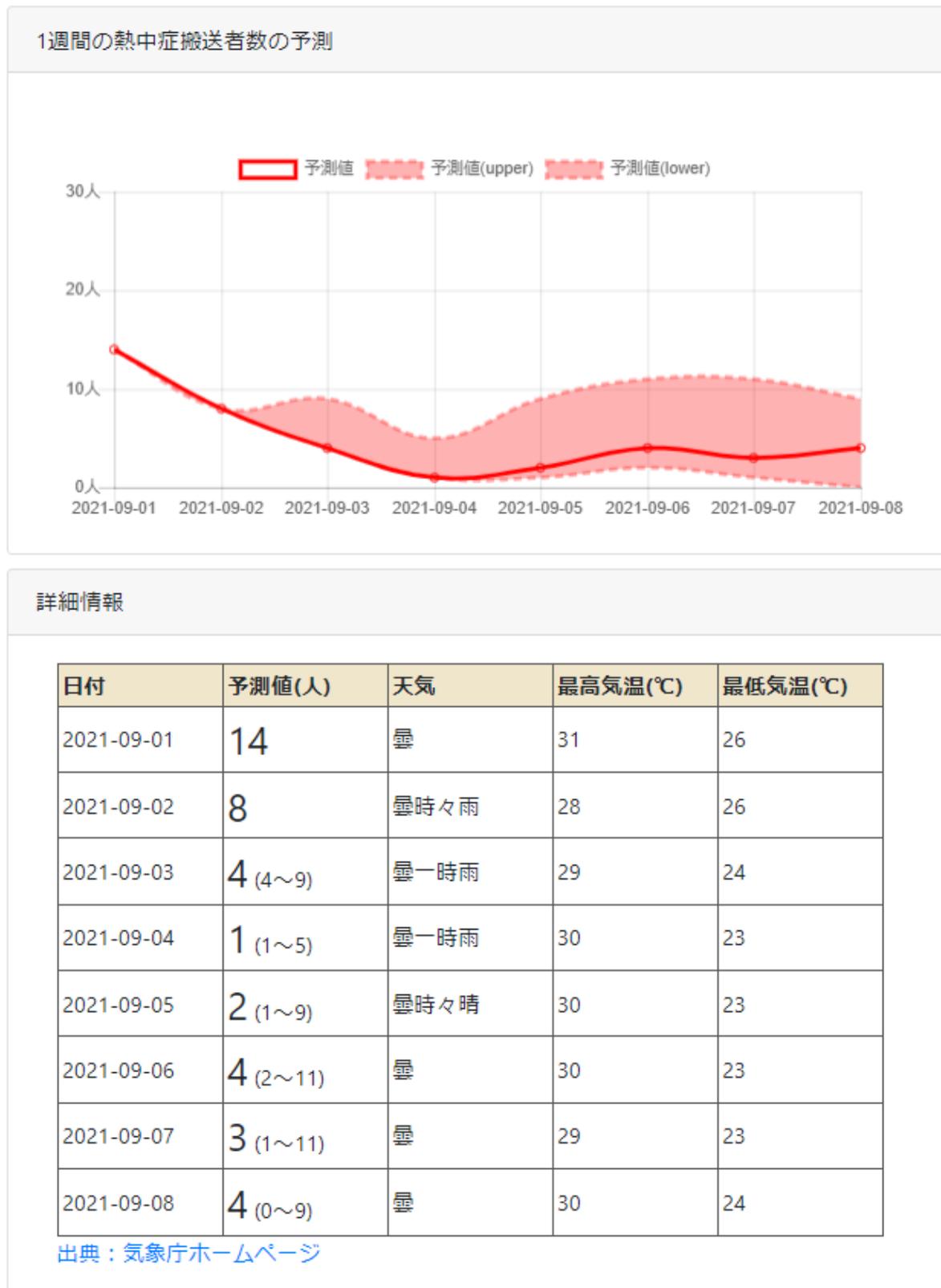


図 30 実際の HeatStroke-Prediction の画面 1

4.4.5 フィードバック

本アプリケーションの有効性を確認するため、神戸市の消防署に勤務し、救急搬送業務に携わる救急隊員に毎日確認して頂いた。加えて、神戸市消防局にてデータ分析や広報に携わる局員にも、同様に毎日確認して頂いた。期間は、2021年7月20日から7月31日までの12日間とした。その後、アプリケーションの使用感や予測値に対する、意見聴取を行い、フィード

気象情報から熱中症を予想する

最低気温(°C) :

最高気温(°C) :

1日前の最高気温(°C) :

2日前の最高気温(°C) :

天気(午前) : 晴れ 曇り 雨

天気(午後) : 晴れ 曇り 雨

雨フラグ :

• 雨フラグ: 少しでも雨が降った、降りそうな場合にチェックを入れる

予想 結果 : 19.0人

特定日時の天気予報を呼び出す

更新

図 31 実際の HeatStroke-Prediction の画面 2

バックを頂いた。

救急搬送業務に携わる救急隊員からのフィードバック

- 予測結果の表示が見やすく、特に問題はない。
- 現場における準備は、熱中症搬送者が増加することも想定しているため、予測データにあわせて変更や増強を行うまでは至らなかった。
- 夏場は気温が低くても、熱中症搬送数が少なくても、救急車に熱中症対策の資材を常時搭載している。

データ分析や広報に携わる局員からのフィードバック

- 気温と予測値の関係は経験則通りであった。

- 予測が具体的な数値で出ていることから、根拠に基づく市民向けの注意喚起に活用できる可能性がある。
- 未来の明らかなリスクが見えていれば、リスクヘッジのため早期に行動することもできる。

得られたフィードバックより、線グラフによる予測値とその値域の表示の有効性が示された。また、予測モデルに対する精度にも、肯定的な評価を得ることができた。

しかしながら、救急搬送業務の現場においては、本アプリケーションによる有益な効果を与えることはできなかった。これに対して局員から「ここ数年の夏場の急激な気温上昇や熱中症搬送者の急増を受け、救急隊員も含め、万全の準備をしている。そのため、現場救急隊員への予測データの提示は、あまり効果は得られないかもしれない。」という追加の意見を頂いた。これより、現場の隊員に対しては、熱中症搬送者数の予測は参考程度の情報であると考えられる。

一方で、データ分析や広報に携わる局員のフィードバックより、本アプリケーションの活用先を見出すことができた。活用例として、「熱中症が多く出ると予想される日に野外イベントを行う場合、数値的根拠を示して熱中症対策を厳重にしよう」や、「計画されている行事などにおいて、事前に熱中症対策の広報を行うことや、行事のやり方の変更を推奨する」などが挙げられた。

このように、ユーザに分かりやすく予測結果を提示することが、ピンポイントの熱中症予防施策を検討することに繋がることから、本アプリケーションの有効性が示されたと考えられる。

5 まとめ

本研究では、消防局員を対象とした、救急ビッグデータ分析・活用プラットフォームに開発に関する研究として、ICT システムに対する専門的な知識を持たない消防局の職員であっても、救急ビッグデータの可視化・分析を様々な視点から行うことが出来るアプリケーションプラットフォームの構築に向けた取り組みを行った。研究では、神戸市消防局と共同研究を行い、実際に現場での活用などを意識した開発を行った。

まず、救急出動記録の時空間情報への変換と救急車の動態可視化では、消防署職員が過去の出動データを振り返りやすくするために、救急出動記録を可視化するシステム ADR を提案、実装した。ADR では、出動記録に含まれる、各イベントの発生時刻、消防署の位置、現場の位置、病院の位置等の情報をもとに、任意の分単位の時刻における各救急隊の動態とおおよその位置を計算し、救急隊の動態を表す時空間情報を生成した。この時空間情報を時系列で地図上に可視化することで、どの救急車がいつどこにどんな動態にあったかをアニメーションで振り返ることが可能になった。さらに、実際に利用することで、可視化することの意義を確認した。

次に、様々な時間粒度で救急活動を振り返るアプリケーションでは、救急車の過去の出動状況をグラフに可視化するシステム ASR を提案、実装した。まず、救急出動記録から各日時の出動隊数を計算し、データベースに保存した。これを分単位で行った後、そのデータを利用して時間単位、日単位でのデータを作成した。ASR はこのデータを API を用いて取得し、web 上に横軸を時間、縦軸を出動数としたグラフ化して表示する。また、開発したシステムを用いて 2013 年～2020 年のデータを可視化した。さらに、開発したシステムを消防局職員の方々に実際に使用していただき、評価を行った。ASR により、出動が多い日、時間など、特徴的な日時の把握が可能になった。

また、消防局の構成を分析・シミュレーションするツールの研究開発では、消防局の構成の分析・シミュレーションを支援するツール FD-CAST を提案した。FD-CAST は、消防局の構成データと町丁目データから、消防署から各町丁目への距離とおおよその所要時間を算出する。これらのデータに基づいて、各町丁目で火災が発生した際に、どの車両が何秒で駆けつけるかを計算によって求め、結果を表形式および地図形式で可視化する。また、ユーザは消防署と車両の情報を変更することで、様々な構成での駆けつけをシミュレーションできる。また、FD-CAST のプロトタイプを実装し、現実的な事例を題材としたケーススタディを行った。FD-CAST により、消防構成の変更が与える影響を把握し、消防署新設等の検討が容易になった。

そして、熱中症予測に関するアプリケーションの研究開発では、最初に中期の熱中症搬送者数を予測する予測モデルの構築を行った。予測モデルは、天気、最高気温、最低気温の情報から、地域特性を持つ新たな特徴量を計算し、説明変数として導入することで構築した。実際の週間天気予報を用いて精度分析を行ったところ、7 日後の熱中症搬送者数を平均誤差 2.79 人で予測できた。次に、構築した予測モデルと週間天気予報を活用した、1 週間の予測値を表示する Web アプリケーション HSP のプロトタイプ開発を行った。アプリが現在の週間天気予報の取得、予測値の出力、線グラフ及び表へのデータ出力を行うため、ユーザは HSP にアクセスするだけで、中期の熱中症搬送者数の予測値を確認できる。HSP により、数値的根拠を元にしたピンポイントの熱中症予防施策を検討するといった、救急施策への補助が行えるようになった。

今後は本研究の成果をもとに、様々な消防局の活動をさらに支援するシステムの開発などを継続して行っていく予定である。

参考文献

- [1] Chart.js. <https://www.chartjs.org/>.
- [2] Phpoffice. <https://github.com/PHPOffice>.
- [3] 気象庁 過去の気象データ検索. <http://www.data.jma.go.jp/obd/stats/etrn/index.php>. (Accessed on 10/12/2020).
- [4] 気象庁 週間天気予報. <http://www.jma.go.jp/jma/kishou/known/kurashi/shukan.html>. (Accessed on 07/23/2021).
- [5] Google Developers. Distance matrix api. <https://developers.google.com/maps/documentation/distance-matrix/overview>.
- [6] Google Developers. Maps javascript api. <https://developers.google.com/maps/documentation/javascript/overview>.
- [7] Sachiko Kodera, Taku Nishimura, Essam A Rashed, Kazuma Hasegawa, Ichiro Takeuchi, Ryusuke Egawa, and Akimasa Hirata. Estimation of heat-related morbidity from weather data: A computational study in three prefectures of japan over 2013–2018. *Environment international*, 130:1–9, 2019.
- [8] Tetsuya Nakai, Sachio Saiki, and Masahide Nakamura. スマートシティ・アナリティクスのための細粒度地図彩色 web サービスの提案. LOIS, March 2020.
- [9] YOLP(地図). Yahoo!ジオコード api - yahoo!デベロッパーネットワーク. <https://developer.yahoo.co.jp/webapi/map/openlocalplatform/v1/geocoder.html>.
- [10] 環境省. 熱中症環境保健マニュアル 2018. https://www.wbgt.env.go.jp/heatillness_manual.php. (Accessed on 10/12/2020).
- [11] 国土交通省. Gis ホームページ. <https://nlftp.mlit.go.jp/>.
- [12] 消防防災科学センター. 消防力の適正配置等事業. "https://www.isad.or.jp/to_fire_department/proper_placement/".
- [13] 神戸市. 神戸市町名一覧表. <https://www.city.kobe.lg.jp/a53715/kurashi/registration/jukyo/ichiran.html>.
- [14] 神戸市. 神戸市：令和元年の消防局の災害・救急出動状況（速報）. <https://www.city.kobe.lg.jp/a21572/fire119/279483012916.html>, January 2020.
- [15] 総務省. 平成 29 年版 救急・救助の現況. http://www.fdma.go.jp/neuter/topics/houdou/h29/12/291219_houdou_2.pdf, December 2017.
- [16] 渡辺幸三. 生産管理・原価管理システムのためのデータモデリング. 日本実業出版社, 2002.

この研究は一般財団法人救急振興財団の「救急に関する調査研究事業助成」を受けて行ったものである。