

# LLM を用いた交通分野固有表現抽出データセットの自動構築

井田龍希 西出隆盛 三輪誠

豊田工業大学

{sd24501,sd21066,makoto-miwa}@toyota-ti.ac.jp

## 概要

LLM を用いた交通分野における新しい固有表現抽出データセットの自動構築について提案する．既存の FindVehicle は 15 種類の限られたテンプレートに固有表現を置換して作成されており，実際の交通状況で見られる多様な表現を十分にカバーしていない．そこで本研究では，FindVehicle で利用された固有表現辞書を活用し，大規模言語モデルにより固有表現を含む文を生成することで，より自然で多様なデータを作成する．生成データの品質を人手で検証し，FindVehicle のみでは対応できない多様な文脈に対する固有表現抽出を実現した．

## 1 はじめに

自動運転システムに大規模言語モデル (Large Language Models; LLM) の判断能力を利用する研究が進められており [1, 2]，それに伴い交通分野における，より正確な言語理解が必要とされている．運転状況の記述テキストには，車両情報・数値情報・位置情報など，様々な固有表現が含まれている．これらの固有表現の適切な抽出は，類似した運転状況の検索や分析において重要な役割を果たし，自動運転システムの安全性向上や交通事故防止の観点からも，運転状況に含まれる固有表現を正確に識別し，構造化された情報として扱う必要がある．

交通分野の固有表現抽出 (Named Entity Recognition; NER) においては FindVehicle データセット [3] が提案されており，近年の固有表現抽出の評価におけるデファクトスタンダードとなっている [4, 5, 6, 7]．このデータセットは，15 種類の限られたテンプレートにおける固有表現 (Named Entity; NE) 部分の置換によって作成されており，実際の交通状況の記述に見られる多様な表現に対応できない．

本研究では，FindVehicle に代わる交通分野の汎用的な固有表現抽出モデルの開発・評価をできるデータセットの構築を目指す．具体的には，図 1 に示す

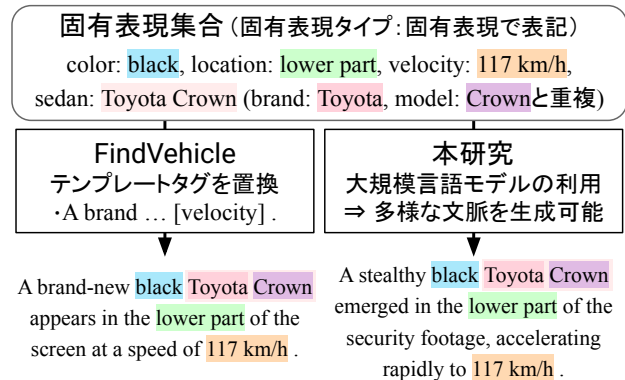


図 1 FindVehicle と提案手法におけるデータ作成の比較

通り，FindVehicle から収集された固有表現辞書と LLM を用いて，多様な文脈を持つ交通分野の固有表現抽出データセットを構築する．近年，固有表現集合から LLM を用いて文を生成する拡張手法が提案されているが，この手法では固有表現の単純な分布のみに注目し，組み合わせの自然さを考慮しない．そこで，組み合わせの自然さを考慮したデータ生成をして，人手による品質評価により生成データの自然さを検証する．

## 2 関連研究

### 2.1 FindVehicle データセット

交通分野における固有表現抽出データセットとして，FindVehicle データセット [3] が広く用いられている．このデータセットは，交通現場で撮影された車両の検出・追跡用の UA-DETRAC [8] の画像を基に作成された道路上の車両の説明文を対象としている．車両の種別・色・ブランド・モデルに加えて，車両の位置や向き・距離・速度を含む 8 種類の粗い固有表現タイプ，および具体的な車両に対して，セダンやトラックなどの車両タイプを表す 13 種類の細かい固有表現タイプをカバーしている．また，“Toyota Crown” 全体が sedan タイプの車両，“Toyota”

はブランド, “Crown” はモデルとしてラベル付けされるように, 粗い固有表現タイプと細かい固有表現タイプの間で重複した固有表現も含まれている。

FindVehicle では, 画像内の車両に対してその属性情報を表にまとめ, それらの属性を一般的な表現へ言い換えることで, 多様な固有表現をカバーしている。特に, ブランド・モデル・車両タイプに関しては, 自動車に関する専門知識を持つ 3 名の協力者による広範な知識と Wikipedia の情報から収集した 65 のブランド・4,793 のモデル・13 の車両タイプを含む包括的な多様性のある固有表現辞書を構築している。一方で, 実際の固有表現抽出データセットを作成する際には, 4 名のボランティアが作成した記述文のテンプレートをベースに, 自動生成フレームワークを用いて固有表現を挿入する手法が採用されており, 実際の交通状況記述に見られる多様な表現を十分にカバーできていない<sup>1)</sup>。

## 2.2 LLM を用いた NER データ拡張

LLM を用いた固有表現抽出データセットの拡張手法が提案されている。主に, 大規模なコーパスからサンプリングした文を LLM に入力して固有表現を抽出する手法 [5, 6] と既知の固有表現集合を入力してそれらを含む自然な文を生成する手法 [9, 10] があるが, ここでは提案手法と関連のある後者の手法について説明する。既知の固有表現集合を利用する手法では, 固有表現タイプを選択して, そのタイプに対応する固有表現辞書から固有表現集合をサンプリングする。この際, 既存研究 [10] では, 固有表現タイプの選択を元データセットの分布に従う方法と一様にサンプリングする方法を比較している。その結果, 低頻度の固有表現ラベルを含む不均衡なデータセットにおいて, 一様なサンプリングが一部の指標で良好な結果を示している。この研究では, データセット全体における固有表現タイプの分布のみ考慮しており, 個々の文における固有表現タイプの組み合わせの自然さは考慮できていない。

## 3 データセットの自動構築

本研究では, FindVehicle から収集した固有表現辞書を活用し, LLM による文生成を通じて, 多様な文脈を持つ新たな交通分野の固有表現抽出データセットを構築する。データセット構築プロセスは, 固有

表現集合の選択と, 選択された固有表現集合を含む文の生成という二つのステップで構成される。それぞれの詳細について, 3.1 節と 3.2 節で述べる。

### 3.1 固有表現集合の選択

固有表現集合の選択では, 構築した固有表現辞書から複数の固有表現の組み合わせを生成する。2.2 節で説明した既存研究 [10] とは異なり, 本研究では文レベルでの固有表現の組み合わせの自然さも考慮する。以下の三つのアプローチを採用し, 各手法によって生成されるデータの特徴および品質を比較する。以降, 各手法は略称で表記する。

- Example-based Generation (EG) : FindVehicle の各事例から得られた固有表現の組み合わせをそのまま活用する手法
- Statistics-based Generation (SG) : FindVehicle の各事例における固有表現タイプの出現頻度に基づいて固有表現辞書から新たな固有表現をサンプリングする手法
- Unconstrained Generation (UG) : FindVehicle の各事例に含まれる固有表現数だけ固有表現辞書からデータセット全体における固有表現タイプの分布に従ってサンプリングする手法

これらの手法は, 既存データセットの特性をどの程度活用するかという点で異なる。EG は既存の事例の固有表現の組み合わせをそのまま活用することで組み合わせの自然さを保証し, SG は固有表現タイプ間の共起関係を維持しながら新しい組み合わせを生成する。一方, UG は完全に自由な組み合わせを生成することで, よりデータの多様性が増すことを期待する。なお, 車両タイプは, ブランド名とモデル名を別々でサンプリングして “BMW Crown” のような不適切な固有表現が生成されないように, 適切な固有表現をまとめてサンプリングする。

### 3.2 プロンプト設計と生成制御

プロンプトは図 2 に示す通り, 三つの要素から構成される: (1) 基本的なインストラクション, (2) Few-shot サンプル, (3) 詳細な生成制御指示である。

インストラクションでは, LLM を「交通に関する自然な文を生成する専門家」として位置づけ, 入力された固有表現とそのタイプから適切な文を生成するという基本タスクを明示する。入力は [(“Toyota Crown”, “sedan”)] のような固有表現とそのタイプ

1) 原著論文では, 完全に人手で複雑なデータも作成したと記述があるが, 実際には固有表現以外の単語レベルでの置換にとどまり, テンプレートは表 3 に示す 15 種類のみである。

表 1 各手法による生成例。背景色は車両の色・タイプ・位置・速度を表す。EG は FindVehicle の固有表現の組み合わせを維持、SG は共起関係に基づく組み合わせ、UG はデータセット全体の分布に従う組み合わせによる生成を示す。

| 生成手法 | 生成例   |
|------|---|
| EG   | A distinctive silver BAIC Saab X5 caught everyone’s attention as it raced across the top of the surveillance frame at 108 kilometers per hour.  |
| SG   | Heavy rain obscured the view as a fuchsia pink vintage-car emerged from the top of the frame, while a barely visible pitch dark pedestrian passed through the top left section of the surveillance feed.  |
| UG   | The busy classic car show featured several eye-catching vehicles, including a soft red and fuchsia-pink display section, where a meticulously restored Chevrolet advance design with a fresh coat of soft-red paint drew admiring glances from enthusiasts. |

のペアをリスト形式で与える。

Few-shot サンプルは、FindVehicle 由来の静的サンプルと LLM 自身が過去に生成した動的サンプルの二種類を組み合わせている。静的サンプルは、固有表現タイプの網羅性と重複する固有表現を含む事例などを考慮して 7 件利用する。動的サンプルでは、常に同じサンプルのみを与えると生成結果が似通ってしまうという観察から、多様な生成を可能にするために最大 10 件追加する。各生成文では、固有表現を<ne type=‘sedan’>Toyota Crown</ne>のようなタグで明示的に組み込むように指示する。

詳細な生成制御では、サンプルの単純な模倣を避けた文生成などの制約を設ける。

## 4 実験

構築したデータセットの有効性を検証するため、生成データセットの品質評価と生成データを用いて追加学習した固有表現抽出モデルによる予測結果の解析の二つの側面から分析を実施する。

### 4.1 データセットの前処理と作成

FindVehicle の公開形式<sup>2)</sup>から JSON 形式を採用し、重複する文のアノテーションを統合して 42,342 文から 22,132 文の一意なデータを作成した。また、データセット中には位置を表す“BOTTM”が大文字で表記されているなど、不自然な大文字使用が含まれるため、すべての文を小文字に統一して処理した。

生成データの作成にあたり、LLM の選定では Claude3.5-Sonnet [11] と GPT4o [12] の出力文を比較して、より多様な文生成が可能であった Claude3.5-Sonnet を採用した。FindVehicle の訓練・開発・評価用データからそれぞれ 1,000 件、100 件、100 件をサンプリングし、これらの事例を利用し、3.1 節で述べた EG・SG・UG を適用して生成した。なお、固有

表現辞書は訓練用データのみを用いて構築した。

### 4.2 データセットの評価

生成データの品質評価として、各サンプリング手法により生成した評価用データに対して分析を実施した。まず、生成文が単純なテンプレートの置換に留まっていないかを確認し、多様な文脈が含まれていることを確認した。

表 1 に示す通り、SG の例では車両と歩行者を同時に含む文、UG では車両展示会における複数の車両の色を含む文など、生成されたデータには FindVehicle には見られない多様な文脈が含まれていた。一方で、元のデータから離れるほど、交通状況から逸脱した展示会などの文脈が生成されたり、FindVehicle の設定では車両タイプとしてラベル付けされるべき“eye-catching vehicles”のような表現がラベル付けされていない<sup>3)</sup>などの課題も確認された。

生成データの品質を定量的に評価するため、自然言語処理を専門とする学生 2 名により、EG・SG・UG の各生成手法から無作為に抽出した 20 文ずつ、計 60 文に対して評価を実施した。評価では brat [13] を用いて生成データを可視化し、(1) 付与された固有表現ラベルの適切性、(2) ラベル付けされるべき表現の見落としの二点について分析を行った。評価の結果、LLM によって付与された固有表現ラベルの適合率は、EG が 96.4%、SG が 95.6%、UG が 95.6%であった。また、再現率は、EG が 93.9%、SG が 92.3%、UG が 87.1%であった。多く見られた誤りとして、人や建物など車両以外の対象の位置や色への固有表現ラベルの付与が挙げられる。また、特に UG では与えられる固有表現集合にタイプの偏りがある場合に、車両に関する固有表現の見落としが発生する傾向が強かった。これらの結果から、EG は FindVehicle の固有表現の組み合わせに基づくため高

2) CoNLL と JSON の二つの形式で公開されており、内容が異なる。 <https://github.com/GuanRunwei/FindVehicle>

3) FindVehicle では、“van”や“truck”などの一般名詞も車両タイプの固有表現として扱うため、“eye-catching vehicles”も固有表現としてラベル付けされるべきである。



い適合率と再現率を維持できる一方で表現の多様性に制限があり、UG は多様な文脈を生成できる一方で固有表現のアノテーション品質の管理が困難であることがわかった。また、SG は EG と UG の中間的な特性を示すことがわかった。

### 4.3 生成データを用いたモデルの評価

GLiNER [7] の `urchade/gliner_large-v2.1` をベースラインモデルとして採用した<sup>4)</sup>。評価実験では、このベースラインモデルに対して、FindVehicle および EG・SG・UG の生成データを学習データとして使用するかどうかなの全組み合わせについて実験を行い、FindVehicle および生成データの両方で性能を評価した。なお、FindVehicle を使用する設定では、FindVehicle で事前に学習したモデルで初期化し、その後 FindVehicle と生成データを組み合わせて学習を行った。評価指標には予測と正解の完全一致に基づくマイクロ F 値を採用した。モデルの性能比較結果とその考察を付録 B に示す。

生成データを用いて学習したモデルの実用性を検証するため、人手で作成された実データにモデルを適用してその結果を解析した。具体的には、自動運転車両とのインタラクションの実現に向けて、画像に対する人からの自然言語による指示文を収集した Talk2Car データセット [14] の指示文を活用した。

20 文に対する予測結果についてアノテータが妥当と判断する固有表現を対象に部分一致も正解として扱うという設定で人手評価を実施した。その結果、FindVehicle のみで学習したモデルでは F 値が 50.8%、生成データを追加学習したモデルでは 68.8% となり、生成データの追加により実データに対する性能向上が確認された。実際の予測結果は、表 2 に示す通りであり、事例 1 では、人物の位置を表す “up ahead” を車両位置として誤って抽出した一方、“slow down” を車両速度として正しく抽出できている。また事例 2 では、“bicycle rack” という文脈において “bicycle” を車両として誤って抽出したものの、駐車位置を示す “next to it” を車両位置として正しく抽出することができた。これらの結果は、多様な表現をカバーする生成データにおける学習により多様な表現からの抽出が可能になった一方で、車両に関係しない表現に対する誤抽出が増える傾向があることを示している。この課題は、生成データにお

表 2 Talk2Car における抽出結果の比較。FindVehicle のみで学習したモデルと生成データで追加学習したモデルの比較を示す。背景色は改善・改悪を示す。

|   |  |
|---|--|
| 事例 1：“up ahead” を誤抽出，“slow down” を正しく抽出 |  |
| 入力文                                     | Watch out for the person crossing up ahead, slow down. |
| FV のみ                                   | 抽出なし   |
| +生成データ                                  | [車両位置：up ahead, 車両速度：slow down,                        |
| 事例 2：“bicycle” を誤抽出，“next to it” を正しく抽出 |  |
| 入力文                                     | There is a bicycle rack. Park next to it.              |
| FV のみ                                   | 抽出なし   |
| +生成データ                                  | [車両種類：bicycle, 車両位置：next to it]                        |

いて車両以外の対象に固有表現が付与されている文や交通分野以外の文が生成されてしまうことに起因すると考えられる。解決策として、生成データのフィルタリングや、LLM による反復的な固有表現ラベル付与の検討が必要である。また、これらの予測結果の変化は、主に SG・UG を追加した場合に観察され、EG のみの追加ではあまり見られないことから、SG・UG の利用により多様な文脈に対応できるようになったと考えられる。

## 5 おわりに

本研究では、FindVehicle データセットの固有表現辞書と LLM を用いて多様な文脈を持つ交通分野における固有表現抽出データセットを自動構築した。固有表現の組み合わせ方として、FindVehicle の組み合わせを活用する EG、共起関係に基づく SG、制約のない UG を比較した。人手評価の結果、EG は高い品質を維持できる一方で表現の多様性に制限があり、UG はその逆の特性を示した。また、生成データを用いて学習した固有表現抽出モデルを Talk2Car データセットで評価した結果、FindVehicle のみで学習したモデルと比較して、多様な表現からの抽出が可能になる一方で、車両以外の対象への誤抽出の増加が確認された。今後の課題として、生成データのフィルタリングや LLM による反復的な固有表現ラベル付与による品質向上、および車両以外の対象への誤抽出を防ぐ手法の開発が挙げられる。また、本研究では固有表現辞書を FindVehicle から再利用したが、より広範な交通分野の固有表現をカバーするために、新たな辞書の構築も検討する必要がある。

4) NER の学習の実装、学習設定は <https://github.com/urchade/GLiNER> を使用。

## 参考文献

- [1] Zhenjie Yang, Xiaosong Jia, Hongyang Li, and Junchi Yan. LLM4Drive: A survey of large language models for autonomous driving, 2023.
- [2] Can Cui, Yunsheng Ma, Xu Cao, Wenqian Ye, Yang Zhou, Kaizhao Liang, Jintai Chen, Juanwu Lu, Zichong Yang, Kuei-Da Liao, Tianren Gao, Erlong Li, Kun Tang, Zhipeng Cao, Tongxi Zhou, Ao Liu, Xinrui Yan, Shuqi Mei, Jianguo Cao, Ziran Wang, and Chao Zheng. A survey on multimodal large language models for autonomous driving. **2024 IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW)**, pp. 958–979, 2023.
- [3] Runwei Guan, Ka Lok Man, Feifan Chen, Shanliang Yao, Rongsheng Hu, Xiaohui Zhu, Jeremy Smith, Eng Gee Lim, and Yutao Yue. FindVehicle and VehicleFinder: a NER dataset for natural language-based vehicle retrieval and a keyword-based cross-modal vehicle retrieval system. **Multimedia Tools and Applications**, Vol. 83, No. 8, pp. 24841–24874, 2024.
- [4] Xiao Wang, Weikang Zhou, Can Zu, Han Xia, Tianze Chen, Yuansen Zhang, Rui Zheng, Junjie Ye, Qi Zhang, Tao Gui, et al. InstructUIE: Multi-task instruction tuning for unified information extraction. **arXiv preprint arXiv:2304.08085**, 2023.
- [5] Wenxuan Zhou, Sheng Zhang, Yu Gu, Muhao Chen, and Hoifung Poon. UniversalNER: Targeted distillation from large language models for open named entity recognition. In **The Twelfth International Conference on Learning Representations**, 2024.
- [6] Sergei Bogdanov, Alexandre Constantin, Timothée Bernard, Benoit Crabbé, and Etienne P Bernard. NuNER: Entity recognition encoder pre-training via LLM-annotated data. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, **Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing**, pp. 11829–11841, Miami, Florida, USA, November 2024. Association for Computational Linguistics.
- [7] Urchade Zaratiana, Nadi Tomeh, Pierre Holat, and Thierry Charnois. GLiNER: Generalist model for named entity recognition using bidirectional transformer. In Kevin Duh, Helena Gomez, and Steven Bethard, editors, **Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)**, pp. 5364–5376, Mexico City, Mexico, June 2024. Association for Computational Linguistics.
- [8] Longyin Wen, Dawei Du, Zhaowei Cai, Zhen Lei, Ming-Ching Chang, Honggang Qi, Jongwoo Lim, Ming-Hsuan Yang, and Siwei Lyu. UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking. **Computer Vision and Image Understanding**, Vol. 193, p. 102907, 2020.
- [9] Ruixiang Tang, Xiaotian Han, Xiaoqian Jiang, and Xia Hu. Does synthetic data generation of LLMs help clinical text mining? **ArXiv**, Vol. abs/2303.04360, , 2023.
- [10] Joan Santoso, Patrick Sutanto, Billy Cahyadi, and Esther Setiawan. Pushing the limits of low-resource NER using LLM artificial data generation. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, **Findings of the Association for Computational Linguistics: ACL 2024**, pp. 9652–9667, Bangkok, Thailand, August 2024. Association for Computational Linguistics.
- [11] Anthropic. The claude 3 model family: Opus, sonnet, haiku. Technical report, 2024.
- [12] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. **arXiv preprint arXiv:2410.21276**, 2024.
- [13] Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun’ichi Tsujii. brat: a web-based tool for NLP-assisted text annotation. In **Proceedings of the Demonstrations Session at EACL 2012**, Avignon, France, April 2012. Association for Computational Linguistics.
- [14] Thierry Deruyttere, Simon Vandenhende, Dusan Grujicic, Luc Van Gool, and Marie Francine Moens. Talk2Car: Taking control of your self-driving car. In **Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)**, pp. 2088–2098, 2019.

## A FindVehicle のテンプレート

FindVehicle は表 3 に示す 15 種類のテンプレートの置換のみにより構築される。

## B モデルの性能比較

学習データの組み合わせによる性能評価の比較を表 4 に示す。FindVehicle のみで学習したモデルは生成データに対する性能が低く、テンプレートベースのアプローチの限界を示している。一方、生成データを追加して学習したモデルは、生成手法の組み合わせによって異なる特性を示した。単一の生成手法を追加した場合、各手法はそれぞれの評価データで高い性能を達成し、複数手法の組み合わせでは全体的に安定した性能を示した。

表 3 FindVehicle のテンプレート一覧。固有表現タイプを [] で示すが、これらのタグは他の固有表現タイプに置換されることもある（例：[SUV] が [sports car] 等に置換）。また、9 番における Mr Green など、固有表現以外の単語レベルでの置換も見られる。

| テンプレートパターン  |
|---|
| 1. A [color of vehicle] [vehicle type] drove [orientation of vehicle], about [vehicle range] away from me, and the speed was not fast, about [vehicle velocity], there were many green grass and purple flowers on both sides of the road, although the sky was gray, but my mood is still very good.       |
| 2. A [vehicle type] going [orientation of vehicle] at the [position of vehicle] of the image at a speed of [vehicle velocity].  |
| 3. All of a sudden, a brand-new [color of vehicle] [SUV] appears in the [position of vehicle] of the screen at a speed of [vehicle velocity].   |
| 4. Find the [vehicle type] located in the [position of vehicle] and [position of vehicle] whose colors are [color of vehicle] and [color of vehicle]. As far as I am concerned, they are [vehicle range] meters away from me.   |
| 5. I need to find out the [vehicle type] that is for util driven in [orientation of vehicle] which is in the [position of vehicle] of the image.  |
| 6. Just now, a [vehicle type] with oranges rolled over. In the [position of vehicle] of the surveillance image, all the oranges on the car fell off, and the ground was full of orange juice. There is a [vehicle type] car and an [vehicle type] driving towards you now.                                  |
| 7. Let the clever boy help find out the [color of vehicle] [bus] and [color of vehicle] [motorcycle] in the [orientation of vehicle] of the image that driven this way.   |
| 8. Maybe there is a [vehicle type] at [vehicle range] and a [vehicle type] goes [orientation of vehicle] about [vehicle range] away.  |
| 9. Mr Green is driving a [color of vehicle] [vehicle type] on the right-side of the highway, following an [color of vehicle] [vehicle type] loaded with many oranges.   |
| 10. Please find the [color of vehicle] [vehicle type] on the [position of vehicle] of the picture and the [color of vehicle] [vehicle type] on the [position of vehicle].   |
| 11. Please help me find the [color of vehicle] [motorcycle] on the [position of vehicle] of the image and the [color of vehicle] [sedan] on the [position of vehicle].  |
| 12. The suspect vehicle is currently driving in [orientation of vehicle] at [vehicle velocity], it is a white [color of vehicle] [vehicle type] from the monitoring point of view, it is currently at the bottom [position of vehicle] the screen. Please stop it as soon as possible.                      |
| 13. There is a [color of vehicle] [vehicle type] driving [orientation of vehicle] with [vehicle velocity] and a [color of vehicle] [vehicle type] driving [orientation of vehicle] at a speed of [vehicle velocity].  |
| 14. There are so many cars on the highway that I cannot recognize but I think there is a [color of vehicle] [sedan] driving [orientation of vehicle] with [vehicle velocity] km/h and a [color of vehicle] [sports car] driving [orientation of vehicle] at a speed of [vehicle velocity].                  |
| 15. Today is a rainy day, the sky is grey, that is like my mood. I am driving my [color of vehicle] [vehicle type], departing away from my friend's house. Mr Black gets up early this morning, he is also driving his big [color of vehicle] [vehicle type] to his company, rich people also have trouble. |

表 4 学習データの組み合わせによる性能比較（FV は FindVehicle, EG・SG・UG はそれぞれの生成手法を表す。）

| 学習データ |    |    |    | 評価データ       |             |             |             |             |
|-------|----|----|----|-------------|-------------|-------------|-------------|-------------|
| FV    | EG | SG | UG | FV          | EG          | SG          | UG          | All         |
| -     |    |    |    | 61.5        | 53.1        | 52.5        | 49.0        | 54.0        |
| ✓     |    |    |    | <b>97.3</b> | 81.5        | 84.4        | 81.7        | 86.2        |
|       | ✓  |    |    | 92.2        | 95.3        | 91.7        | 86.8        | 91.5        |
|       |    | ✓  |    | 87.9        | 91.0        | 93.3        | 87.3        | 89.9        |
|       |    |    | ✓  | 88.6        | 88.9        | 92.6        | 91.8        | 90.5        |
| ✓     | ✓  |    |    | 96.4        | <b>96.8</b> | 93.6        | 88.8        | <b>93.9</b> |
| ✓     |    | ✓  |    | 92.6        | 93.3        | <b>95.2</b> | 90.3        | 92.9        |
| ✓     |    |    | ✓  | 93.8        | 91.6        | 94.9        | 94.2        | 93.6        |
|       | ✓  | ✓  |    | 88.2        | 91.0        | 93.4        | 87.8        | 90.1        |
|       | ✓  |    | ✓  | 88.2        | 89.0        | 92.9        | 91.1        | 90.3        |
|       |    | ✓  | ✓  | 89.1        | 89.4        | 92.3        | 90.8        | 90.4        |
| ✓     | ✓  | ✓  |    | 91.8        | 92.7        | 95.0        | 90.0        | 92.4        |
| ✓     | ✓  |    | ✓  | 93.7        | 91.5        | 93.9        | <b>94.4</b> | 93.4        |
| ✓     |    | ✓  | ✓  | 92.5        | 91.1        | 95.0        | 94.2        | 93.2        |
|       | ✓  | ✓  | ✓  | 88.4        | 88.7        | 91.4        | 91.5        | 90.0        |
| ✓     | ✓  | ✓  | ✓  | 93.7        | 92.1        | 94.7        | <b>94.4</b> | 93.7        |

## C 生成に使用したプロンプト

データセット構築に使用したプロンプトを以下に示す。

図 2 データセット自動構築のためのプロンプト

|   |
|---|
| <p><b>基本的なインストラクション：</b><br/>You are an expert at generating natural sentences about traffic.<br/>I will provide you with named entities and their types, and you should generate a natural sentence that includes all of them.<br/>You must output only one or more sentences, and you must not use any quotation marks in the output.</p> <p><b>Few-shot サンプル：</b><br/>Input: [{"van", "vehicle type"}, {"top-left", "position of vehicle"}, {"silver", "color of vehicle"}]<br/>Output: Find the &lt;ne type='vehicle type'&gt;van&lt;/ne&gt; in the &lt;ne type='position of vehicle'&gt;top-left&lt;/ne&gt; that is &lt;ne type='color of vehicle'&gt;silver&lt;/ne&gt;.</p> <p><b>詳細な生成制御指示</b><br/>IMPORTANT: Your task is to generate a completely new and unique sentence that is DIFFERENT from all examples above.<br/>DO NOT Copy or imitate the sentence structures from examples<br/>DO NOT Use similar phrases or expressions<br/>DO NOT Follow the same patterns of description<br/>The sentence should:<br/>1. Include all provided named entities with appropriate XML-like tags<br/>2. Be natural and fluent in English<br/>3. Handle partially overlapping spans:<br/>- Some entities may share part of their text (e.g., "upper part" for position of vehicle, "upper" for orientation of vehicle)<br/>- Use nested tags to indicate overlapping spans like: &lt;ne type='position of vehicle'&gt;&lt;ne type='orientation of vehicle'&gt;upper&lt;/ne&gt;part&lt;/ne&gt;<br/>4. Only output one or more sentences<br/>5. Do not use any quotation marks in the output<br/>IMPORTANT: Only use the following entity types in your &lt;ne&gt; tags:<br/>1. Generic vehicle types (when mentioning general categories):<br/>- vehicle type<br/>2. Specific vehicle types (when mentioning actual car models):<br/>- SUV (for SUV vehicles)<br/>- hatchback (for hatchback vehicles)<br/>- sports car (for sports cars)<br/>- sedan (for sedan vehicles)<br/>- bus (for buses)<br/>- truck (for trucks)<br/>- vintage car (for vintage cars)<br/>- coupe (for coupe vehicles)<br/>- roadster (for roadster vehicles)<br/>- estate car (for estate cars)<br/>- motorcycle (for motorcycles)<br/>- MPV (for MPV vehicles)<br/>- van (for vans)<br/>3. Other types:<br/>- position of vehicle<br/>- color of vehicle<br/>- orientation of vehicle<br/>- vehicle velocity<br/>- vehicle range<br/>- brand of vehicle<br/>- vehicle model<br/>DO NOT create or use any other entity types.</p> |
|---|