REPORT

Can language models be used for real-world urban-delivery route optimization?

Yang Liu,[1] Fanyou Wu,[1] Zhiyuan Liu,[2] Kai Wang,[3] Feiyue Wang,[4] and Xiaobo Qu[3,*]

*Correspondence: xiaobo@tsinghua.edu.cn

Received: June 22, 2023; Accepted: September 27, 2023; Published Online: September 29, 2023; https://doi.org/10.1016/j.xinn.2023.100520

GRAPHICAL ABSTRACT

PUBLIC SUMMARY

- Revealing that language models can learn human behavioral sequences.
- Integrating the implicit knowledge of drivers in the route optimization process.
- A novel algorithm to optimize delivery routes emulating real-world driving behaviors.
- Broadening the scope of language models to diverse domains with certain grammar rules.

Can language models be used for real-world urban-delivery route optimization?

Yang Liu,[1] Fanyou Wu,[1] Zhiyuan Liu,[2] Kai Wang,[3] Feiyue Wang,[4] and Xiaobo Qu[3,*]

[1]State Key Laboratory of Intelligent Green Vehicle and Mobility, Tsinghua University, Beijing 100084, China

[2]Jiangsu Key Laboratory of Urban ITS, Jiangsu Province Collaborative Innovation Center of Modern Urban Traffic Technologies, School of Transportation, Southeast University, Nanjing 211189, China

[3]School of Vehicle and Mobility, Tsinghua University, Beijing 100084, China

[4]Institute of Automation, State Key Laboratory for Management and Control of Complex Systems, Chinese Academy of Sciences, Beijing 100190, China

*Correspondence: xiaobo@tsinghua.edu.cn

Received: June 22, 2023; Accepted: September 27, 2023; Published Online: September 29, 2023; https://doi.org/10.1016/j.xinn.2023.100520

Citation: Liu Y., Wu F., Liu Z., et al., (2023). Can language models be used for real-world urban-delivery route optimization? The Innovation **4(6)**, 100520.

Language models have contributed to breakthroughs in interdisciplinary research, such as protein design and molecular dynamics understanding. In this study, we reveal that beyond language, representations of other entities, such as human behaviors, that are mappable to learnable sequences can be learned by language models. One compelling example is the real-world delivery route optimization problem. We here propose a novel approach based on the language model to optimize delivery routes on the basis of drivers' historical experiences. Although a broad range of optimization-based approaches have been designed to optimize delivery routes, they do not capture the implicit knowledge of complex delivery operating environments. The model we propose integrates this knowledge in the route optimization process by learning from driving behaviors in experienced drivers. A real-world delivery route that preserves drivers' implicit behavioral patterns is first analogized to a sentence in natural language. Through unsupervised learning, we then learn the vector representations of words and infer the drivers' delivery chains on the basis of the tailored chain-reaction-based algorithm. We also provide insights into the fusion of language models and operations research methods. In our approach, language models are applied to learn drivers' delivery behaviors and infer new deliveries at the delivery zone level, while the classic traveling salesman problem (TSP) model is embedded into the hybrid framework for intra-zone optimization. Numerical experiments performed on real-world data from Amazon's delivery service demonstrate that the proposed approach outperforms pure optimization, supporting the effectiveness, efficiency, and extensibility of our model. As a versatile approach, the proposed framework can easily be extended to various disciplines in which the data follow certain grammar rules. We anticipate that our work will serve as a stepping stone toward the understanding and application of language models in tackling interdisciplinary research problems.

INTRODUCTION

Language models, originally designed to solve problems in computational linguistics, such as speech recognition,[1,2] machine translation,[3,4] and document generation,[5,6] treat human languages as sequences of words governed by specific grammar rules. By following this approach, it was revealed that entities other than languages that are mappable to learnable sequences can also be solved by language models.[7] For example, protein sequences share many similarities with human language; amino acids can be considered letters, and their multiple combinatorial arrangements can form structures with functions, i.e., words and sentences.[8,9] Similarly, human behaviors are also analogous to human languages. For example, a traveler's trip chain can be represented as a sentence consisting of departure time, origin and destination coordinates, and travel mode.[10,11] Another typical example is real-world delivery route optimization problem studied in this paper (see the research background section in the supplemental information for details), in which a delivery route is abstracted as a sentence, with its nodes represented by words. The sentence must comply with the grammar rules set up by various real-world constraints, such as an illegal U-turn, road closures for construction, and unpleasant traffic conditions.

In real-world delivery route optimization, the challenges go beyond identifying the shortest path.[12–16] They encompass intricate real-world constraints to take into account genuine delivery conditions.[17] Easa's pioneering work introduces the shortest path algorithm incorporating movement prohibitions, hence offering a simpler network representation.[18] However, the dynamic interplay between real-world factors, including traffic, parking, and customer preferences, is often neglected theoretical models. Although manual removal of the complex road segments may be feasible for small networks, it becomes unmanageable on a larger scale. Such divergence often results in drivers equipped with local knowledge deviating from optimized paths. By learning and understanding drivers' delivery behaviors, the planned routes will be refined to better accommodate complex real-world urban environment.

Motivated by recent interdisciplinary research about language models and grounded in the theoretical foundations of geometric deep learning,[19] in this paper we proposes a novel language-model-based approach to solve real-world urban delivery route optimization problem. The primary function of the language model is to learn and understand drivers' delivery behaviors, allowing high-fidelity approximations of the real-world delivery routes optimized by drivers. This approach contributes to the development of more efficient and safer routes, as well as improved service quality. A real-world delivery route is analogously defined as a sentence in language, which preserves the implicit knowledge of drivers' behavioral patterns without involving intensive computations. By learning the behaviors of experienced drivers, the proposed model facilitates the transfer of this knowledge to delivery route planning, which cannot be achieved by using only optimization-based approaches. Instead of performing sequence-to-sequence or step-by-step prediction using supervised learning, we first use unsupervised learning to learn the vector representations of words, and then design a tailored chain-reaction-based algorithm to infer the complete delivery sequence.

This study also provides insights into the fusion of language models and operations research methods. The devised hybrid architecture consists of two parts: the first part uses language model to learn drivers' delivery behaviors and infer the zone sequence of new deliveries, while the second part uses the classic traveling salesman problem (TSP) model to optimize routing within each zone in the sequence. To validate the effectiveness of the proposed model, we conduct extensive numerical experiments using actual delivery routes. The experimental results show that the proposed method outperforms pure optimization approach. It is worth noting that our approach is not limited to the urban delivery route optimization problem. Adaptations can be seamlessly implemented in interdisciplinary domains, including but not limited to understanding the dynamics of complex stochastic molecular systems[20–25] and protein design,[9,26–29] with different features, such as word encoding schemes and sentence definitions.

RESULTS

Real-world delivery routes can be mapped into sentences of language

The fundamental idea behind the proposed framework lies in the analogy that a real-world delivery route can be represented as a sentence in language. This idea stems from the principles of geometric deep learning, particularly the paradigm of "learning on string representations."[19] In this context, we equate the ID of an element in a route—which can be a node, a delivery stop, or a delivery service zone—to a word in the sentence. The rationale behind this analogy is that the generated sequences, or "sentences," have two key characteristics: syntax and semantic. Drawing an intellectual connection with molecular strings,[30] the syntax aspect dictates that not all combinations of characters result in valid molecules, analogous to how not all sequences of delivery stops form feasible routes, because of real-world limitations. Semantically, the corresponding chemical

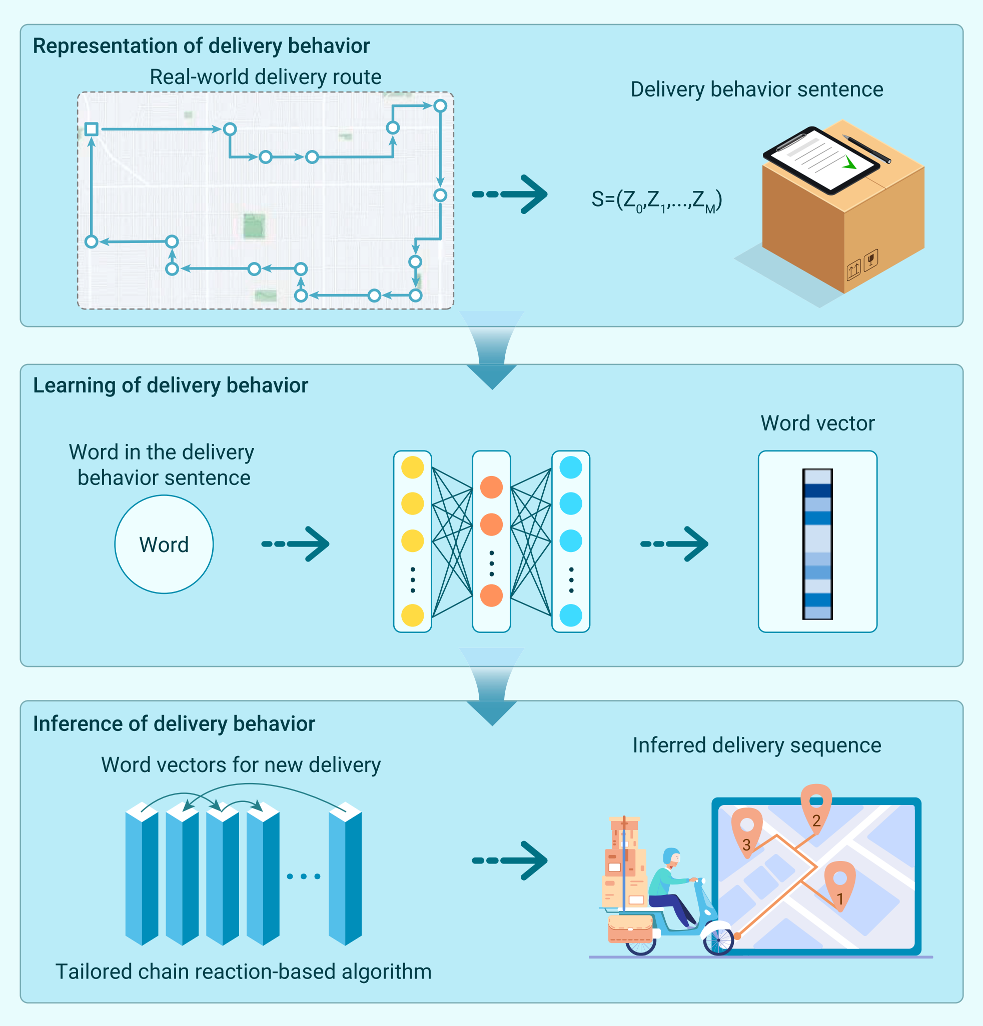
Representation of delivery behavior
Real-world delivery route
Delivery behavior sentence
$S=(Z_0,Z_1,\dots,Z_M)$
Learning of delivery behavior
Word in the delivery behavior sentence
Word
Word vector
Inference of delivery behavior
Word vectors for new delivery
Tailored chain reaction-based algorithm
Inferred delivery sequence
3
2
1

Figure 1. Flowchart of the learning-based model The proposed model consists of three main steps: (1) representing real-world delivery routes as "delivery behavior sentences," (2) learning delivery behaviors using a machine learning model, and (3) inferring delivery sequences from word vectors.

compounds possess different physicochemical and biological properties depending on how the elements of the string are combined. Likewise, different delivery routes have different efficiencies in route optimization. Hence, the arrangement of delivery stops in our "sentences" carries semantic information about the route's efficiency. The flowchart of the proposed learning-based model is illustrated in Figure 1.

In the theory of language models, it is widely acknowledged that word sequences with higher occurrence probabilities are more consistent with the grammar criteria.[31,32] Thus, the probability of observing a word sequence can function as an indicator of its grammatical correctness. For instance, assume that we obtain the statistics of a large corpus and find:

$$P(\text{join, us, in,building, a, better, world}) > P(\text{world, better, a, building, in, us, join}). \qquad \text{(Equation 1)}$$

It can be inferred that the sentence "Join us in building a better world" is more grammatically correct. Similarly, the probability of element sequences in the route is used to characterize the driver's delivery behavior pattern, i.e., the sequences with higher occurrence probabilities are more consistent with the drivers' behaviors.

In this paper, we focus on Amazon's delivery service for a better exposition. Detailed descriptions of the dataset are given in the data description section in the supplemental information, as well as in Tables S1–S3. An Amazon delivery route starts from a delivery station (i.e., depot) and then serves several delivery stops, each with a varying number of packages. To facilitate deliveries, Amazon also delineates zone IDs on the basis of the geo-planned area where the stop is located, where each zone contains a varying number of stops. Taking the delivery zone sequence as an example, it is assumed that the statistics of the real-world delivery data show:

$$P(D-3.2A, D-3.1A, \dots, D-2.3C) > P(D-2.2A, D-3.1A, \dots, D-2.1B), \qquad \text{(Equation 2)}$$

where "*D-3.2A*, *D-3.1A*,..., *D-2.1B*" are the IDs of the delivery zones in the real-world route. We can infer that zone sequence "*D-3.2A*, *D-3.1A*,..., *D-2.3C*" is closer to the real-world delivery zone sequence.

It should be noted that the words in our approach are not restricted to delivery zone IDs; they can also be fine-grained delivery stop IDs. If we analogize the delivery stop ID to an amino acid, then the use of the delivery zone ID as the word is similar to the three-dimensional (3D) secondary structural element formed by amino acids. We define the main notations used in the paper in Table S4. A real-world delivery route consists of *M* zones and one delivery station. The delivery station is the starting point of a route, and its index is 0. The zones are labeled with the number 1, ..., *M*. The real-world delivery behavior sentence is extracted from the delivery route, defined as follows:

$$s = (z_0, z_1, \dots, z_M), \qquad \text{(Equation 3)}$$

where each element of the sentence is a "word" that represents a zone or the delivery station in a route. Each sentence represents a zone sequence, with the order of the words in the sentence being the same as that of the driver's delivery. The probability that this sentence exists in $\mathcal{H}$ (i.e., a set of zone ID sequence in the training set) is:

$$P(z_0, z_1, \dots, z_M) = \prod_{m=0}^{M} P(z_m | z_0, z_1, \dots, z_{m-1}), \qquad \text{(Equation 4)}$$

where

$$P(z_m | z_0, z_1, \dots, z_{m-1}) = \frac{\text{Count}(z_0, z_1, \dots, z_m)}{\sum_{z \in D} \text{Count}(z_0, z_1, \dots, z_{m-1}, z)}, \qquad \text{(Equation 5)}$$

and z_m is the *m*th word in the sentence, and Count($\cdot$) operation indicates the number of occurrences of a subsequence in the training set.

In the actual calculation, the Markov assumption[33] is introduced to further reduce the computational effort, i.e., only the previous *k* zones are considered when calculating the conditional probabilities in Equation 6.

$$P(z_m | z_0, z_1, \dots, z_{m-1}) \approx P(z_m | z_{m-k}, \dots, z_{m-1}). \qquad \text{(Equation 6)}$$

Training the network is equivalent to learning the grammar rules of real-world logistics delivery

We use the Word2Vec approach in natural language processing (NLP)[34,35] to learn vector representations of words in delivery behavior sentences. We construct the delivery behavior sentences on the basis of the actual routes in the dataset, thereby providing an accurate reflection of the real-world delivery

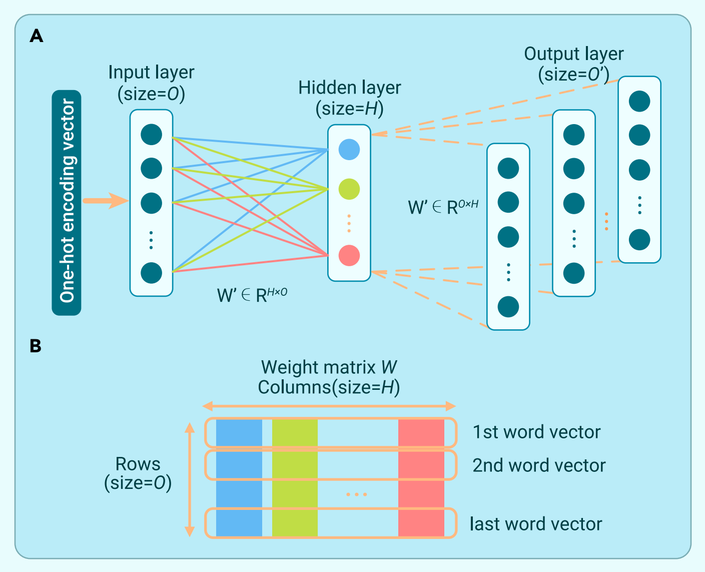

Figure 2. Visualization of the neural network and the associated weight matrix (A) Structure of the neural network. (B) Structure of the weight matrix.

behaviors of drivers. To learn the driver's delivery behavior, our objective is to learn an optimal mapping $\mathcal{F}(\cdot)$ by maximizing the co-occurrence probability of the zones in each delivery behavior sentence.

Word2Vec has two typical modeling ideas, namely continuous bag-of-words (CBOW) and skip-gram.[34,35] Taking the skip-gram structure as an example, the idea is to give a central word and then to predict its context within a certain range, as outlined by the following objective function:

$$\mathcal{L}(\theta) = \prod_{m=0}^{M} \prod_{-k \le i \le k, i \ne 0} P(z_{m+i} | z_m; \theta), \qquad \text{(Equation 7)}$$

where θ denotes the parameters to be optimized in the model (e.g., the weight matrix in a neural network), and z_m is the current word. Its negative log likelihood form is generally used as a loss function:

$$\mathcal{L}(\theta)' = -\frac{1}{M} \sum_{m=0}^{M} \sum_{-k \le i \le k, i \ne 0} \log P(z_{m+i} | z_m; \theta), \qquad \text{(Equation 8)}$$

where

$$P(z_{m+i} | z_m) = \frac{\exp\left(v_{z_{m+i}}^T v_{z_m}\right)}{\sum_{z \in D} \exp\left(v_z^T v_{z_m}\right)}, \qquad \text{(Equation 9)}$$

Algorithm 1. Tailored chain-reaction-based algorithm

Input: Word vector for the delivery station v_0, word vector for zone $v_m \in V \backslash \{v_0\}$, delivery station ID z_0^+, and zone ID $z_m^+ \in D^+ \backslash \{z_0^+\}$.

Output: Zone ID sequence.

```
Initialization: $R \leftarrow \{z_0^+\}$, $S \leftarrow \{v_0\}$, $r \leftarrow v_0$
while $D^+ - R \neq \emptyset$ do
    Find $\max_{v_m \in V \backslash S} similarity(r, v_m)$
    Find zone ID $z_m^+$ that corresponds to $v_m$, then output $z_m^+$
    $R \leftarrow R \cup \{z_m^+\}$
    $S \leftarrow S \cup \{v_m\}$
    $r \leftarrow v_m$
end while
```

and v_z is the word vector corresponding to word z, and D is the set of all words.

The learning process of optimal mapping $\mathcal{F}(\cdot)$ can be explained by a neural network with a hidden layer. Figure 2A illustrates the neural network, and Table S5 introduces parameters needed for the neural network structure.

The steps of data processing, training, and learning outcomes of the model are as follows.

Step 1. (Encoding) Transform all zone IDs and station IDs in the training set into one-hot encoding vectors.

Step 2. (Constructing samples) Taking skip-gram mode as an example, we need to construct training samples in the form of a "feature-label" pair on the basis of the real-world route sequences. The input feature is the one-hot encoding vector of z_m, and the label is formed by concatenating the one-hot encoded vectors of its context.

$$\text{feature} = \text{one\_hot}(z_m). \qquad \text{(Equation 10)}$$

$$\begin{aligned}\text{label} = \; & \text{concat}(\text{one\_hot}(z_{m-k}), \ldots, \text{one\_hot}(z_{m-1}), \\ & \text{one\_hot}(z_{m+1}), \ldots, \text{one\_hot}(z_{m+k})).\end{aligned} \qquad \text{(Equation 11)}$$

Step 3. (Training) On the basis of the defined loss function, the network weights $W \in \mathbb{R}^{H \times O}$ and $W' \in \mathbb{R}^{O \times H}$ are updated using an optimizer based on the constructed training samples.

Step 4. (Learning outcomes) After training, the learned word vector for the ith zone is obtained by multiplying its one-hot encoded vector with the weight matrix $W \in \mathbb{R}^{H \times O}$. The matrix structure is shown in Figure 2B.

The complete real-world delivery route can be inferred from the word vector using a tailored chain-reaction-based algorithm

The theoretical foundation of our tailored chain-reaction-based algorithm stem from the notion that functional similarities between entities can be deduced on the basis of their mapping and proximity in the word embedding space.[30]

By projecting delivery routes to the word embedding space, we aim to infer complete delivery routes on the basis of their proximity and relational mapping in this space. For instance, let us consider an actual route "*DES4*, *D-3.2A*,..., *D-2.3C*" from historical data, where "*DSE4*" denotes the delivery station, i.e., the starting point of the route. First, we extract the word vectors of station "*DSE4*." Given that "*DSE4*, *D-3.2A*" co-occurs in historical data, it is inferred that the word vectors of the two elements are similar. Therefore, "*D-3.2A*" can be identified by finding the zone corresponding to the next most similar vector. From the perspective of the discrete choice model, this operation essentially equates to selecting the zone with the maximized utility.[36] In NLP, vector similarity is usually measured using cosine similarity.[37] Given the word vectors v_1 and v_2 for the two zones, respectively, the cosine similarity is defined as follows:

$$\text{similarity}(v_1, v_2) = \frac{v_1 \cdot v_2}{\|v_1\| \|v_2\|}. \qquad \text{(Equation 12)}$$

For an unsorted zone ID sequence $s^+ = \{z_0^+, z_1^+, \ldots, z_M^+\}$, where z_0^+ is the delivery station ID, $z_1^+, \ldots, z_M^+$ are the zone IDs to be sorted, we apply Algorithm 1 to obtain a sorted zone ID sequence from the word vector. Inspired by the basic steps of chain reactions, namely initiation, propagation, and termination,[38] the algorithm operates in a similar phased manner. The initiation phase starts with the delivery station, treating it as the initial starting point. The propagation phase then invokes an iterative process that identifies the subsequent delivery zones to be served. The identification relies on finding the word vector most similar to the previous zone. This exploitation of word vector similarity, a by-product of the Word2Vec model trained on historical delivery routes, positions zones frequently and consecutively appearing in historical data in close proximity in the vector space. Essentially, each step in the propagation phase retraces the paths embedded in the word vector space. The algorithm continues its propagation phase until all delivery zones are covered, which marks the termination of the algorithm. The final result is a complete delivery sequence that reflects the historical patterns embedded in the word vectors.

The details of the tailored chain-reaction-based algorithm are elaborated in Algorithm 1.

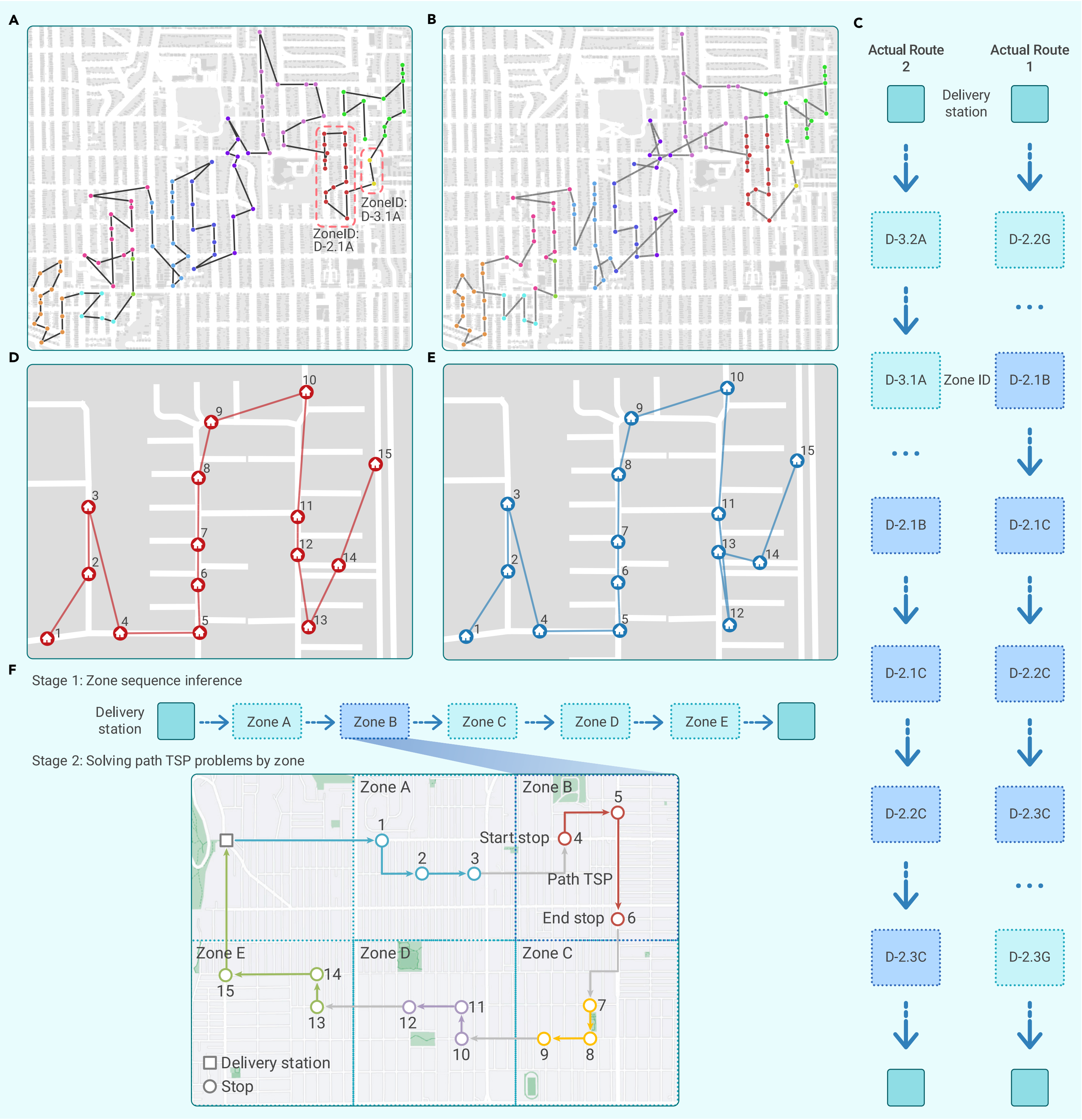

Figure 3. Comparison between real-world delivery route and theoretically optimized route (A) A real-world delivery route of Amazon's delivery service, where each point represents a delivery stop in the route. Stops within the same delivery service zone share the same color. (B) A delivery route optimized by the standard TSP model. (C) Two delivery routes in the same city sharing a common zone subsequence "*D-2.1B*, *D-2.1C*, *D-2.3C*." (D) A real-world intra-zone delivery route. (E) An optimal intra-zone delivery route. (F) Overview of the real-world delivery route optimization system.

Fusing language model and operations research method

In Amazon's package delivery business, a delivery service area is normally divided into zones to facilitate delivery. Figure 3A shows a real-world Amazon delivery route, where the driver completes deliveries in zone "*D-3.1A*" before proceeding to "*D-2.1A*," indicating a tendency to finish one zone before moving to another. Drivers make these decisions on the basis of their knowledge of roadway conditions, infrastructure design, and customer preferences. With extensive experience in delivery in practice, drivers may "reoptimize" the route sequences themselves instead of following the optimal routes provided by the routing system, as the latter may lack consideration of real-world road conditions and drivers' preferences. Figure 3B displays an optimized delivery route derived from solving the standard TSP that aims at minimizing the travel time. Nevertheless, the optimized route may conflict with drivers' experience in delivery. In Figure S2, combined with satellite imagery, we present a real-world case showing that delivery routes optimized by the standard TSP model might not be convenient for drivers given actual road conditions. Such kind of knowledge, unfortunately, can

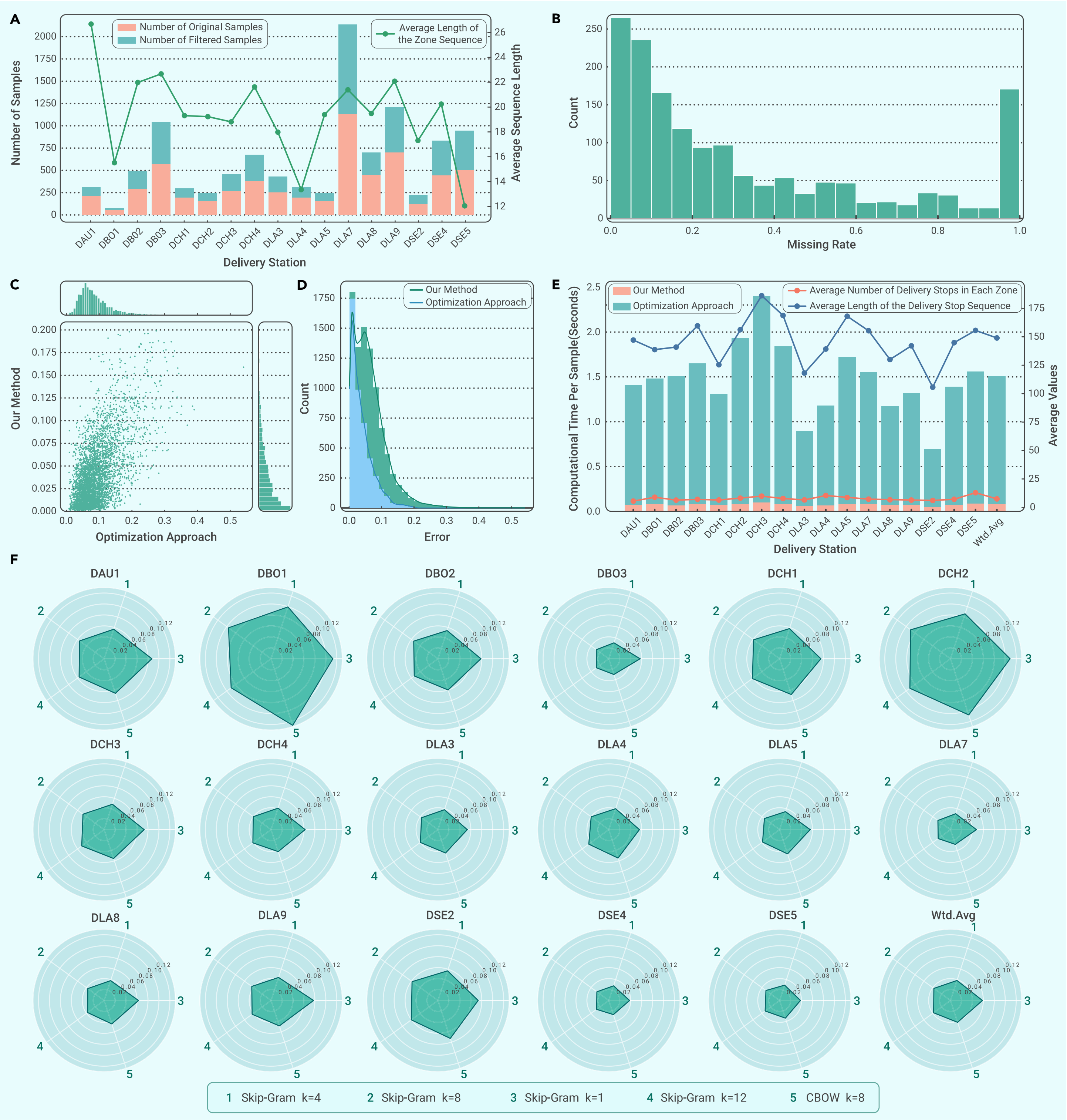

Figure 4. Performance comparison of different models (A) Statistics of samples in the data. (B) Missing rate of zone ID sequences. (C) Scatterplot showcasing sample errors across different methods. (D) Error distribution of our method and the optimization approach. (E) Efficiency metrics of our method and the optimization approach. (F) Error analysis for 15 delivery stations with different model parameters. Here, ***k*** represents the range of contextual information, often termed as window size in NLP.

hardly be quantified or incorporated by the prevailing route planning system in the industry, resulting in deviated route sequences in practice from the optimal routes.

Through the analysis of the actual delivery route data, it is found that a large number of identical subsequences existed in the routing data at the delivery zone level (see Figure 3C), indicating that drivers have an inherent behavioral pattern. However, within each zone, drivers tend to follow the shortest path, as illustrated in Figure 3D. This behavior is theoretically explained by the principle of local optimality.[39,40] The drivers would strive for the shortest path within each zone, essentially optimizing the route locally within each zone.

Motivated by the analysis above, we design a two-step hybrid framework that fuses the language model and operations research method to solve the real-world delivery route optimization problem. In the first step, the language model is used to extract the drivers' tacit knowledge about optimal delivery patterns, which may not be explicitly incorporated in the traditional delivery routing

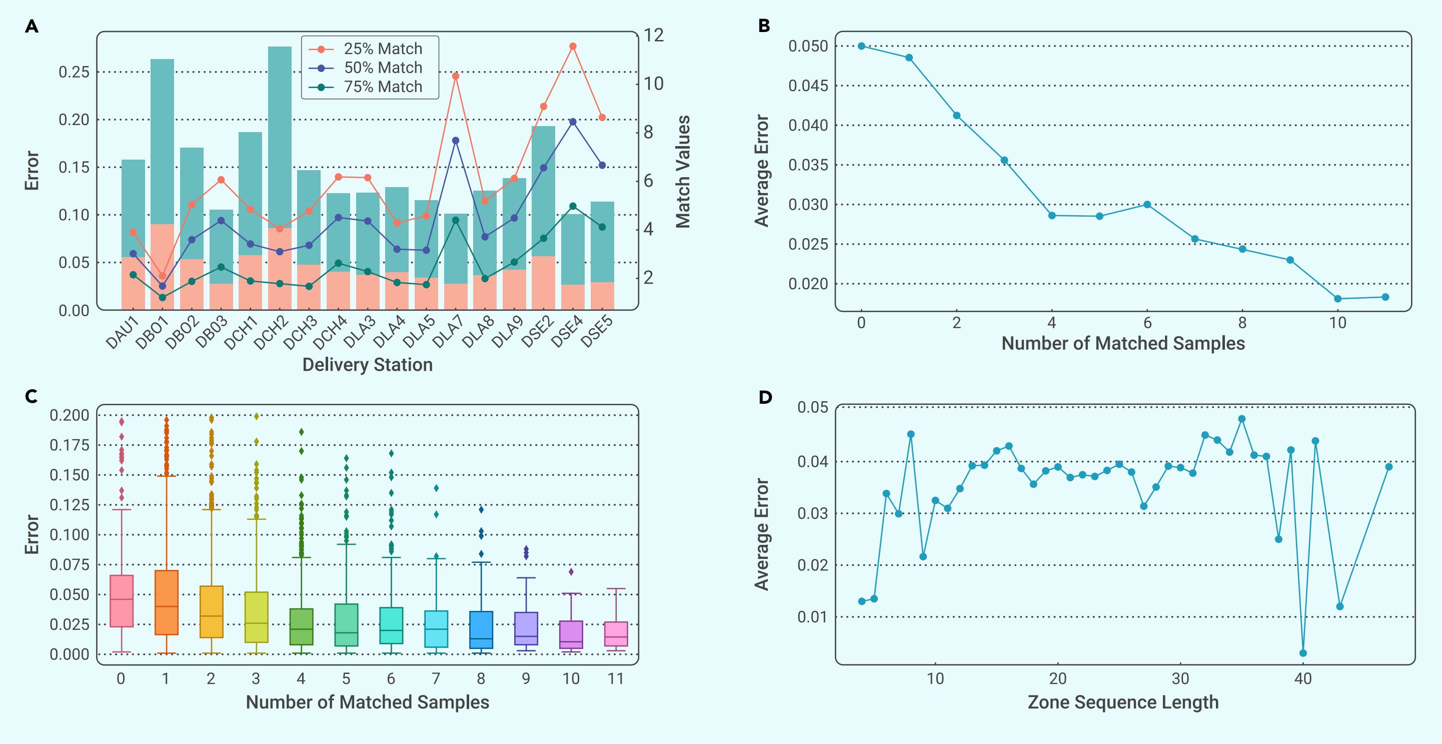

Figure 5. Model performance analysis (A) Prediction error across different delivery stations. (B) Relationship between the number of matched samples and average error. (C) Distribution of individual error values. (D) Relationship between the length of the zone sequence and the average error.

optimization algorithms. It provides a zone sequence that aligns with drivers' intuitive preferences. Upon procuring the sequence of zones, the second step treats each zone's delivery as the classic TSP problem with specific start and end stops (see the method section in the supplemental information). This approach offers a theoretically efficient route that also aligns with the on-ground experiences and preferences of drivers. The general idea of the complete delivery route optimization system is illustrated in Figure 3F.

Comparison of the proposed and traditional optimization model

The real-world data used in our experiments are extracted from Amazon's delivery service, encompassing critical information such as route information, travel time data, and actual delivery sequence information (see Tables S1–S3).[41] The data contain 6,112 routes from 17 distinct delivery stations scattered across various regions in the United States. We filtered outlier samples and divided the dataset on the basis of the delivery stations, in order to create subsets that align with unique geographical and operational characteristics of each station. To overcome possible limitations imposed by sample size and ensure robust model evaluation, we adopt a leave-one-out cross-validation strategy in our experiments. Detailed statistical information on the dataset is shown in Figures 4A and 4B.

We use a hybrid evaluation metric to determine the error between the sequences of the model output and the drivers' high-quality delivery sequences (considered to be the true values). The details of this metric and the sample creation are elaborated in the evaluation metric and sample construction process sections of the supplemental information. Figure 4C illustrates the scatterplots of individual sample errors for both our method and the conventional optimization approach. Figure 4D further shows the error distributions for both methods. Table S6 provides a comparative analysis of the error statistics for the two methods. The proposed approach of generalized sequence modeling shows a clear advantage over the standard TSP model from operations research in terms of both mean error and variance, indicating both the robustness and generalizability of our method. In contrast, the operations research approach, tailored specifically for a single task and dataset, lacks the adaptability needed for broader applications such as protein structure prediction.

Our method decomposes a large-scale route optimization problem into several small-scale route optimization problems. Figure 4E shows a comparison between the computational time of the proposed model and the traditional optimization approach. Although the total number of stops in the route is the same in both solutions, the computational time of our method is much shorter than that of the optimization approach.

The proposed model has several adjustable parameters, and the impact of these parameters on the error is illustrated in Figure 4F and Table S7. Window size has a significant impact on the results and the weighted average error when $k = 8$ is much lower than that when $k = 1$. From the perspective of feature engineering, $k = 8$ implies that the current word, along with eight preceding and eight succeeding words, constitutes a training sample, which contains more information than $k = 1$ and leads to better results. However, it should be noted that the length of the zone ID sequence in our study is approximately 20, and there is a bottleneck in improving the model performance by adjusting the window size. This bottleneck manifests itself in the observation that the weighted average error when $k = 12$ is larger than that when $k = 8$. Despite the benefit of richer information with a larger window, it also introduces more missing values in the training data.

The observed disparity in error between the skip-gram and CBOW models in Figure 4F can be attributed to the distinctions in their underlying architectures. The CBOW model predicts a target word from its context, while skip-gram does the opposite, predicting context words from a target word. This fundamental difference means that skip-gram often ends up with a more detailed representation for less frequent words, as it treats each context-target pair as a new observation. Because of the aforementioned mechanism, skip-gram virtually has more training data, as it learns from multiple context-target pairs for a single center word. In contrast, CBOW aggregates all context words for predicting a single center word, which can sometimes lead to a loss of detailed information. Skip-gram tends to perform better on infrequent words because its emphasis is on predicting contexts for specific target words. On the other hand, CBOW might neglect such words because of its averaging mechanism over contexts.

Performance analysis of the proposed method

In the experiments, an individual model was trained for each delivery station. The errors of corresponding subsets are further analyzed and showcased in Figure 5A. Here, we find that the error is related not only to the quantity of training data but also to the quality of the samples, which we refer to as the matching rate of the samples. See the sample construction process section in the supplemental information for detailed formulas used to calculate this matching rate. The relationship between the number of matched samples (75% match) and the average error is shown in Figures 5B and 5C. It can be found that the average error of the machine learning model decreases as the number of matched samples increases. The potential and practical values of machine learning methods are also revealed relation to the efforts required in algorithm design. Compared with *ad hoc* approaches, machine learning methods can automatically learn the drivers' behaviors without extensive expert experience in problem analysis and solution algorithm design for optimization problems.

A common challenge in the sequence prediction task resides in the potential rapid escalation of error as the length of the sequence targeted for prediction extends because of weak correlations between inputs and outputs in longer sequences. Figure 5D shows the relationship between the length of the zone sequence and the average error in our model. It is found that the average error fluctuates around 0.04 as the length of the zone sequence increases without an observable tendency to increase.

CONCLUSION

Our research introduces a pioneering attempt to apply language models, traditionally developed for NLP, to a broader range of tasks. In particular, any tasks involving understanding systems organized by certain grammar rules could potentially benefit from this research. It could be pertinent to fields in which complex systems are typically involved, such as biology, chemistry, and physics. Specifically, we examined the real-world delivery route optimization problem, in which drivers may need to "reoptimize" planned route sequences themselves on the basis of actual scenarios such as road conditions, infrastructure design, and driving preferences rather than strictly following the optimal routes derived from the routing system. In this direction, we have developed a hybrid architecture integrating a language model with an embedded TSP model, seeking to tackle real-world delivery route optimization problems that are challenging to address with merely operations research methods. Numerical experiments were performed on the real-world data from Amazon's delivery service, and a comparison was made between the proposed method and a pure optimization method. The results shed light on the feasibility, effectiveness, efficiency, and potential adaptability of the proposed method.

The application of the proposed approach can extend far beyond the scope of delivery route optimization. Potential applications span a variety of fields, such as intelligent transportation, smart cities, and autonomous driving.[42–47] As our research progresses, we are optimistic about the broader applicability of this model in tackling complex sequence prediction problems across diverse disciplines. This study serves as a stepping stone toward harnessing the power of language models in practical real-world scenarios, and we look forward to the remarkable advancements that lie ahead.

MATERIALS AND METHODS

See the supplemental information for details.

REFERENCES

1. Hovsepyan, S., Olasagasti, I., and Giraud, A.-L. (2020). Combining predictive coding and neural oscillations enables online syllable recognition in natural speech. Nat. Commun. **11**, 3117.
2. Lu, Y., Tian, H., Cheng, J., et al. (2022). Decoding lip language using triboelectric sensors with deep learning. Nat. Commun. **13**, 1401.
3. Popel, M., Tomkova, M., Tomek, J., et al. (2020). Transforming machine translation: a deep learning system reaches news translation quality comparable to human professionals. Nat. Commun. **11**, 4381.
4. Makin, J.G., Moses, D.A., and Chang, E.F. (2020). Machine translation of cortical activity to text with an encoder–decoder framework. Nat. Neurosci. **23**, 575–582.
5. Schramowski, P., Turan, C., Andersen, N., et al. (2022). Large pre-trained language models contain human-like biases of what is right and wrong to do. Nat. Mach. Intell. **4**, 258–268.
6. Ive, J., Viani, N., Kam, J., et al. (2020). Generation and evaluation of artificial mental health records for natural language processing. NPJ Digit. Med. **3**, 69.
7. Webb, T., Holyoak, K.J., and Lu, H. (2023). Emergent analogical reasoning in large language models. Nat. Human Behav. 1–16.
8. Ferruz, N., and Höcker, B. (2022). Controllable protein design with language models. Nat. Mach. Intell. **4**, 521–532.
9. Huang, T., and Li, Y. (2023). Current progress, challenges, and future perspectives of language models for protein representation and protein design. Innovation **4**, 100446.
10. Liu, Y., Wu, F., Lyu, C., et al. (2022). Behavior2vector: Embedding users' personalized travel behavior to Vector. IEEE Trans. Intell. Transport. Syst. **23**, 8346–8355.
11. Liu, Y., Lyu, C., Liu, Z., et al. (2021). Exploring a large-scale multi-modal transportation recommendation system. Transport. Res. C Emerg. Technol. **126**, 103070.
12. Zhang, T., Wang, J., and Meng, M.Q.-H. (2022). Generative adversarial network based heuristics for sampling-based path planning. IEEE/CAA J. Autom. Sinica **9**, 64–74.
13. Yue, L., and Fan, H. (2022). Dynamic scheduling and path planning of automated guided vehicles in automatic container terminal. IEEE/CAA J. Autom. Sinica **9**, 2005–2019.
14. Zhao, J., Ma, X., Yang, B., et al. (2022). Global path planning of unmanned vehicle based on fusion of A* algorithm and Voronoi field. J. intell. connect. veh. **5**, 250–259.
15. Zhang, W., Zhao, H., and Xu, M. (2021). Optimal operating strategy of short turning lines for the battery electric bus system. Commun. Trans. Res. **1**, 100023.
16. Yin, J., Li, L., Mourelatos, Z.P., et al. (2023). Reliable Global Path Planning of Off-Road Autonomous Ground Vehicles Under Uncertain Terrain Conditions. IEEE Trans. Intell. Veh. 1–14.
17. Liao, F., Arentze, T., and Timmermans, H. (2013). Incorporating space–time constraints and activity-travel time profiles in a multi-state supernetwork approach to individual activity-travel scheduling. Transp. Res. Part B Methodol. **55**, 41–58.
18. Easa, S.M. (1985). Shortest route algorithm with movement prohibitions. Transp. Res. Part B Methodol. **19**, 197–208.
19. Atz, K., Grisoni, F., and Schneider, G. (2021). Geometric deep learning on molecular representations. Nat. Mach. Intell. **3**, 1023–1032.
20. Tsai, S.-T., Kuo, E.-J., and Tiwary, P. (2020). Learning molecular dynamics with simple language model built upon long short-term memory neural network. Nat. Commun. **11**, 5115.
21. Pan, J. (2023). Large language model for molecular chemistry. Nat. Comput. Sci. **3**, 5.
22. Born, J., and Manica, M. (2023). Regression Transformer enables concurrent sequence regression and generation for molecular language modelling. Nat. Mach. Intell. **5**, 432–444.
23. Moret, M., Pachon Angona, I., Cotos, L., et al. (2023). Leveraging molecular structure and bioactivity with chemical language models for *de novo* drug design. Nat. Commun. **14**, 114.
24. Ross, J., Belgodere, B., Chenthamarakshan, V., et al. (2022). Large-scale chemical language representations capture molecular structure and properties. Nat. Mach. Intell. **4**, 1256–1264.
25. Kuenneth, C., and Ramprasad, R. (2023). polyBERT: a chemical language model to enable fully machine-driven ultrafast polymer informatics. Nat. Commun. **14**, 4099.
26. Ferruz, N., Schmidt, S., and Höcker, B. (2022). ProtGPT2 is a deep unsupervised language model for protein design. Nat. Commun. **13**, 4348.
27. Jiang, T.T., Fang, L., and Wang, K. (2023). Deciphering "the language of nature": A transformer-based language model for deleterious mutations in proteins. Innovation **4**, 100487.
28. Hie, B.L., Shanker, V.R., Xu, D., et al. (2023). Efficient evolution of human antibodies from general protein language models. Nat. Biotechnol.
29. Brandes, N., Goldman, G., Wang, C.H., et al. (2023). Genome-wide prediction of disease variant effects with a deep protein language model. Nat. Genet. 1–11.
30. Manica, M., Mathis, R., Cadow, J., et al. (2019). Context-specific interaction networks from vector representation of words. Nat. Mach. Intell. **1**, 181–190.
31. Bengio, Y., Lodi, A., and Prouvost, A. (2021). Machine learning for combinatorial optimization: a methodological tour d'horizon. Eur. J. Oper. Res. **290**, 405–421.
32. Bengio, Y., Ducharme, R., Vincent, P., et al. (2003). A Neural Probabilistic Language Model. J. Mach. Learn. Res. **3**, 1137–1155.
33. Markov, A.A., Markov, A.A., Levin, V., et al. (1960). The Theory of Algorithms (Springer).
34. Mikolov, T., Chen, K., Corrado, G., et al. (2013). Efficient estimation of word representations in vector space. Preprint at arXiv. 1301.3781.
35. Mikolov, T., Sutskever, I., Chen, K., et al. (2013). Distributed representations of words and phrases and their compositionality. NeurIPS **26**.
36. Ben-Akiva, M.E., and Lerman, S.R. (1985). Discrete Choice Analysis: Theory and Application to Travel Demand (MIT press).
37. HGomaa, W., and A Fahmy, A. (2013). A survey of text similarity approaches. Int. J. Comput. Appl. **68**, 13–18.
38. Laidler, K.J., and Keith, J. (1965). Chemical Kinetics (McGraw-Hill).
39. Bellman, R. (1966). Dynamic programming. Science **153**, 34–37.
40. Sutton, R.S., and Barto, A.G. (2018). Reinforcement Learning: An Introduction (MIT press).
41. Merchán, D., Arora, J., Pachon, J., et al. (2022). 2021 Amazon last mile routing research challenge: Data set. Transport. Sci.
42. Ortúzar, J.d.D. (2021). Future transportation: Sustainability, complexity and individualization of choices. Commun. Trans. Res. **1**, e100010.
43. Lin, H., Liu, Y., Li, S., et al. (2023). How generative adversarial networks promote the development of intelligent transportation systems: A survey. IEEE/CAA J. Autom. Sinica **10**, 1781–1796.
44. Sun, Y., Hu, Y., Zhang, H., et al. (2023). A parallel emission regulatory framework for intelligent transportation systems and smart cities. IEEE Trans. Intell. Veh. **8**, 1017–1020.
45. Lin, Y., Hu, W., Chen, X., et al. (2023). City 5.0: Towards Spatial Symbiotic Intelligence via DAOs and Parallel Systems. IEEE Trans. Intell. Veh. **8**, 3767–3770.
46. Xu, Q., Li, K., Wang, J., et al. (2022). The status, challenges, and trends: an interpretation of technology roadmap of intelligent and connected vehicles in China (2020). J. intell. connect. veh. **5**, 1–7.
47. Olovsson, T., Svensson, T., and Wu, J. (2022). Future connected vehicles: Communications demands, privacy and cyber-security. Commun. Trans. Res. **2**, 100056.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (grants 72322002, 52221005, and 52220105001).

AUTHOR CONTRIBUTIONS

Y.L., Fanyou Wu, Z.L., Feiyue Wang, and X.Q. conceived the research idea. Y.L. and Fanyou Wu designed the experiment. Y.L. and K.W. conducted the main analysis. Y.L. and K.W. wrote the manuscript. Z.L., Feiyue Wang, and X.Q. reviewed and edited the manuscript.

DECLARATION OF INTERESTS

The authors declare no competing interests.

SUPPLEMENTAL INFORMATION

It can be found online at https://doi.org/10.1016/j.xinn.2023.100520.

LEAD CONTACT WEBSITE

http://www.svm.tsinghua.edu.cn/essay/80/1920.html