Michael Ukhterov 212238091 & Shaked Fried 308338888

Interactive AI | Winter 2023

Pick Me Up

Meeting point Coordination-Interface for Ride-Sharing

PROJECT SUMMARY

PROBLEM AND MOTIVATION

The carpooling problem is usually concerned with matching a *ride-provider* to a *traveler*, nevertheless, when sharing a journey, *travelers* can be picked up and dropped off at their origin and destination or at a **meeting point within a certain distance from their original point which can lead to a better solution[1].** Furthermore, traditional travel planning platforms can also enhance their service pair of *ride-provider* and *traveler*. That issue is also been partially discussed in the framework of distribution platforms and other last-mile problems[2].

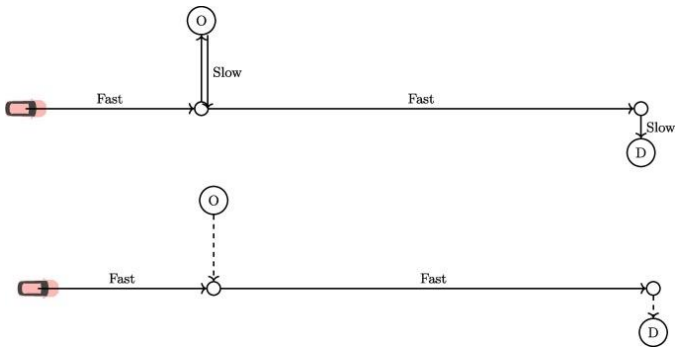

Figure and description are borrowed from: Fielbaum, A., Bai, X., & Alonso-Mora, J. (2021).

Figure 1: How can walking enhance ride-sharing systems? there are two main sources of improvement: prioritizing high-speed and highly connected streets, and avoiding detours. The former is illustrated - in which walking (bottom row) allows the vehicle to stay in the main avenue, keeping away from three slow segments. (see Fielbaum, A., Bai, X., & Alonso-Mora, J. (2021).

MAIN CHALLENGES

Currently, a *ride provider* and a *traveler* wishing to coordinate a meeting point have no proper solution, they are facing three main challenges:

1. Lack of clear information on meeting-point options.
2. Limited ability to integrate one's preferences into existing routes\pick-up options, as well as limited ability to realize the impact of pick-up points on the fellow traveler and other expenditures (e.g., cost, gas consumption, emissions, total time spent, etc.)
3. Given the obscurity mentioned above – communicating and comprehending each other's needs to coordinate optimal meeting point - is a nontrivial mission.

INTERACTIVE-AI POTENTIAL IMPACT

Ideally, coordinating meeting points holds both algorithmic challenges and Human-communication challenges. Hopefully, empowering and easing dialogue that considers realistic constraints with both sides' preferences could be greatly beneficial and promote pro-social and pro-environmental behaviors. For instance, being exposed to more information provides an opportunity for both sides to perform acts of goodwill to optimize the overall outcome that includes cost and reducing environmental impact. Our goal is to manifest a dialogue-centric coordination platform for a pair of *ride-provider*|*traveler* users.

1 Fielbaum, A., Bai, X., & Alonso-Mora, J. (2021). On-demand ridesharing with optimized pick-up and drop-off walking locations. Transportation research part C: emerging technologies, 126, 103061.

2 Emrah Demir, Aris Syntetos, Tom van Woensel, Last mile logistics: Research trends and needs, IMA Journal of Management Mathematics, Volume 33, Issue 4, October 2022, Pages 549–561,

SYSTEM DESCRIPTION

Interface Demonstration

INTERFACE

Respectively to the identified challenges, the interface core components are:

- ✓ Integration of user's ride preferences into possible pick-up options.
- ✓ Impact description of the chosen meeting point (implications on the fellow traveler and other expenditures (e.g., cost, gas consumption, emissions, total time spent).
- ✓ Dialogue – communicating mutual needs with the fellow traveler easily to coordinate optimal meeting point.

The main supported tasks are (1) Coordinating a meeting point together with a fellow traveler (2) Offering a meeting point to a fellow traveler and (3) Looking into assorted options for meeting points that are enriched with relevant information (as described above).

See figure 2 for interface flow:

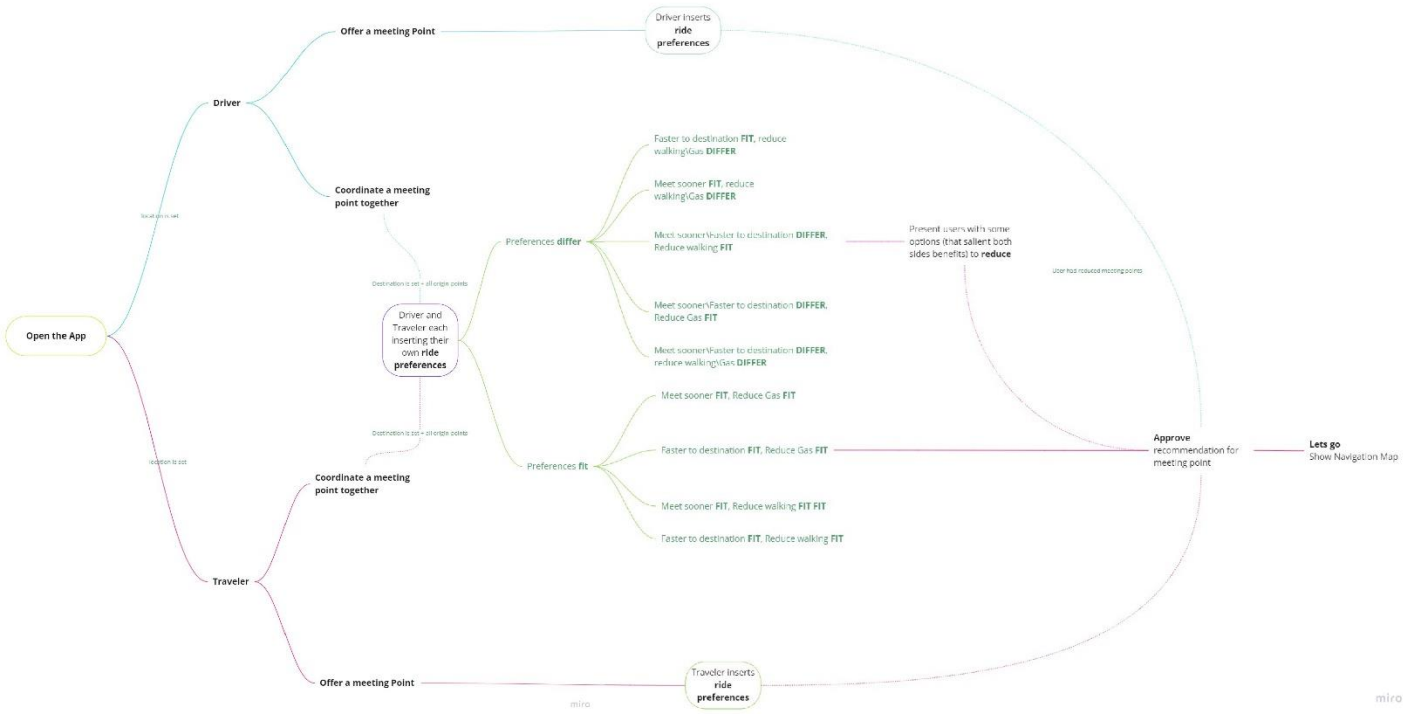

Figure 2

DESIGN PRINCIPLES

- AI in the loop experience (rather than human in the loop)
    - Transparency – we let users control main parameters of optimization and therefore better feel and understanding how the AI works. We also present other viable meeting points.
    - Agency - engaging users and giving them control. (Adding and removing points, sliding to see more travel parameters)
- Dialogue-Centric
    - Our AI major merit is empowering dialogue and mutual reciprocity rather than deciding what is the best solution for our users.
- Nudge for social impact
    - Platforms and technologies that are widely spread have strong potential for social and environmental impact. Along with commercial and technological interests, social impact should not be overlooked. The prospect of designing a system with such potential led us to think of how to take a step further into promoting sustainability (emissions) and health (walking).
    - Our system integrates a simple and non-aggressive encouragement for users to prefer walking (conventionally positive) over emissions (conventionally negative). See figure 3

Figure 3: Colors were chosen to promote walking over gas spending. Walking\gas preferences were chosen to be presented one against the other to emphasize their effect.

ALGORITHM

The backhand utilizes 2 python libraries:

- OSMNX
- RADAR

Overview of libraries and functionality:

Both allow users to work with geospatial data, OSMNX is adequate for meeting points discovery and radar is more applicative. Geospatial data is in converted by OSMNX to a graph $G = (V, E)$ where the node V are potential meeting points (usually places where roads/streets intersect) and E are streets/ roads. Each edge $e \in E$ has certain information like travel time and distance which is used to calculate the route time in the optimization. This algorithm works well and displays the path on the graph for the user. We then use to RADAR to turn coordinates into readable address and deliver it to the user.

Backhand main flow:

Our system's backhand consists of 3 main parts:

1. Converting address to coordinates/node in graph. Input – human readable address, Graph G from osmnx. The function finds closest point in the graph by coordinates. Output – node in G
2. Converting meeting point coordinates to address. Input – driver start node, passenger start node, destination node, graph for driver G, graph for passenger G2, criterion 1, criterion 2. Output – node of meeting point.
3. Finding meeting point based on user input. Input – graph G, node of meeting point. Output – coordinates of meeting point.

First, both users enter their starting points, destination, and criteria for the route. W convert those addresses to coordinates. The user start node is taken for finding the meeting point that optimizes route time based on user criteria and considering user preferences. The algorithm iterates over all nodes in the graph and finds the best options according to optimization: best k routes which will be displayed for both users.

$$result = \arg\min(\max\{X_1, X_2\} + X_3)$$

$$\text{s.t } X_4 \leq T_3 \text{ or } X_5 \leq T_4$$

Where 1 ,2 - time to meeting point. 3 – time from meeting point to destination, 4 – walking distance, 5 – gas milage

Then it returns the node id of the meeting point which we convert back to an address for user convenience.

Prototype GitHub link: https://github.com/shaked-fried/Interactive_AI_PickMeUP

Code demonstration (we are not planning on making changes to the code)

USER EVALUATION

5 users have tried to use our interface

#1 male, Israeli 23, data scientist
#2 male, Israeli, 21, data scientist
#3 female, Chinese, 26, cognitive psychologist
#4 female, Israeli, 29 architect
5 male, Israeli, 27, mathematician

Tasks:

- {#1,#2} were asked to act as if they were drivers to examine the code (we have acted as passengers)
  - Insert specified origin and destinations points
  - Choose your preferences
  - Accept or decline the system offer
  - See recommended route
- {#5} were asked to act as if they were drivers to examine the code. (We have acted as drivers)
  - Insert specified origin and destinations points
  - Choose your preferences
  - Accept or decline the system offer
  - See recommended route
- {#3,#4, #5} were asked act as if they were passengers \ driver (We have acted as the fellow traveler) to examine the Interface
  - Get "in" the app
  - Choose your preferences
  - Accept or decline the interface offer[3]

Overall, the users were curious, entertained, and pleased with the experience. It was stated that both the system\interface were quite fluid to use.

Interface:

Worked well:

1. Pretty
2. Fluid and clear
3. not complicated
4. answering their needs that they didn't even knew they had

Can be improved:

1. not all presented information was at once clear (see figure 4)

Figure 4 graph wasn't noticeably clear

[3] *some functionality wasn't implemented so when user #3 asked to decline the offer he was drawn out of the flow.

2. Information about traffic is lacking

Code:

Worked well:

3. Code worked well and mostly fluidly.
4. The map with the route was rational solution
5. Easy \ effortless
6. Clear logic and AI is not hidden.

Can be improved:

1. Part of the interface\widget wasn't very intuitive.
2. Freedom was limited.
3. Part 2 of the code took 4 sec, might be better if will be shorter [Make the system faster].

SUMMARY

Notable things we have learned:

- We've experienced an in-and-out up-and-down scale of thinking. We were exposed to a simultaneously diverse way of creating and figuring out a project \ technology.
- We have felt how asking the question of who is going to use our technology changes algorithmic approach.
- We have started to be more minded to how technologies we use are designed in the context of XAI, for example, we will be more than happy to that EGEED TICKTAK designers will go to Interactive AI course in the Technion.
- We have found ourselves discussing whether we are 'allowed' to create a system that directly aims to 'make the world better', even if 'better' is negotiable.

Further work and improvement:

- Exploration vs exploitation – maybe showing a user extra one meeting point that doesn't maximize one's preferences will lead them to find attractive meeting points out of the box (e.g, its near a parrot shelter can adopt the parrot I've always wanted to have..)
- Learning preferences of the user and slowly reduce questions\clicks.
- Learning relationship between users and slowly recommend meeting point with criteria that match their history.