A Mobile Decision Support System for Tourist Planning

Haakon Borch
Master Thesis
University of Bergen

Goals

Produce a Decision Support Systems (DSS)

On Mobile

For tourists

To solve the problem:

> “What can I do **today** while at **this location**?

Decision Support Systems

Intended to aid humans in making decisions as opposed to AI

Personal vs enterprise

Intelligent - uses AI techniques to generate suggestions

Mobile DSS (MDSS)

- Availability of location-data
- Lack of research

Research Question

Research Question 1: Can an intelligent MDSS using location-dependent information for aiding tourists be built?

Research Method

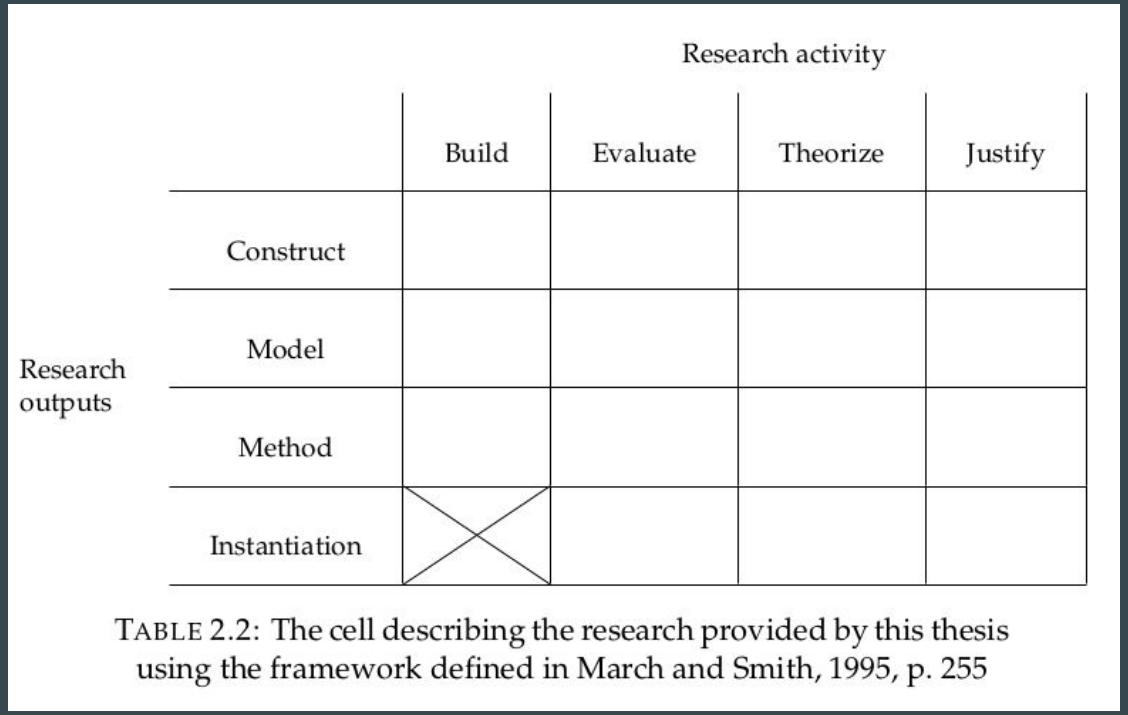
March and Smith (1995)

- Build an instantiation

Prototyping (Davis, 1992)

- Throwaway
- Evolutionary

March and Smith



Research outputs	Research activity: Build	Evaluate	Theorize	Justify
Construct				
Model				
Method				
Instantiation	X			

TABLE 2.2: The cell describing the research provided by this thesis using the framework defined in March and Smith, 1995, p. 255

Specifications

All calculations done on device

On Android using Java

Using survey data from Innovation Norway

Using location data from Google Maps

A day is defined as consisting of five activities

Bayes theorem and Naive Bayes

Bayes theorem - updating beliefs based on evidence

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

Naive Bayes - assuming conditional independence between variables

Conditional independence - knowledge of probability distribution of A does not provide knowledge of same for B given some third event C

Vocabulary and height are not independent, but given age they are conditionally independent

Genetic Algorithms

Population of Individuals

Each generation:

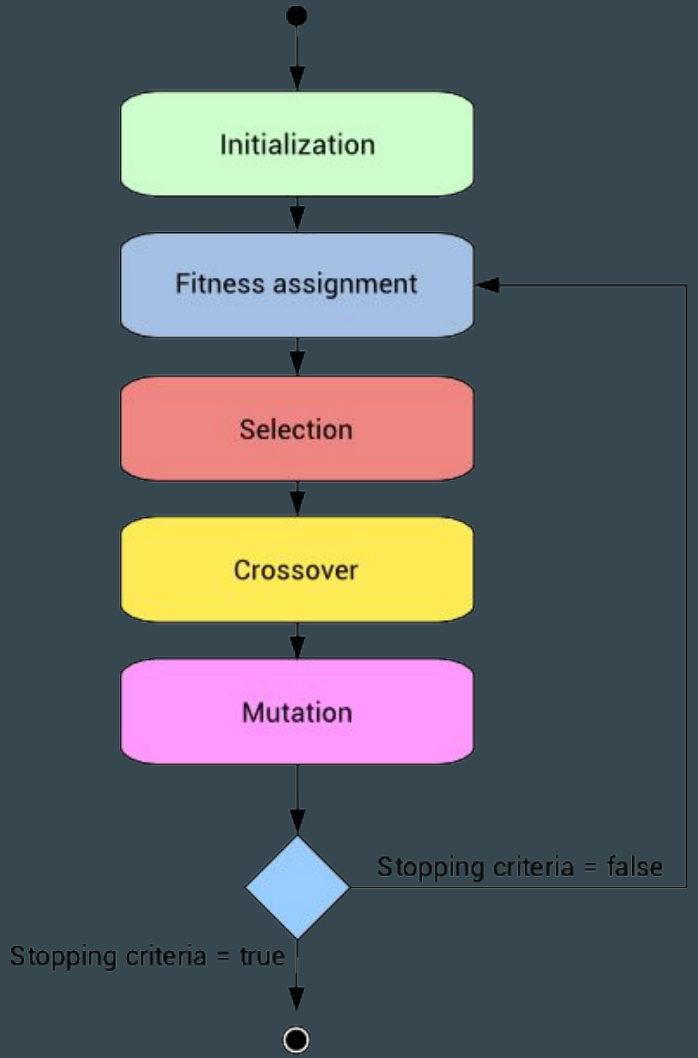
Ranked according to Fitness function

Used to generate new Population by:

Crossover

Mutation

Until Fitness criteria or time



How does the artifact work?

1. User answers Questionnaire
2. Answers used to generate probabilities of interest
3. First round of GA - find Google Maps categories of interest
4. Information gathered from Google Maps
5. Second round of GA - find specific locations and a path between these
6. Suggestion presented to user
7. User accepts or rejects

Constraints

For each GA there were constraints that modified the fitness-function

- Reward for food as 3rd
- Reward for food as any
- Penalty for “non-interesting”
- Penalty for repeat categories
- Penalty for laybacks / inefficient paths

Inefficient Path

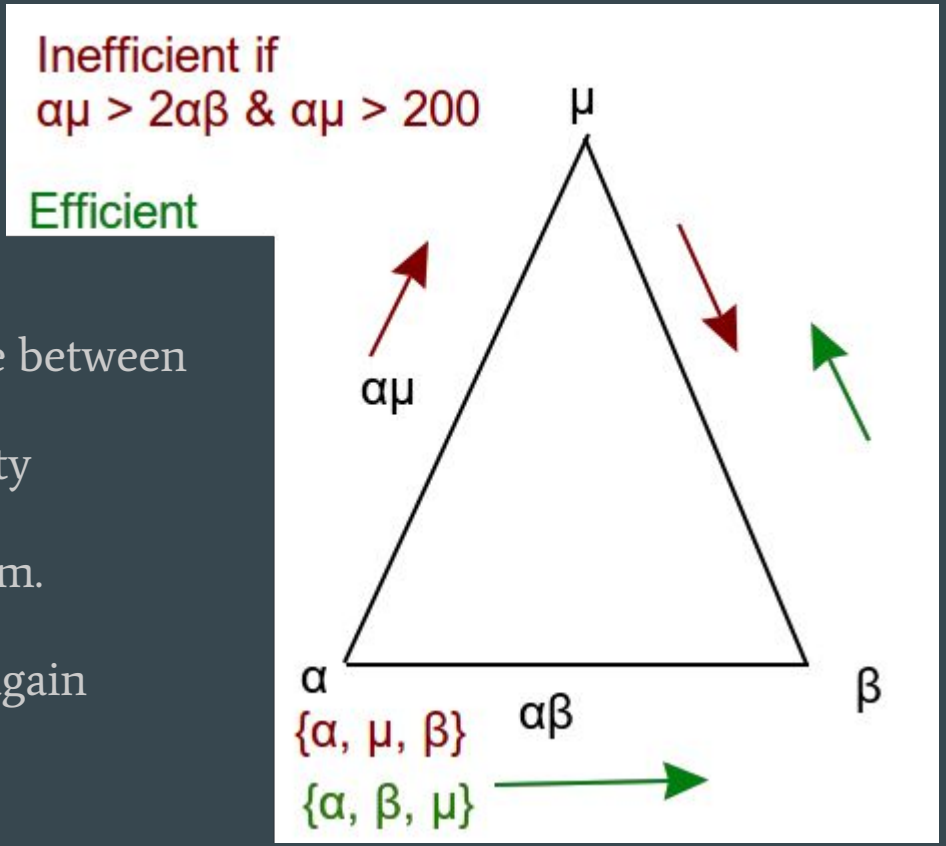
Given a path between three points {α, μ, β},

an inefficient path is one where the distance between

the points is such that the distance-inequality

αμ > 2αβ holds and the distance of αμ > 200m.

- Intended to avoid going out and back again



Finding the distance between two points on a sphere

Definition: Inefficient Paths

Haversine Formula

$$hav\theta = \sin^2\left(\frac{\theta}{2}\right)$$

Given point p 1 at latitude φ 1 and longitude λ 1 and point p 2 at latitude φ 2 and longitude λ 2 the haversine formula for the distance(d) between p 1 and p 2 is

$$d = 2r\sin^{-1}(\sqrt{hav(\phi_2 - \phi_1) + \cos(\phi_1)\cos(\phi_2)hav(\lambda_2 - \lambda_1)})$$

Termination Condition

Time-based

Described by:

$$2\omega + \theta$$

ω is the maximal running time of a GA round, 5 seconds

θ is the timeout value of the connection-request to Google Maps API

Tools

Statistical Package for Social Sciences statistics (SPSS statistics)

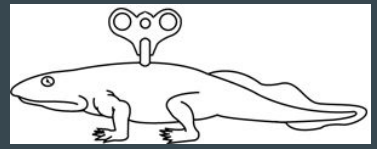
R and Rstudio

Java and Android

Watchmaker

Dev tools:

Git, CircleCI, Slack, RxJava and StarUML



Preparing the data from Innovation Norway

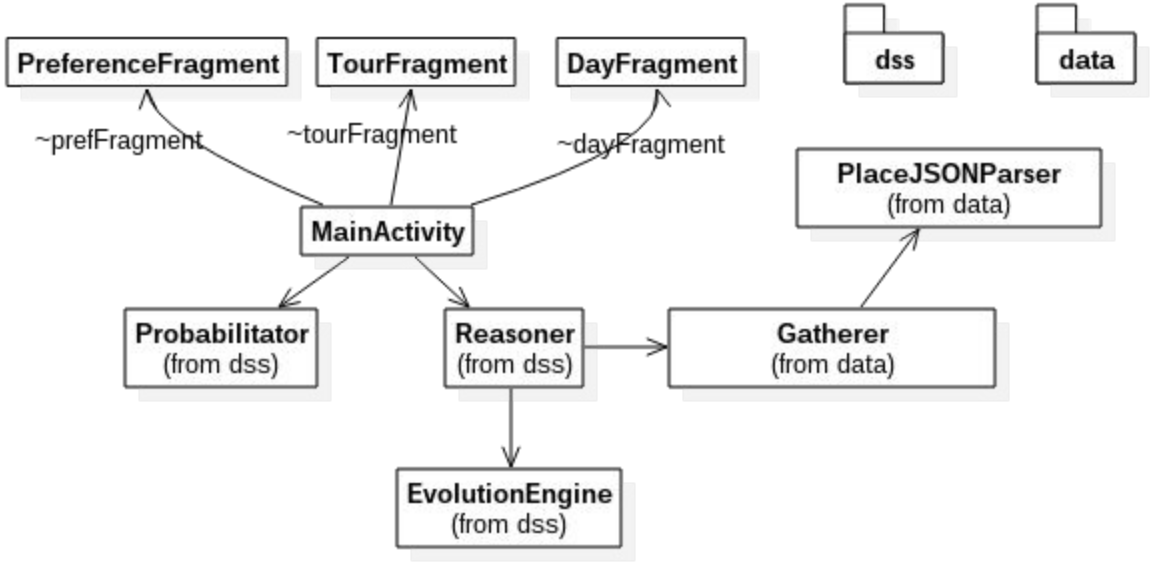
164 variables in survey

Reduced to 17, 11 activities and six sociological variables

- Purpose of the trip
- Permanent residence
- Age
- Gender
- Education Level
- Employment status

Mapped the 90 available Google Map categories to 11 bins of activities

Overview


PreferenceFragment
TourFragment
DayFragment
dss
data
~prefFragment
~tourFragment
~dayFragment
PlaceJSONParser
(from data)
MainActivity
Probabilitator
(from dss)
Reasoner
(from dss)
Gatherer
(from data)
EvolutionEngine
(from dss)

Probabilitator

For each of the 11 activity types:

- Generating priors of interest from data
- Generating conditional probability of interest from Sociological values of user
- Multiplying probabilities with assumption of conditional independence
- Naive Bayes
- Normalizing probabilities for each activity

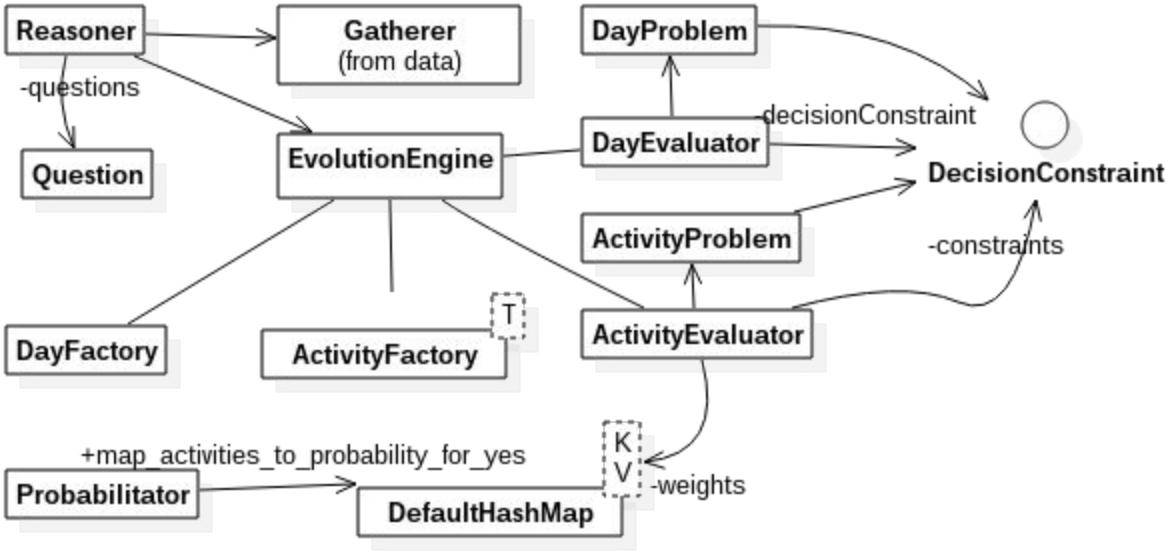
Watchmaker

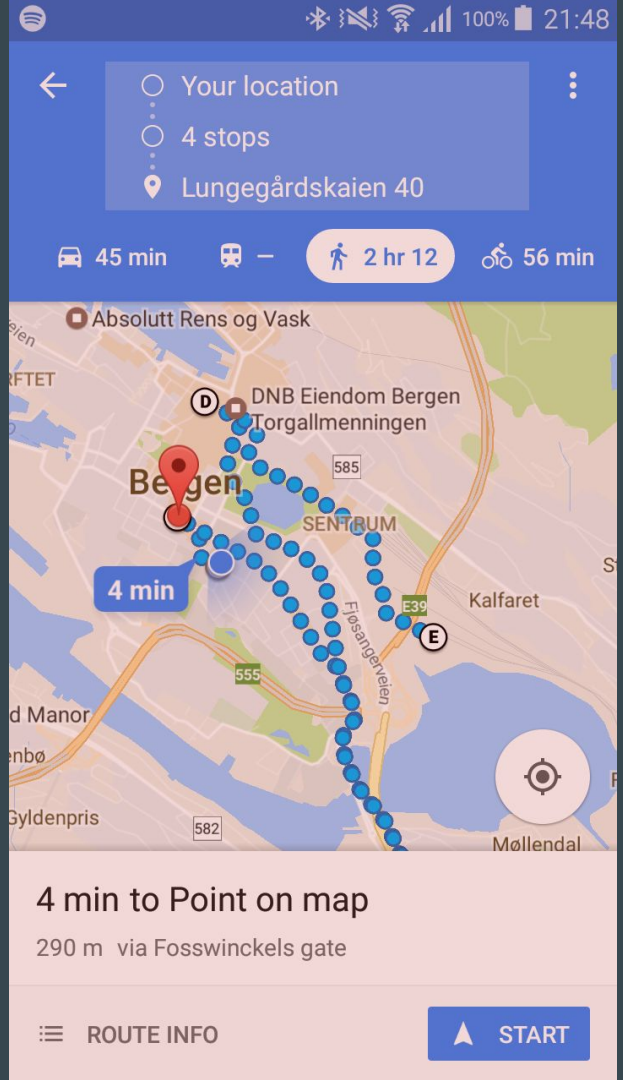
Implemented in EvolutionEngine class

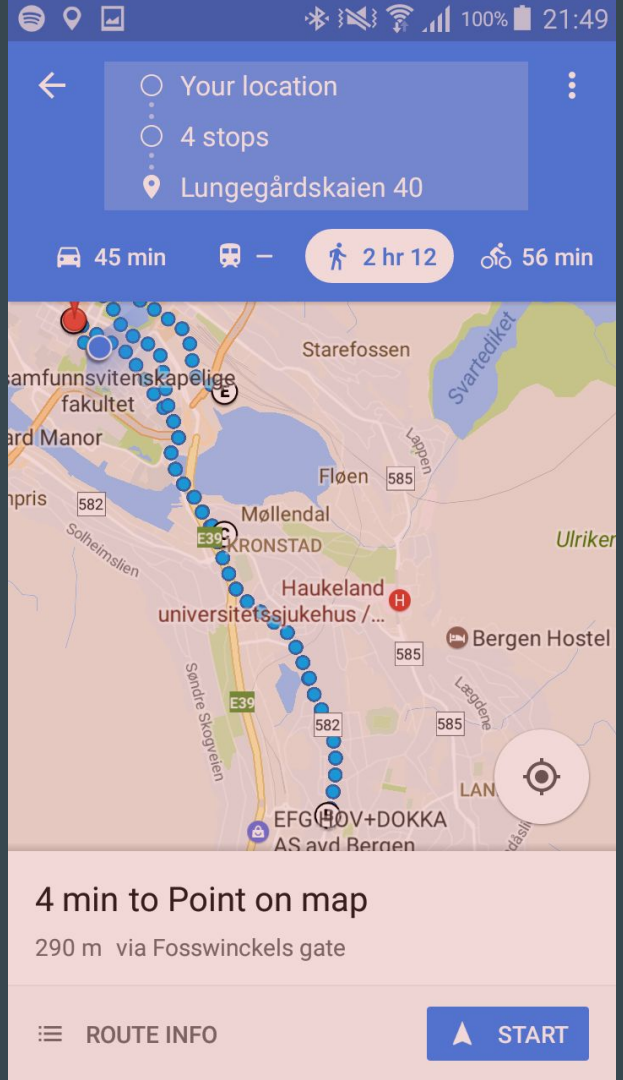
For each round of GA, Activity and Day:

- Factories - Produced initial population
- Problems - Contained constraints
- Evaluators - Provided fitness-function

Evolution Engine




100% 21:48
Your location
4 stops
Lungegårdskaien 40
45 min
2 hr 12
56 min
Absolutt Rens og Vask
DNB Eiendom Bergen Torgallmenningen
Bergen
SENTRUM
4 min
585
555
582
E39
Kalfaret
Møllendal
4 min to Point on map
290 m via Fosswinckels gate
ROUTE INFO
START


100% 21:49
Your location
4 stops
Lungegårdskaien 40
45 min
2 hr 12
56 min
Starefossen
fakultet
Fløen
585
582
Møllendal
KRONSTAD
E39
Haukeland universitetssjukehus /...
Bergen Hostel
585
582
585
4 min to Point on map
290 m via Fosswinckels gate
ROUTE INFO
START

Evaluation

Purpose

- Answering the research question

Method

- User-test
- Ideal user-test

User-Test

Three users

- Expert domain-knowledge in IT, availability and mobile application development

Task-completion

Provided feedback

- Two: positive
- One: needs work - got recommended to visit a funeral home

Feedback used for improvements - additional penalty and reward constraints

Defining an Ideal User

- Aged between 30-60
- From Norway
- Other vacation activities
- Employed
- Not a student or Researcher

Each of these values has 50% coverage in the data

Ideal User-Test

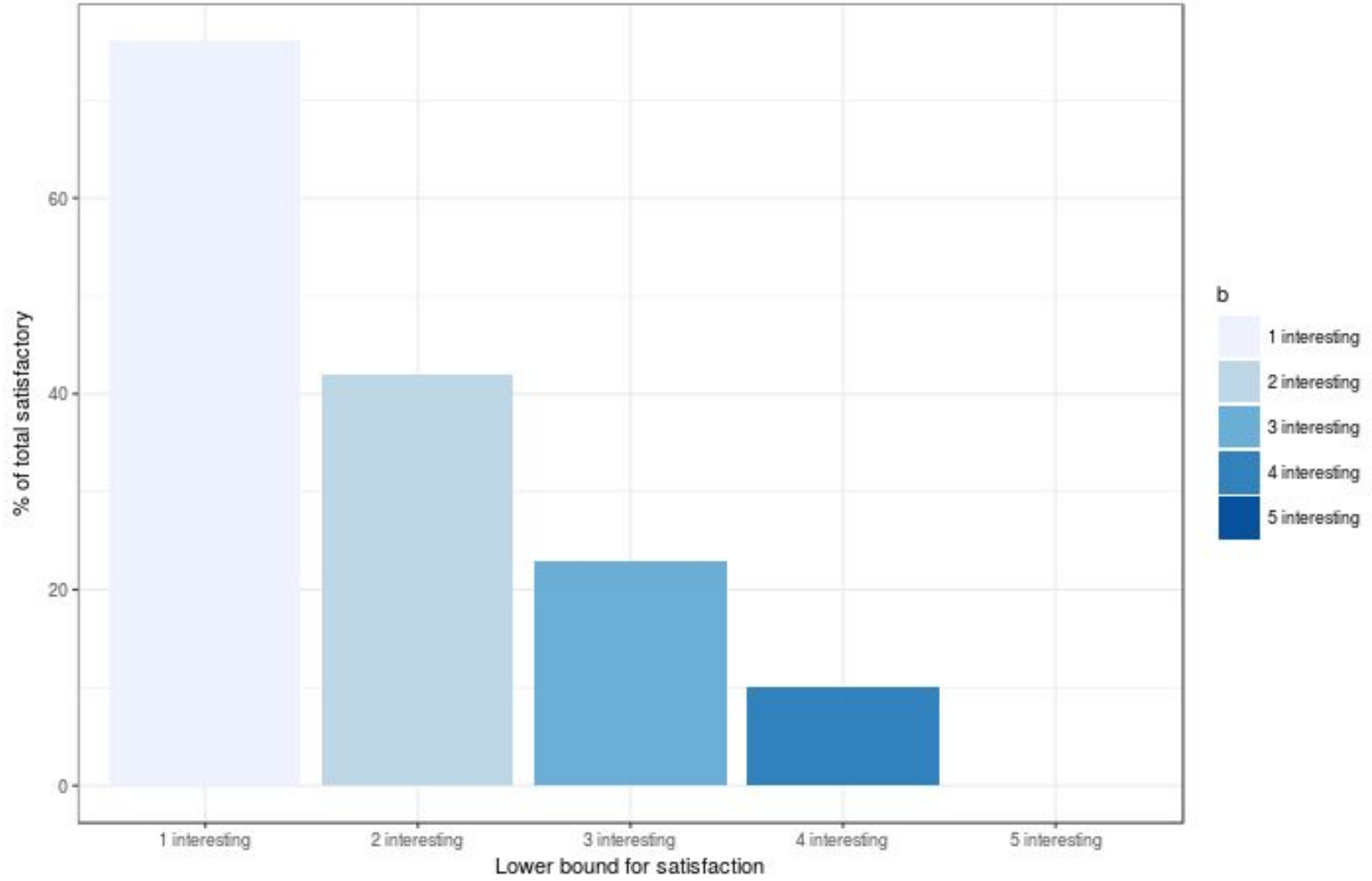
26 suggestions generated using definition of Ideal User

Each suggestion rated on amount of “interesting” activities

- Interesting defined as estimated to be relevant to tourist visiting Norway

Results:

- 76% of suggestions contain at least one “interesting” activity


% of total satisfactory
60
40
20
0
1 interesting
2 interesting
3 interesting
4 interesting
5 interesting
Lower bound for satisfaction
b
1 interesting
2 interesting
3 interesting
4 interesting
5 interesting

A “non-interesting” suggestion

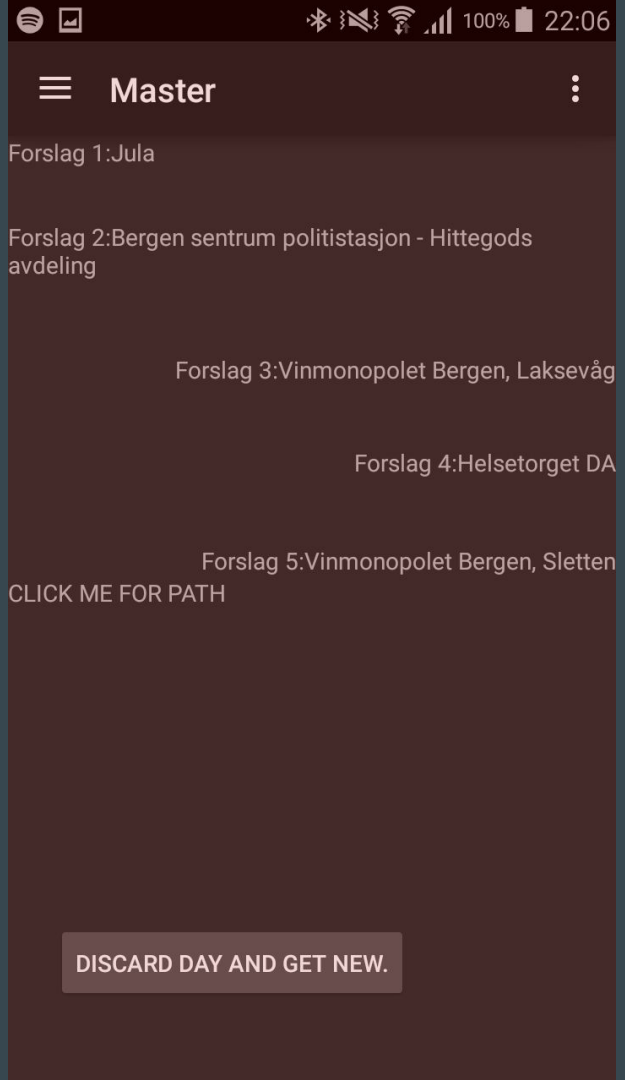
Location 1: Jula (Hardware Store)

Location 2: Bergen Police station- lost and found

Location 3: Vinmonopolet (liquor store)

Location 4: Helsetorget (doctors office)

Location 5: Vinmonopolet (a different liquor store)



Research Question

Research Question 1: Can an intelligent MDSS using location-dependent information for aiding tourists be built?

Answering the Research Question

Properties of artifact needed for affirmative answer:

- Is an intelligent MDSS
- Uses location-dependent information
- Intends to aid Tourists
- It works

Assuming lowest floor for “it works”, the RQ can be answered affirmatively

Validity

Research Question and Background

- Group-data for predicting individuals
- Bias from mapping 11 activity types to Google Maps categories
- Arbitrary constraints
- Tuning the settings of the GA

Alternatives

- Bayes & GA vs only GA
- On-device vs Server

Performance

- Threading
- Target platform of library
- Garbage-collection and real time termination

Evaluation

- Overlap between real testers
- Evaluation without testing
- Users only tested one suggestion
- Arbitrariness of ideal user

Future Work

Taking into account memory-limitations

Developing with the computation being done on server

Using a tuning-aware methodology for GA for improved quality

Exploring tourist preference prediction and modelling

A / B testing with random suggestions for evaluation

Thank you