T U M

I N S T I T U T F Ü R I N F O R M A T I K

Proactivity in a Mobile, Context-Aware Recommender System

Johannes Huebner, Wolfgang Woerndl

TUM-I11
April 11

T E C H N I S C H E U N I V E R S I T Ä T M Ü N C H E N

TUM-INFO-04-I11-0/1.-FI

Druck: Institut für Informatik der
Technischen Universität München

Abstract. Traditional recommender systems usually follow a request-response pattern, i.e. these systems only return item suggestions when a user makes an explicit request. Proactivity means that the system pushes recommendations to the user when the current situation seems appropriate. This is conceivable in mobile scenarios such as restaurant or gas station recommendations. However, proactivity has not gained much attention in recommender system research or has been put into practice. In this paper, we present a new model for proactivity in mobile, context-aware recommender systems. The model relies on domain-dependent context modeling in several categories. We have implemented a prototype gas station recommender and conducted a survey for evaluation. Results showed good correlation of the output of our system with the assessment of users regarding the question when to generate recommendations.

Table of Contents

1. Introduction

Mobile computing, context-awareness, and recommender systems are well-explored research areas that have led to a great amount of successful real-world applications. Due to the capabilities of modern mobile devices, such as smartphones, as well as the price decrease in mobile Internet plans, there is great potential for mobile computing applications that make use of a client-server architecture or other services available over the Internet. Mark Weiser first proposed applications of what he called "ubiquitous computing" [We1991] about 20 years ago. What has remained a vision for a long time since then, is now not only relatively easily feasible, but in the process of becoming affordable for a broad public. While there are some privacy concerns regarding pervasive computing (cf. [Hi2005]) that have to be taken into consideration when designing such systems, great benefits could be achieved by assisting humans in everyday situations.

The focus of this paper lies on the aspect of proactivity in mobile recommender-systems, on which there seems to be relatively little research so far. For this purpose, the existing work on that topic will be analyzed in chapter 2. Subsequently, a model of how to build such a system will be proposed and implemented in a lightweight prototype that will be evaluated in an online questionnaire in a last step. In the final chapter, conclusions from this study will be drawn. In order to generate a practical model that can be applied to further scenarios, two real-world scenarios were analyzed in detail to derive a more general model.

1.1. Restaurant Scenario

In the first scenario, the user receives recommendations to visit a restaurant on her smartphone. The system includes context attributes like the user's geographic distance to each point of interest (POI), or the POI's opening hours. Furthermore, the system factors in the user's current mobility. If the user is currently walking, a smaller radius of maybe 300m would be the geographic range of choice than if the user were driving in a car. If the user, on the other hand, is riding the subway, bus, train, or other public transportation means, the system should try to determine the stations the user could get off and include restaurants in a range of 300m around each station in the process. Ideally, the recommender would also have access to the user's calendar to enable the system to determine how much spare time the user currently has. That way, recommendations could be made much more sensitively, namely when the user's interruptability (cf. [Ho2005]) is highest. In addition to these and further context attributes, a set of user preferences should be considered. Similar to [Ri2010] three POI-based criteria for restaurants will be proposed in this work: cuisine, price, and atmosphere. As such, the system would automatically create a user profile based on the user's (positive and negative) feedback on recommendations. For instance, if the user receives the proposition to visit a fish restaurant, one possible feedback for the user to give would be "I do not particularly like fish", which would "penalize" all fish restaurants in future recommendations.

In the restaurant context, user preferences will probably play a very important role, since the decision on whether or not to visit a restaurant greatly depends on the match between the user's preferences in terms of cuisine, atmosphere and price, and the corresponding attributes of each restaurant.

1.2. Gas Station Scenario

In contrast to the explained restaurant scenario, dynamic context attributes are likely to have more impact in the gas station scenario. In this case, the user will receive recommendations to select gas stations via the built-in navigation system while driving. There are still a few non-contextual attributes, as for instance the user's preferences regarding the brand of gas stations (see [Ba2011]), or his or her average fuel tank level when visiting a gas station, but the focus clearly lies on context attributes. [Ba2011] determined the gas price and the detour required to reach a gas station as important influence factors. Particularly important is to make recommendations only when the driver is currently not in very stressful situations – even more so than in the first-described scenario. Instead, a good point of time might be when waiting for red lights to turn green, or in long, straightforward track sections that do not require the user to get actively involved very much. For that reason, numerous attributes are factored in to determine the driver's interruptability. Apart from that, other attributes like the current tank level certainly play an important role. Moreover, it will be discussed whether the number of co-drivers or the question if the user is currently on the way towards a target or heading back home make a difference in acceptance towards recommendations.

2. Related Work

While a large amount of research on recommender systems exists, mobile computing applications, location-, or context-awareness (e.g. [De1999]) as well as any combination of the above areas (e.g. [Wo2007]), there is relatively sparse information about proactivity in such applications. Most recommender systems require the user to input his or her preferences manually (cf. [Ho2009], [Ye2010]).

That is particularly surprising when taking into consideration that through the constraints of mobile devices (i.e. reduced computing capacities, restrictions regarding user interface and interaction capabilities) [Ri2010], it could add very much to the user experience when the user would not be required to become active anymore at all.

[Ri2010] made a first step towards proactivity when proposing that instead of actually typing text on the mobile phone, the users will be offered alternatives to choose from via simply tapping on buttons. Other approaches for recommendation provisioning require the user to enter an extensive user profile on first usage (cf. [Ho2009], [Ru2009]) or only work proactively after the user formulated search queries beforehand (cf. [Em2009]). Hong et al. then created a model according to which a user profile is deduced from a user's context history, which enables proactive recommendations in the future. However, in the model proposed by [Ho2009], training time is very important.

This paper therefore aims at developing a proactive recommender system that can be put into practice without initial user input as well as without prior user history data and focuses on the point of time when recommendations are communicated to the user. The model proposed in chapter 3 will also include user preferences that help to improve the model over time. However, the majority of this recommender's input factors are extracted from context parameters that change dynamically rather than relying very much on static user preferences.

2.1. Definition of Context

There has been much research on context-awareness and the different facets of context (cf. [Ad2008]). Due to that, various definitions of the term "context" exist (cf. [Ba2005]). For instance, [De1999] defined context as "any information that can be used to characterize the situation of entities (i.e. whether a person, place or subject) that are considered relevant to the interaction between a user and an application, including the user and the application themselves". [Wo2009] concluded that this "means, context is very dynamic and transient. By contrast, user profile information such as preferences or ratings are somewhat static and longerlasting". [Ma2006] takes an opposing position by also including user preferences into context – in spite of their rather static nature.

This work follows the distinction between context and static attributes (such as user preferences) according to the volatility of a criterion. Following this, context is defined as the collection of all properties that change dynamically (e.g. the distance between the user and a POI). In contrast to that, more static properties (e.g. the user's taste in restaurants or the location of a POI) are regarded as static user or POI attributes.

Furthermore, four context dimensions are defined, each containing any number of parameters.

2.2. Context dimensions

After having defined context, the question arises how to populate the rather abstract term "context" with concrete parameters. [Pi2005] states that "location is currently the most dominant context parameter". [De1999] identified location, identity, activity, and time as to be the most important context dimensions.

Based on these categories, the two previously described scenarios were analyzed in detail and a set of common context categories could be identified as main influence factors:

- *Time Context*
- *Geographical Context*
- *Social Context*
- *User Context*

These are very similar to the criteria proposed by [Sa2008]. However, since the personal behavior is difficult to predict in the present cases, the User Context category will be populated by attributes such as how busy the user currently is (based on the user's calendar or driving behavior).

Each of those categories will be dynamically populated with specific context attributes depending on the recommender type. For a restaurant recommender, for instance, the user's distance to a restaurant would be a geographic context attribute while in a gas station recommender, the detour from the current route would most likely be better-suited.

3. Model

In this work, a two-phase model is used for a proactive, context-aware recommender system. In the first phase (called *Filtering*), it will be determined whether or not the user should receive a recommendation. In case this question is answered in the affirmative, the second phase (*Recommendation*) will be initiated and a recommendation might be communicated to the user.

The recommender application will be executed periodically in the background, e.g. once every minute. In a next step, the recommender could derive information on exactly when the next execution would be useful.

3.1. Point of Time of Display

As described before, the recommender system will be executed in regular intervals and decide on whether or not to make a recommendation. The point of time when the message is displayed does not necessarily have to match the point of time of the application execution.

Instead of displaying messages immediately, a proactive recommender should in any case try to determine, how receptive the user currently is towards recommendations in order to avoid annoyance.

According to [Ho2005], users react to proactively delivered messages more positively when received during the transition of two physical activities instead of at random times. While this is not directly applicable to every recommender implementation, it is a valuable fact to bear in mind when designing the communication strategy. In the restaurant scenario, information on the user's interruptability could be gained by analyzing data generated by the cellphone's gravity sensor. For the gas station recommender, however, other indicators have to be taken into the equation. Suitable indicators are described in chapter 3.6. For instance, a good policy might be to propagate suggestions to the driver when waiting for a red light to turn green, or on long passages without any turns. In general, the model attempts to determine the ideal point of time for suggestions based on criteria in all context dimensions.

3.2. Score Calculation

Both phases utilize the four context dimensions as defined in chapter 2.2. Each attribute in these categories (context dimensions and user data) will be weighted according to the relative importance of the parameter to the user in the given scenario, and is based on how much information is available in the particular category. For instance, an in-car navigation system can access a lot of data from sensors (e.g. fuel tank level, seat occupancy, current speed) and the user might even provide additional information by specifying a destination. In contrast to the restaurant scenario, however, there will be only little data available regarding the user herself. For instance, the navigation system cannot know what the driver's plans for the rest of the day are. Hence, the overall weight of each context category differs between our two application scenarios.

3.3. Phase I: Filtering

In the Filtering phase, the system needs to determine whether or not the user should receive a recommendation. The Filtering phase can have various outcomes. The overall model of the proposed two-phase process is illustrated in Figure 1.

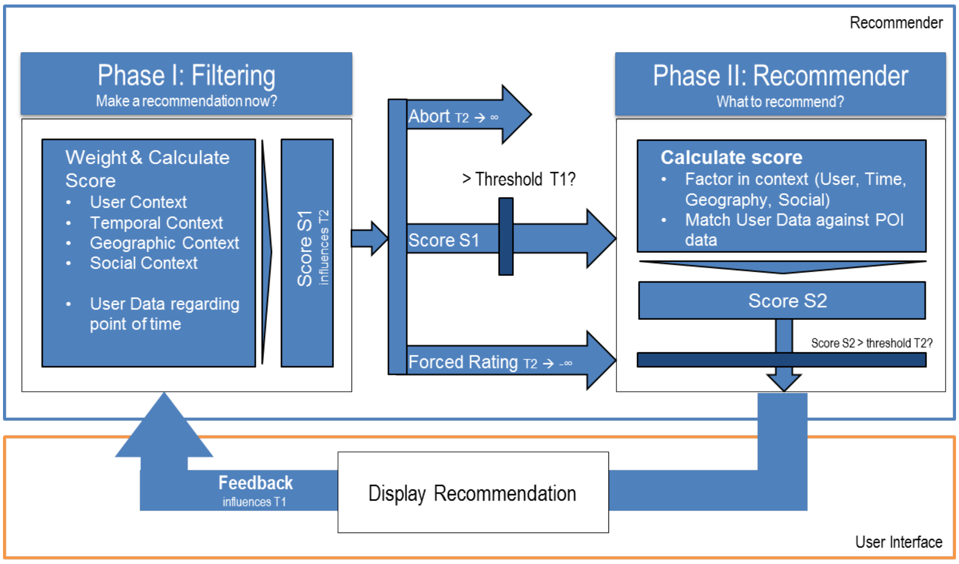


Figure 1: Model

In the normal case, several criteria will be considered in order to calculate an overall score S1 that will determine whether or not the second phase (recommendation) will be initiated and hence, whether a recommendation will be communicated to the user in the end, or not. If S1 exceeds a certain threshold T1, the recommendation phase will be started. Otherwise, the process will be aborted. Furthermore, S1 has an impact on the second threshold T2. For example if the timing for a recommendation is perfect, T2 will be decreased in order to make sure that at least the score S2 of one POI reaches the required threshold and will be recommended to the user. Following this, T2 is a function of S1:

$$\mathrm{T1}(\mathrm{S1}) = |1 - \mathrm{S1}|$$

To give an example, in the gas station scenario the likelihood to initiate the Recommender phase II is be higher at a tank level of 10% than with a tank that is 40% filled. The criteria to be taken into consideration at this point have to be defined for each recommender scenario separately. Generally, all four previously-mentioned context dimensions (time, geography, user, social) as well as time-relevant user profile data should be examined to determine relevant criteria.

In addition to that, there are two special cases. First of all, the entire lifecycle can be aborted when triggered correspondingly by certain criteria, e.g. when the navigation system in the car recognizes that the fuel level is at almost 100%, Score S1 will be set to $-\infty$ and hence, the recommendation application will be aborted since threshold T1 will always be in the range of 0 and 1. Following this, there is a possibility that the application will run every minute and abort itself in many consecutive cases. On the other hand, there are criteria that will force a recommendation by setting S1 to $+\infty$. That means, (1) the recommendation phase will be initiated and (2) the application will be forced to communicate a recommendation to the user. This will be achieved by setting Threshold T1 to 0, when S1 is $+\infty$. Thus, regardless of the best-ranked POI's score, a recommendation would be always made in such a case. For instance, if the fuel tank of the car is almost empty and there is only one known gas station in range, this particular gas station would be recommended to the user regardless of all other parameters, e.g. of how big the detour is.

Concluding, each criterion can trigger a lifecycle abort, force a recommendation, or yield to a score S1 between 0 and 1.

3.4. Phase II: Recommendation

Similar to the score calculation in phase I, at the Recommendation stage there will also be a score (S2) calculated based on criteria in all four context dimensions as well as by determining the match between POI attributes and user attributes, e.g. the match of a restaurant's cuisines and the user's taste.

Again, each POI can be immediately eliminated from the recommendation process if triggered so by certain criteria (e.g. due to very high spatial distance to the user).

If a forced recommendation was triggered in phase I, the best-ranked POI will be proposed to the user since the threshold T2 will be exceeded by every single POI. Otherwise, even if the recommendation phase is executed, it could happen that no recommendation is made, depending on whether or not at least one POI reaches a score greater than the previously-defined threshold T2.

3.5. User Feedback

As shown in Figure 1, there is a feedback loop used to improve the user satisfaction over time. Obviously, if the user chooses not to follow a recommendation made by the system, he or she will be asked about the reasons for the negative response. In a restaurant scenario, for instance, a user could choose whether it was the cuisine, the atmosphere, other POI attributes (cf. critique-based interfaces, [Ri2010]), or other criteria not represented in the recommender that led him or her to not visit the restaurant. The given feedback will be reflected in the user attributes and is taken into consideration accordingly in future recommendation processes.

Apart from that, the user is offered the choice to just ignore the recommendation in case he or she is very busy. It is possible for the user to influence the frequency of recommendations. This is important in light of the recommender's proactive nature. If the user provides feedback regarding the number of recommendations made, threshold T1 will be altered accordingly. Per default, T1 is set to 0.5. Each time, the user wants to receive less frequent recommendations, the threshold will be increased by 10% (e.g. to 0.55, 0.605, and so on).

Keeping Ricci's research ([Ri2010]) in mind, when declining a recommendation, the user will be able to enter feedback by simple tapping on a button offering to reduce the recommendation frequency in the future.

In the opposite case, i.e. when a user follows the recommendation and visits the recommended restaurant or approaches the proposed gas station, the application will also offer to increase the frequency, or rather probability, of recommendations by lowering the threshold T1.

Both, positive and negative feedback on the recommendation frequency will not reflect on the time span between two executions of the program, but will rather alter the threshold T1 accordingly.

In conclusion, the user will have the possibility to influence both frequency (i.e. alter T1) and quality (i.e. update user preferences) of recommendations. The threshold T2 will be set to a default value of 0.5 and be adjusted in dependence of S1 (cf. chapter 3.3).

3.6. Prototype Implementation

In a next step, a lightweight prototype for the gas station scenario was built to simulate phase I of the presented model. In this chapter, the considered criteria, their weights and the overall score calculation will be illustrated in detail in Table 1. Subsequently, evaluation results will be described in chapter 4.

[Ba2011] highlights important influence factors, such as the tank level or the detour required to reach a gas station, which are included in the implementation. In addition to that, several further context attributes are factored in to determine the stress level and other factors that are potentially important to the user.

a	Criterion (Category)	Input Parameters	Description and function f_a	Weight w_a
1	**Tank Level** (User Context)	• TankLevel [%]	The function will return $-\infty$ when the tank level is 50% or greater. Otherwise, the return value is calculated by the linear function $$f(TankLevel) = 1 - \frac{TankLevel}{50}$$ Following this, a recommendation becomes more likely as the tank empties.	10%
2	**Direction** (User Context)	• Direction (home|away|unknown)	An assumption was made that drivers tend to visit a gas station more likely when being on the way back home instead of heading towards the desired location, since they might have less time pressure then. $$f(Direction) = \begin{cases} 1, if\ Direction = home \\ 0.5\ if\ Direction = unknown \\ 0\ if\ Direction = away \end{cases}$$ A further assumption is made that this criterion is not always measurable, hence the little weight.	5%
3	**Stress-Level** (Time Context)	• CurrSpeeding [km/h] • AvgSpeeding [km/h]	The recommender tries to determine how stressed the driver currently is based on the speeding behavior. If the driver usually drives 5km/h faster than allowed (all time average), but he or she is currently driving 10km/h faster than allowed (current drive average), then he or she might be in a hurry. Hence, a recommendation is less likely. If the difference between CurrSpeeding and AvgSpeeding is 10km/h or more, the function will return 0. If, on the other hand, CurrSpeeding – AvgSpeeding is -10km/h or less, the value 1 will be returned. Otherwise, the following linear function is presumed:	15%

			$f(CurrSpeeding, AvgSpeeding) = 0.5 - \frac{CurrSpeeding - AvgSpeeding}{20}$	
4	**Traffic** (Time Context)	• CurrSpeed [km/h] • AvgSpeed [km/h, in last 5 minutes] • curvesAhead [#, in next 2 minutes] • crossingsAhead [#, in next 2 minutes]	A further influence factor assumed crucial is the current traffic situation. In general, a recommendation becomes less likely the more stressful the current traffic is for the driver. If the driver is currently standing (i.e. the current speed is less than 3 km/h), for instance because he or she is waiting for a light to turn green, the value 1 is returned. The same applies to situations in which the driver is in a traffic jam or stop-and-go traffic (i.e. AvgSpeed < 5km/h). Otherwise, the return value is based on how many curves and crossings lie ahead on the track, i.e. how often the driver needs to become active in the next few minutes: $f(curvesAhead, crossingsAhead) = 0.5 * \left(1 - \frac{curvesAhead}{5}\right) + 0.5 * (1 - \frac{crossingsAhead}{10})$	20%
5	**POIs-InRange** (Geographic Context)	• POIsInRange [#]	The function will return $+\infty$ when there is only one known gas station in range. If there are two gas stations in range, it might be a very good idea to communicate a recommendation, but it is not absolutely necessary. Due to that, the value 1 will be returned as partial score in that case. If, on the other hand, there are ten or more gas stations in range, 0 will be returned, since apparently there is no danger of running out of fuel in the current location.	5%

			Otherwise, i.e. with between 2 and 10 gas stations in range, the return value will be calculated using the following linear function: $$f(POIsInRange) = \frac{10 - POIsInRange}{8}$$	
6	**Minimum Detour** (Geo-graphic Context)	• MinDetour [m] • TrackLength [m]	As described by [Ba2011], the detour is one of the most important factors in the gas station selection process. When the driver entered his or her destination in the navigation system, the minimum detour/track ratio will be factored in, because people might be willing to make a 2km detour when driving 200km; when driving for only 5km, though, a detour of 2km seems unacceptable. If the minimum detour would be more than 10km, or if the detour/track ratio is more than 10%, the return value is 0, since a longer detour seems unrealistic even for 1000km drives. Otherwise, the return value is based on the following function: $$f(MinDetour, TrackLength) = 1 - 10 * \frac{MinDetour}{TrackLength}$$ In case the route is unknown, the minimum distance is considered in the model. If the minimum distance is less than 20m, the return value is 1; if it is more than 2km, the return value is 0. Between that, following linear function will be used: $$f(MinDetour) = \frac{2000 - MinDetour}{1980}$$	15%
7	**PersonsInsonsIn-Car** (Social	• PersonsInCar [#]	It is assumed that people avoid visiting gas stations when there is one co-driver in the car, since that person would be bored while fueling. Hence, the function will return 0 when there are two persons	5%

	Context)		including the driver in the car. In all other cases, 1 will be returned – also if there are more than two persons in the car, because then all co-drivers can still have a conversation while the driver leaves the car to fuel it. Since the perception of social awkwardness as well as its impact on the decision when to tank is presumed highly individual, this criterion is only considered in the overall score with a weight of 5%.	
8	**Average Tank Level when fueling** (User Preferences)	• AvgTankLevel [%] • CurrTankLevel [%]	The recommender presumes that people have different fueling behavior patterns, i.e. there are some people who visit a gas station very early and others, who wait until the last possible moment. To factor this in, the absolute difference between the tank's current level and the average level when fueling is calculated. The further apart those two values are, the less likely a recommendation will be made: $f(AvgTankLevel, CurrTankLevel) = \|1 - \|AvgTankLevel - CurrTankLevel\|\|$	25%

Table 1: Parameters used in the gas station recommender prototype

Each function f_a either returns a value in the range of 0 and 1 or ±∞ to abort the process or force a recommendation. The weight w_a of each criterion is also in the range between 0 and 1. The sum of all weights w_a is 1 ($\sum w_a = 1$). The overall score S1 will be calculated as follows:

$$S1 = \sum w_a * f_a$$

Using the above formula, Score S1 will also be in the range of 0 and 1 as long as f_a is not ±∞. In this case, the function f_a with the value ±∞ will outweigh all other partial scores, i.e. the triggers to abort the process or force a recommendation will be propagated to the score S1. In the unlikely event that there is one f_a with the value +∞ and another f_a with the value –∞, precedence will be given to the parameter a with the higher weight w_a.

4. Evaluation

As a next step, we evaluated the implementation of the model introduced in chapter 3 in the scenario of a gas station recommender. For this purpose, we set up an online questionnaire.

4.1. Methodology

The included questions can be distinguished into three parts[1]. First, all participants were asked to enter demographic data (age, sex, kilometers traveled per year). In the subsequent section, 13 scenarios were constructed (cf. chapter 7.2, Appendix) in each of which the test subjects were asked to rate on a 5-point Likert scale (1 = very bad, 5 = very good), how convenient they would find a recommendation made by the navigation system in each case. The parameters for each scenario were input into the prototype to find out in which cases the proposed model would lead to a recommendation (cf. chapter 3). Afterwards, the prototype's decisions were compared to the results of the user study. Furthermore, to make the influence of single parameters measurable, most scenarios came in couples or triples that only differed by one influence factor.

The target of the third part of the questionnaire was to determine the overall acceptance and usefulness of such a recommender system. To assess these properties, a semantic differential with nine adjective pairs was used [La1997]. Additionally, users had two free text input fields to describe how often they would like to receive recommendations and another one for comments, annotations, and feedback.

4.2. Findings

The proposed model was evaluated in a user study amongst 35 students of the Technische Universität München. The evaluation was targeted at the first phase of the suggested model, i.e. the point of time when to make recommendations. The second phase - the recommendation process itself - was neglected, since that was not the focus of this paper.

For simplicity's sake, the only user profile value contained in our model was given in the survey, i.e. the average tank level when the user tanks, was set to 1/8, which, in many cars, marks the beginning of the tank's reserve. In order to make these judgments easier comparable to the recommender's output, these values were rescaled to fit into the range of 0 and 1.

[1] For screenshots of the questionnaire, please refer to chapter 7.1 (Appendix)

Scenario No.	Prototype result	Evaluation MEAN (μ)	Evaluation STD (σ)	Results / Interpretation
1	-∞	0,39	0,23	Due to a tank level of more than 50%, the prototype returns $-\infty$ as score S1, which leads to the same decision as in the survey (no recommendation).
2	0,53	0,74	0,26	A recommendation would be made, although the tank is still relatively full and the user usually refills at 12.5%. However, all other criteria are almost ideal preconditions for a recommendation.
3	0,46	0,8	0,2	The different results indicate that the number of co-drivers does not have such a great impact as assumed in the model. Furthermore, the underlying function for this attribute did not hold up in the survey. Apparently it is not true that there is a local minimum at "2 persons in the car". It rather seems, the more co-drivers there are, the less people want to tank (although the standard deviation is higher than in scenario no. 3, so there seem to be different opinions on that matter).
4	0,51	0,74	0,28	
5	+∞	0,95	0,1	The prototype returns +∞ since there is only one gas station in range and would, hence, make a recommendation. The evaluation revealed the same result.
6	0,46	0,63	0,29	The online survey's results are relatively strong deviated, so one should be careful interpreting these results. However, it seems like the questioned users' wish to receive recommendations depends less on the traffic situation than assumed. For this purpose it might be beneficial to decrease the weight of this
7	0,57	0,68	0,25	

				context dimension.
8	0,39	0,46	0,25	In scenario 8, there is a good match between prototype and survey results. However, scenario 9 leads to the conclusion that the tank level seems to have greater impact on the user's decision than assumed in the model.
9	0,64	0,86	0,23	
10	0,79	0,57	0,27	Real users seem to take the absolute detour as an indicator rather than the detour/route quota, or at least have a lower "maximum acceptable detour" threshold, which was assumed to be 10km in the model.
11	0,5	0,49	0,25	Although the recommender and survey would lead to different decisions, both values match almost perfectly. The difference between scenario 11 and 12 confirms the interpretation from scenarios 8 and 9: The tank level seems to be more important to users than assumed.
12	0,72	0,8	0,23	
13	0,53	0,76	0,23	As indicated in the interpretation regarding scenario no. 10, the required detour seems to be very important.

Table 2: Evaluation results

Although the recommender algorithm is very straightforward and rather simple, the online questionnaire showed that it yields good results (cf. Table 2). Only in 2 out of 13 scenarios, the recommender's decision clearly differed from the one made by the survey participants. To conclude, it seems like real drivers give more weight to the tank level and only factor in rather few context information. This also reflects in the answers provided in a free input field and regarding the question of how often recommendations should be made. Instead of naming frequencies, most participants entered that they wish to receive suggestions on the event that their tank level reaches a certain threshold (e.g. ¼ of the tank, or when the tank will only last for another 100km).

As can be seen in Figure 2, the semantic differential question revealed a very positive attitude towards such recommender systems, as long as they are not too invasive and disturbing whilst driving.

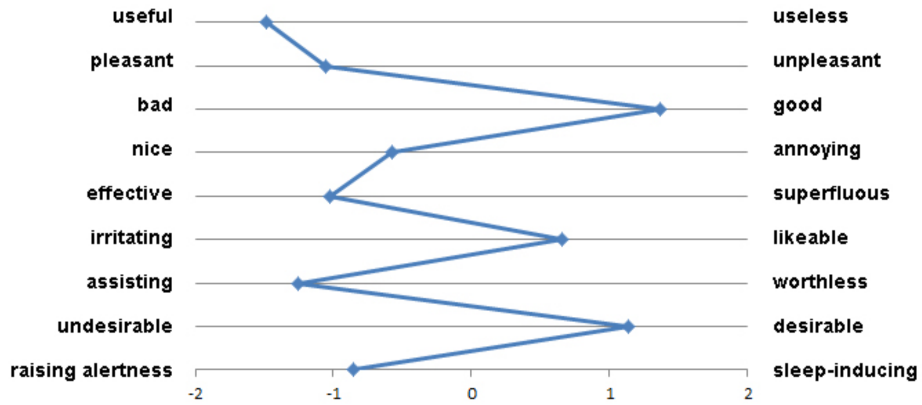


Figure 2: Semantic Differential

5. Conclusions

In this paper, a general model for proactive recommenders was developed and built into a first prototype for a gas station recommender. A subsequent online questionnaire proved high acceptance for such applications. Furthermore, many participants expressed appreciation for this support function as long as suggestions are communicated in a sensitive manner, which was the core motivation of this study. The proposed model offers instructions for how to implement a proactive recommender that makes propositions only when the recipient is ready to perceive them. In any case, however, it is crucial to analyze the relevant input factors to determine the user's current receptiveness for suggestions.

In conclusion, as [Pi2005] also constitutes, future research should be devoted on how to exploit more contextual information to achieve good timing for proactively generated suggestions.

The current technological capabilities (e.g. of smartphones) provide access to a large quantity of helpful data and it will be interesting to see to what extent the progress in sensor technologies will change the game in this regard. Current research on sound-based real-world event recognition (cf. [Ne2007]) and RFID or visual sensors to detect user's actions (cf. [Sa2008]) offers promising approaches that could be incorporated in proactive recommender systems in the future.

6. References

[Ad2008] Gediminas Adomavicius and Alexander Tuzhilin. Context-aware recommender systems. In Proceedings of the 2008 ACM Conference on Recommender systems, RecSys'08, ACM, New York, NY, USA, pp. 335–336, 2008.

[Ba2005] Mary Bazire and Patrick Brzillon. Understanding Context Before Using It. In Modeling and Using Context, pp. 29–40, 2005.

[Ba2011] Roland Bader, Eugen Neufeld, Wolfgang Woerndl and Vivian Prinz. Context-Aware POI Recommendations in an Automotive Scenario using Multi-Criteria Decision Making Methods. In Proc. Workshop on Context-aware Retrieval and Recommendation (CaRR 2011), Conference on Intelligent User Interfaces (IUI 2011), Palo Alto, CA, USA, 2010.

[De1999] Anind K. Dey and Gregory D. Abowd. Towards a Better Understanding of Context and Context-Awareness. In In HUC 99: Proceedings of the 1st International Symposium on Handheld and Ubiquitous Computing, Springer, pp. 304–307. 1999.

[Em2009] Andreas Emrich, Alexandra Chapko and Dirk Werth. Context-Aware Recommendations on Mobile Services: The m:Ciudad Approach. In Payam Barnaghi, Klaus Moessner, Mirko Presser and Stefan Meissner (Eds.), Smart Sensing and Context, Lecture Notes in Computer Science 5741, Springer, pp. 107–120, 2009.

[Hi2005] Lorenz M. Hilty. Pervasive Computing - A Case for the Precautionary Principle? In Dieter Hutter and Markus Ullmann (Eds.), Security in Pervasive Computing, Lecture Notes in Computer Science 3450, Springer, pp. 1-2, 2005.

[Ho2005] Joyce Ho and Stephen S. Intille. Using Context-Aware Computing to Reduce the Perceived Burden of Interruptions from Mobile Devices. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '05), ACM, New York, NY, USA, pp. 909–918, 2005.

[Ho2009] Jongyi Hong, Eui-Ho Suh, Junyoung Kim and SuYeon Kim. Context-Aware System for Proactive Personalized Service Based on Context History. Expert Syst. Appl., Vol. 36, pp. 7448–7457, May 2009.

[La1997] Jinke D. Van Der Laan, Adriaan Heino and Dick De Waard. A Simple Procedure for the Assessment of Acceptance of Advanced Transport Telematics. Transportation Research Part C: Emerging Technologies, Vol. 5, No. 1, pp. 1–10, 1997.

[Ma2006] Qusay H Mahmoud. Provisioning Context-Aware Advertisements to Wireless Mobile Users. In IEEE International Conference on Multimedia and Expo, pp. 669–672, 2006.

[Ne2007] Yuya Negishi and Nobuo Kawaguchi. Instant Learning Sound Sensor: Flexible Real-World Event Recognition System for Ubiquitous Computing. In Haruhisa Ichikawa, We-Duke Cho, Ichiro Satoh and Hee Youn (Eds.), Ubiquitous Computing Systems, Lecture Notes in Computer Science 483, Springer, pp. 72–85, 2007.

[Pi2005] Tapio Pitkäranta, Oriana Riva and Santtu Toivonen. Designing and Implementing a System for the Provision of Proactive Context-aware Services, In Workshop on Context Awareness for Proactive Systems (CAPS'05), 2005.

[Ri2010] Francesco Ricci. Mobile Recommender Systems. International Journal of Information Technology and Tourism, 2010.

[Ru2009] Almudena Ruiz-Iniesta, Guillermo Jimenez-Diaz and Mercedes Gomez-Albarran. Recommendation in Repositories of Learning Objects: A Proactive Approach that Exploits Diversity and Navigation-by-Proposing. In Advanced Learning Technologies (ICALT 2009), pp. 543 –545, 2009.

[Sa2008] Somkiat Sae-Ueng, Sineenard Pinyapong, Akihiro Ogino and Toshikazu Kato. Personalized Shopping Assistance Service at Ubiquitous Shop Space. In Advanced Information Networking and Applications (AINAW), pp. 838–843, 2008.

[We1991] Mark Weiser. The Computer for the 21st Century. SIGMOBILE Mob. Comput. Commun. Rev., Vol. 3, pp. 3–11, 1991.

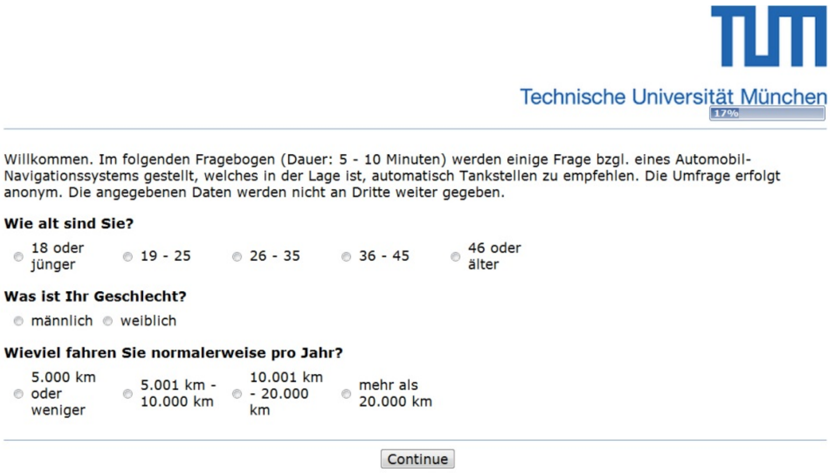
[Wo2007] Wolfgang Woerndl, Christian Schueller and Rolf Wojtech. A Hybrid Recommender System for Context-aware Recommendations of Mobile Applications. In. Proc. of the 3rd International Workshop on Web Personalization, Recommender Systems and Intelligent User Interfaces, 2007.

[Wo2009] Wolfgang Woerndl, Henrik Muehe, Stefan Rothlehner and Korbinian Moegele. Context-Aware Recommendations in Decentralized, Item-Based Collaborative Filtering on Mobile Devices. In Proc. Workshop on Innovative Mobile User Interactivity, MobiCASE Conf., San Diego, USA, 2009.

[Ye2010] Kam Fung Yeung and Yanyan Yang. A Proactive Personalized Mobile News Recommendation System. In Developments in E-systems Engineering (DESE), pp. 207–212, 2010.

7. Appendix

7.1. Online Questionnaire Screenshots


Technische Universität München

17%

Willkommen. Im folgenden Fragebogen (Dauer: 5 - 10 Minuten) werden einige Frage bzgl. eines Automobil-Navigationssystems gestellt, welches in der Lage ist, automatisch Tankstellen zu empfehlen. Die Umfrage erfolgt anonym. Die angegebenen Daten werden nicht an Dritte weiter gegeben.

Wie alt sind Sie?

18 oder jünger | 19 - 25 | 26 - 35 | 36 - 45 | 46 oder älter

Was ist Ihr Geschlecht?

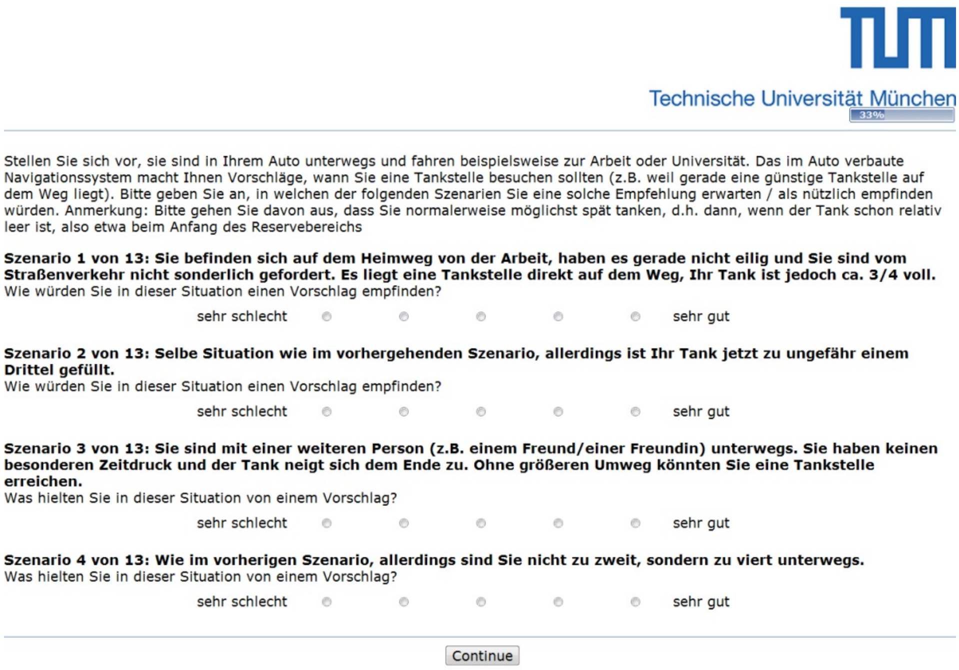
männlich | weiblich

Wieviel fahren Sie normalerweise pro Jahr?

5.000 km oder weniger | 5.001 km - 10.000 km | 10.001 km - 20.000 km | mehr als 20.000 km

Continue

Figure 3: Online questionnaire, page 1: Demographic data



Technische Universität München

33%

Stellen Sie sich vor, sie sind in Ihrem Auto unterwegs und fahren beispielsweise zur Arbeit oder Universität. Das im Auto verbaute Navigationssystem macht Ihnen Vorschläge, wann Sie eine Tankstelle besuchen sollten (z.B. weil gerade eine günstige Tankstelle auf dem Weg liegt). Bitte geben Sie an, in welchen der folgenden Szenarien Sie eine solche Empfehlung erwarten / als nützlich empfinden würden. Anmerkung: Bitte gehen Sie davon aus, dass Sie normalerweise möglichst spät tanken, d.h. dann, wenn der Tank schon relativ leer ist, also etwa beim Anfang des Reservebereichs

Szenario 1 von 13: Sie befinden sich auf dem Heimweg von der Arbeit, haben es gerade nicht eilig und Sie sind vom Straßenverkehr nicht sonderlich gefordert. Es liegt eine Tankstelle direkt auf dem Weg, Ihr Tank ist jedoch ca. 3/4 voll.
Wie würden Sie in dieser Situation einen Vorschlag empfinden?

sehr schlecht ○ ○ ○ ○ ○ sehr gut

Szenario 2 von 13: Selbe Situation wie im vorhergehenden Szenario, allerdings ist Ihr Tank jetzt zu ungefähr einem Drittel gefüllt.
Wie würden Sie in dieser Situation einen Vorschlag empfinden?

sehr schlecht ○ ○ ○ ○ ○ sehr gut

Szenario 3 von 13: Sie sind mit einer weiteren Person (z.B. einem Freund/einer Freundin) unterwegs. Sie haben keinen besonderen Zeitdruck und der Tank neigt sich dem Ende zu. Ohne größeren Umweg könnten Sie eine Tankstelle erreichen.
Was hielten Sie in dieser Situation von einem Vorschlag?

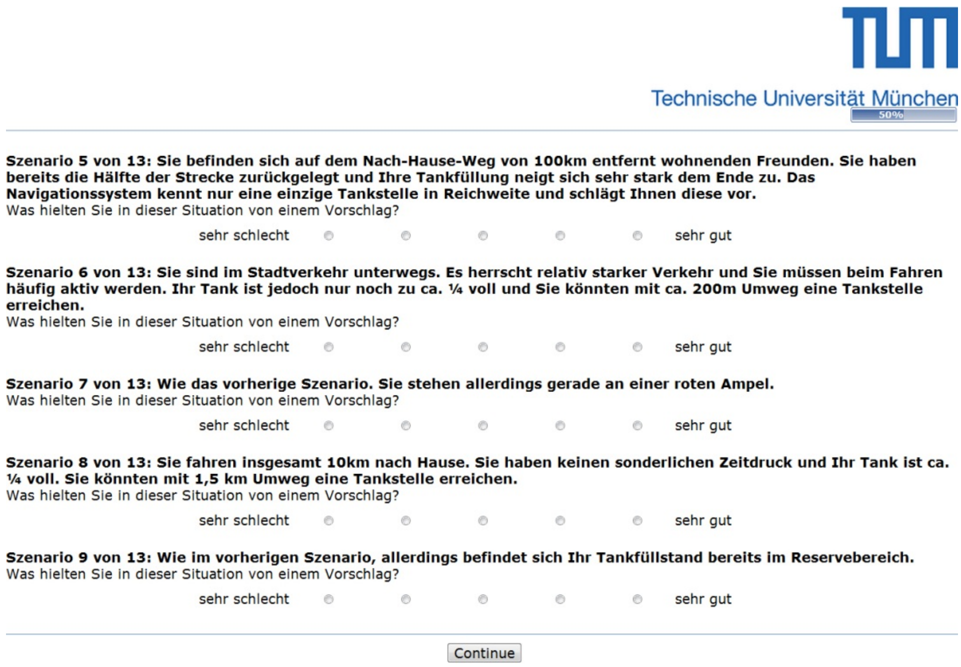
sehr schlecht ○ ○ ○ ○ ○ sehr gut

Szenario 4 von 13: Wie im vorherigen Szenario, allerdings sind Sie nicht zu zweit, sondern zu viert unterwegs.
Was hielten Sie in dieser Situation von einem Vorschlag?

sehr schlecht ○ ○ ○ ○ ○ sehr gut

Continue

Figure 4: Online questionnaire, page 2: Scenarios 1 - 4



Technische Universität München

50%

Szenario 5 von 13: Sie befinden sich auf dem Nach-Hause-Weg von 100km entfernt wohnenden Freunden. Sie haben bereits die Hälfte der Strecke zurückgelegt und Ihre Tankfüllung neigt sich sehr stark dem Ende zu. Das Navigationssystem kennt nur eine einzige Tankstelle in Reichweite und schlägt Ihnen diese vor.
Was hielten Sie in dieser Situation von einem Vorschlag?

sehr schlecht ○ ○ ○ ○ ○ sehr gut

Szenario 6 von 13: Sie sind im Stadtverkehr unterwegs. Es herrscht relativ starker Verkehr und Sie müssen beim Fahren häufig aktiv werden. Ihr Tank ist jedoch nur noch zu ca. ¼ voll und Sie könnten mit ca. 200m Umweg eine Tankstelle erreichen.
Was hielten Sie in dieser Situation von einem Vorschlag?

sehr schlecht ○ ○ ○ ○ ○ sehr gut

Szenario 7 von 13: Wie das vorherige Szenario. Sie stehen allerdings gerade an einer roten Ampel.
Was hielten Sie in dieser Situation von einem Vorschlag?

sehr schlecht ○ ○ ○ ○ ○ sehr gut

Szenario 8 von 13: Sie fahren insgesamt 10km nach Hause. Sie haben keinen sonderlichen Zeitdruck und Ihr Tank ist ca. ¼ voll. Sie könnten mit 1,5 km Umweg eine Tankstelle erreichen.
Was hielten Sie in dieser Situation von einem Vorschlag?

sehr schlecht ○ ○ ○ ○ ○ sehr gut

Szenario 9 von 13: Wie im vorherigen Szenario, allerdings befindet sich Ihr Tankfüllstand bereits im Reservebereich.
Was hielten Sie in dieser Situation von einem Vorschlag?

sehr schlecht ○ ○ ○ ○ ○ sehr gut

Continue

Figure 5: Online questionnaire, page 3: Scenarios 5 - 9

Technische Universität München

67%

Szenario 10 von 13: Sie fahren insgesamt 200km nach Hause. Sie haben keinen sonderlichen Zeitdruck und Ihr Tank ist ca. ¼ voll. Sie könnten mit 5 km Umweg eine Tankstelle erreichen.
Was hielten Sie in dieser Situation von einem Vorschlag?

sehr schlecht ○ ○ ○ ○ ○ sehr gut

Szenario 11 von 13: Sie fahren insgesamt 10km nach Hause. Sie haben keinen sonderlichen Zeitdruck und Ihr Tank ist ca. ¼ voll. Sie könnten mit 1,5 km Umweg eine Tankstelle erreichen.
Was hielten Sie in dieser Situation von einem Vorschlag?

sehr schlecht ○ ○ ○ ○ ○ sehr gut

Szenario 12 von 13: Wie im vorherigen Szenario, allerdings befindet sich Ihr Tankfüllstand im Reservebereich.
Was hielten Sie in dieser Situation von einem Vorschlag?

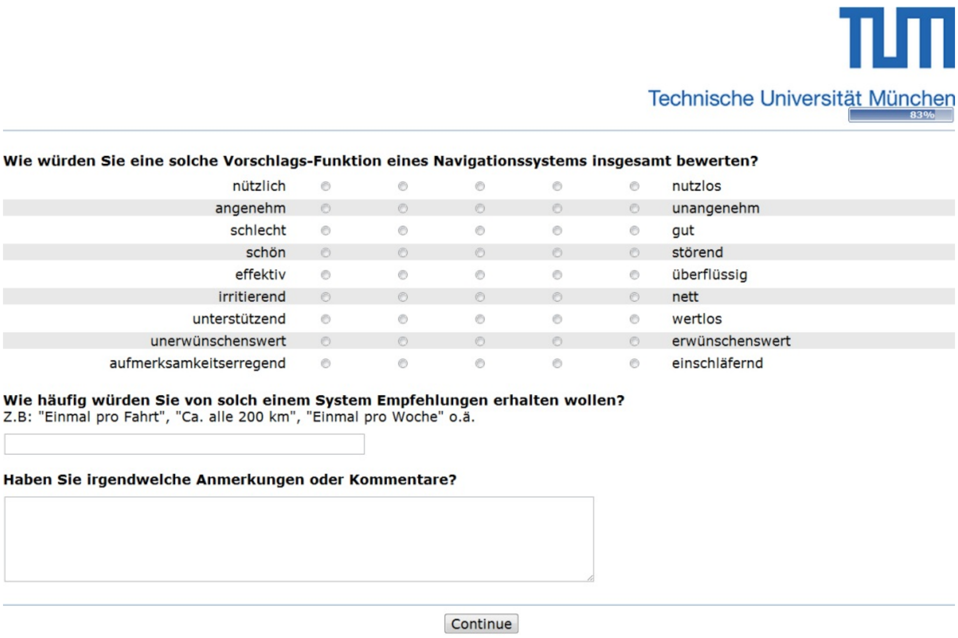
sehr schlecht ○ ○ ○ ○ ○ sehr gut

Szenario 13 von 13: Wie im vorletzten Szenario, d.h. Ihr Tank ist ¼ voll. Sie könnten eine Tankstelle mit 500m Umweg erreichen.
Was hielten Sie in dieser Situation von einem Vorschlag?

sehr schlecht ○ ○ ○ ○ ○ sehr gut

Continue

Figure 6: Online questionnaire, page 4: Scenarios 10 - 13



Technische Universität München

83%

Wie würden Sie eine solche Vorschlags-Funktion eines Navigationssystems insgesamt bewerten?

nützlich	○	○	○	○	○	nutzlos
angenehm	○	○	○	○	○	unangenehm
schlecht	○	○	○	○	○	gut
schön	○	○	○	○	○	störend
effektiv	○	○	○	○	○	überflüssig
irritierend	○	○	○	○	○	nett
unterstützend	○	○	○	○	○	wertlos
unerwünschenswert	○	○	○	○	○	erwünschenswert
aufmerksamkeitserregend	○	○	○	○	○	einschläfernd

Wie häufig würden Sie von solch einem System Empfehlungen erhalten wollen?
Z.B: "Einmal pro Fahrt", "Ca. alle 200 km", "Einmal pro Woche" o.ä.

Haben Sie irgendwelche Anmerkungen oder Kommentare?

Continue

Figure 7: Online questionnaire, page 5: Semantic differential and frequency of recommendations

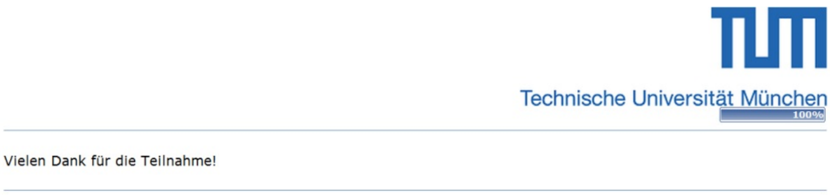


Figure 8: Online questionnaire, page 6: Completion confirmation

7.2. Scenarios

#	Scenario Description
1	You are currently on the way home from work. You are not in a hurry and driving is not particularly challenging at the moment, since there is only little traffic. There is a gas station right on your route, but your tank is still ¾ filled.
2	Like scenario #1, but your tank level is ⅓.
3	You are driving together with one further person (e.g. a friend). You are not in a hurry and your tank is almost empty. You could reach a gas station without a large detour.
4	Like scenario #3, but you have three persons on board (instead of one).
5	You are on your way home from visiting a friend who is living 100km away from your place. You already drove more than half of the track. Your tank will be empty soon. The navigation system knows only one gas station in your range and proposes it to you.
6	You are driving in the city and there is relatively heavy traffic so that you must interfere frequently. However, your current tank level is only at about ¼ and you could reach a gas station with a 200m detour.
7	Like scenario #6, but you are currently waiting for a red light to turn green.
8	You are on a 10km drive on your way home. You are not particularly rushed and your tank level is at about ¼. You could reach a gas station with a 1.5km detour.
9	Like the previous scenario, but your tank level indicator is already in the reserve area.

10	You are on a 200km drive to your home. You do not have particularly much time pressure and your tank level is about ¼. With a detour of 5km you could reach a gas station.
11	You are on a 10km drive to your home. You do not have particular much time pressure and your tank level is about ¼. With a detour of 1.5km you could reach a gas station.
12	Like the previous scenario, but your tank level is in the reserve area.
13	Like in the scenario before the previous, i.e. your tank level is ¼. However, you could reach a gas station with a 500m detour.

Table 3: Scenario Descriptions