UDC 004.62:654.164
Survey Paper

Automated motive-based user review analysis in the context of mobile app acceptance: Opportunities and applications

Elisabeth Pergler *elisabeth.pergler@evolaris.net*
evolaris next level GmbH

Abstract

The increasing amount and importance of user-generated content that is available in the web necessitates methods of its effective and efficient utilization. In this paper a system is presented that enables automated classification of user reviews concerning the usage motives mentioned in the reviews. Four possible applications of the system are discussed in detail in the course of this paper: a learning environment for mobile app development, a download prognosis mechanism, an app prizing decision support system, and a recommendation system. These applications are evaluated concerning state-of-the-art methods that currently address the challenge as well as advantages of the implementation of a motive-based system in the particular business processes.

Keywords: Technology acceptance research, user-generated content mining, motive classification

1. Introduction

Mobile apps constitute a growing market due to the fact that smart phones experienced a rapid global diffusion in the course of the past years. The demand for mobile apps is growing and at the same time the variety of offered apps is increasing too. The vast range of available apps is hardly ascertainable and reaches from simple programs such as alarm clocks to very sophisticated software products such as car navigation tools. The heterogeneity of the apps market and its expansion cause uncertainty among developers of apps as well as investors regarding the potential success of apps.

Technology acceptance research has a long tradition in the field of information systems research. This is due to the fact that people always wanted to know in advance whether a certain technology will be successful and understand causalities leading to this acceptance. The main objective of technology acceptance research is gaining insights into factors that affect the behavioral intention to further use a certain technology. Usually the outcome of technology acceptance research projects is an explanatory model of technology usage mostly limited to a specific domain. Classical technology acceptance research shows three major shortcomings [19]:

1. It is static either because of static theory or because of static methods. This static nature neglects the dynamic nature of technology acceptance where time may change perceptions [10], [21], [24] respectively that the time of study conduction (e.g. before or after hands-on experience [e.g. 2]) may influence the result. The lack of dynamics hampers the ability to come up with the especially dynamic mobile service market.
2. The results are highly aggregated fuzzy acceptance factors that do not allow intersubjective understanding. Aggregation enables simplification of models like the most often used Technology Acceptance Model [6] tries to explain acceptance using only two main antecedents of acceptance, namely „perceived ease of use“ and „perceived usefulness“. Further understanding or convention of what makes a technological artifact easy to use or useful is still missing [1]. Therefore designers and developers are not able to derive requirements or design implications for new service development from results of technology acceptance research.

3. Technology acceptance research methods that include real-world experience with the technology such as field tests or experimental studies require prototypes on a ready-to-use-level in order to obtain hands-on experience of respondents. Therefore only minor changes or incremental innovations are caused by the results as investment concerning product development and infrastructure has already been made. Radical innovations are nearly impossible.

These shortcomings lead to technological innovations in the field of mobile apps that are not accepted by the users and therefore are not economically successful. Examples of such unsuccessful apps are numerous on the market and reach from economic disasters like mobile TV to simple false estimations of users' willingness to pay for an app. These failures are caused by the fact that developers did not know or could not interpret what really makes somebody use this app – what are his motives and which motives are satisfied by the app.

These problems are addressed by a new framework where technology acceptance research by means of user-generated content analysis is included in the very first phase of the innovation process. This approach will support the understanding of usage motives by means of already successful apps and therefore enable the creation of apps that do address real user motives.

Two main research questions arise within this context and will be dealt with in the remainder of this paper:

4. Is it possible to extract usage motives from user reviews of mobile apps by means of automated content analysis? Respectively is a machine capable of extracting meaningful motive classes from unstructured text that resembles text message style with regard to wording, grammar, abbreviations, typing errors etc.?
5. Which problems in the context of app design can be solved or at least mitigated by this method? Respectively what are the most reasonable applications of the proposed system and where are its potentials?

The framework has been implemented in an exemplary prototype that analyses acceptance of mobile apps in Apple's AppStore. As this is only one possible technological implementation of the system there exist several further application scenarios that go beyond mere extraction of motives. Four of them are discussed in this paper and the opportunities and advantages of the presented system are analyzed in detail.

The remainder of this paper is organized as follows: In chapter 2 the conceptual framework is presented. Chapter 3 deals with relevant state of the art methods of automated user-generated content analysis. The system proposal itself is outlined in chapter 4 followed by an outlook on further applications of the system that address more detailed challenges of app design, development and distribution. The paper concludes with some general remarks and an outlook on further research that is to be done in the context of the presented approach.

2. Conceptual framework

The common procedure of mobile service innovation begins with creativity tools that aid idea generation. User requirements are hardly considered in this phase. As soon as prototypes are at hand it is possible to conduct acceptance research or market research. By this time the innovation process has moved to a phase where incremental innovations are more likely than radical ones though it is the first consideration of actual user opinions. Analysis of user-generated content usually starts after market launch.

In the course of the reshaped procedure user reviews are analyzed before idea generation starts. User-generated content offers opinions from a diffuse mass market that are hard or even impossible to assess by other data gathering methods (e.g. questionnaire-based surveys). This offers the opportunity to learn from successful apps that are similar to the one that is to be designed and developed. User reviews are provided by users with actual usage experience and are not biased by questionnaire specific issues. These biases may occur when opinions are not provided on voluntary bases or due to limited intersubjectivity of surveyed constructs. The fact that authors of user reviews have actual usage experience avoids results that are founded on mere speculations concerning a certain technology (e.g. „Imagine you could watch TV on your mobile phone. What would you like best about that?“). The high number of reviews reduces the effect of statistical outliers on the final results. The digital nature of user-generated content enables its immediate usage and automated processing.

The reviews are then classified according to usage motives that are addressed in the text. This allows comparison of reviews concerning a heterogeneous set of apps. Reiss' model of motivation [20] was chosen as it is independent from technological changes over the time and also includes motives for any kind of human activities. This was considered to be the second best solution as the optimum would include a prior clustering of the reviews and motive model derivation from the obtained clusters. As the main goal of this research is not theory building in the area of motivational psychology we decided for the alternative of choosing an existing model. Reiss’ model of motivation is not being justified or validated in the course of its application. It serves as a tool for the classification process but can be substituted by any other independent motivational model. Some of its characteristics made it especially useful for this purpose such as the relatively big number of motives. This enables conclusions concerning the ability of our system to differentiate between many classes that might be overlapping. The 16 basic desires of the model and the corresponding intrinsic feelings are listed in table 1 below.

Motive class	**Motive**	**Intrinsic feeling**
Power	Desire to influence (including leadership; related to mastery)	Efficacy
Curiosity	Desire for knowledge	Wonder
Independence	Desire to be autonomous	Freedom
Status	Desire for social standing (including desire for attention)	Self-importance
Social Contact	Desire for peer companionship (desire to play)	Fun
Vengeance	Desire to get even (Including desire to compete, to win)	Vindication
Honor	Desire to obey a traditional moral code	Loyalty
Idealism	Desire to improve society (including altruism, justice)	Compassion
Physical exercise	Desire to exercise muscles	Vitality
Romance	Desire for sex (including courting)	Lust
Family	Desire to raise own children	Love
Order	Desire to organize (including desire for ritual)	Stability
Eating	Desire to eat	Satiation (avoidance of hunger)
Acceptance	Desire for approval	Self-confidence
Tranquility	Desire to avoid anxiety, fear	Safe, relaxed
Saving	Desire to collect, value of frugality	Ownership

Table 1. 16 basic desires of Reiss’ motivational model

An automated system then learns to attach these motives to the user reviews. This is done based on training of a limited number of reviews that were previously labeled by hand. The automation allows continuous real-time monitoring of the huge amount of reviews available for mobile apps.

3. State of the art methods

The methods described in this chapter are well-known techniques as well as state of the art approaches to the task of user-generated content analysis.

3.1. Sentiment analysis

A commonly used method for classification of user reviews is sentiment analysis. The basic functionality of sentiment analysis is to differentiate positive and negative semantic orientation of the reviews in order to aggregate the available text pieces. Therefore the main task is „identification and assessment of opinions“ [25] which are not explicitly phrased. This task is often performed by means of sentiment word lists or sentiment rules that are based on part-of-speech tags that were obtained from previously annotated corpora [e.g. 15 and 26]. Valence shifters [e.g. 9] can enhance sentiment classification by integration of intensity of sentiments into the classification decision. Besides word level sentiment classification efforts have been made to classify text on phrase level [30].

Usage of sentiment word lists is not useful for the task of mobile app review classification as app reviews are written on mobile devices with limited input options and consequently often show text message style text. Text message style is changing dynamically and fast. Moreover there is no solution to noisy text features such as irony or sarcasm. “GREAT job you Facebook people! I want my money back.” is not a positive opinion for instance and “This is too addicting. I want my life back!!!” is not negative at all. Additionally the mere classification into positive and negative reviews will not provide app designers and developers with highly useful and design relevant information.

3.2. Feature based opinion mining

A form of sentiment analysis that connects sentiment information with particular product characteristics is feature based opinion mining. The product features are either predefined by usage of word lists [e.g. 16] or assessed by association sets [e.g. 22] which include opinion words as well as product categories and with that allow feature identification without explicit verbalization of feature words. Another method that is independent from actual appearance of feature words is shallow parsing [29]. It takes advantage of phrases appearing in the text and then utilizes them to extract relations between feature and opinion. A completely different approach to obtain product features and sentiment orientation is the assessment of domain knowledge. Domain knowledge can be obtained by mining structured or semi-structured reviews [e.g. 14]. This form of rule-based method can also be applied in the feature and sentiment association task [23]. Text clustering and text summarization [28] are further approaches to obtain features but in contrast to the other methods the feature list is created based on information obtained from the reviews. These approaches are based on the assumption that there are product features which are intersubjectively comparable. The definition of such features is not possible for mobile apps due to two main reasons. First of all mobile apps are a tremendously heterogeneous collection of different products unlike the products where review mining has already been conducted in this way (e.g. movies in [23] or hotels in [16]). This is the reason why it is impossible to define a fixed set of features that is applicable to all apps. Secondly the determination of certain features would lead to a static analysis and therefore neglect dynamic changes of technological possibilities and customer requirements. Static feature concepts are even used for dynamic opinion mining approaches [11]. Additionally most of these methods require misspelling corrections and stemming of the text which results in loss of valuable information (e.g. the information value of “fun” is different to that of “FUUUNNN”).

3.3. Latent semantic indexing

Latent semantic indexing is an approach that aims to reveal latent constructs or topics in language text by help of occurrence matrixes. Extensions of the probabilistic model have been applied to the task of detection of characteristics that are important to particular communities in order to like or dislike a product or service [8]. One shortcoming of latent semantic indexing with regard to the review classification task that should be applicable to numerous data sets is its varying performance for different data sets. The reasons for these performance variations are not fully detected yet [12].

3.4. Review quality evaluation

Another task in the context of the presented research project is evaluation of review quality [4]. The aim is the identification of important reviews that should be used for further analysis while noisy text pieces are filtered out. This approach introduces a further classification problem to the task of review classification.

4. System proposal

In this chapter the exemplary application of a motive-based acceptance analysis system to Apple's AppStore is presented as one form of technological implementation. The suggested method and the system could be applied to any data source that fulfills certain requirements. The general process from appropriate choice of a data source to final interpretation of the obtained results is depicted in figure 1 and each step is further explained in detail in the remainder of this chapter.

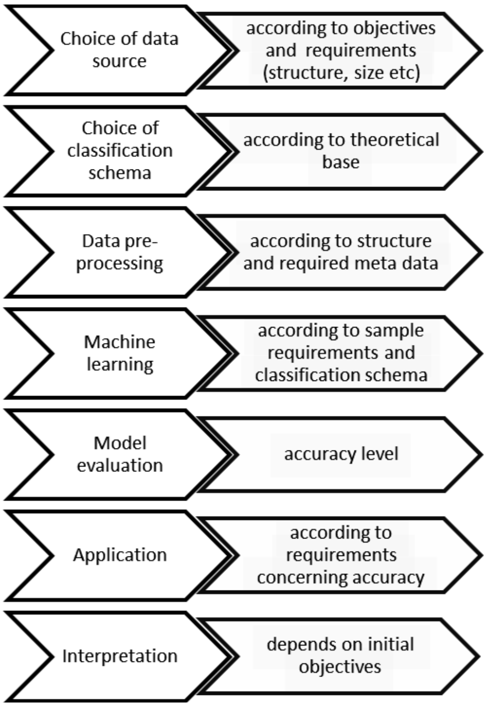
Choice of data source
according to objectives and requirements (structure, size etc)
Choice of classification schema
according to theoretical base
Data pre-processing
according to structure and required meta data
Machine learning
according to sample requirements and classification schema
Model evaluation
accuracy level
Application
according to requirements concerning accuracy
Interpretation
depends on initial objectives

Figure 1: general process of the suggested system

4.1. Data collection

The data source needs to meet three criteria in order to be eligible for choice:

- The number of available user opinions respectively reviews has to be relatively high. This is important for the statistical relevance of the obtained results and also for the avoidance of biases caused by outliers.
- Reviews need to be provided only by people with actual usage experience in order to avoid extraction of alleged motives that are rather imaginations or expectations that actual usage motives.
- The data needs to be structured (e.g. reviews must be distinguishable from other text passages) and indicators concerning the success of an app (e.g. download numbers) need to be included. This is required by the technical platform that is used. There exist approaches to the challenge of extracting reviews from unstructured text but this

would include another classification task and therefore mitigate the informative value of the results obtained in this research project as more sources of error would be added to the process.

Apple's AppStore is the app distribution platform that meets these needs at the best. More than 400000 apps are available and there were more than 10 million downloads by the beginning of 2011. Reviews can only be written after downloading a certain app which ensures actual usage experience of the authors. The data is structured and includes meta data such as name, category, download rank and reviews. As it is intended to learn from successful apps only the top 100 apps of each category (top free, top paid and top grossing) were included. The data was transferred into xml files for further processing. The files concerning very successful apps that consequently obtained a higher number of reviews were partitioned. A threshold of 200 reviews per file was chosen to simplify the machine learning process and its evaluation.

This procedure led to 1588 xml files including 277345 reviews. These files were then screened for duplicates as a top paid app can also be top grossing. The filtering resulted in 1132 unique files in the xml structure depicted in figure 2.

```
<application>
    <name/>
    <author/>
    <rank/>
    <category/>
    <app-reviews>
        <app-review>
            <title/>
            <author/>
            <date/>
            <version/>
            <rating/>
            <review/>
        </app-review>
    </app-reviews>
</application>
```

Figure 2: Structure of the xml files after read-out

4.2. Classification schema

Definition of classes is a crucial decision with regard to the possible results that are provided by the system. The classes serve as coding categories and determine which kind of output can be obtained. The classification schema could be obtained by inductive formation out of the data (e.g. clustering methods) or by deductive application of theories. In this case a motivational model [20] that is widely used in psychology research is applied. In fact any motivational model could be used but Reiss' model of motivation was applied in order to test the abilities of the system to cope with large numbers of classes. The obtained classification should be more expressive than mere positive-negative-neutral differentiation. Some examples of manual classification are listed in table 2 below.

Original review text	**Class**
„What a BLAST Tap Zoo has been to play! Our whole family plays-Talk about Family Bonding Time! It gives us something to talk about everyday! It‘s just a silly, Fun game for everyone!“	Family
„This game keeps me entertained forever!! I‘m always competing with	Vengeance

my friends for high scores!“	
„Waaaaaaay better than paying $2 for a stupid rington. Awesum app!!!!!□“	Saving
„Like Angry Birds but don‘t have anything against pigs? How bout America? Do you love America? Well then Fragger is the game for you!“	Honour
„Please add the ability to delete Friends. I already tired of people‘s drama.“	none

Table 2. Examples of manually annotated user reviews.

4.3. Data pre-processing

The data needs some pre-processing in order to be useful for further analysis. Most important is tokenization of the text which is done by means of finite-state transducers. The process mentioned above results in single tokens that are annotated with text strings and information concerning upper and lower case spelling. We refrained from further pre-processing such as stemming as this might cause a significant loss of information regarding actual usage motives. As mentioned above several methods that are based on word lists or ontologies might require such extensive pre-processing but the in the case of usage motive annotation the trade-off between information loss and higher level semantic analysis was decided in favor of the preservation of information.

A random set of 60 xml files is chosen out of the 1132 existing files in order to train the system. The 9510 reviews included in these files are manually labeled with usage motives that are addressed within the review. In order to avoid researcher bias this annotation procedure was done by two researchers independently. Only the 3431 corresponding matches were resumed for the machine learning. The annotation was supported by the platform in GATE [5].

4.4. Machine learning

A machine learning algorithm was configured to be able to learn from the data labeled by hand. The support vector machine implemented in GATE [14] met all requirements concerning data structure and distribution present in the actual review data. Naive Bayes and other standard classifiers were also computed in order to get reference values.

The multiclass problem that arises from the 16 desires in the motivational model needed to be transferred into several binary problems. A one-versus-another approach was chosen in order to obtain better classification results at the expense of system performance. In contrast to the one-versus-others approach classification probabilities are compared for each possible pair of classes. Unigrams were formed out of the single token strings and upper/lower case spelling in order to design kernels for the machine learning process. Threshold probability of classification was set to the value of 0.4 in order to obtain a satisfying number of classified reviews but at the same time keep classification results meaningful. Estimates of classification confidence are not easy to obtain especially for Support Vector Machines and other classifiers that are not probabilistic in a strict sense [7]. This is why several estimates were computed and test applications of the trained model to the remaining reviews (approximately 270000) were run. These showed that both requirements were met at a confidence level of 0.4.

4.5. Evaluation

A hold-out test was performed to evaluate the prototype. The hold-out test was preferred to k-fold cross-validation due to additional computing expenses that would be caused by cross-validation [17]. Therefore the training data set was split into two parts. Two thirds of it served as a new training set for learning the classification model which was then applied to the remaining third of the data. In the course of the application process the machine tries to annotate the reviews of the new test set and then compares these annotations to the manually labeled annotations. This procedure is the reason for the 200 reviews per xml file threshold. The limited number of reviews per file ensures a balanced number of reviews in the two parts of the split training data set which is necessary for meaningful evaluation results. Without this precaution the randomly chosen files for the training could contain a very low number of reviews and therefore show bad performance when the learned model is applied to the remaining high number of reviews.

The learned model obtained an accuracy level (F1 measure) of 0.6699 over all classes. Precision and recall were almost equal at 0.6692 and 0.6705 respectively. The accuracy level is more than acceptable with regard to the high number of classes and the complexity of language in user reviews that were written on mobile devices. Detailed evaluation results on class level are provided in table 3 for each usage motive except for idealism that was not present in the training data set.

Motive class	Precision	Recall	F1
Power	0.0000	0.0000	0.0000
Curiosity	0.7134	0.8146	0.7602
Independence	0.3536	0.2874	0.3160
Status	0.6008	0.6210	0.5977
Social Contact	0.8593	0.8374	0.8480
Vengeance	0.4288	0.2447	0.3026
Honor	0.0000	0.0000	0.0000
Physical exercise	0.6593	0.5634	0.5910
Romance	0.0000	0.0000	0.0000
Family	0.7250	0.7292	0.7259
Order	0.3867	0.2344	0.2844
Eating	0.0000	0.0000	0.0000
Acceptance	0.3417	0.3616	0.3483
Tranquility	0.6308	0.5626	0.5931
Saving	0.4525	0.5650	0.4920

Table 3. Detailed evaluation results of the learned model

Some classes obtained better accuracy than others mainly due to the higher number of available positive examples in the training set. Manual annotation of further reviews addressing motives that were rare in the training set such as power, honor, eating, romance and idealism could improve overall accuracy of the learned model. Naive Bayes that was computed for reasons of comparison achieved an accuracy level 0.2775 probably because of inadequate data structure.

5. Applications

The scenarios described in this chapter depict possible applications and useful interpretations of the system. Currently a learning environment is available as a semi-automated prototype. It was used as an exemplary application to evaluate the general applicability of the basic ideas of the approach. The results of the evaluation indicate that it is possible to extract usage motives from user reviews of apps even though their structure is challenging. A second step of the evaluation in which the learned model was applied to the remaining data set of 300.000+ reviews and then a random sample of the annotations was manually checked for consistency did prove the meaningfulness of the automated classification.

5.1. A learning environment

App developers and designers are reliant on information concerning user requirements. These requirements are rarely explicitly stated and people involved in the app development process often conduct so-called “quick research” in order to find apps that are similar to the one that is to be developed. The resulting list of apps often lacks similarity with the motives that should be addressed by the new app. It is also not very clear what similarities lead to inclusion in the list. A structured representation of successful apps can aid this process and benefit the possible output. Expert interviews with app developers [18] indicated that a general overview of motive structures in reviews of successful apps is highly useful to avoid development of “more of the same apps” and niche services.

The possibility to repeat motive analysis at any moment allows trend analysis of motive structures which is useful for long-term projects. The results of motive annotation also enable the derivation of best practice apps that can then be analyzed in a creative-iterative learning environment. Designers and developers are provided with the information that aids them figuring out characteristics of apps that are highly accepted and include these in their own apps. Moreover it is possible to evaluate target achievement after market launch by comparing the motive structure of the present reviews associated to the own app to the motives that were previously set as targets.

5.2. A download prognosis mechanism

Some services require high investment during the development process. This is why developers need to know if the investment will be worthwhile. A prediction concerning potential downloads of the app will be useful to avoid investment in apps that are not demanded by users as they do not really address frequent usage motives.

When all reviews are annotated with motives it is possible to derive download ranks from motive structure by using motives as prediction variables of the target value download rank. The motives that are addressed by the new app can be obtained by means of early user integration and testing of prototypes. The result of this process is a download rank prognosis. Another approach would be to use text strings as input for the learning model instead of motives and predict download numbers on this basis. Of course the explanatory power of this prognosis is limited as the relationship between motives and willingness to pay is not documented yet but still it can help avoid high cost and time effort. Prior predictive applications of automated text analysis obtained very promising results. Sentiment analysis of blogs that was used to predict movie revenue numbers showed good accuracy [27]. Another study that analyzed sentiments in twitter posts to predict the development of stock market indexes achieved an accuracy level of 87.6 % which is also very accurate [3].

5.3. A prizing decision support

A very difficult task in the context of app development is finding an acceptable price for the app [18]. There exist various approaches from free demo versions in combination with paid extended versions to price changes over time. These strategies are useful to find acceptable prices but include image dangers. Price changes over time might result in unsatisfied users as the value for money aspect will be emphasized after price rises or reductions. An example for that is a game that did cost 20 USD in the beginning. Obviously it did not attract as many users as planned but those who downloaded it were happy with the game performance, design and other features regarding their initial reviews of the game. After a certain period the price was reduced to 5 USD which actually did lead to more downloads. An unintended outcome of this price reduction was that many early users did revise their reviews and changed them into value for money calculations because they did pay 20 USD for a game that was now obviously only worth 5 USD. Prior happiness among the user community turned into a critical mass of unsatisfied comments including complaints concerning lower perceived performance, etc. even though the game itself did not change at all.

As the meta data of each app also includes its price it is possible to use the motive information within the reviews to suggest a price that is suitable for the particular motive structure. Similar to the learning mechanism that is used for the automated motive labeling one can also set the price as a target value of the machine learning process. The obtained suggestions might help to find a reasonable price for the market launch before information concerning the own particular app is at hand. Therefore threats to app or even brand image are reduced.

5.4. A recommendation system

Whereas traditional recommender systems are based on explicit customer or user data the integration of motive-based content analysis offers new opportunities for product or service recommendation within online distribution platforms. Usually recommendations are derived from correlations between sales numbers such as “This article is often bought in combination with this article” or “People who bought this article also bought this article” or from artificial categories such as product classes (e.g. a customer bought an article from a certain category and therefore might also like other products in the same category). Recommender systems in social networks follow the same logics. Contacts are recommended as “This person is in contact with one of your friends and therefore you might know him as well” and activities are also recommended by analyzing existing connections. A motive-based approach introduces a broader range of possible recommendations. Content analysis of reviews and annotation with motives enables recommendation of products or services with similar motive structure. Another possibility is to recommend products or services that address the same motive as the particular user addressed in his or her own review.

In the context of social networks this procedure allows recommendations apart from already known real-world social contacts. It is possible to recommend a new contact based on his or her motive structure which is similar to the user’s motive structure. This takes recommendations from “you might know this person” to the “you might not know but will potentially like this person” level.

6. Discussion and outlook

The presented system was evaluated by means of functional testing and obtained convincing results that are also promising concerning further applications of the system as presented in this paper. The prototypical implementation enabled the proof of the concept of utilizing user reviews for acceptance research at an early stage. The general feasibility of the classification approach concerning usage motives addressed in app reviews is indicated by the satisfying accuracy of the implemented machine learning model. The first of the two main research questions concerning the general possibility that a machine can extract meaningful motives from user reviews with particular characteristics can therefore be answered positively.

Possible applications of the system address various aspects of mobile app development and design and could save money and time in the process. This might be a preliminary answer to the second research question regarding potentials of reasonable applications that aid solving problems in the context of app development. The usefulness of the applications can be evaluated in their practical implementation.

The next steps in the course of this research project arise in the context of its evaluation. First of all the system needs to be evaluated concerning its usefulness in the app development process. This will be done by means of comparison to state of the art methods in the first place. Later, after implementation in practice usage, it will also possible to evaluate the results obtained by application of the system. Precondition for this kind of evaluation is that the system is applied in real app development processes and causes superior results in terms of successful apps that were developed using the system there.

Another aspect is the evaluation of the usefulness in the context of acceptance research. The system will be tested on dynamic models of acceptance research and compared to other methods with dynamic aspects such as open innovation or experimental studies. These dynamic models will require retraining of the leaning model according to the particular model. The predictive power of the prognosis applications can be evaluated against other predictive methods such as prediction markets or predictive analytics by means of TAM. This is planned for a follow-up project. In the course of this project a limited set of new apps from the same domain will be used for an experimental comparison. The apps will be tested by a sample of users who will then give feedback by means of user reviews as well as the standardized TAM questionnaire. Next to that they will also be asked to trade shares of these apps on a virtual market during the test period. This procedure is well known from foresight research as prediction markets method. The results will give evidence for the usefulness and appropriateness of the suggested system regarding other established methods in the context of usage prediction.

References

[1] Benbasat, I.; Barki, H. Quo vadis, TAM? *Journal of the Association for Information Systems*, 8 (4):211-218, 2007.

[2] Bhattacherjee, A.; Premkumar, G. Understanding Changes in Belief and Attitude Toward Information Technology Usage: A Theoretical Model and Longitudinal Test. *MIS Quarterly*, 28 (2):229-254, 2004.

[3] Bollen, J.; Mao, H.; Zeng, X. Twitter mood predicts the stock market. *Journal of Computational Science*, 2 (1):1-8, 2011.

[4] Chen, C.; Tseng, Y. Quality evaluation of product reviews using an information quality framework. *Decision Support Systems*, 50 (4):755-768, 2011.

[5] Cunningham, H.; Maynard, D.; Bontcheva, K.; Tablan, V. GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02)*, Philadelphia, July 2002.

[6] Davis, F. Perceived Usefulness, Perceived ease of Use, and User Acceptance of Information Technology. *MIS Quarterly*, 13 (3):319-340, 1989.

[7] Delany, S.; Cunningham, P.; Doyle, D.; Zamolotskikh, A. Generating Estimates of Classification Confidence for a Case-Based Spam Filter. *Lecture Notes in Computer Science* 3620:177-190, 2005.

[8] Kagie, M.; Van der Loos, M.; Wezel, M. v. Including item characteristics in the probabilistic latent semantic analysis model for collaborative filtering. *AI Communications*, 22 (4):249-265, 2009.

[9] Kennedy, A.; Inkpen, D. Sentiment Classification of Movie Reviews Using Contextual Valence Shifters. *Computational Intelligence*, 22 (2):110-125, 2006.

[10] Kollmann, T. *Akzeptanz individueller Nutzungsgüter und –systeme. Konsequenzen für die Einführung von Telekommunikations- und Multimediasystemen* (Acceptance of individual goods and systems. Consequences for the introduction of telecommunication and multimedia systems.), Gabler, Wiesbaden, Germany, 1998.

[11] Ku, L.; Ho, H.; Chen, H. Opinion mining and relationship discovery using CopeOpi opinion analysis system. *Journal of the American Society for Information Science & Technology*, 60 (7):1486-1503, 2009.

[12] Li, D.; Kwong, C. Understanding latent semantic indexing: A topological structure analysis using Q-analysis. *Journal of the American Society for Information Science & Technology*, 61 (3):592-608, 2010.

[13] Li, Y.; Bontcheva, K.; Cunningham, H. Adapting SVM for Data Sparseness and Imbalance: A Case Study on Information Extraction. *Natural Language Engineering*, 15 (02):241-271, 2009.

[14] Miao, Q.; Li, Q.; Zeng, D. Fine-grained opinion mining by integrating multiple review sources. *Journal of the American Society for Information Science & Technology*, 61 (11):2288-2299, 2010.

[15] Na, J.; Khoo, C.; Wu, P. Use of negation phrases in automatic sentiment classification of product reviews. *Library Collections, Acquisitions, & Technical Services*, 29 (2):180-191, 2005.

[16] Pekar, V.; Ou, S. Discovery of subjective evaluations of product features in hotel reviews. *Journal of Vacation Marketing*, 14 (2):145-155, 2008.

[17] Platt, J. Probabilities for SV Machines. In *Advances in Large Margin Classifiers*, pages 61-74, MIT Press, 1999.

[18] Platzer, E.; Petrovic, O. A Learning Environment for Developers of Mobile Apps. In *Proceedings of the IEEE Engineering Education (EDUCON)*, pages 14-19, 4th – 6th April, Amman, Jordan, 2011.

[19] Platzer, E. A framework to support the design of mobile applications. In *Proceedings of the International Conference on Computer Networks and Mobile Computing* , pages 290- 296, 24th – 26th November, Venice, Italy, 2010.

[20] Reiss, S. Multifaceted Nature of Intrinsic Motivation: The Theory of 16 Basic Desires. *Review of General Psychology*, 8 (3):179-193, 2004.

[21] Schwarz, A.; Chin, W. Looking Forward: Toward an Understanding of the Nature and Definition of IT Acceptance. *Journal of the Association for Information Systems*, 8 (4):230-243, 2007.

[22] Su, Q.; Zhu, Y.; Swen, B.; Yu, S. Mining Feature-based Opinion Expressions by Mutual Information Approach. *International Journal of Computer Processing of Oriental Languages*, 20 (2/3):137-150, 2007.

[23] Thet, T.; Na, J.; Khoo, C. Aspect-based sentiment analysis of movie reviews on discussion boards. *Journal of Information Science*, 36 (6):823-848, 2010.

[24] Waarts, E.; van Everdingen, Y. The dynamics of factors affecting the adoption of innovations. *Journal of Product Innovation Management*, 19 (6):412-423, 2002.

[25] Wright, A. Our Sentiments, Exactly. *Communications of the ACM*, 52 (4):14-15, 2009.

[26] Ye, Q.; Zhang, Z.; Law, R. Sentiment classification of online reviews to travel destinations by supervised machine learning approaches. *Expert Systems with Applications*, 36 (3):6527-6535, 2009.

[27] Yu, X.; Liu, Y.; Huang, X.; An, A. Mining Online Reviews for Predicting Sales Performance: A Case Study in the Movie Domain. *IEEE Transactions on Knowledge and Data Engineering*, PP (99):1, 2010.

[28] Zhan, J.; Loh, H.; Liu, Y. Gather customer concerns from online product reviews – A text summarization approach. *Expert Systems with Applications*, 36 (2):2107-2115, 2009.

[29] Zhang, Q.; Wu, Y.; Li, T.; Ogihara, M.; Johnson, J.; Huang, X. Mining Product Reviews Based on Shallow Dependency Parsing. In *Proceedings of the SIGIR(Special Interest Group on Information Retrieval) Forum*, pages 726-727, 2009.

[30] Zhang, Z. Weighing Stars: Aggregating Online Product Reviews for Intelligent E-commerce Applications. *IEEE Intelligent Systems*, 23 (5):42-49, 2008.