
This is an electronic reprint of the original article.
This reprint may differ from the original in pagination and typographic detail.

Hitz, Andreas; Naas, Si Ahmed; Sigg, Stephan

Sharing geotagged pictures for an Emotion-based Recommender System

Published in:

2021 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events, PerCom Workshops 2021

DOI:

[10.1109/PerComWorkshops51409.2021.9430978](https://doi.org/10.1109/PerComWorkshops51409.2021.9430978)

Published: 25/05/2021

Document Version

Peer-reviewed accepted author manuscript, also known as Final accepted manuscript or Post-print

Please cite the original version:

Hitz, A., Naas, S. A., & Sigg, S. (2021). Sharing geotagged pictures for an Emotion-based Recommender System. In *2021 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events, PerCom Workshops 2021* (pp. 68-73). Article 9430978 (2021 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events, PerCom Workshops 2021). IEEE. <https://doi.org/10.1109/PerComWorkshops51409.2021.9430978>

This material is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of the repository collections is not permitted, except that material may be duplicated by you for your research use or educational purposes in electronic or print form. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone who is not an authorised user.

© 2021 IEEE. This is the author's version of an article that has been published by IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Sharing geotagged pictures for an Emotion-based Recommender System

Andreas Hitz^{*†}, Si-Ahmed Naas[†], Stephan Sigg[†]

^{*}{firstname.lastname}@tum.de [†]{firstname.lastname}@aalto.fi

Abstract—Recommender systems are prominently used for movie or app recommendation or in e-commerce by considering profiles, past preferences and increasingly also further personalized measures. We designed and implemented an emotion-based recommender system for city visitors that takes into account user emotion and user location for the recommendation process. We conducted a comparative study between the emotion-based recommender system and recommender systems based on traditional measures. Our evaluation study involved 28 participants and the experiments showed that the emotion-based recommender system increased the average rating of the recommendation by almost 19%. We conclude that the use of emotion can significantly improve the results and especially their level of personalization.

Index Terms—Artificial Intelligence, Recommender Systems, Emotions, City tour guide.

I. INTRODUCTION

Recommender systems have been extensively used by online services [1]. Popular applications are movie, app or product recommendations [2], [3]. Recommender systems predict the popularity of certain items and aim to give personalized recommendations. In the area of tourism, recommending places to visit is a common application. Most existing systems are built on analyzing general facts about its users, characteristics of the places, or contextual information such as location, time of day, or weather [4].

Besides these aspects, emotions are another important personal factor that can help to optimize the recommendations.

In this paper, we design and implement a location- and emotion-based recommender system to enhance personalized recommendations. The system is implemented in Python on the backend side, where all the data is processed and the recommendations are calculated, and in HTML and JavaScript on the frontend side. Emotion is used as a further parameter for recommendations, in addition to location. A user of the system can select her desired emotion (e.g. happy, calm, excited, relaxed), to experience during the trip, and based on this, nearby events that feature similar sentiment are recommended from recently shared geotagged pictures taken by other users of the system.

A user study is carried out, which finds that using the emotion component significantly improves the quality of recommendations. In combination with other parameters, this can lead to recommender systems delivering more personalized results than state-of-the-art.

II. RELATED WORK

Recommender systems have been invented and implemented first in the 1990s, and are nowadays used for various purposes.

Recommender systems are as well subject to many research studies [5]. We distinguish two main approaches: content-based and collaborative filtering. In content-based filtering, the recommendation takes place by analyzing and categorizing specifically defined characteristics of an item, e.g. through keywords, and finding others with similar properties, thus requiring appropriate means to measure similarity [6].

In contrast, in collaborative filtering, predictions are based on decisions of the past, and on decisions which have been made by others. This means that a high amount of data is to be analyzed to understand user preferences and to eventually provide precise recommendations [6]. One example for collaborative filtering is, in online stores, the feature "Customers who bought this item also viewed/purchased" [7]. Disadvantages are, for instance, the cold start issue, sparsity of ratings among a large number of items and scalability (computational complexity for large amounts of items and users) [1].

Combining these two approaches into a hybrid system can make recommendations more accurate [8]. The authors in [9] systematically compare various recommender systems, and define seven different hybridization techniques. These include weighted, cascade, and feature combination. Netflix, for instance, is using a mixed technique by both comparing watching habits of similar users and recommending movies or series with user-preferred characteristics [10].

According to [2], there are three different use cases for mobile recommender systems: *Goal-oriented search*: "Where can I eat around here tonight?" *Location discovery*: "Which places around here are interesting? What can I do tonight?" *Routing and transport*: "How can I get from here to there?"

In pull-based recommender systems, the delivery of recommendations is only performed upon user request [11]. Reactive recommender systems react to changing situational context when generating recommendations, independent from explicit interventions through the user [12]. Proactive recommender systems use not only historic and current context, but pre-cache appropriate content from a server on the mobile device by trying to predict future context [13]. Table I summarizes features utilized by mobile recommender systems.

The use of emotion in recommender system is yet a novel domain. Emotion can be recognized from textual messages, or determined from a picture with any content. According to [14], it has become common to express feelings and opinions in social media. In their approach, emotional states are represented in a two dimensional circular space presented by Russell's Circumplex model [15] and Twitter messages are used as the input data for emotion recognition and their

Tasks	Data sources	Place recomm. types	Prediction categories	Mobility features	Applied algorithms	Evaluation methods
<ul style="list-style-type: none"> - Goal-oriented search - Location discovery - Routing and transport 	<ul style="list-style-type: none"> - Explicit - Implicit - Sensors 	<ul style="list-style-type: none"> - Categories - Next place - New places - Routes - Events 	<ul style="list-style-type: none"> - Popularity - Proximity 	<ul style="list-style-type: none"> - Place - User - Structure 	<ul style="list-style-type: none"> - Supervised learning - Random walks - Similarity measures - Classification and regression - Clustering 	<ul style="list-style-type: none"> - Quantitative - Qualitative

TABLE I: Features of mobile recommender systems

hashtags as labels. A database of over 130.000 tweets is analyzed exploiting vocabulary, emoticons, punctuation and negations to categorize the tweets to assign hashtags to one of the emotion classes. Similarly, [16] uses Twitter as the source of analyzable text. In addition to the six basic emotions happy, sad, fear, anger, surprise and disgust, as defined by Ekman [17], love is supplemented. The paper creates a system that discovers and identifies these emotions within tweets and analyzes their distribution within text. It is found that disgust and happiness are the most commonly used emotions in tweets. In [18], a database of six million tweets is analyzed regarding their emotion. The described model selects bipolar pairs of emotions, which are joy and sadness, anticipation and surprise, anger and fear, trust and disgust. Whissell's "Dictionary of Affect in Language" [19] was created to measure emotions in any verbal material, like freely produced text, word lists or literature passages. It originally contained over 4.000 specifically emotional words. Later, the dictionary was revised to include 8.742 words, which increased its applicability for analyzing natural language.

For image-based emotion recognition, [20] shows means how to predict emotional reaction of people towards images, using computer vision techniques and Convolutional Neural Networks (CNN). The paper builds a large image data set with photos from Instagram and Flickr, using hashtags as weak labels for eight emotions, which are similar to Ekman's basic emotions. Over 23.000 photos are used to train two neural networks: the first by fine-tuning an existing pre-trained model with the verified pictures, and the second by fine-tuning another, using the weakly labeled pictures from the beginning. The first network recognizes over 58% of the pictures tagged by humans correctly, outperforming the 32% of the existing pre-trained model. For three public datasets of exemplary photos, the accuracy rate is over 80% for most of the emotions. The authors in [21] use theoretical and empirical concepts from psychology and art for image-based emotion classification. Various low-level features of the images, such as color (e.g. saturation, brightness), texture, image composition (e.g. level of detail, dynamics), and image content (e.g. faces, visible skin), are extracted and combined to represent the emotional content of an image. The eight emotions used for classification are the same as in [20]. Furthermore, [22] uses Deep Neural Networks (DNN) to analyze object, background and semantic information of images and to predict emotions based on it. The paper uses the Circumplex model of Russell. As the source of data, the authors use both pictures from Flickr tagged with emotion keywords, and pictures from [20], adding up to a total of over 10.000 images in the database.

Current developments in fields like web technologies, wireless networks, and social networking make it possible to deliver effective and accurate recommendations to the tourist, which consider the users' interests and preferences, as well as their social and environmental context. Existing mobile recommender systems for the support of tourists recommend attractions in cities, suggest tourist services, offer collaborative user-generated content, or multi-day trips [23]. Table II summarizes features of tourism recommender systems.

III. EMOTION FOR TOURISM RECOMMENDER SYSTEMS

It has become common to express feelings and opinions in social media [14]. Consequently, automatically classifying text messages can help to identify anxiety or depression, or measure the mood of both individuals and groups. The emotion states can be shown in a two dimensional circular space presented by Russell's Circumplex model.

In the following, the architecture and implementation of our recommender system for city visitors is described.

The system consists of two parts, which are communicating: A Python-based backend, which collects the pictures to be recommended and which calculates the recommendations within the recommender system, and a HTML- and JavaScript-based frontend, which displays the data in an accessible way. The recommender system fetches various data and analyzes them to generate personalized recommendations, taking additionally the user's emotion into account. The connection between frontend and backend is realized via a Python-based server.

Figure 1 shows the architecture of the system and all its components, illustrated in a class diagram. The HTML client and Python server communicate via Ajax. The capabilities of the backend are accessed via the JavaScript functions `flickr()` and `recommend()`.

A. Data collection

To collect data for the recommendations, databases of geo-tagged photos are needed. These photos are processed step by step: First, their content is analyzed, then they are categorized,

Recommendation areas	Architectural styles	User involvement	Recommendation criteria
<ul style="list-style-type: none"> - Attractions - Tourist services - User-generated content - Routes and tours - Multiple-day trips 	<ul style="list-style-type: none"> - Web-based - Standalone - Web-to-mobile 	<ul style="list-style-type: none"> - Pull-based - Reactive - Proactive 	<ul style="list-style-type: none"> - User constraints-based - Pure location-aware - Context-aware

TABLE II: Features of mobile recommender systems



Fig. 1: UML class diagram: System architecture and structure

i.e. tagged with emotions. Eventually, all information is stored in CSV files. The whole process is implemented in Python.

1) *Sources for Collecting Images*: The main data sources for the system are photos with geotags of locations for which recommendations are performed. Therefore databases of photos from online services were created - in this case, Flickr was selected. In the course of this work, three different picture databases were built up:

- 1) **flickr.csv** information and pictures from Flickr taken in and around Helsinki
- 2) **local.csv** information and pictures from Flickr taken in and around Munich
- 3) **own.csv** information and pictures taken by ourselves.

2) *Image Content Analysis*: After collecting the pictures and parsing the API response into XML for further processing, the images are analyzed. The aim is to obtain a textual description of the contents of the image for each photo, thus requiring image analysis. We considered the use of solutions provided by Google¹ and especially Clarifai² (eventually selected for our implementation), Amazon Rekognition³, Microsoft Azure Cognitive Services Computer Vision API⁴, and IBM Watson Visual Recognition⁵.

Clarifai is a provider for image content analysis. It was selected due to its many advantages: a simple to use JSON API⁶, 5000 free requests per billing cycle, and the visual recognition feature that delivers a high number of accurate, precise textual predictions (keywords) for the contents of a picture, called "concepts".

To use it in Python, the package `clarifai`⁷ was installed, on Windows along with the required Microsoft Visual C++ Build Tools⁸, including Windows 8/8.1 SDK. Hereafter, an API key was obtained, in the following referred to as

word	pleasure	activity	imagery
architecture	1.8333	1.8333	2.4
building	1.8571	1.8750	3.0
winter	2.3333	2.3750	3.0
city	2.1000	2.3000	2.6
travel	2.5714	3.0000	1.6
snow	2.3333	1.4000	3.0
outdoors	not found	not found	not found
sight	2.2857	1.7500	1.8
old	1.5000	1.2000	2.4
town	2.1667	1.6000	2.8
∅	2.109	1.9259	not used

TABLE III: Example contents of dict.txt

CLARIFAI_API_KEY. The API uses so-called models for the prediction of image content: Interesting models for this use case are the General model, which is able to recognize over 11.000 different concepts, and the Travel model.

The Python code gets the response from the Clarifai API request, using the General model, limits the amount of concepts being returned to 10 (reached for every picture), and parses it as JSON. It then selects the concepts from the JSON response. The array `answers` stores all information collected in the further process of analyzing the picture. In this way, the ten concepts which most likely represent the contents of the picture are saved, sorted by probability.

3) *Sentiment and Emotion*: Two approaches to estimate emotional content of text are *sentiment analysis* (determining the attitude regarding a certain topic) and *emotion analysis* (determining basic emotions). Another method is to use dictionaries⁹. For instance, DepecheMood¹⁰ contains 37.000 English words, each automatically assigned a probability for the emotions *afraid*, *amused*, *angry*, *annoyed*, *dont_care*, *happy*, *inspired* and *sad* adding up to 100%.

We selected "Whissell's Dictionary of Affect in Language" [19] as it is multi-dimensional and comprehensive. It contains 8.742 English words, each rated on a scale from one to three for pleasure, activity and imagery. Table III shows the contents of the dictionary for an example picture.

The Python code looks into the dictionary and calculates the average of the values for pleasure (p) and activity (a) from the "concepts" that were previously assigned to the pictures. The last parameter, imagery, is not used in our system.

Our system evaluates the values of pleasure and activity and assigns a set of two emotional values to each picture, similar to [14], and which translates to emotions according to Russell's Circumplex model of affect. In this way, all images can also be categorized according to a total of 16 emotions.

IV. IMPLEMENTATION OF EMOTION-BASED RECOMMENDATIONS

After collecting all the information about the pictures in the databases, the recommendations can be calculated. This requires building an underlying recommender system performing all necessary calculations.

⁹<http://saifmohammad.com/WebPages/lexicons.html>

¹⁰<https://github.com/marcoguerini/DepecheMood/releases>

¹<https://cloud.google.com/vision/>

²<https://clarifai.com/>

³<https://aws.amazon.com/rekognition/>

⁴<https://azure.microsoft.com/en-us/services/cognitive-services/computer-vision/>

⁵<https://www.ibm.com/watson/services/visual-recognition/>

⁶<https://clarifai.com/developer/quick-start/>

⁷<https://github.com/Clarifai/clarifai-python>

⁸<https://go.microsoft.com/fwlink/?LinkId=691126>

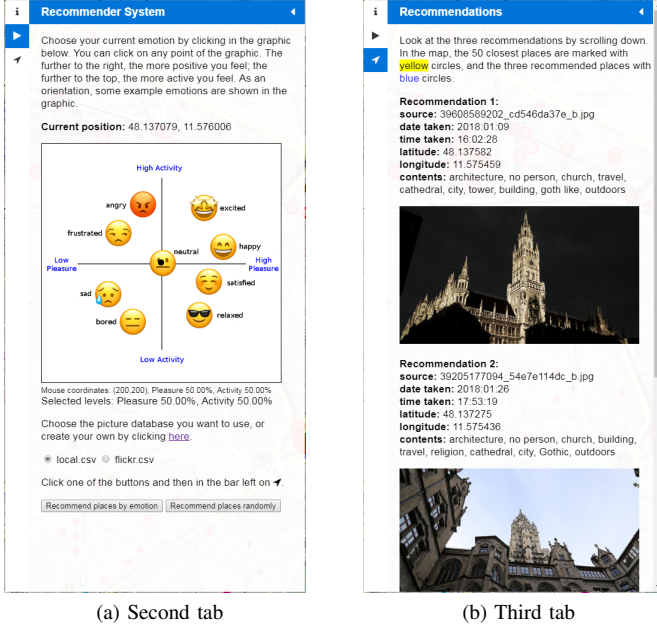


Fig. 2: Exemplary tabs from our implementation.

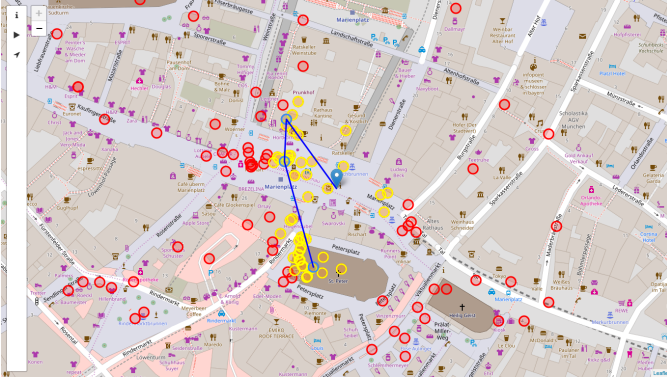


Fig. 3: Recommendations on the map

The recommender system used here has been built from scratch in Python using NumPy, SciPy and scikit-learn. Our recommender system features collaborative filtering, as it considers the user behavior, i.e. data, as well as the pictures, i.e. item data, for the recommendations. Furthermore, it is memory-based, because the system memorizes and stores the data for the generation of recommendations.

The system computes all pleasure and activity data of the 50 pictures that are closest to the selected location. It then performs a k-nearest neighbor search for this data. We used the *ball tree* algorithm¹¹ due to its simplicity and fast calculation. In this way, the three best matching locations, sorted by closeness in location and selected emotion, are found. All calculations are performed locally. The focus of the recommendations is on the influence of the factors location, defined by latitude and longitude coordinates, and emotion, defined by pleasure and activity values.

¹¹<http://scikit-learn.org/stable/modules/neighbors.html>

Figures 2a and 2b show two of the three sidebar screens in the application. Figure 3 shows the three calculated recommendations, and how they are displayed on the map. The map visualizes the position specified by the user with a marker, and all geotagged photos from the selected database (e.g. here `local.csv` or `flickr.csv`) in red.

The other sidebar screen, the system's Info tab, shows general information about the recommender system and its usage.

After choosing the emotion on the second tab, the closest 50 locations are marked in yellow and the top three in blue, i.e. showing the photos matching the selected emotion best. The location tab shows three recommendations together with all its information, a route between them is drawn on the map.

Since the recommendations are performed via the Python server, the system runs entirely without an internet connection after the creation of a tagged picture database, except for loading and caching of the map tiles from OpenStreetMap; otherwise, no network connection is needed. This communication is sketched in Figure 4 for the two use cases of creating an own picture database, and of calculating the recommendations. It lists the information that is transferred when the user sends requests to the server via the frontend functions, and the responses that are sent to the client when the server has finished the execution of the backend methods.

V. EVALUATION

To ensure that the additional emotional component helps to improve the place recommendations, a user study was conducted.

A. User Study Design

Two possible recommendation variants were compared in the study: (1) recommending nearby places considering the emotion vs. (2) recommending nearby places randomly, based on distance only. To compare the recommendations unbiased, we did not inform the participants which of the variants they were seeing. All visual elements and parameters of the system, such as used picture database and the number of shown recommendations were identical in both versions of the system.

B. Conducting the Study

The study was carried out amongst 28 male and female participants (age 16 to 65 years). Figure 5a shows the gender distribution of the participants, Figure 5b shows the age distribution. To get a broad spectrum of opinions from a wide demographic range, it was tried to consult participants with different genders, ages and also interests.

Only the database `local.csv` containing pictures from Munich was used for the study, and all texts in the explanations and the emotions graphic were furthermore translated to German. The participants were guided step by step at using the system via the instructions on the screen, especially they were asked to take a close look at the pictures while also considering the map showing the markers.

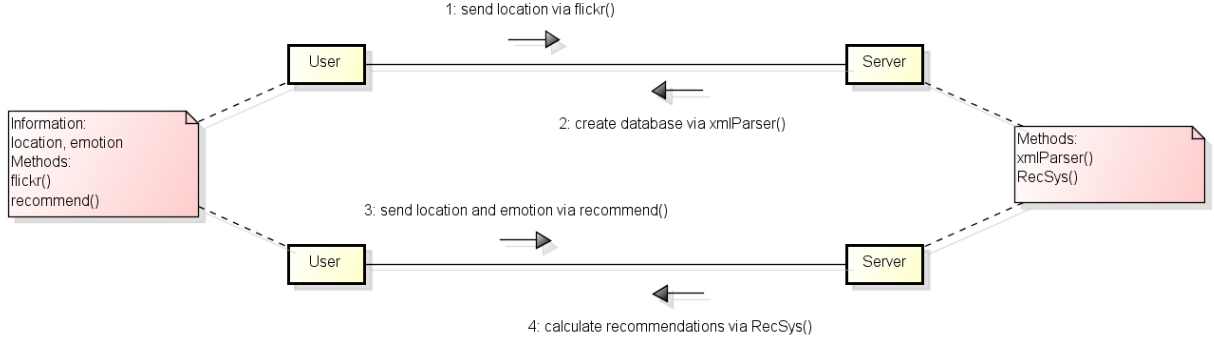


Fig. 4: UML communication diagram showing requests and responses between user and server

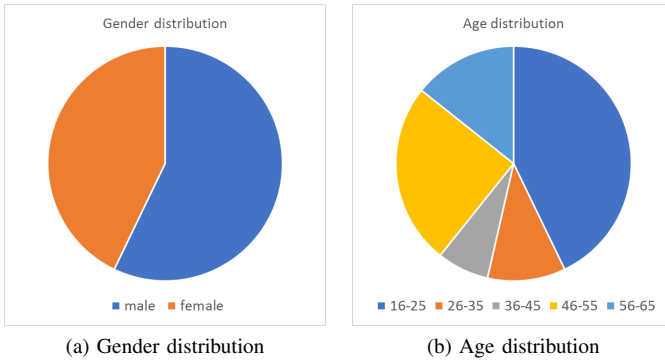


Fig. 5: Gender and age distribution of participants of the user study

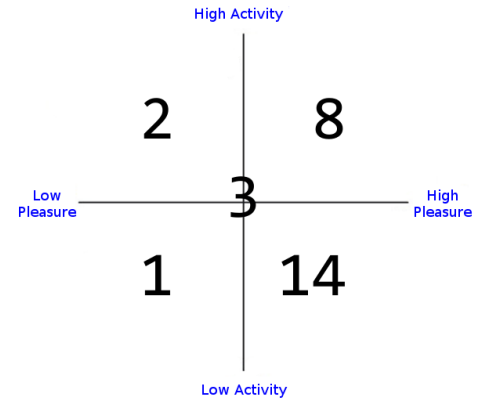


Fig. 7: Emotions selected by the participants

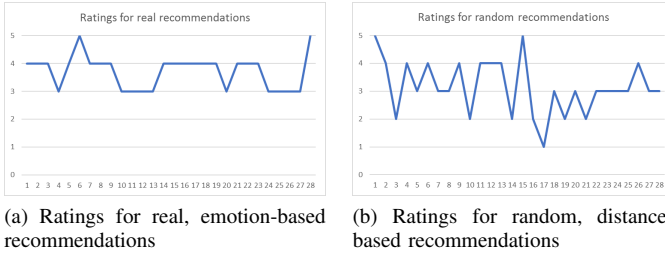


Fig. 6: Participant ratings for emotion-based vs. distance-based recommendations

The questions which were to answer on the last tab were, besides gender and age, the participants' rating for recommendations, on a scale from one to five stars. The corresponding question was formulated like this: "How suitable - in terms of your chosen emotion - did you find the three recommendations? How do you rate the quality and the accuracy of the recommendations?" This was testing the quality and usefulness of the system. The participants could also give additional comments and suggestions for improvements.

C. Results

Figure 6a and Figure 6b show the ratings of the participants for the emotion-based and for the location-based recommendations. The average of the emotion-based ratings is 3.71, while

the average of the location-based ratings is 3.14, thus showing an increase of more than half a star for the recommendations that consider the emotions.

Figure 7 reveals from which areas the participants selected emotions. The most popular choices were *satisfied* (10x) and *happy* (7x). 10 choices featured a rather active emotion (upper half), 15 choices featured a rather passive emotion (lower half). 22 choices were rather positive (right half), 3 choices were rather negative (left half).

D. Interpretation

An interesting aspect was that almost only positive emotions were chosen. Consequently the question arised about how to deal with negative emotions: Does it, for instance, make sense to recommend sad pictures, i.e. places where sad pictures were taken, when the user is sad, or would it in general also sometimes be beneficial to get recommendations for an emotion different from the one selected? Due to its construction, the recommender system tries to suggest pictures that express the same emotion as the user's. Therefore it was interesting to know which recommendations the participants wanted to receive, and how useful the participants perceived the suggestions they received.

Many participants commented that, if negative emotions are selected, recommendations should rather suggest a positive location instead. Some desired only positive recommendations, also when the selected emotion was negative, or at least a slightly more positive recommendation than the selected emotion. Another participant stated that the recommendation should rather be a calm location, or a location which makes the user feel more positive. However, others suggested to leave a choice for the user to select what kind of recommendations are given, especially when the selected emotion is negative, and that in the case of positive selected emotion, also neutral suggestions would be all right. Some participants found that recommendations should always match the chosen emotion.

Further opinions were that not only the negative-positive, but also the passive-active gap could be addressed: If a user is, for example, bored, the given recommendations could be more active. The recommended places themselves and the user's familiarity with them are also a factor for rating the usefulness of the results: If the system is used at a known location, it would be preferred to receive recommendations for new or not only touristic, but rather special places; the distance between the places might also matter in terms of physical activity. This perception might be different in an environment people are not familiar with: Then they might be willing to move and explore more and also want to see the highlights, i.e. more touristic places.

VI. SUMMARY AND CONCLUSION

An emotion-based recommender system was built, combining many different components, techniques and technologies. Its emphasis is on the analysis and interpretation of emotions of users and in pictures, but the system also covers aspects from fetching the pictures to presenting the calculated data.

This paper focused on the usefulness of emotion, defined by the two dimensions pleasure and activity, for recommendations. The conducted study was able to prove that adding emotion can help to improve the recommendations (by 19% in the proposed use case), even though there are several aspects that make the process of recommending complicated, such as the subjectivity of emotions and the multistage procedure of analyzing and tagging pictures with emotions.

As future works, we are planning to include more data sources to our databases such as recent pictures from Twitter, videos and user preferences. Additionally, involving other criteria to the recommendation process such as time of day, weather, or type of activity (indoor or outdoor) should improve the user experience. Another future direction is to involve wearable devices such as pupil glasses to track the user's eyes. The collected data could be integrated with virtual and augmented reality to determine emotion.

It would therefore be beneficial for existing recommender systems with many features to include emotion in general and both the emotion of users and emotions expressed in pictures in particular as an extra factor for analyzing places to visit and calculating recommendations from them. In systems where recommendations are already based on other information as well, the use of emotion as an additional emotion component

can significantly improve the results and especially their level of personalization [21].

REFERENCES

- [1] F. Ricci, L. Rokach, and B. Shapira, *Recommender Systems Handbook*. Springer, 2015, ch. Recommender Systems: Introduction and Challenges, pp. 1–35.
- [2] N. Lathia, *Recommender Systems Handbook*. Springer, 2015, ch. The Anatomy of Mobile Location-Based Recommender Systems, pp. 493–510.
- [3] S. Sigg, E. Peltonen, E. Lagerspetz, P. Nurmi, and S. Tarkoma, “Exploiting usage to predict instantaneous app popularity: Trend filters and retention rates,” *ACM Transactions on the WEB*, 2019.
- [4] G. Adomavicius and A. Tuzhilin, *Recommender Systems Handbook*. Springer, 2015, ch. Context-Aware Recommender Systems, pp. 191–226.
- [5] J. Beel, B. Gipp, S. Langer, and C. Breiteringer, “Research-paper recommender systems: a literature survey,” *International Journal on Digital Libraries, Volume 17, Issue 4*, pp. 305–338, 2016.
- [6] P. Melville and V. Sindhwani, *Encyclopedia of Machine Learning and Data Mining*. Springer, 2017, ch. Recommender Systems, pp. 1056–1066.
- [7] G. Linden, J. Jacobi, and E. Benson, “Collaborative recommendations using item-to-item similarity mappings,” 1998. [Online]. Available: <https://patents.google.com/patent/US6266649B1/en>
- [8] D. Jannach, M. Zanker, and A. Felfernig, *Recommender Systems*. Cambridge University Press, 2010, ch. Hybrid Recommendation Approaches, pp. 124–142.
- [9] R. Burke, *The Adaptive Web*. Springer, 2007, ch. Hybrid Web Recommender Systems, pp. 377–408.
- [10] C. A. Gomez-Urbe and N. Hunt, “The netflix recommender system: Algorithms, business value, and innovation,” *ACM Transactions on Management Information Systems, Volume 6, No. 4*, pp. 1–19, 2015.
- [11] M. Kenteris, D. Gavalas, and D. Economou, “An innovative mobile electronic tourist guide application,” *Personal and Ubiquitous Computing, Volume 13, Issue 2*, pp. 103–118, 2007.
- [12] V. Bellotti and B. Begole, “Activity-based serendipitous recommendations with the magitti mobile leisure guide,” *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1157–1166, 2007.
- [13] K. McCarthy, M. Salamó, L. Coyle, L. McGint, B. Smyth, and P. Nix, “Group recommender systems: A critiquing based approach,” *Proceedings of the 11th international conference on Intelligent user interfaces*, pp. 267–269, 2006.
- [14] M. Hasan, E. A. Rundensteiner, and E. Agu, “Emotex: Detecting emotions in twitter messages,” *Academy of Science and Engineering*, pp. 1–10, 2014.
- [15] J. Posner, J. A. Russell, and B. S. Peterson, “The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology,” *Development and psychopathology*, pp. 715–734, 2005.
- [16] K. Roberts, M. A. Roach, J. Johnson, J. Guthrie, and S. M. Harabagiu, “Empatweet: Annotating and detecting emotions on twitter,” *Proceedings of the Eight International Conference on Language Resources and Evaluation*, pp. 3806–3813, 2012.
- [17] P. Ekman, *Handbook of Cognition and Emotion*. John Wiley & Sons Ltd., 1999, ch. Basic Emotions, pp. 46–60.
- [18] J. Suttles and N. Ide, “Distant supervision for emotion classification with discrete binary values,” *CICLing 2013: Computational Linguistics and Intelligent Text Processing*, pp. 121–136, 2013.
- [19] C. Whissell, “Using the revised dictionary of affect in language to quantify the emotional undertones of samples of natural language,” *Psychological Reports, Volume 105, Issue 2*, pp. 509–521, 2009.
- [20] Q. You, J. Luo, H. Jin, and J. Yang, “Building a large scale dataset for image emotion recognition: The fine print and the benchmark,” *Thirtieth AAAI Conference on Artificial Intelligence*, pp. 308–314, 2016.
- [21] J. Machajdik and A. Hanbury, “Affective image classification using features inspired by psychology and art theory,” *Proceedings of the 18th ACM international conference on Multimedia*, pp. 83–92, 2010.
- [22] H.-R. Kim, Y.-S. Kim, S. J. Kim, and I.-K. Lee, “Building emotional machines: Recognizing image emotions through deep neural networks,” *Computing Research Repository*, pp. 1–11, 2017.
- [23] D. Gavalas, C. Konstantopoulos, K. Mastakas, and G. Pantziou, “Mobile recommender systems in tourism,” *Journal of Network and Computer Applications, Volume 39*, pp. 319–333, 2014.