RESEARCH ARTICLE OPEN ACCESS

Comprehensive study on Opinion Mining for User Reviews for Mobile App Comparisons

Kavitha Gopu[1], Alampally Sreedevi[2]

[1,2]Assistent Professor, Dept of CSE, Sri Indu College of Engineering and Technology, Hyderabad, Telangana, India

Abstract.

Existing approaches for opinion mining for the most part center around re-sees from Amazon, space particular audit sites or online networking. Little efforts have been spent on fine-grained analysis of opinions in audit texts from portable advanced cell applications. In this paper, we propose a perspective and subjective expression extraction demonstrate for German reviews from the Google Play store. We break down the effect of deferent highlights, including space particular word embeddings. Our best model design demonstrates a performance of 0.63 F1 for viewpoints and 0.62 F1 for subjective expressions. Further, we perform cross-area analyzes: A model prepared on Amazon reviews and tried on app reviews accomplishes bring down performance (drop by 27 rate focuses for angles and 15 rate focuses for subjective expressions). The outcomes demonstrate that there are solid deference's in the way closely-held convictions on item angles are communicated in the specific spaces.

Keywords: Sentiment Analysis, Reviews, German, App Reviews, Opinion Mining.

I. INTRODUCTION

The analysis of sentiment articulations and opinions in text picked up a ton of consideration inside the most recent decade [23]. Contemplated kinds of texts incorporate item reviews, Twitter messages or blog entries [36]. Opinion mining (sometimes known as sentiment analysis or emotion AI) refers to the use of natural language processing, text analysis, computational linguistics, and biometrics to systematically identify, extract, quantify, and study affective states and subjective information. Sentiment analysis is widely applied to voice of the customer materials such as reviews and survey responses, online and social media, and healthcare materials for applications that range from marketing to customer service to clinical medicine.
A basic task in sentiment analysis is classifying the polarity of a given text at the document, sentence, or feature/aspect level—whether the expressed opinion in a document, a sentence or an entity feature/aspect is positive, negative, or neutral. Advanced, "beyond polarity" sentiment classification looks, for instance, at emotional states such as "angry", "sad", and "happy".

Precursors to sentimental analysis include the General Inquirer,[1] which provided hints toward quantifying patterns in text and, separately, psychological research that examined a person's psychological state based on analysis of their verbal behavior.[2]
Subsequently, the method described in a patent by Volcani and Fogel,[3] looked specifically at sentiment and identified individual words and phrases in text with respect to different emotional scales. A current system based on their work, called EffectCheck, presents synonyms that can be used to increase or decrease the level of evoked emotion in each scale.

Many other subsequent efforts were less sophisticated, using a mere polar view of sentiment, from positive to negative, such as work by Turney,[4] and Pang[5] who applied different methods for detecting the polarity of product reviews and movie reviews respectively. This work is at the document level. One can also classify a document's polarity on a multi-way scale, which was attempted by Pang[6] and Snyder[7] among others: Pang and Lee[6] expanded the basic task of classifying a movie review as either positive or negative to predict star ratings on either a 3- or a 4-star scale, while Snyder[7] performed an in-depth analysis of restaurant reviews, predicting ratings for various aspects of the given restaurant, such as the food and atmosphere (on a five-star scale).

First steps to bringing together various approaches—learning, lexical, knowledge-based, etc.—were taken in the 2004 AAAI Spring Symposium where linguists, computer scientists, and other interested researchers first aligned interests and proposed shared tasks and benchmark data sets for the systematic computational research on affect, appeal, subjectivity, and sentiment in text.[8] Even though in most statistical classification methods, the neutral class is ignored under the assumption that neutral texts lie near the boundary of the binary classifier, several researchers suggest that, as in every polarity problem, three categories must be identified. Moreover, it can be proven that specific classifiers such as the Max Entropy[9] and the SVMs[10] can benefit from the introduction of a neutral class and improve the overall accuracy of the classification. There are in principle two ways for operating with a neutral class. Either, the algorithm proceeds by first identifying the neutral language, filtering it out and then assessing the rest in terms of positive and negative sentiments, or it builds a three-way classification in one step.[11] This second approach often involves estimating a probability distribution over all categories (e.g. naive Bayes classifiers as implemented by the NLTK). Whether and how to use a neutral class depends on the nature of the data: if the data is clearly clustered into neutral, negative and positive language, it makes sense to filter the neutral language out and focus on the polarity between positive and negative sentiments. If, in contrast, the data are mostly neutral with small deviations towards positive and negative affect, this strategy would make it harder to clearly distinguish between the two poles.

A different method for determining sentiment is the use of a scaling system whereby words commonly associated with having a negative, neutral, or positive sentiment with them are given an associated number on a −10 to +10 scale (most negative up to most positive) or simply from 0 to a positive upper limit such as +4. This makes it possible to adjust the sentiment of a given term relative to its environment (usually on the level of the sentence). When a piece of unstructured text is analyzed using natural language processing, each concept in the specified environment is given a score based on the way sentiment words relate to the concept and its associated score.[12] This allows movement to a more sophisticated understanding of sentiment, because it is now possible to adjust the sentiment value of a concept relative to modifications that may surround it. Words, for example, that intensify, relax or negate the sentiment expressed by the concept can affect its score. Alternatively, texts can be given a positive and negative sentiment strength score if the goal is to determine the sentiment in a text rather than the overall polarity and strength of the text.[13]

Subjectivity/objectivity identification

This task is commonly defined as classifying a given text (usually a sentence) into one of two classes: objective or subjective.[14] This problem can sometimes be more difficult than polarity classification.[15] The subjectivity of words and phrases may depend on their context and an objective document may contain subjective sentences (e.g., a news article quoting people's opinions). Moreover, as mentioned by Su,[16] results are largely dependent on the definition of subjectivity used when annotating texts. However, Pang[17] showed that removing objective sentences from a document before classifying its polarity helped improve performance.

Feature/aspect-based

It refers to determining the opinions or sentiments expressed on different features or aspects of entities, e.g., of a cell phone, a digital camera, or a bank.[18] A feature or aspect is an attribute or component of an entity, e.g., the screen of a cell phone, the service for a restaurant, or the picture quality of a camera. The advantage of feature-based sentiment analysis is the possibility to capture nuances about objects of interest. Different features can generate different sentiment responses, for example a hotel can have a convenient location, but mediocre food.[19] This problem involves several sub-problems, e.g., identifying relevant entities, extracting their features/aspects, and determining whether an opinion expressed on each feature/aspect is positive, negative or neutral.[20] The automatic identification of features can be performed with syntactic methods, with topic modeling,[21][22] or with deep learning.[23] More detailed discussions about this level of sentiment analysis can be found in Liu's work.[24]

Methods and features

Existing approaches to sentiment analysis can be grouped into three main categories: knowledge-based techniques, statistical methods, and hybrid approaches.[25] Knowledge-based techniques classify text by affect categories based on the presence of unambiguous affect words such as happy, sad, afraid, and bored.[26] Some knowledge bases not only list obvious affect words, but also assign arbitrary words a probable "affinity" to particular emotions.[27] Statistical methods leverage on elements from machine learning such as latent semantic analysis, support vector machines, "bag of words" and Semantic Orientation — Pointwise Mutual Information (See Peter Turney's[4] work in this area). More sophisticated methods try to detect the holder of a sentiment (i.e., the person who maintains that affective state) and the target (i.e., the entity about which the affect is felt).[28] To mine the opinion in context and get the feature about which the speaker has opined, the grammatical relationships of words are used. Grammatical dependency relations are obtained by deep parsing of the text.[29] Hybrid approaches leverage on both machine learning and elements from knowledge representation such as ontologies and semantic networks in order to detect semantics that are expressed in a subtle manner, e.g., through the analysis of concepts that do not explicitly convey relevant information, but which are implicitly linked to other concepts that do so.[30]

Open source software tools deploy machine learning, statistics, and natural language processing techniques to automate sentiment analysis on large collections of texts, including web pages, online news, internet discussion groups, online reviews, web blogs, and social media.[31] Knowledge-based systems, on the other hand, make use of publicly available resources, to extract the semantic and affective information associated with natural language concepts. Sentiment analysis can also be performed on visual content, i.e., images and videos. One of the

first approach in this direction is SentiBank[32] utilizing an adjective noun pair representation of visual content. In addition, the vast majority of sentiment classification approaches rely on the bag-of-words model, which disregards context, grammar and even word order. Approaches that analyses the sentiment based on how words compose the meaning of longer phrases have shown better result,[33] but they incur an additional annotation overhead.

A human analysis component is required in sentiment analysis, as automated systems are not able to analyze historical tendencies of the individual commenter, or the platform and are often classified incorrectly in their expressed sentiment. Automation impacts approximately 23% of comments that are correctly classified by humans.[34] However, humans often disagree, and it is argued that the inter-human agreement provides an upper bound that automated sentiment classifiers can eventually reach.[35]

Sometimes, the structure of sentiments and topics is fairly complex. Also, the problem of sentiment analysis is non-monotonic in respect to sentence extension and stop-word substitution (compare THEY would not let my dog stay in this hotel vs I would not let my dog stay in this hotel). To address this issue a number of rule-based and reasoning-based approaches have been applied to sentiment analysis, including defeasible logic programming.[36] Also, there is a number of tree traversal rules applied to syntactic parse tree to extract the topicality of sentiment in open domain setting.[37][38]

The accuracy of a sentiment analysis system is, in principle, how well it agrees with human judgments. This is usually measured by variant measures based on precision and recall over the two target categories of negative and positive texts. However, according to research human raters typically only agree about 80%[39] of the time (see Inter-rater reliability). Thus, a program which achieves 70% accuracy in classifying sentiment is doing nearly as well as humans, even though such accuracy may not sound impressive. If a program were "right" 100% of the time, humans would still disagree with it about 20% of the time, since they disagree that much about any answer.[40] On the other hand, computer systems will make very different errors than human assessors, and thus the figures are not entirely comparable. For instance, a computer system will have trouble with negations, exaggerations, jokes, or sarcasm, which typically are easy to handle for a human reader: some errors a computer system makes will seem overly naive to a human. In general, the utility for practical commercial tasks of sentiment analysis as it is defined in academic research has been called into question, mostly since the simple one-dimensional model of sentiment from negative to positive yields rather little actionable information for a client worrying about the effect of public discourse on e.g. brand or corporate reputation.[41][42][43]

In recent years, to better fit market needs, evaluation of sentiment analysis has moved to more task-based measures, formulated together with representatives from PR agencies and market research professionals. The focus in e.g. the RepLab evaluation data set is less on the content of the text under consideration and more on the effect of the text in question on brand reputation.[44][45][46]

One step towards this aim is accomplished in research. Several research teams in universities around the world currently focus on understanding the dynamics of sentiment in e-communities

through sentiment analysis.[50] The CyberEmotions project, for instance, recently identified the role of negative emotions in driving social networks discussions.[51]
The problem is that most sentiment analysis algorithms use simple terms to express sentiment about a product or service. However, cultural factors, linguistic nuances and differing contexts make it extremely difficult to turn a string of written text into a simple pro or con sentiment.[47] The fact that humans often disagree on the sentiment of text illustrates how big a task it is for computers to get this right. The shorter the string of text, the harder it becomes.
Even though short text strings might be a problem, sentiment analysis within microblogging has shown that Twitter can be seen as a valid online indicator of political sentiment. Tweets' political sentiment demonstrates close correspondence to parties' and politicians' political positions, indicating that the content of Twitter messages plausibly reflects the offline political landscape.[52]
Generally speaking, sentiment analysis aims to determine the attitude of a speaker, writer, or other subject with respect to some topic or the overall contextual polarity or emotional reaction to a document, interaction, or event. The attitude may be a judgment or evaluation (see appraisal theory), affective state (that is to say, the emotional state of the author or speaker), or the intended emotional communication (that is to say, the emotional effect intended by the author or interlocutor).The analysis of portable applications (otherwise called apps) and their client reviews in app stores, for example, the Apple App Store3, Google Play Store 4, BlackBerry World5 or Windows Store6, has just increased exceptionally restricted consideration up until now. Be that as it may, app reviews over intriguing attributes which the amusement is extremely incredible and simply fun. Space lack was at last dispensed with.

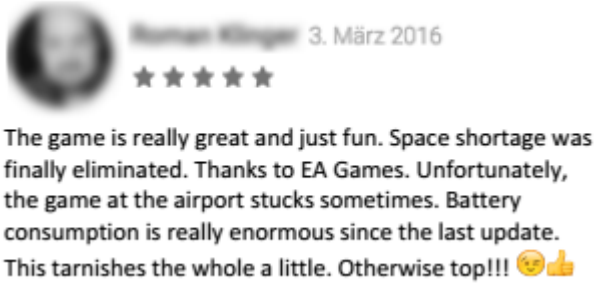

Fig. 1. Case of a client audit for a portable application. The audit contains helpful information and criticism for app designers, e.g. that amusement when all is said in done is simply fun. Be that as it may, the battery utilization, which is a part of the application, is extremely colossal.
merit unique examination: On the one side, they share properties with Tweets and other web-based social networking texts, e.g., similarly short and informal language [6]. On the opposite side they are like item reviews from different spaces or platforms, e.g., reviews about family unit appliances, shopper hardware or books on Amazon, as they ordinarily portray the client's opinion about particular perspectives. In the illustration client survey in Figure 1, the assignment is identify for example the viewpoints "amusement" with the assessment "extraordinary" and "fun". It additionally features that the angle "battery utilization" is assessed contrarily, as "huge" demonstrates.

The analysis of app reviews is additionally intriguing from a business perspective. The reviews form a rich asset of information, since they hold the client's opinions about the application. In addition the reviews regularly contain dissensions about issues and blunders of the app and notices of desired highlights. Joining this input into the development procedure can affect the accomplishment of the application [22]. Be that as it may, the mind-boggling measure of client reviews challenges app designers. An application can get hundreds or thousands of reviews every day, which make a manual examination and analysis extremely tedious and unfeasible. A computerized analysis of the reviews would be beneficial for app clients also since this would empower them to break down the favorable circumstances and hindrances of one or different applications all the more effortlessly. For instance, they could contrast two wellness trackers agreeing with particular perspectives like the exactness of the followed course or the representation of the preparation advance.

II. RELATED WORK

Late year's work in opinion mining delivered countless [16, 31, 34]. The dominant part of approaches centers on the investigation of item reviews [5], Twitter messages [32] and blog entries [18]. Just not very many approaches examine versatile applications and client reviews in app stores.

An early approach is finished by Harman et al. [14]. They break down the value, client rating and the rank of app downloads of apps in the BlackBerry App Store. Assessment comes about demonstrate a solid connection between's the client rating and the rank of app downloads. Interestingly, Jacob and Harrison [17] consequently identify highlight asks for in app reviews. They utilize a corpus of 3,279 reviews from deferent applications in the BlackBerry App Store and physically make an arrangement of 237 etymological examples (e.g. "Including <request> would be <POSITIVE-ADJECTIVE>"). Fu et al. [8] center on negative reviews and the recognizable proof of reasons which prompt poor evaluations. For this reason the use Latent Dirichlet distribution (LDA) [2] to separate themes from negative reviews and think about the principle purposes behind poor reviews of applications from deferent classifications. Chen et al. [3] utilize deferent subject models in a semi-directed classifier to recognize informative and non-informative reviews. A deferent approach to the analysis of app reviews is followed in [13]. They remove application highlights in light of thing, verb and modifier collocations. They utilize SentiStrength [33], a sentence based opinion mining strategy, to decide the client opinions about the removed highlights. Besides, the perceived highlights will be joined to more broad themes utilizing LDA.

Different approaches around there incorporate extortion recognition [9], grouping of app reviews to distinguish bug reports and highlight demands [24], and coarse-grained sentiment analysis [11, 13]. Additionally explore examines theme and watchword distinguishing proof strategies [10, 37] and audit affect analysis [28]. Table 1 outlines work around there. Most of the approaches depends on physically made English corpora which aren't accessible to the exploration group. For different languages just a couple of data sets exist, e.g. for German Maalej and Nabil [24] make their audit data accessible yet just give archive level explanations. Sänger et al. [30] as of late distributed a corpus of German app reviews commented on with perspectives, subjective expressions and extremity. Be that as it may, they just give results to a standard model. In this

paper, we perform additionally investigates this asset. To the best of our insight, this is the main corpus accessible which contains such fine-grained explanations from the space.

Table 1. Outline of existing work on app store audit mining and analysis. For each approach the general target, the quantity of applications and reviews utilized and also the app store (Apple App Store (A), Google Play Store (G) or BlackBerry World (B)) they start from are given. All approaches utilize English language reviews.

Authors	Objective	Store	#Apps	#Reviews
Harman et al. [14]	Identification of correlations between price, rating and download rank	B	32,108	—
Iacob, Harrison [17]	Pattern-based detection of feature requests	B	270	137,000
Galvis et al. [10]	Identification of topics and keywords using LDA	G	3	327
Fu et al. [8]	Analysis of negative reviews and their reasons	G	171,493	13,286,706
Pagano, Maalej [28]	Analysis of the impact of app user reviews	A	1,100	1,100,000
Guzman, Maalej [13]	Extraction of application features / characteristics	A,G	7	32,210
Chen et al. [3]	Identification of informative and non-informative reviews	G	4	241,656
Vu et al. [37]	Identification of topic and keywords using topic modeling techniques	G	95	2,106,605
Maalej, Nabil [24]	Classification of app review into bugs, feature requests and simple appraisals	A,G	1,140	1,303,182

The utilization of word implanting based highlights has indicated extensive effect on the performance on an assortment of NLP undertakings, for example lumping [4] or named element acknowledgment [35]. Existing approaches either utilize word embeddings directly [32] or get discrete highlights [7] from them. For instance, Guo et al.[12] perform k-implies bunching to get paired component vectors. In constrast, Turian et al.[35] use the interims in which the estimations of the vector parts mislead create discrete highlights. We don't know about any past work that has explored word embeddings and highlights in light of word embeddings in the context of app reviews, particularly in a cross-area setting.

III. BASELINE MODEL

We demonstrate the acknowledgment of subjective expressions and application viewpoints as arrangement naming undertaking, i.e., each expression of an audit text is appointed a class mark from the set L = {O, B-Subj , I-Subj, B-Asp, I-Asp} . We utilize a straight chain contingent arbitrary field [21] and the MALLET toolbox [25] to actualize the model. To take in the parameters of the model, the most extreme probability technique is applied. Derivation is performed utilizing the Viterbi calculation [29].

Our gauge show takes lexical, morphological and linguistic highlights from each word (e.g. the token itself, grammatical form tag, capitalization, 3-character pre-and suffix) into record to catch the attributes of application perspectives what's more, evaluative expressions. The highlights are inspired by [19]. We additionally incorporate nullification word location and also smiley and emoji acknowledgment. For refutation word location we physically incorporated a rundown of

German terms, which infer the nonappearance of specific issues or complete a nullification of a genuine circumstance, and match them with the audit text. We utilize a physically collected rundown of smileys and emojis for acknowledgment in view of the rundowns GreenSmilies (http://www.greensmilies.com/smilie-lexikon/) and Smiley dictionary (http://home. allgaeu.org/cwalter/smileys.html).

Notwithstanding the textual highlights of the right now thought to be token, the qualities of the context words are considered. For this reason, all highlights of the words with a separation of two positions when the present token are added to the element vectors. Each component will be set apart with the separation to the at present thought about token. All highlights of our model are spoken to as boolean qualities.

IV. WORD EMBEDDING MODEL

We create highlights from word embeddings to improve our model. We select deduction of discrete highlights to have the capacity to pick up bits of knowledge about the effect and effectiveness of such highlights. The highlights are inspired by past work [15, 38].

4.1 Synonym Expansion

The first component classification that depends on embeddings speaks to the utilization of equivalent words and semantically related words. All the more formally, for a word w up to 10 different words w0 from the vocabulary V with a cosine-comparability more noteworthy than a limit t (as indicated by their embeddings v(w) and v(w0)) are included as equivalent word highlights.

$syn(w) = \{w' | w' 2 € V / \{w\} \wedge sim(v_w, v_{w'}) \geq t\}$.

We set t = 0. 8 empirically in light of a hold out set. Comparative words are probably going to speak to the same or comparable ideas and ought to in this way get a similar name. For example, if the term app is perceived as a marker of a perspective, it is likely that terms, for example, application, program or apparatus ought to likewise be considered as viewpoints since they depict comparative ideas.

4.2 Clustering

Equivalent word includes just model connections certainly between gatherings of comparable words. To influence this express, we to perform progressive bunching of the word embeddings and include the record of the most comparable group focus to the present word and also the full way and all way prefixes in the bunch order. Utilizing the way prefixes empowers the model to consider changing levels of granularity and consequently test deferent reflection layers and bunch sizes.

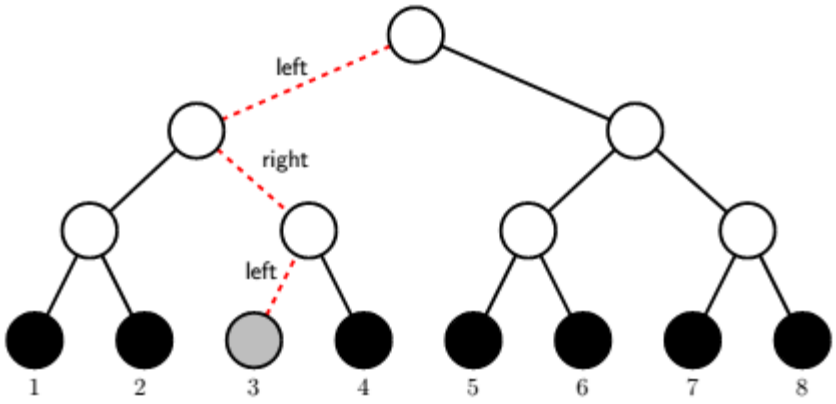

Fig. 2. Case for grouping based component extraction. The dim leave relates to the nearest group to a thought about word. Highlights for the way from the root are hence left, left-right, left-right-left, and bunch id=3.

than singular words and in this manner accomplish a higher review. For instance, subjective articulations like energizing, entrancing or superb, possibly used to depict the UI of an app, ought to be dealt with proportionately and in this manner ought to get a similar mark. To construct the group tree we apply a recursive best down approach. Toward the starting all word embeddings form one bunch, which is separated into two sub-groups utilizing the k-implies bunching calculation and cosine similitude. Each sub - bunch is then recursively partitioned into two sub-groups until the point when a profundity of 10 layers is come to.

CONCLUSION

In this paper, we introduced a fine-grained sentiment analysis show for German for app reviews from the Google play store. The model depends on contingent irregular fields and takes lexical, morphological and linguistic highlights and in addition space particular qualities into record to separate subjective articulation and application perspectives from the client audit texts. To demonstrate connections amongst words and gatherings of words we advance our approach with discrete highlights in view of word embeddings. The assessment of the model shows aggressive figures as indicated by aftereffects of comparative extraction approaches created on other item spaces. Besides, we show that the performance of our model can be enhanced by 2 % with highlights in light of space particular word embeddings. A cross-area analyze uncovered that there are clear differences in the way sincere beliefs and item angles are communicated in app reviews rather than Amazon item reviews. This demonstrates the need of space particular models for fine-grained app audit mining which take the linguisitic quirks of the short and informal survey texts into account. Our approach speaks to a first step towards more definite analysis of reviews which will bolster application engineers and in addition app clients to break down and look at the focal points and downsides of one or various apps.

References

1. Al-Rfou, R., Perozzi, B., Skiena, S.: Polyglot: Distributed word representations for multilingual nlp. In: Proceedings of the Seventeenth Conference on Computational Natural Language Learning. pp. 183–192. Association for Computational Linguistics, Sofia, Bulgaria (August 2013)

2. Blei, D., Ng, A.Y., Jordan, M.: Latent dirichlet allocation. Journal of Machine Learning Research 3, 993–1022 (2003)

3. Chen, N., Lin, J., Hoi, S.C., Xiao, X., Zhang, B.: Ar-miner: mining informative reviews for developers from mobile app marketplace. In: Proceedings of the 2014 International Conference on Software Engineering. pp. 767–778. Hyderabad, India (2014)
4. Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P.: Natural language processing (almost) from scratch. Journal of Machine Learning Research 12(Aug), 2493–2537 (2011)
5. Cui, H., Mittal, V., Datar, M.: Comparative experiments on sentiment classification for online product reviews. In: Proceedings of the Eighteenth Conference on Inno-vative Applications of Artificial Intelligence. vol. 6, pp. 1265–1270. Boston, MA, USA (2006)
6. Derczynski, L., Maynard, D., Rizzo, G., van Erp, M., Gorrell, G., Troncy, R., Petrak, J., Bontcheva, K.: Analysis of named entity recognition and linking for tweets. Information Processing & Management 51(2), 32–49 (2015)
7. Faruqui, M., Tsvetkov, Y., Yogatama, D., Dyer, C., Smith, N.: Sparse overcomplete word vector representations. In: Proceedings of Association for Computational Linguistics. Beijing, China (2015)
8. Fu, B., Lin, J., Li, L., Faloutsos, C., Hong, J., Sadeh, N.: Why people hate your app:making sense of user feedback in a mobile app store. In: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 1276–1284. Association for Computing Machinery, Chicago, USA (2013)
9. Gade, T., Pardeshi, N.: A survey on ranking fraud detection using opinion mining for mobile apps. International Journal of Advanced Research in Computer and Communication Engineering 4(12) (2015)

10. Galvis Carreno, L., Winbladh, K.: Analysis of user comments: an approach for soft-ware requirements evolution. In: Proceedings of the 2013 International Conference on Software Engineering. pp. 582–591. San Francisco, CA, USA (2013)
11. Gu, X., Kim, S.: What parts of your apps are loved by users? In: Proceedings of the 30th IEEE/ACM International Conference on Automated Software Engineering. pp. 760–770. IEEE, Lincoln, USA (2015)
12. Guo, J., Che, W., Wang, H., Liu, T.: Revisiting embedding features for simple semi-supervised learning. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. pp. 110–120. Doha, Qatar (2014)
13. Guzman, E., Maalej, W.: How do users like this feature? a fine grained sentiment analysis of app reviews. In: Proceedings of the 22nd International Requirements Engineering Conference. pp. 153–162. Karlskrona, Sweden (2014)
14. Harman, M., Jia, Y., Zhang, Y.: App store mining and analysis: Msr for app stores. In: Proceedings of the 9th IEEE Working Conference on Mining Software Repositories. pp. 108–111. Zurich, Switzerland (2012)
15. Hintz, G., Biemann, C.: Delexicalized supervised german lexical substitution. In:Proceedings of GermEval 2015: LexSub. pp. 11–16 (2015)
16. Hutto, C.J., Gilbert, E.: Vader: A parsimonious rule-based model for sentiment analysis of social media text. In: Eighth International AAAI Conference on Weblogs and Social Media. Ann Arbor, MI, USA (2014)
17. Iacob, C., Harrison, R.: Retrieving and analyzing mobile apps feature requests from online reviews. In: Proceedings of the 10th IEEE Working Conference on Mining Software Repositories. pp. 41–44. San Francisco, CA, USA (2013)

18. Jakob, N., Gurevych, I.: Extracting opinion targets in a single-and cross-domain setting with conditional random fields. In: Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing. pp. 1035–1045. Association for Computational Linguistics, Stroudsburg, PA, USA (2010)
19. Klinger, R., Cimiano, P.: Joint and pipeline probabilistic models for fine-grained sentiment analysis: extracting aspects, subjective phrases and their relations. In: IEEE 13th International Conference on Data Mining Workshops. pp. 937–944. Dallas, TX, USA (2013)
20. Klinger, R., Cimiano, P.: The usage review corpus for fine grained multi lingual opin-ion analysis. In: Proceedings of the Nineth International Conference on Language Resources and Evaluation. pp. 2211–2218. Reykjavik, Iceland (2014)
21. Lafferty, J., McCallum, A., Pereira, F.: Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: Proceedings of the 18th International Conference on Machine Learning. Morgan Kaufmann, Williamstown, MA, USA (2001)
22. Liang, T.P., Li, X., Yang, C.T., Wang, M.: What in consumer reviews affects the sales of mobile apps: A multifacet sentiment analysis approach. International Journal of Electronic Commerce 20(2), 236–260 (2015)
23. Liu, B.: Sentiment analysis: Mining opinions, sentiments, and emotions (2015)
24. Maalej, W., Nabil, H.: Bug report, feature request, or simply praise? on auto-matically classifying app reviews. In: Proceedings of the IEEE 23rd International Requirements Engineering Conference. pp. 116–125. IEEE, Karlskrona, Sweden (2015)
25. McCallum, A.: Mallet: A machine learning for language toolkit (2002), http: //mallet.cs.umass.edu (Online; Last access 08.02.2017)
26. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word represen-tations in vector space. In: Proceedings of Workshop at International Conference on Learning Representations. Scottsdale, AZ, USA (2013)
27. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed represen-tations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems. pp. 3111–3119. South Lake Tahoe, NV, USA (2013)
28. Pagano, D., Maalej, W.: User feedback in the appstore: An empirical study. In: Pro-ceedings of the 2013 21st IEEE International Requirements Engineering Conference. pp. 125–134. IEEE, Rio de Janeiro, Brazil (2013)
29. Rabiner, L.: A tutorial on hidden markov models and selected applications in speech recognition. Proceedings of the IEEE 77(2), 257–286 (1989)
30. Sänger, M., Leser, U., Kemmerer, S., Adolphs, P., Klinger, R.: SCARE - the sentiment corpus of app reviews with fine-grained annotations in german. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016). Portorož, Slovenia (2016)
31. Täckström, O., McDonald, R.: Discovering fine-grained sentiment with latent variable structured prediction models. In: Advances in Information Retrieval, pp. 368–374. Springer (2011)
32. Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T., Qin, B.: Learning sentiment-specific word embedding for twitter sentiment classification. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. pp. 1555–1565. Baltimore, MD, USA (2014)

33. Thelwall, M., Buckley, K., Paltoglou, G., Cai, D., Kappas, A.: Sentiment strength detection in short informal text. Journal of the American Society for Information Science and Technology 61(12), 2544–2558 (2010)
34. Titov, I., McDonald, R.: A joint model of text and aspect ratings for sentiment summarization. In: Proceedings of 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Columbus, OH, USA (2008)
35. Turian, J., Ratinov, L., Bengio, Y.: Word representations: a simple and general method for semi-supervised learning. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. pp. 384–394. Uppsala, Sweden (2010)
36. Vinodhini, G., Chandrasekaran, R.: Sentiment analysis and opinion mining: a survey. International Journal 2(6) (2012)
37. Vu, P.M., Nguyen, T.T., Pham, H.V., Nguyen, T.T.: Mining user opinions in mobile app reviews: a keyword-based approach. In: Proceedings of the 30th IEEE/ACM International Conference on Automated Software Engineering. pp. 749–759. IEEE, Lincoln, NE, USA (2015)
38. Yu, M., Zhao, T., Dong, D., Tian, H., Yu, D.: Compound embedding features for semi-supervised learning. In: Proceedings of Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics. pp. 563–568. Atlanta, GA, USA (2013)