SCARE – The Sentiment Corpus of App Reviews with Fine-grained Annotations in German

Mario Sänger[*], **Ulf Leser**[*], **Steffen Kemmerer**[§], **Peter Adolphs**[§], **Roman Klinger**[‡]
[*] Department of Computer Science, Humboldt Universität zu Berlin, Rudower Chaussee 25, 12489 Berlin, Germany
[§] Neofonie GmbH, Robert-Koch-Platz 4, 10115 Berlin, Germany
[‡] Institut für Maschinelle Sprachverarbeitung, Universität Stuttgart, Pfaffenwaldring 5b, 70569 Stuttgart, Germany
{saengerm,leser}@informatik.hu-berlin.de,
{steffen.kemmerer,peter.adolphs}@neofonie.de,
roman.klinger@ims.uni-stuttgart.de

Abstract

The automatic analysis of texts containing opinions of users about, *e.g.*, products or political views has gained attention within the last decades. However, previous work on the task of analyzing user reviews about mobile applications in app stores is limited. Publicly available corpora do not exist, such that a comparison of different methods and models is difficult. We fill this gap by contributing the Sentiment Corpus of App Reviews (SCARE), which contains fine-grained annotations of application aspects, subjective (evaluative) phrases and relations between both. This corpus consists of 1,760 annotated application reviews from the Google Play Store with 2,487 aspects and 3,959 subjective phrases. We describe the process and methodology how the corpus was created. The Fleiss-κ between four annotators reveals an agreement of 0.72. We provide a strong baseline with a linear-chain conditional random field and word-embedding features with a performance of 0.62 for aspect detection and 0.63 for the extraction of subjective phrases. The corpus is available to the research community to support the development of sentiment analysis methods on mobile application reviews.

Keywords: sentiment analysis, app reviews, German corpus, conditional random field, word embedding

1. Introduction

Mobile devices, such as smartphones and tablets, are widespread in our society. Applications for these devices (also known as *apps*) become increasingly popular and gain a lot of attention in our daily lives. Applications are typically downloaded via app stores, *i.e.*, application distribution platforms, such as the Apple App Store[1], Google Play Store[2], BlackBerry World[3] or the Windows Store[4]. These platforms offer their users the possibility to assess applications with a 5-star rating and a textual review. An example for such a review is depicted in Figure 1.

These reviews form a rich resource of information for app developers, since they hold the user's opinions about the application itself and important aspects, like design and usability. Moreover, the reviews often contain complaints about problems and errors of the application as well as feature requests. Incorporating this feedback in the development process may have influence on the success of the app. However, one challenge for the developers is to deal with the overwhelming amount of reviews. Applications can have hundreds of thousands or even millions of reviews. A manual inspection and analysis of all these reviews is very time consuming and impractical. The app stores themselves offer only basic analytical capabilities.

The analysis of opinions in reviews has been widely investigated within the last decade (Pang and Lee, 2008) and is typically referred to as *sentiment analysis* or *opinion mining* (Liu, 2012). However, previous work in the area of analyzing user reviews in app stores is limited. Unlike product reviews of other domains, *e.g.* household appliances, consumer electronics or movies, application reviews offer a couple of peculiarities which deserve special treatment: The way in which users express their opinion in app reviews is shorter and more concise than in other product reviews. Moreover, due to the frequent use of colloquial words and a flexible use of grammar, app reviews can be considered to be more similiar to Twitter messages ("Tweets") than reviews of products from other domains or platforms like Amazon. However, the comparison of existing methods in this area is difficult, because there is (to the best of our knowledge) no corpus of annotated application reviews available to the research community. With this paper, we contribute to this situation: We publish the first corpus with fine-grained sentiment information (*i.e.*, annotations of subjective phrases, aspects, and their relations) of German mobile app reviews from the Google Play Store. The corpus is available for future research[5].

2. Previous Work

A plethora of approaches for opinion mining has been proposed in the last decades. Many of them are based on statistically trained, supervised models (Klinger and Cimiano, 2013b; Li et al., 2010, for instance), incorporate weakly supervised machine learning techniques (Titov and McDonald, 2008; Täckström and McDonald, 2011) or employ rules (Reckman et al., 2013) or dictionaries (Waltinger, 2010b) to

[1] https://itunes.apple.com/us/genre/ios/id36?mt=8
[2] https://play.google.com/store/
[3] https://appworld.blackberry.com/webstore/
[4] https://www.microsoft.com/en-us/windows/apps-and-games
[5] The corpus and further information, including the complete app list and annotation guidelines, are available at http://www.romanklinger.de/scare/

Authors	Store	#Apps	#Reviews
Vasa et al. (2012)	A	17,330	8,701,198
Harman et al. (2012)	B	32,108	—
Iacob and Harrison (2013)	B	270	137,000
Khalid (2013)	A	20	6,390
Galvis Carreno et al. (2013)	G	3	327
Fu et al. (2013)	G	171,493	13,286,706
Pagano and Maalej (2013)	A	1,100	1,100,000
Chen et al. (2014)	G	4	241,656
Khalid et al. (2014)	A	20	6,390
Guzman and Maalej (2014)	A,G	7	32,210
Vu et al. (2015)	G	95	2,106,605
Martin et al. (2015)	B	15,095	2,729,103
Maalej and Nabil (2015)	A,G	1,140	1,303,182

Table 1: Overview of existing work on app store review mining and analysis. For each approach the number applications and reviews used as well as the app store (Apple App Store (A), Google Play Store (G) or BlackBerry World (B)) they originate from are given. All approaches use English language reviews.

detect sentiment in text. One focus is the study of product reviews (Klinger and Cimiano, 2014), Twitter messages (Liu et al., 2012; Tang et al., 2014) and blog posts (Klinger and Cimiano, 2013a; Kessler et al., 2010).

Only a limited number of approaches focused on mobile applications and their user reviews. Early work found a strong correlation between the customer rating and the rank of app downloads by analyzing over 32,000 apps in the BlackBerry App Store (Harman et al., 2012). These results also show that there is no correlation between price and download as well as price and rating. Iacob and Harrison (2013) automatically detect feature requests. They use a corpus of 3,279 reviews from different applications of the BlackBerry App Store to manually create a set of 237 linguistic patterns (*e.g.* "*Adding <request> would be <POSITIVE-ADJECTIVE>*"). Fu et al. (2013) focus on negative reviews and the identification of reasons which lead to a poor rating. They create a linear regression model to identify inconsistencies between text and rating of an app review. Moreover, they use Latent Dirichlet allocation (LDA) (Blei et al., 2003) to extract topics from negative reviews. Building on that, a comparison of the main reasons for poor reviews of applications from different categories is made. For example, they found that games are mainly criticized due to their lack of attractiveness, price and insufficient stability. The main criticism of sport, social networks and finance applications are connectivity issues.

Similarly, Chen et al. (2014) employ topic models in a semi-supervised expectation maximization classifier (Nigam et al., 2000) to distinguish informative and non-informative reviews. Informative reviews include feature requests or specific error descriptions. Emotional expressions or unclear error descriptions are considered non-informative. After filtering of non-informative reviews they extract and rank topic of the remaining reviews using two topic models (LDA and Aspect and Sentiment Unification Model (Jo and Oh, 2011)). Other topics in this area include fraud detection (Gade and Pardeshi, 2015), classification of app reviews to identify bug reports and feature requests (Maalej and Nabil, 2015), and coarse-grained sentiment analysis (Gu and Kim, 2015). Further research includes topic (Galvis Carreno and Winbladh, 2013) and feature detection (Guzman and Maalej, 2014), keyword extraction (Vu et al., 2015), and review impact analysis (Pagano and Maalej, 2013). Table 1 provides an overview of app store mining approaches and the review corpora used within them.

Figure 1: Example review from Google Play Store for the app MAPS.ME (personally identifiable information blurred for depiction).

For fine-grained sentiment analysis and opinion mining in other domains than app reviews, a plethora of manually annotated corpora is available (Lakkaraju et al., 2011; Nakov et al., 2013; Wiebe et al., 2005, for instance). Hu and Liu (2004) constructed a corpus of Amazon reviews annotated with aspects, subjective phrases and a polarity score for each sentence. Spina et al. (2012) provide a data set containing 9,396 Tweets annotated with offsets for aspect mentions (of predefined categories) and evaluative phrases. Annotated blog posts about cars and cameras are the content of the JDPA sentiment corpus from Kessler et al. (2010).

A few corpora exist for other languages than English. Examples are a corpus of sentences from German web texts with subjectivity and polarity annotations on sentence, phrase and word level (Clematide et al., 2012). Klinger and Cimiano (2014) published the USAGE corpus containing the annotation of product aspects and evaluative phrases in German and English Amazon reviews.

3. A Corpus for Fine-grained Sentiment Analysis of Mobile Application Reviews

In the following, we present the Sentiment Corpus of App Reviews (SCARE) consisting of mobile application reviews annotated with aspects, subjective (evaluating) phrases, polarities and their relation.

3.1. Corpus Selection

We select eleven application categories, which represent typical use-cases of mobile applications. The categories are *instant messengers*, *fitness tracker*, *social network platforms*, *games*, *news applications*, *alarm clocks*, *navigation and map applications*, *office tools*, *weather apps*, *sport news* and *music players*. We further choose 10–15 widely-used applications from each category, leading to 148 applications in total. A complete list of these applications and categories is available at the corpus website.[6]

[6] http://www.romanklinger.de/scare

We use the Android Market API[7], a programming interface for the Google Play Store, to retrieve reviews of mobile applications. This API provides a sub-sample with up to 5,000 reviews per application which contains both the latest and older reviews of an application. Repeating requests allows for collecting more reviews, if available. We retrieved all reviews for the selected list of applications in the time period of December 2014 to June 2015 leading to over 800,000 German app reviews.

3.2. Annotation Guidelines

We distinguish the classes *aspect* and *subjective (evaluative) phrase*. An aspect is part of an app or related to it, *e.g.*, separate features of the application, usability, design, price, required authorizations or displayed advertisement. Additionally, we regard the whole application itself as well as errors and feature requests as aspects. Beyond that, we annotate the relationship of the aspect to the main application discussed in the review. Aspects which refer to an app or an aspect of an app other than the app in discussion are marked as "foreign". This is often the case in cross-application comparisons. All other aspects are "related".
Subjective phrases express opinions and statements of a personal evaluation regarding the app or a part of it, that are not based on (objective) facts but on individual opinions of the reviewers. Each subjective phrase is assigned a polarity (positive, negative, neutral) and may have a set of aspects it refers to.
The following sentences (with aspects and *subjective* phrases) illustrate the entity classes and annotation guidelines:

- *Sehr gute* und *übersichtliche* App.
  (Very good and well-arranged app.)
  - *App* is a target of *Sehr gute* and *übersichtliche*. Both evaluations are positive.
- Die Verbindungsanzeige *funktioniert nicht*.
  (The connection indicator does not work.)
  - *Verbindungsanzeige* is a target of *funktioniert nicht*, which represents a negative evaluation.
- Die App ist *cool*, aber das Design ist *schrecklich*.
  (The app is cool, but the design is terrible.)
  - *cool* is a positive evaluation of *App*, *schrecklich* represents a negative opinion for *Design*.

In addition, the annotators were instructed to annotate aspects and subjective phrases as fine-grained as possible and to avoid overlapping annotations. The annotations should be as short as possible, as long as the meaning is understandable if only the annotations were given (without the sentence itself).
We performed a two-step annotation process. Firstly, the actual annotation is performed by the annotator. Secondly, the annotator checks and improve the annotations created by himself in the first step and examines the review text for more aspects, subjective phrases and relations between them. The complete guidelines are available on the corpus website.

3.3. Annotation Process

Annotation was performed by four annotators using the program *Brat*[8] in version v1.3 (Stenetorp et al., 2012). One of the annotators is an author of this paper. The group was composed exclusively of men aged 25 to 35 years. The training of the four annotators and optimization of the guidelines has been conducted in three iterations. To quantify the inter-annotator agreement, Fleiss' κ has been measured (Fleiss et al., 2003). In each iteration, 20 reviews were randomly sampled from the complete review corpus and given to the annotators.
The agreement between the annotators reached a κ-value of 0.57 (on token level using an in-out classification scheme) in the first annotation round. An analysis of the pairwise results showed that three of the annotators had a relatively high agreement (average κ-value of 0.66). The agreement between the three annotators and the fourth was comparably low (κ-value of 0.48). Within a meeting with all four annotators, problems and ambiguities in the annotations of the first round were discussed. In the second iteration, an agreement of 0.76 has been achieved. The differences between individual annotators did not reoccur in this iteration. To confirm the result, a third iteration was carried out, which led to an agreement of 0.72.
After completion of this training phrase, the actual annotation was performed. The annotation took place in June and July 2015 over a period of four weeks. For each application category, we randomly sampled 160 reviews from the complete corpus and gave them to the four annotators. The distribution of reviews was designed such that ≈20% of the reviews of each category were annotated by two annotators. This enabled us to monitor the development of the agreement. Each annotator worked on 36 reviews per category. Further, one annotator (one of the authors of this paper) annotated another 52 reviews per category. To build the final corpus, we harmonized the annotations from reviews which were annotated by more than one annotator by considering all identical annotations as well as the intersection of all overlapping but not completely identical annotations, if the meaning was still understandable, in a manual process.

4. Analysis

In the following, we present an overview of the corpus. Furthermore, we provide a prediction baseline for future models to be evaluated using SCARE.

4.1. Collected App Reviews

As mentioned in Section 3.1., the overall corpus consists of 802,860 German app reviews, retrieved in six months. The number of reviews varies greatly between the individual application categories. Only 15,000 reviews were collected for office tools and alarm clocks. In contrast, over 150,000 reviews were retrieved for games and instant messengers. These differences result from different degrees of popularity:

[7] https://code.google.com/p/android-market-api/

[8] http://brat.nlplab.org/

		App categories											
		Alarm Clocks	Fitness Tracker	Games	Instant Messenger	Navigation / Maps	News Apps	Music Player	Office Tools	Social Networks	Sport News	Weather Apps	Total
	# reviews	160	160	160	160	160	160	160	160	160	160	160	1,760
	avg. length	18.6	17.8	17.2	16.9	21.0	22.0	21.6	23.0	13.5	16.9	14.6	18.5
	# entities	648	572	547	527	610	662	623	623	464	595	575	6,446
	# relations	217	176	146	146	202	237	182	193	104	173	193	1,969
Subj. phrases	num.	393	339	357	327	378	375	383	368	315	374	350	3.959
	avg. length	1.69	1.73	1.86	1.91	1.82	1.79	1.95	1.79	1.98	1.88	1.77	1.83
	positive	309	250	230	159	233	211	228	221	158	211	253	2,463
	neutral	6	2	4	5	6	6	9	10	5	2	8	63
	negative	78	87	123	163	139	158	146	137	152	161	89	1,433
Aspects	num.	255	233	190	200	232	287	240	255	149	221	225	2,487
	avg. length	1.21	1.31	1.16	1.28	1.33	1.20	1.25	1.33	1.19	1.22	1.17	1.25
	related	245	224	183	177	220	274	225	235	146	212	216	2,357
	foreign	10	9	7	23	12	13	15	20	3	9	9	130
Agreement	Fleiss' κ	0.73	0.74	0.68	0.73	0.73	0.71	0.69	0.71	0.73	0.72	0.74	0.72
	F_1 subj. phrases	0.70	0.73	0.61	0.64	0.77	0.71	0.65	0.68	0.67	0.71	0.76	0.69
	F_1 aspects	0.86	0.82	0.67	0.67	0.79	0.81	0.73	0.79	0.79	0.77	0.82	0.78
	F_1 relations	0.71	0.71	0.44	0.43	0.68	0.64	0.53	0.58	0.38	0.68	0.80	0.62

Table 2: Statistics of the full corpora as well as separated into different app categories. In total, the corpus contains 6,446 entities (aspects and subjective phrases) and 1,969 relations between them. The provided average pairwise F_1 measures refer to exact matches between the annotations of two annotators.

For instance, the instant messenger *Threema* is installed on one to five million devices compared to 500,000 to 1,000,000 installations of *Smart Office 2* (office tools, numbers as of February 27, 2016)[9].
The collected reviews are short in contrast to other product domains (Pollach, 2006) with an average length of 17 token. Negative reviews, *i.e.*, reviews with a 1 or 2 star rating, are generally longer with 25 tokens on average than positive ones (4 or 5 stars, 13 tokens on average).
The average star rating of the reviews is 3.75 (on a scale between 1 and 5). The average values for the different categories vary from 3.46 (instant messenger) to 4.22 (alarm clocks). Nearly 70 % of all reviews have a minimal or maximal star rating. In contrast, ratings with three stars, which supposedly reflect a neutral or mixed assessment, are rare (about 8%). This distribution pattern is common in user reviews and has as well been observed in other product domains (Filatova, 2012).

4.2. Annotations

Table 2 summarizes the annotated corpus. The corpus consists of 1,760 annotated application reviews. On average, a review contains 3.66 entities. The number of aspects is in total 2,487. The most frequent aspects are referring to the whole application itself. Less frequent aspects are often more specific. The majority of aspects are directly related to the application in discussion. Only 130 aspects are marked as "foreign". There are 3,959 subjective phrases from which 2,463 are positive, 1,433 are negative, and 63 are neutral. Subjective phrases are, with an average length of 1.83 tokens per phrase, longer than the annotated aspects (avg. 1.25 tokens). The most frequent subjective expressions are "Super", "Top", "Gut" (good), "Sehr gut" (very good) und "cool". The number of subjective phrase-target relations is 1,969.

The number of subjective phrases within the different application categories is relatively constant. In contrast, the number of application aspects (from 149 to 287) and relations (104 to 237) varies much more. Especially reviews for social network applications contain less application aspects (149) and relations (104) than reviews of other categories. This indicates that reviews of this category do not outline detailed information about features or properties of the application. Examples are *"Sehr gut" (Very good)* or *"Einfach nur lächerlich" (Just ridiculous)*.

The inter-annotator agreement for the final corpus is the same as during the training phase with a Fleiss κ of 0.72. The highest agreement holds within weather and fitness apps. The annotation of games and music players shows the lowest agreement. Reviews from social networks contain less application aspects, more slang words and are generally more often plain praising or blaming of the application, *e.g.* *"Facebook sucks!"* or *"Ich liebe instagram!!" (I love instagram)*, than reviews of other categories. On the contrary, reviews for weather and fitness apps consist of clear and rec-

[9]See information on `https://play.google.com/store/apps/details?id=ch.threema.app` and `https://play.google.com/store/apps/details?id=com.picsel.tgv.app.smartoffice`

Lexicon/Authors	#Pos	#Neg
SentiWS (Remus et al., 2010)	13,910	13,825
GermanLex (Clematide and Klenner, 2010)	2,812	4,677
German Senti Spin (Waltinger, 2010b)	42,276	63,284
German Subj. Clues (Waltinger, 2010b)	3,336	5,742
German Polarity Clues (Waltinger, 2010a)	2,994	5,749
German Polarity Clues (Waltinger, 2010a)	17,627	19,962
App-Domain-1vs5 (AD-1vs5)	1,101	326
App-Domain-12vs45 (AD-12vs45)	3,449	975

Table 3: Overview of the used polarity lexicons. The first six are previously published, general-purpose polarity lexicons. The last two are domain-specific lexicons build on the collected app reviews. The figures represent the actual determined word numbers from the freely available files.

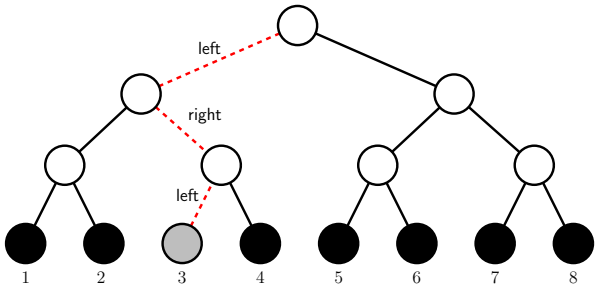

Figure 2: Example for feature extraction based on word embeddings. The grey leaf corresponds to the closest cluster to the word for which features are extracted. Features for the path from the root are therefore `left`, `left-right`, `left-right-left`, and `cluster-id=3`.

ognizable expressions of sentiments, application aspects and links between both. In particular, many reviews of the two categories focus on the accuracy (*e.g.* of a weather forecast or a tracked running route) of the application.

Pairwise F_1 scores can serve as an upper bound for automatic analysis tools: If the agreement between two human is lower than between a tool and a human, the result should be interpreted critically. We observe a higher agreement on aspects (F_1 score of 0.78) than subjective expressions (0.69).

4.3. Prediction Baseline

In the following, we provide baseline results on SCARE, achieved with a linear-chain conditional random field model (CRF, Lafferty et al. (2001)). We use MALLET[10] for implementation.

The following features are implemented to capture the characteristics of aspects and subjective phrases in our IOB (inside, outside, begin) sequence prediction setting (inspired by previous work, *e.g.* Klinger and Cimiano (2013a)):

Token-based features: Each token is represented by a set of features, *i.e.*, the token itself, its POS tag, the combination of the token and the POS tag, the token in lower case letters as well as whether the token contains numbers or non-ASCII characters. In addition, we check whether the token is an emoticon, smiley or negation word based on manually created lists[11].

Polarity lexicons: We use eight polarity lexicons to detect positive or negative words, each leading to a separate feature (*cf.* Table 3). Two out of these eight are domain-specific dictionaries we compiled based on their pointwise mutual information with respect to the star-rating[12]. This procedure has been performed on the full crawled corpus, but without taking advantage of the annotations. Therefore, for unseen data, these dictionaries can be adapted analogously.

Word embeddings: To capture characteristics of infrequent terms (like typos or slang words), we opted for the creation of word embeddings-based features (Turney and Pantel, 2010). For each token, all other tokens with a cosine-similarity greater than 0.8 are added as features. In addition, the index of the most similar cluster center of a hierarchical clustering is added as well as the full path and all path prefixes in the cluster hierarchy. An example for this procedure is shown in Figure 2.

We use the CBOW model of Word2Vec[13] (Mikolov et al., 2013a; Mikolov et al., 2013b) to estimate word embeddings with a context size of 5 on the complete corpus of collected app reviews. We omit tokens with less than 10 occurrences. All other parameters of the model are set to the default values.

Context features: To capture the context of a token, all features of tokens in a left and right-window of 2 are taken into account.

4.4. Results

We perform the following two experiments to evaluate our model:

1. **10-fold cross-validation on the full corpus** including all reviews from all application categories. Cross-validation is performed on the document level (not on sentence-level) to ensure that no characteristics of one review is shared between the respective training and validation set.
2. **Cross-category validation:** training on the reviews from all but one application category and test on reviews of the hold-out category. This setup is performed for each application category. The goal of this evaluation is to get insights on how homogeneously opinions and application aspects are expressed within different categories and how easy a model trained on app reviews of certain categories can be transferred to reviews of a new application category.

[10] `http://mallet.cs.umass.edu/`

[11] The created lists can be found on our corpus website at `http://www.romanklinger.de/scare/`

[12] For *AD-1vs5* and *AD-12vs45* we treated 1 resp. 1 and 2 star ratings as negative contexts and 5 resp. 4 and 5 star ratings as positive contexts.

[13] `https://code.google.com/archive/p/word2vec/`

		Aspects						Subjective phrases					
		Exact			Partial			Exact			Partial		
		P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
cross-category validations	Alarm Clocks	0.69	0.56	0.62	0.78	0.63	0.70	0.71	0.62	0.66	0.87	0.76	0.82
	Fitness Tracker	0.74	0.58	0.65	0.79	0.62	0.69	0.73	0.65	0.69	0.89	0.73	0.84
	Games	0.73	0.59	0.65	0.81	0.66	0.73	0.68	0.58	0.63	0.86	0.73	0.79
	Instant Messenger	0.68	0.57	0.62	0.75	0.62	0.68	0.68	0.62	0.64	0.88	0.79	0.84
	Navigation / Maps	0.65	0.51	0.57	0.74	0.58	0.65	0.68	0.60	0.64	0.87	0.76	0.81
	News Apps	0.65	0.58	0.61	0.75	0.66	0.70	0.65	0.57	0.60	0.85	0.72	0.78
	Music Player	0.67	0.58	0.62	0.78	0.67	0.72	0.62	0.58	0.60	0.85	0.76	0.80
	Office Tools	0.69	0.53	0.60	0.79	0.60	0.69	0.66	0.58	0.62	0.85	0.75	0.78
	Social Networks	0.67	0.57	0.62	0.73	0.62	0.67	0.65	0.56	0.60	0.84	0.72	0.78
	Sport News	0.67	0.53	0.60	0.78	0.62	0.69	0.66	0.58	0.62	0.87	0.76	0.81
	Weather Apps	0.66	0.55	0.60	0.78	0.66	0.71	0.63	0.61	0.62	0.86	0.82	0.84
	10-fold cross-val.	0.69	0.56	0.62	0.77	0.62	0.69	0.67	0.59	0.63	0.86	0.75	0.80

Table 4: Evaluation results of the CRF-based model. "*10-fold cross-val.*" refers to a 10-fold cross-validation experiment on the full corpus. The "*cross-category validations*" correspond to an evaluation setting in which the model is trained on all reviews except for the application category indicated in the table. The reviews of the left-out category are used for testing, the results of which are shown in the table. We further distinguish aspect and subjective phrase prediction as well as exact and partial matches for each experiment.

The evaluation results are shown in Table 4. We report precision, recall and F_1 measures and further distinguish exact and partial matches between prediction and annotation. In exact mode the predicted text spans of aspects and subjective phrases must exactly match those of the gold standard. A partial match true positive holds if gold and prediction overlap by at least one token.

The results of the 10-fold cross-validation experiment are similar to the figures of other sentiment analysis systems on reviews of other product domains (Klinger and Cimiano, 2014). Considering only exact matches the model achieves similiar values on the extraction of application aspects (F_1 score of 0.62) and subjective evaluations (0.63). Taking partial matches into account, the model reaches higher results on the detection of subjective expressions (0.80) than aspects (0.69). Precision is higher than recall throughout the experiments. In order to achieve a better understanding and comparability of the results, we further run the model described in Klinger and Cimiano (2013b), expanded to include the polarity lexicons and word embeddings of our model, on SCARE. The model from Klinger and Cimiano (2013b) achieves a slightly lower performance (F_1 score of 0.67 on the aspect extraction and 0.78 on subjective evaluations regarding partial matches) on SCARE than the CRF-based model.

The results of the cross-category experiment show relatively homogeneous performance for aspect detection and extraction of subjective phrases. Therefore, we expect a good adaptability to novel domains, unseen at the time of estimating the model.

5. Summary and Conclusion

We present, to the best of our knowledge, the first manually annotated resource for fine-grained sentiment analysis of German mobile application reviews. The reviews are annotated with aspects, evaluative (subjective) phrases and relations between them. The corpus consists of 1,760 annotated application reviews containing 2,487 aspects and 3,959 subjective phrases. During annotation we achieved an inter-annotator agreement of 0.72 (Fleiss' κ). We further provide a strong prediction baseline by applying a CRF-based model on the corpus resulting in an F_1 score of 0.62 for aspect detection and 0.63 for the extraction of subjective phrases. During the construction of the corpus, we further collected a data set of over 800,000 reviews from apps of 11 different application categories, which is (as far as we known) the first German corpus in this domain available. These data will motivate and enable an array of novel research questions to be investigated and foster the development of sentiment analysis methods on mobile application reviews and, in general, on German text.

6. Acknowledgments

We thank Heiko Ehrig, Hilmi Yildirim and Abdoulaye Dramé for their annotation work and feedback during the optimization of the annotation guidelines. We thank Christian Scheible for fruitful discussions.

7. Bibliographical References

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022.

Chen, N., Lin, J., Hoi, S. C., Xiao, X., and Zhang, B. (2014). Ar-miner: Mining informative reviews for developers from mobile app marketplace. In *Proceedings of the 2014 International Conference on Software Engineering*, pages 767–778, Hyderabad, India.

Clematide, S. and Klenner, M. (2010). Evaluation and extension of a polarity lexicon for german. In *Proceedings*

of the First Workshop on Computational Approaches to Subjectivity and Sentiment Analysis, Workshop at European Conference on Artifical Intelligence, pages 7–13, Lisbon, Portugal.

Clematide, S., Gindl, S., Klenner, M., Petrakis, S., Remus, R., Ruppenhofer, J., Waltinger, U., and Wiegand, M. (2012). MLSA – A Multi-layered Reference Corpus for German Sentiment Analysis. In *Proceedings of the 8th International Conference on Language Ressources and Evaluation (LREC)*, pages 3551–3556, Marrakesh, Marocco. European Language Resources Association (ELRA).

Filatova, E. (2012). Irony and sarcasm: Corpus generation and analysis using crowdsourcing. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey. European Language Resources Association (ELRA).

Fleiss, J., Levin, B., and Paik, M. C. (2003). *Statistical Methods for Rates and Proportions*. John Wiley and Sons New York.

Fu, B., Lin, J., Li, L., Faloutsos, C., Hong, J., and Sadeh, N. (2013). 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*, pages 1276–1284, Chicago, USA. Association for Computing Machinery.

Gade, T. and Pardeshi, N. (2015). A survey on ranking fraud detection using opinion mining for mobile apps. *International Journal of Advanced Research in Computer and Communication Engineering*, 4(12).

Galvis Carreno, L. and Winbladh, K. (2013). Analysis of user comments: an approach for software requirements evolution. In *Proceedings of the 2013 International Conference on Software Engineering*, pages 582–591, San Francisco, CA, USA.

Gu, X. and Kim, S. (2015). What parts of your apps are loved by users? In *Proceedings of the 30th IEEE/ACM International Conference on Automated Software Engineering*, pages 760–770, Lincoln, USA. IEEE.

Guzman, E. and Maalej, W. (2014). How do users like this feature? a fine grained sentiment analysis of app reviews. In *Proceedings of the 22nd International Requirements Engineering Conference*, pages 153–162, Karlskrona, Sweden.

Harman, M., Jia, Y., and Zhang, Y. (2012). App store mining and analysis: Msr for app stores. In *Proceedings of the 9th IEEE Working Conference on Mining Software Repositories*, pages 108–111, Zurich, Switzerland.

Hu, M. and Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177, New York, NY, USA. Association for Computing Machinery.

Iacob, C. and Harrison, R. (2013). Retrieving and analyzing mobile apps feature requests from online reviews. In *Proceedings of the 10th IEEE Working Conference on Mining Software Repositories*, pages 41–44, San Francisco, CA, USA.

Jo, Y. and Oh, A. (2011). Aspect and sentiment unification model for online review analysis. In *Proceedings of the 4th ACM International Conference on Web Search and Data Mining*, pages 815–824, Kowloon, Hong Kong. Association for Computing Machinery.

Kessler, J. S., Eckert, M., Clark, L., and Nicolov, N. (2010). The 2010 ICWSM JDPA Sentiment Corpus for the Automotive Domain. In *4th International AAAI Conference on Weblogs and Social Media Data Workshop Challenge (ICWSM-DWC 2010)*, Washington, DC, USA.

Khalid, H., Nagappan, M., Shihab, E., and Hassan, A. (2014). Prioritizing the devices to test your app on: A case study of android game apps. In *Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering*, pages 610–620, Hong Kong, China. Association for Computing Machinery.

Khalid, H. (2013). On identifying user complaints of ios apps. In *Proceedings of the 35th International Conference on Software Engineering (ICSE)*, pages 1474–1476, San Francisco, CA, USA. IEEE.

Klinger, R. and Cimiano, P. (2013a). Bi-directional interdependencies of subjective expressions and targets and their value for a joint model. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 848–854, Sofia, Bulgaria. Association for Computational Linguistics.

Klinger, R. and Cimiano, P. (2013b). Joint and pipeline probabilistic models for fine-grained sentiment analysis: Extracting aspects, subjective phrases and their relations. In *IEEE International Conference on Data Mining Workshops (ICDMW)*, Dallas, TX, USA.

Klinger, R. and Cimiano, P. (2014). The USAGE review corpus for fine-grained multi-lingual opinion analysis. In *Proceedings of the Nineth International Conference on Language Resources and Evaluation*, pages 2211–2218, Reykjavik, Iceland.

Lafferty, J., McCallum, A., and Pereira, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *International Conference on Machine Learning*, pages 282–289, Williamstown, MA, USA.

Lakkaraju, H., Bhattacharyya, C., Bhattacharya, I., and Merugu, S. (2011). Exploiting coherence for the simultaneous discovery of latent facets and associated sentiments. In *SIAM International Conference on Data Mining*, pages 498–509, Mesa, AZ, USA. SIAM / Omnipress.

Li, F., Huang, M., and Zhu, X. (2010). Sentiment analysis with global topics and local dependency. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, pages 1371–1376, Atlanta, Georgia, USA.

Liu, K.-L., Li, W.-J., and Guo, M. (2012). Emoticon smoothed language models for twitter sentiment analysis. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, Toronto, ON, Canada.

Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1):1–167.

Maalej, W. and Nabil, H. (2015). Bug report, feature request, or simply praise? on automatically classifying app

reviews. In *Proceedings of the IEEE 23rd International Requirements Engineering Conference*, pages 116–125, Karlskrona, Sweden. IEEE.

Martin, W., Harman, M., Jia, Y., Sarro, F., and Zhang, Y. (2015). The app sampling problem for app store mining. In *Proceedings of the 12th IEEE Working Conference on Mining Software Repositories*, pages 123–133, Florence, Italy. IEEE.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, et al., editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

Nakov, P., Rosenthal, S., Kozareva, Z., Stoyanov, V., Ritter, A., and Wilson, T. (2013). Semeval-2013 task 2: Sentiment analysis in twitter. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 312–320, Atlanta, Georgia, USA. Association for Computational Linguistics.

Nigam, K., McCallum, A., Thrun, S., and Mitchell, T. (2000). Text classification from labeled and unlabeled documents using EM. *Machine learning*, 39(2-3):103–134.

Pagano, D. and Maalej, W. (2013). User feedback in the appstore: An empirical study. In *Proceedings of the 2013 21st IEEE International Requirements Engineering Conference*, pages 125–134, Rio de Janeiro, Brazil. IEEE.

Pang, B. and Lee, L. (2008). *Foundations and Trends in Information Retrieval 2(1-2), pp. 1–135, 2008.*, volume 2 (1-2) of *Foundations and Trends in Information Retrieval*. Now Publishers.

Pollach, I. (2006). Electronic word of mouth: a genre analysis of product reviews on consumer opinion web sites. In *Proceedings of the 39th Annual Hawaii International Conference on System Sciences*, volume 3, Kauia, HI, USA. IEEE.

Reckman, H., Baird, C., Crawford, J., Crowell, R., Micciulla, L., Sethi, S., and Veress, F. (2013). teragram: Rule-based detection of sentiment phrases using SAS sentiment analysis. In *Proceedings of the 7th International Workshop on Semantic Evaluation Exercises*, pages 513–519, Atlanta, Georgia, USA.

Remus, R., Quasthoff, U., and Heyer, G. (2010). Sentiws - a publicly available german-language resource for sentiment analysis. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Spina, D., Meij, E., de Rijke, M., Oghina, A., Bui, M. T., and Breuss, M. (2012). Identifying entity aspects in microblog posts. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '12, pages 1089–1090, New York, NY, USA. Association for Computing Machinery.

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). brat: a web-based tool for nlp-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107, Avignon, France. Association for Computational Linguistics.

Täckström, O. and McDonald, R. (2011). Semi-supervised latent variable models for sentence-level sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 569–574, Portland, Oregon, USA. Association for Computational Linguistics.

Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T., and Qin, B. (2014). Learning sentiment-specific word embedding for twitter sentiment classification. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 1555–1565, Baltimore, MD, USA.

Titov, I. and McDonald, R. (2008). A Joint Model of Text and Aspect Ratings for Sentiment Summarization. In *Proceedings of ACL-08: HLT*, pages 308–316, Columbus, Ohio. Association for Computational Linguistics.

Turney, P. and Pantel, P. (2010). From frequency to meaning: vector space models of semantics. *Journal of Artificial Intelligence Research*, 37(1):141–188.

Vasa, R., Hoon, L., Mouzakis, K., and Noguchi, A. (2012). A preliminary analysis of mobile app user reviews. In *Proceedings of the 24th Australian Computer-Human Interaction Conference*, pages 241–244, Melbourne, VIC, Australia. Association for Computing Machinery.

Vu, P. M., Nguyen, T. T., Pham, H. V., and Nguyen, T. T. (2015). Mining user opinions in mobile app reviews: a keyword-based approach. In *Proceedings of the 30th IEEE/ACM International Conference on Automated Software Engineering*, pages 749–759, Lincoln, NE, USA. IEEE.

Waltinger, U. (2010a). Germanpolarityclues: A lexical resource for german sentiment analysis. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Seventh conference on International Language Resources and Evaluation*, Valletta, Malta, May. European Language Resources Association (ELRA).

Waltinger, U. (2010b). Sentiment analysis reloaded - a comparative study on sentiment polarity identification combining machine learning and subjectivity features. In *Proceedings of the 6th International Conference on Web Information Systems and Technologies*, pages 203–210, Valencia, Spain.

Wiebe, J., Wilson, T., and Cardie, C. (2005). Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation*, 39(2-3):165–210.