Anger and Its Direction in Collaborative Software Development

Daviti Gachechiladze*, Filippo Lanubile†, Nicole Novielli†, Alexander Serebrenik*
* Eindhoven University of Technology, The Netherlands
† University of Bari, Italy

Abstract

Recent research has provided evidence that communication between software developers reflects a wide range of emotions. We argue that among those emotions anger deserves special attention as it can serve as an onset for tools supporting collaborative software development. This, however, requires a fine-grained model of the anger emotion, able to distinguish between anger directed towards self, others, and objects. Detecting anger towards self could be useful to support developers experiencing difficulties; detection of anger towards others might be helpful for community management; detecting anger towards objects might be helpful to recommend and prioritize improvements. As a first step towards automatic identification of anger direction, we built a classifier for anger direction, based on a manually annotated gold standard of 723 sentences that were obtained by mining comments in Apache issue reports.

1 Introduction

Software development is an inherently social activity, involving a large amount of interaction, as programmers often need to cooperate with others. Recent research has provided evidence that communication between software developers reflects a wide range of emotions [12]. So far, the majority of studies addressing the role of emotions in software development rely on polarity as the only dimension to operationalize affect.

Proceedings of the Seminar Series on Advanced Techniques and Tools for Software Evolution SATToSE 2017 (sattose.org). 07-09 June 2017, Madrid, Spain.

However, polarity is only one of the possible dimensions of affect [9, 12]. Among others, negative affective states recently received particular attention [3, 8] due to their detrimental impact on developers productivity and ability to react [1].

We envision emergence of tools monitoring communication between the developers, analysing the negative affect expressed in this communication and translating the analysis results into actionable insights.

To support this vision we focus on anger and all its nuances, and advocate a fine-grained model of the anger emotion distinguishing between anger directed towards self, others, and objects. Detecting anger towards *self* could be useful to design tools for supporting developers in their daily programming tasks [8]. Conversely, timely detection of anger towards *others*, such as peers, in developers' communication messages [3], might be exploited for detection of code of conduct violations [15]. Finally, detecting the expression of anger towards *objects* might be helpful to recommend and prioritize improvements based on the complaints about frameworks, programming languages or documentation [3].

Our work is therefore intended as the first step towards creation of tools monitoring communication between the developers, analysing the negative affect expressed in this communication and translating the analysis results into actionable insights.

2 A Psychological Model of Anger Direction

Psychologists worked at decoding emotions for decades, developing theories aiming at classification of emotions and their functioning. As far as emotion mining from text is concerned,emotions are either considered as a continuous function of two dimensions, valence (affect polarity) and arousal (level of activation) [13], or as a finite set of individual emotions. The latter is represented by the framework of Shaver et

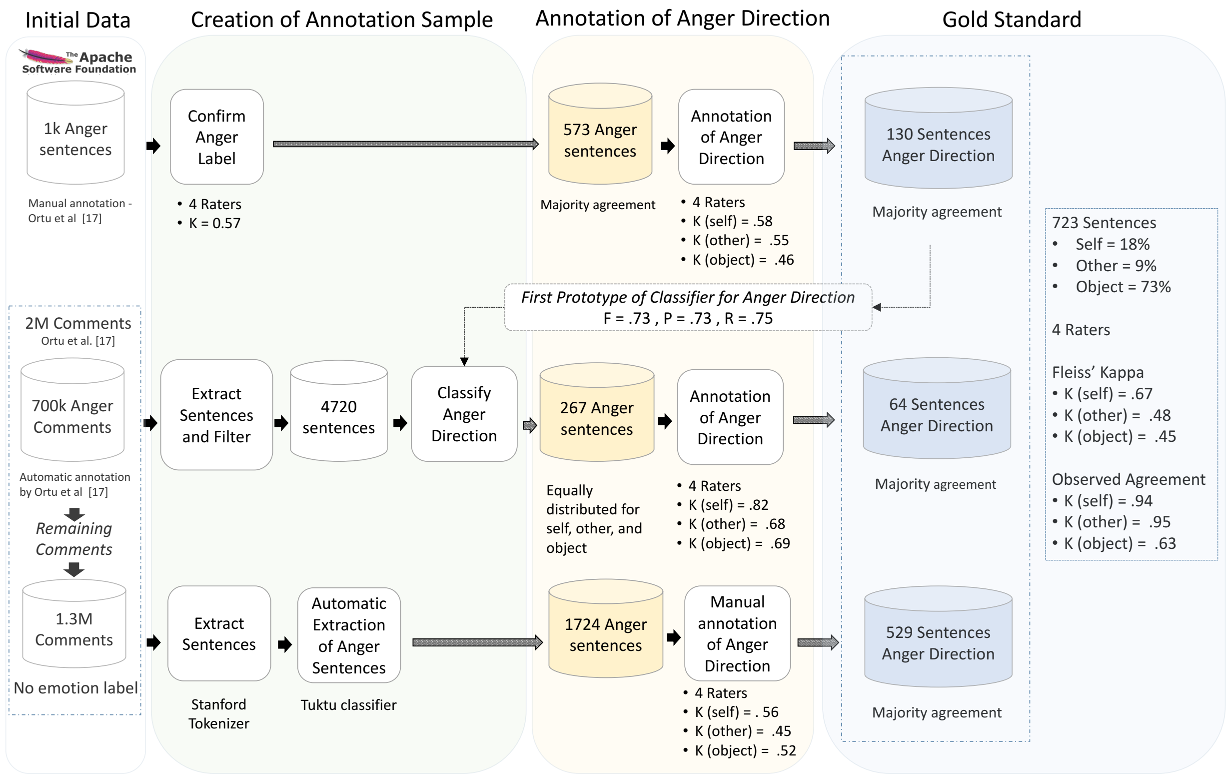

Figure 1: Creating the Gold Standard through Manual Annotation of Anger Direction.

al. [14], which includes *love*, *joy*, *anger*, *sadness*, *fear*, and *surprise*.

When modeling anger direction, we combine the Shaver et al. definition of anger with further specification of its direction. In particular, we follow the OCC model [10] according to which emotions can be a reaction that focuses on *self*, on the *other* agent, or some properties of an *object*. For instance, a comment taken from Apache Jira "I don't have to ensure that the classloader knows groovy classes, *you* must do that"[1] expresses anger directed towards *other*.

3 A Gold Standard for Anger Direction

As our goal is to create a tool for automatic detection of anger direction, we first create a collection of angry statements and manually label each one of the statement with one of the three directions (self, other, object). Later on we use this data to train and evaluate an automatic classifier (Section 4).

Starting from the comments in the Apache issue reports, we built a *gold standard* dataset by annotating angry sentences within comments with the anger direction, as shown in Figure 1. We adopt *sentences* as unit of analysis rather than each comment as a whole. Being able to analyse comments at such a fine-grained level we aim at developing a classifier which is able to clearly identify the location of the anger trigger, being it *self*, the *other* interlocutor or a specific *object*.

We started with the manually labeled emotion dataset of Ortu et al. [11], which is the best dataset available for our purpose. Since emotion annotation is a subjective task [2], before proceeding with the annotation of the anger direction, we preliminary assessed the validity of the anger label in the original dataset. We obtained 130 sentences where the authors, acting as raters, agree both on the presence of anger and on its direction. Using these sentences we developed the first prototype of the classifier for anger direction, trained in a supervised setting exploiting Support Vector Machines [6] on the features described in Section 4.

In the second step we applied this classifier to a noisier anger dataset of sentences derived from 700K comments automatically classified by the tool of Ortu et al. [11]. The classification granularity in this dataset is sentence-based. However, comments are released with just the indication of the number of sentences for which the emotion is detected. To reduce noise, we decided to use only comments composed by at most two sentences, out of which at least one was labeled as containing anger. Using this dataset and manual annotation of the anger direction we extended the annotated collection with 64 additional sentences.

Finally, in the third step we have considered sentences derived from 1.3M comments. These comments

[1] `https://issues.apache.org/jira/browse/GROOVY-1885`

are released without any information about the emotional content [11]. We created the annotation sample using the emotion classification tool Tuktu[2] [16]. We decided to use Tuktu after comparing it against other tools for anger classification, namely Syuzhet[3] and Alchemy, on the first 130 sentences of our gold standard. In particular, we observed the highest precision for Tuktu (P=.73, R=.12, F=.21), the highest recall for Alchemy (P=.36, R=.51, F=.42), and a more balanced performance for Syuzhet, which shows the highest F-measure (P=.44, R=.5, F=.47). By optimising for precision, we reduce the number of neutral sentences misclassified as expressing anger, and then avoid annoying raters with useless annotation of neutral cases. The raters were CS graduate students trained by the first author. The final gold standard consists of 723 anger sentences with direction labels. In particular, we have 18% sentences annotated as *self*, 9% as *other* and 73% as *object*. The interrater agreement was assessed by measuring the Fleiss' Kappa values and percentage of observed agreement among raters, that is the percentage of cases for which the raters provided the same label. Values for both metrics are reported in Figure 1. In particular, Kappa values range from moderate to substantial agreement, indicating a higher level of interrater agreement with respect to previous research on emotion annotation in Apache Jira developer comments [9].

4 A Classifier for Anger Direction

We investigate the feasibility of building an anger direction classifier by exploiting machine learning techniques in a supervised setting, using our gold standard for training and validation. We used Weka[4], a library of machine learning algorithms. As for features, we automatically extracted uni- and bi-grams using the unsupervised Weka filter *StringToWordVector*.

In Table 4 we report the results obtained in a 10-fold cross validation setting with the SMO Weka implementation of Support Vector Machines (SVM), J48, and Naive Bayes. We built our baseline using both the ZeroR Weka classifier, which always predicts the majority class, and random guessing.

The best performance is obtained with SVM. However, the SVM performance reflects a bias towards the majority class *object*. By looking at the confusion matrix, we observe that low recall for *self* is due to the misclassification of 40% *self* sentences as *object*. Similarly, 79% of *other* sentences are misclassified as *object*.

[2]http://www.tuktu.io/

[3]https://github.com/mjockers/syuzhet

[4]`http://www.cs.waikato.ac.nz/ml/weka/`

Classifier	Class	P	R	F
SVM	Self	0.89	0.60	0.72
	Other	0.80	0.18	0.30
	Object	0.83	0.98	0.90
	Overall	0.84	0.84	0.81
J48	Self	0.69	0.57	0.62
	Other	0.38	0.24	0.29
	Object	0.83	0.91	0.87
	Overall	0.76	0.78	0.77
Naive Bayes	Self	0.53	0.82	0.64
	Other	0.30	0.57	0.39
	Object	0.91	0.68	0.78
	Overall	0.78	0.69	0.72
Baseline: Majority Class	Self	0.00	0.00	0.00
	Other	0.00	0.00	0.00
	Object	0.73	1.00	0.84
	Overall	0.53	0.73	0.61
Baseline: Random Guessing	Self	0.18	0.33	0.23
	Other	0.09	0.33	0.14
	Object	0.73	0.33	0.46
	Overall	0.33	0.33	0.33

Table 1: Anger Direction Classification Results. P, R, F correspond to precision, recall and the F-measure respectively.

5 Related Work

This submission is a summary of the recent paper by the authors [4]. Similarly to the analysis of emotions in software artifacts of Murgia et al. [9], we use the framework by Shaver et al. [14]. This study was followed-up by development of a classifier for automatic detection of joy, love, sadness, and anger in issue tracking comments [11]. Similarly to the tools we envision, Keertipati et al. [7] have included information about presence of negative emotion to prioritize feature improvements. Differently from these works, we stress the importance of the emotion direction, specifically the anger direction.

Our approach to detection of emotions is based on the way their are expressed in the developers' written communication. An alternative approach based on the bio-metrics have been recently advocated by Ḿuller and Fritz [8].

6 Discussion and Conclusions

In this work we envision emergence of tools monitoring communication between the developers, analysing the negative affect expressed in this communication and translating the analysis results into actionable insights. To support this vision we have conducted a preliminary study towards automatically detecting the direction of anger when developers communicate by exchanging text messages.

Our preliminary results confirm that all the three anger directions (anger towards self, others, and object) are present within comments from Apache issue reports. The preliminary classifier showed reasonable performance, suggesting that the automatic detection of the emotion direction is a realistic but challenging instrument to investigate the communication behavior among software developers.

Several directions are considered as a future work. First, as indicated above, the gold standard is highly unbalanced with *object* representing the 73% of sentences. This suggests that it is probably easier to express frustration towards something (e.g., tools or programming languages) rather than towards somebody who could be hurt and react negatively. This unbalanced distribution of labels affects the performance of our automatic classification of anger direction. In particular, we observe that the best performing algorithm (SVM) shows high precision values for all classes, while reporting low recall for the *other* class. Thus, we highlight the need for a richer, more balanced dataset to train a robust classifier. Alternatively, more advanced machine learning techniques such as *deep learning* might be considered. Still, early results are encouraging and suggest that automatically detecting anger and its direction is feasible other than worth of the effort.

Second, the taxonomy of the anger direction can be further refined towards more specific classes of objects, corresponding, e.g., to components of the software being developed or software development environment (e.g., the build system or the issue tracker). Orthogonally, the taxonomy can be refined to take into account different kinds of maintenance activities [5].

Finally, leveraging on the results of the classifier designed recommender systems envisioned in the introduction can be built.

Acknowledgment

This work is partially supported by the project "Emo-Quest - Investigating the Role of Emotions in Online Question & Answer Sites", funded by the Italian Ministry for Education, University and Research (MIUR) under the program "Scientific Independence of young Researchers" (SIR).

References

[1] P. J. Denning. Moods. *Commun. ACM*, 55(12):33–35, Dec. 2012.

[2] H. A. Elfenbein and N. Ambady. On the universality and cultural specificity of emotion recognition: a meta-analysis. *Psychological bulletin*, 128(2):203–235, 2002.

[3] D. Ford and C. Parnin. Exploring causes of frustration for software developers. In *CHASE*, pages 115–116. IEEE Press, 2015.

[4] D. Gachechiladze, F. Lanubile, N. Novielli, and A. Serebrenik. Anger and its direction in collaborative software development. In *ICSE NIER*, 2017. accepted.

[5] A. Hindle, D. M. Germán, M. W. Godfrey, and R. C. Holt. Automatic classication of large changes into maintenance categories. In *ICPC*, pages 30–39, 2009.

[6] T. Joachims. Text categorization with suport vector machines: Learning with many relevant features. In *ECML*, pages 137–142. Springer, 1998.

[7] S. Keertipati, B. T. R. Savarimuthu, and S. A. Licorish. Approaches for prioritizing feature improvements extracted from app reviews. In *EASE*, pages 33:1–33:6, New York, NY, USA, 2016. ACM.

[8] S. C. Müller and T. Fritz. Stuck and frustrated or in flow and happy: Sensing developers' emotions and progress. In *ICSE*, pages 688–699. IEEE, 2015.

[9] A. Murgia, P. Tourani, B. Adams, and M. Ortu. Do developers feel emotions? an exploratory analysis of emotions in software artifacts. In *MSR*, pages 262–271. ACM, 2014.

[10] A. Ortony, G. L. Clore, and A. Collins. *The cognitive structure of emotions*. Cambridge University Press, 1990.

[11] M. Ortu, B. Adams, G. Destefanis, P. Tourani, M. Marchesi, and R. Tonelli. Are bullies more productive?: empirical study of affectiveness vs. issue fixing time. In *MSR*, pages 303–313. IEEE, 2015.

[12] M. Ortu, A. Murgia, G. Destefanis, P. Tourani, R. Tonelli, M. Marchesi, and B. Adams. The emotional side of software developers in Jira. In *MSR*, pages 480–483. ACM, 2016.

[13] J. Russell. A circumplex model of affect. *Journal of personality and social psychology*, 39(6):1161–1178, 1980.

[14] P. Shaver, J. Schwartz, D. Kirson, and C. O'connor. Emotion knowledge: further exploration of a prototype approach. *Journal of personality and social psychology*, 52(6):1061–1066, 1987.

[15] P. Tourani, B. Adams, and A. Serebrenik. Code of conduct in open source projects. In *SANER*, pages 24–33, 2017.

[16] E. Tromp and M. Pechenizkiy. Pattern-based emotion classification on social media. In M. M. Gaber, M. Cocea, N. Wiratunga, and A. Goker, editors, *Advances in Social Media Analysis*, pages 1–20. Springer, 2015.