Automatically Extracting and Analyzing Customer Needs from Twitter: A "Needmining" Prototype

Niklas Kühl[1], Dominik Martin[1], and Gerhard Satzger[1]

[1] Karlsruhe Institute of Technology (KIT), Karlsruhe Service Research Institute (KSRI), Karlsruhe, Germany
{kuehl,martin,gerhard.satzger}@kit.edu

Abstract. Automated and scalable elicitation of customer needs is still in its infancy. With the proposed "Needmining" prototype, we aim to enable automated customer need extraction from the micro blog platform Twitter.

Keywords: Customer Needs, Twitter, Machine Learning, Need Elicitation

1 Introduction

For modern businesses to thrive, they must design customer-centric, innovative products and services [1]. To achieve this goal, providers first need to identify the needs of existing and potential customers—so they can directly address them within their marketing activities [2], and, more importantly, create innovative offerings which are tailored to those needs [3].

For *customer need elicitation*, the application of different methods like interviews, focus groups or conjoint analyses is typical [4]. These methods, however, can be very expensive and time-consuming [5]. This is mainly due to the manual effort required, as these traditional methods do neither scale well nor can they be automated.

Fortunately for businesses, customers are in most cases well aware of what they actually need [6]. Today, it is very common for customers to voluntarily and intrinsically share personal information via social media. Some of these social media instances may contain valuable insights about their concerns, wants, demands and problems [7]. These insights are publicly available and free of cost—but any manual review would be laborious and would only allow a snapshot at the time of the analysis.

In the proposed prototype, we show the feasibility to automatically identify and quantify customer needs from a publicly available source like Twitter. The developed artifact utilizes natural language processing and machine learning techniques to provide innovation managers with an aggregated overview of customer needs—and does so on an ongoing, "always current" basis. This contribution has the potential to drive a step change towards automated, analytical support for customer need elicitation.

2 The Needmining artifact

Within an overall Design Science Research (DSR) effort of automatically identifying customer needs from Twitter data [8], a first cycle evaluated the feasibility of "need tweet" identification as a basis. We implemented an automated identification

14th International Conference on Wirtschaftsinformatik,
February 24-27, 2019, Siegen, Germany

of tweets containing customer needs with supervised machine learning models for the exemplary domain of e-mobility [9]. The results from the technical experiment of this first cycle show the feasibility by providing superior statistical classification results compared to all regarded baselines [10]. Additionally, the best-working classification model is deployed as a single service. The second DSR cycle requires this service to output “need tweets” only, as we analyze possibilities on how to identify the needs themselves from this output. As a result of the second cycle, we achieve an additional service which is able to automatically assign tweets to multiple need categories for the domain of e-mobility [11]. Therefore, the previous two cycles demonstrate the feasibility of (a) automatic “need tweet” identification and (b) automatic quantification of the needs themselves. Apart from showing the statistical evaluations for each cycle, two isolated services are designed: One capable of identifying tweets containing needs, one capable of assigning these tweets to pre-defined need categories—based on supervised machine learning. Both services, however, are not tailored to a non-IT focused user, since all other activities would need to be undertaken manually.

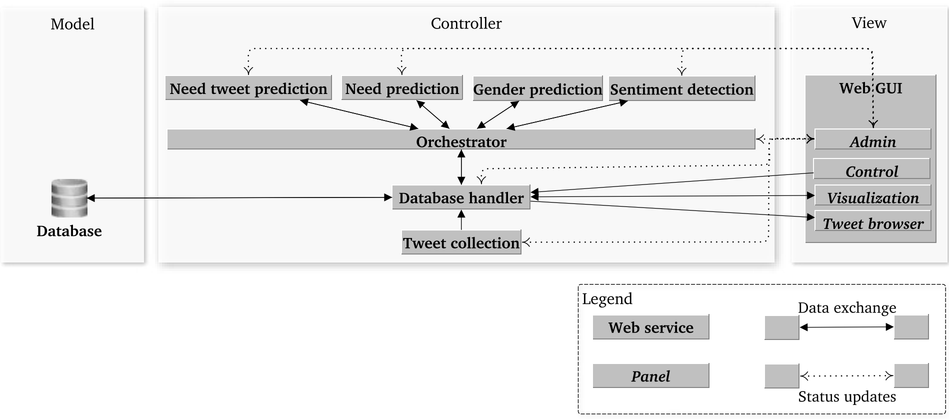

Figure 1. General architecture of the Needmining artifact

For the proposed prototype, we add a third DSR cycle to integrate both services into a comprehensive Needmining artifact. With this resulting artifact we later aim to evaluate not just statistical performance, but also (real) user acceptance. The Needmining artifact aims at automatically visualizing the most frequently expressed customer needs, which are autonomously posted on Twitter.

For a general architecture of the software design, we follow two principles: model-view-controller (MVC) [12] and service-oriented architecture (SOA) [13]. Figure 1 depicts the general overview of the architecture and its different web services. We structure our general architecture into three layers (model, view, controller) and a number of web services as independent instances, which can be accessed via a well-defined interface.

First, the tweets need to be received (*tweet collection*) and then stored for further processing. As multiple services need to access the storage, it is important to implement a *database handler*, which allows access to the database and monitors these accesses.

The *database* itself is part of the model layer. Apart from the tweet collection and the database handler, the controller layer consists of multiple other services. The *need tweet prediction* classifies whether a tweet from the database contains a need or not. In order to aggregate the needs themselves and achieve a distinct representation, they are automatically assigned to need categories as part of the *need prediction*. Moreover, the possibility of monitoring a change in needs over a specified period and differences due to demographic factors—like gender—can provide further insights into the development and the characteristics of certain needs. As an example, we additionally implement a self-built *gender prediction.* To automatically identify the sentiment of a tweet, we also utilize a *sentiment prediction* [14]. All these services (need tweet identification, need clustering, sentiment detection and gender detection) need to be orchestrated with regard to their access to databases as well as their sequence and error

Figure 2. Screenshot from the Needminer GUI

handling. To address this, we implement a central *orchestrator* service, which orchestrates the main functions within the controller layer. As the orchestrator ensures a coherent processing of the data, the *Web GUI* from the view layer can directly access the data and visualize it for the end user (see figure 2). As the view layer is independent of the controller and the model, it can be enhanced or exchanged at any time, e.g. with a smartphone app interface.

3 Evaluation

As it is important to evaluate the artifact in a real-world scenario, we conduct an industry workshop with a large German utility and e-mobility services provider. The participants are from different divisions with different roles in the company. In total, seven employees take part in the three-hour workshop. The group consists of a product & innovation manager, a senior project manager, the head of a competence center, a group leader, a user researcher and an R&D project manager as well as an intern.

To evaluate the prototype, all participants can use the Needmining artifact for up to 30 minutes. Then, they fill out a further questionnaire regarding the experiences with using the software and we openly discuss positive and negative feedback. The questionnaire contains questions to assess (a) the helpfulness of the software, (b) its possible application within the company, (c) its ability to replace existing methods as well as (d) feedback regarding functionalities and design.

Out of seven participants, six regard the Needmining artifact as a helpful software. Furthermore, five participants can imagine using it for their daily business within their company. None of the participants states that the Needmining artifact in its current implementation can fully replace existing need elicitation methods. We further analyze the additional feedback, separated into functional and design feedback.

References

1. Teece, D.J.: Business Models, Business Strategy and Innovation. Long Range Plann. 43, 172 194 (2010).
2. Evans, J., Laskin, R.: The relationship marketing process: A conceptualization and application. Ind. Mark. Manag. 23, 439–452 (1994).
3. Slater, S.F.: Developing a customer value-based theory of the firm. J. Acad. Mark. Sci. 25, 162–167 (1997).
4. Hauser, J.R., Griffin, A.: The Voice of the Customer. Mark. Sci. 12, 1–27 (1993).
5. Blindheim, J., Wulvik, A., Steinert, M.: Using secondary video material for user observation in the needfinding process for new product development and design. In: Proceedings of the DESIGN 2016 14th International Design Conference (2016).
6. Mills, P.K., Margulies, N.: Toward a core typology of service organizations. Acad. Manag. Rev. 5, 255–266 (1980).
7. Misopoulos, F., Mitic, M., Kapoulas, A., Karapiperis, C.: Uncovering customer service experiences with Twitter: the case of airline industry. Manag. Decis. 52, 705–723 (2014).
8. Kuehl, N.: Needmining: Towards Analytical Support for Service Design. In: IESS 2016, Bucharest, Romania, May 25-27, 2016, Proceedings. pp. 187–200. Springer International Publishing, Cham (2016).
9. Scheurenbrand, J., Engel, C., Peters, F., Kühl, N.: Holistically Defining E-Mobility: A Modern Approach to Systematic Literature Reviews. In: 5th Karlsruhe Service Summit, Karlsruhe, Germany (2015).
10. Kühl, N., Scheurenbrand, J., Satzger, G.: Needmining: Identifying micro blog data containing customer needs. In: ECIS 2016 (2016).
11. Kühl, N., Mühlthaler, M., Goutier, M.: Automatically Quantifying Customer Need Tweets : Towards a Supervised Machine Learning Approach. In: HICSS-51, Waikoloa Village, Hawaii, United States, 3rd - 6th January 2018 (2018).
12. Kransner, G.E., Pope, S.: Cookbook for using the model-view-controller user interface paradigm. Object Oriented Program. 26–49 (1988).
13. Huhns, M.N., Singh, M.P.: Service-oriented computing: key concepts and principles. IEEE Internet Comput. 9, 75–81 (2005).
14. Thelwall, M.: Heart and soul: Sentiment strength detection in the social web with sentistrength. Proc. CyberEmotions. 5, 1–14 (2013).