Building an Integrated Comment Moderation System – Towards a Semi-automatic Moderation Tool

Dennis M. Riehle(✉), Marco Niemann, Jens Brunk, Dennis Assenmacher, Heike Trautmann, and Jörg Becker

ERCIS, University of Münster, 48149 Münster, Germany
dennis.riehle@ercis.uni-muenster.de

Abstract. The past decade has been characterized by a strong increase in the use of social media and a continuous growth of public online discussion. With the failure of purely manual moderation, platform operators started searching for semi-automated solutions, where the application of Natural Language Processing (NLP) and Machine Learning (ML) techniques is promising. However, this requires huge financial investments for algorithmic implementations, data collection, and model training, which only big players can afford. To support smaller or medium-sized media enterprises (SME), we developed an integrated comment moderation system as an IT platform. This platform acts as a service provider and offers Analytics as a Service (AaaS) to SMEs. Operating such a platform, however, requires a robust technology stack, integrated workflows and well-defined interfaces between all parties. In this paper, we develop and discuss a suitable IT architecture and present a prototypical implementation.

Keywords: Comment moderation · Machine learning · Business model · IT platform · Analytics as a service

1 Introduction

The internet and especially the web 2.0 disrupted the way how people communicate and exchange information. Where sporadic letters to the editor have been the most prominent form of reader feedback for decades, newspapers nowadays offer discussion fora and comment sections, allowing their audience to directly interact and engage with each other and the journalists [23,32,34]. In times of decreasing sales of physical newspaper copies, reader engagement in the digital realm (and hence dwelling time) is perceived as central for economic sustainability [34]. However, the promising concept turned out to be less shiny in reality: instead of being spaces for lively exchange that create value for the reader, many comment sections turned out to be collections of appalling and vile content [23]. With estimates ranging from 2% to 80% of abusive user-generated comment

G. Meiselwitz (Ed.): HCII 2020, LNCS 12195, pp. 71–86, 2020.
https://doi.org/10.1007/978-3-030-49576-3_6

content [4,11,23,28,36,44], many journalists and newspapers turn their back on comment sections and even urge their readers to refrain from reading comments at all [3,23]. Despite being an unpleasant experience that might scare off people (and hence subscription and advertisement money), especially hateful or insulting comments can even cause substantial legal issues [48]. Considering the massive amounts of incoming comments that newspapers face (e.g., The New York Times reports 9,000 comments per day [17]), manual filtering and moderation turn out to be a "*great deal of extra work*" [48]. The necessary workforce to handle these comments is a substantial investment without any direct returns (as comments are typically a free feature). Consequently, many newspapers either decided to lock the comment sections for the most debated topics or even closed them completely [4,31,48,57]. While economically reasonable, many journalists prefer to refrain from these radical approaches because of ethical considerations [4,14,23,47]. However, if newspapers decide to keep the functionalities for direct user feedback, enforcing the typically more rigid guidelines often worsens the already problematic expenses. For example, moving from post-moderation (*checking when reported*) to strict pre-moderation (*checking before publishing*) can be a prohibitive investment for many newspapers, especially if comments should be released timely. While some – especially large – newspapers might be able to afford this type of debate, many other opportunities for exchange with readers will disappear eventually. To keep these virtual spaces of discourse available, both practitioners and academics began to investigate, whether comment moderation could be (semi-)automated [3,41]. Recent advances in the areas of machine learning (ML) and natural language processing (NLP) show promising results and provide reason to believe that such a solution is indeed feasible. However, there are still several substantial challenges left that range from a precise definition of the problem at hand to the actual training of the ML models and ensuring their acceptance. Furthermore, competent analysts and developers that could create and maintain such a tool are sparse and expensive. This situation makes it hard for small- and medium-sized enterprise (SME) media companies to benefit from these developments that could make comment sections more economically promising again. To address the mentioned challenges, we propose the creation of a novel Analytics as a Service (AaaS) platform, which would help to bundle extant resources to both speed up the problem-solving process as well as to reduce the cost and competencies required from SME newspapers. After introducing the business model in [6], we shed some light on the proposed technical architecture and its implementation.

2 Current Challenges

Despite the fact that abusive language received increasing interest from academia over the last decade, there are still some open research challenges left.

One of the open challenges is the fundamental question of *what constitutes abusive language*. While being widely discussed, no consistent and commonly used definition emerged [41,60] so far, which is partly due to the complexity

of clearly delineating those concepts [22,25,40,52]. As a consequence, there is neither uniformity regarding the terms used to name the concepts nor regarding the use of the terms themselves. To ensure a common terminological ground we propose to increasingly work towards a standard definition or at least to agree on a common process to create definitions as, e.g., proposed by [40]. Since abusive language detection tools will be applied by a diverse set of platforms, this process should be able to integrate platform-specific restrictions [46] as well as the (inter-)national legal perspective(s) under which they are operating [20,40].

A second, closely related, challenge is the *accurate labeling* of large data sets. The underlying issue is the abundance of user comments generated each day that are not only natural language and hence unstructured data, but also lack any form of natural labeling. As abusive comments are usually filtered by algorithms that require those labels to train a suitable model (supervised learning algorithms), this implies a lack of necessary data for algorithms to separate clean from problematic comments. To overcome this issue, data is currently often manually annotated record by record which is time-consuming, expensive, and prone to biases [31,41]. Even though options such as crowd-sourcing exist and have been shown to be effective for the problem at hand [13,29,60], only financially powerful actors will be able to afford a substantial amount of labeling—plus the diversity of the annotating people is often detrimental to the annotation quality and consistency. As the quality of any detection tool substantially depends on the quality of its training material, we propose to use "custom crowds" [35] with carefully selected and curated annotators—which, however, increases the already substantial financial cost. Furthermore, pre-labelled data only reflects the status quo at a specific point in time.

Even with a suitable data set at hand further pitfalls and challenges await. Similar to many other machine learning problems, a plethora of different algorithms and algorithm classes are used to obtain promising results. These range from classical algorithms, such as logistic regression [1,7,13], tree-based approaches [8,10,13,15,37,49], Naïve Bayes [10,13,15,33], and SVMs [7,10,13,15,33,37,49,55,61] to neural networks, e.g. Recurrent Neural Networks [39,46,53], Convolutional Neural Networks [37,45,46,56] or Long Short-Term Memory Networks [1,32,37,56]. Currently, many publications do not report all major metrics (precision [1,7,37,41], recall [1,7,9,37,41], F-score [1,7,37,41], accuracy [16,46,60]), nor does the domain have an agreed-upon representation for the comments [41,50]. Furthermore, questions such as classifier configuration, parameter tuning or even interpretability [59] are still largely unanswered. Our proposal is to ensure greater clarity in result communication by incentivizing the reporting of all relevant parameters and metrics while assessing the potential of AutoML as an option to reduce the human bias [18,19,30].

The concluding challenge will revolve around achieving acceptance of all stakeholders which are primarily the community managers and the community members. In countries with codified freedom of speech many people—so far—perceive (semi-)automatic moderation as censorship and opinion dictatorship. Extant research tells us that people are more likely to trust and accept systems

they can understand [12,24,38,54,59]—especially in cases where they behave in an unexpected manner (e.g., by blocking a comment) [27]. The EU regulation on algorithmic transparency, which grants every user of an "intelligent system" the right to receive an explanation for each automated decision, will further increase the need for interpretable results [5,21,26,43]. However, the associated domain of explainable artificial intelligence [51] is still in its early stages, respectively in its infancy considering abusive language detection.

3 A Business Model Canvas for an Integrated Platform

The aforementioned challenges require the adoption of (semi-) automated comment moderation tools. Especially for smaller and medium sized organizations, which do not have the human resources to perform a post-moderation of all user comments, (semi-) automated comment moderation is essential. First successful attempts to deploy such a tool have been reported in the literature (e.g., [56]). However, most of the published solutions only address some of the four challenges outlined above. While some seminal papers like [41] handle up to three of the challenges, many others at best deal with two [2], leaving large parts of the overall issue unaddressed. Additionally, especially for SMEs, further issues arise. Not every website operator or newspaper organization is capable of developing a system of their own. Therefore, the question arises, how a viable service-oriented business model could look like that provides this analytic functionality also to SME customers.

In previous work [6], we have adopted the Business Model Canvas (BMC) by Osterwalder [42] to develop a business model for operating a central comment moderation platform. The BMC is a strategic management tool, which uses a visual chart of nine building blocks to describe a business' value propositions, infrastructure, customer and finances. The resulting canvas is depicted in Fig. 1, where the nine building blocks identify as follows (cf. [6]):

"The **Customer Segments** building block defines the different groups of people or organizations an enterprise aims to reach and serve" [42, p. 20]. In this case, the AaaS platform serves a segmented market, because it distinguishes between different types of customers with different needs and problems. There will be small, medium and large enterprise customers. These customer types differ in the amount of comments that need to be processed as well in the frequency of performing additional ad-hoc analyses.

The **Value Proposition** building block describes which value the platform offers the customers through products or services. The AaaS platform creates value for its differently sized customers in three ways: First, since manual moderation reflects a major cost factor, automatic comment processing reduces costs for the customers. Second, the platform supports its customers in processing comments in a structured and documented manner to ultimately improve the quality of comment moderation. Third, the platform allows its customers to reopen comment sections on topics which had to be closed before and, as such, resurrects and increases their interaction within the community.

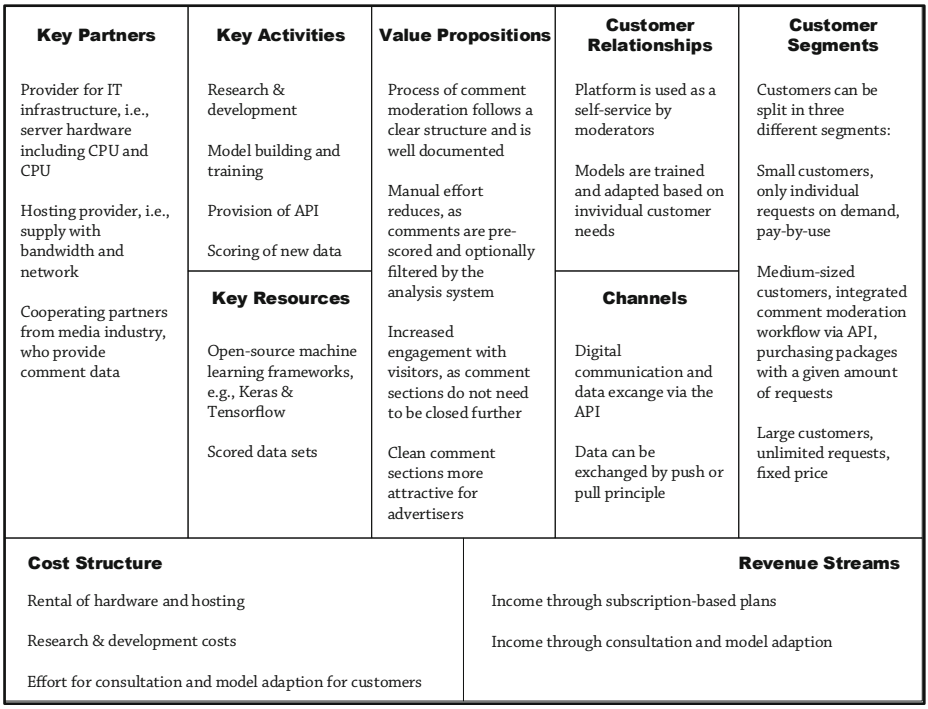

Fig. 1. Business Model Canvas for an integrated platform [6]

The **Channels** building block explains how a firm delivers its value proposition to their customers but also how it reaches new customers and communicates with existing ones. In this work we abstract from communication and marketing channels, as this is an issue of the exact instantiation of this conceptual business model. The delivery of the value proposition, which is the comment evaluation and the ability to moderate comments in an appropriate dashboard, is provided digitally. The data of the customers' own content management system and the AaaS platform is exchanged live through pre-defined application programming interfaces (API), which use push, pull and receive protocols.

The building block of **Customer Relationships** describes which kind of relationships the business maintains with its respective customer segments. In general, all customer segments (small to large customers) are provided with a self-service interface. They can submit the comments that need to be evaluated, receive and display the results and possibly manage them in the provided interface. On top of that, however, specific customers may want the trained evaluation models to be adapted to particular needs. This represents a second type of customer relationship that goes beyond the previously mentioned self-service infrastructure.

The **Revenue Streams** represent all incoming turnover that the business generates from its customers. Similar to the previous building block, the revenue

stream is twofold. The first includes consulting, adapting and also implementing on a customer basis; e.g., the previously mentioned customization of the evaluation models or an API-provision for individual content management systems. The second and major revenue stream represents the subscription model that the different customer segments utilize to receive evaluations for their comments.

The essential resources needed to implement the business model are reflected in the building block of the **Key Resources**. The AaaS platform makes use of a variety of machine learning methods, tools, frameworks and libraries to calculate the evaluation models. Many of the applied assets are open source or available under different accessible licenses. Besides, training data sets are needed. These data sets include real world comments annotated with labels and additional information that the machine learning algorithms leverage to extract their decision making patterns. These data sets can stem from partners, customers or might be even self-developed.

Key Activities. The key activities building block describes the most important tasks that the business needs to perform in order to deliver its business value to the customers. For our platform, the most important activity is the evaluation of comments submitted by the customers. To be able to do this, two secondary activities need to be performed. On the one hand, the APIs need to be implemented and provided beforehand. And on the other hand, the evaluation model needs to be trained, which is a continuous challenge, as language is not a static but evolving construct.

Key Partnerships include all the relations and strategic partnerships that enable the business model to function. Considering the previous two building blocks, a strategic partnership with the developers of the machine learning tools and frameworks is of importance. Through this, the platform can ensure a timely and continuous delivery of necessary updates. Similarly, data set providers can be important partners. For example, some customers might also be data deliverers at the same time and therefore the relationship should receive special attention. Another more basic, but as important, partnership is the IT infrastructure and hosting of the platform. The calculation and provision of the machine learning evaluation models requires great amounts of calculation power and the availability of the service must always be guaranteed.

Last but not least, the **Cost Structure** includes all the cost that accumulate by executing the business model. Of course, the previously mentioned infrastructure, hosting, and computing power incurs major costs on the business. Furthermore, research and continuous development on how to improve, adapt, and optimize the evaluation models are very important. And ultimately, the staff that executes the consulting and adaption actions with (future) customers must be noted here as well.

4 Building an IT Architecture

To address the previously outlined challenges and to implement the presented AaaS business model a suitable IT artifact is needed. Given the aaS approach

the platform needs to be web-based and multi-tenant-ready (#tenants=n) (each tenant being a media operator). As most media companies cater their content (and hence comments) to multiple systems (#systems=m) (ranging from proprietary CMS to social media platforms such as Facebook) the AaaS platform will have to support up to $n \times m$ interfaces. The concept of the platform is depicted in Fig. 2.

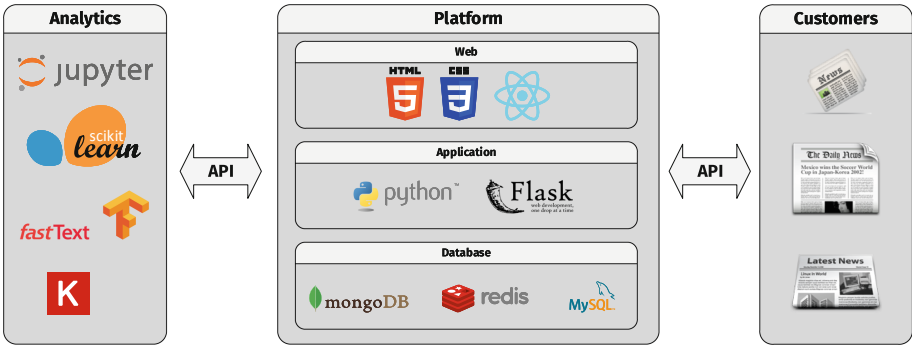

Fig. 2. Conceptualization of an integrated platform

Between these systems and the IT platform, data needs to be exchanged via Application Programming Interfaces (API). Here, two different data exchange patterns need to be distinguished: Following the *push principle*, the data sources actively push new user comments to the platform. This likely requires customization and/or programming on the side of the data source, as the software there needs to implement the API of the platform. An advantage of the push principle is the immediate availability of user comments on the platform, as the push activity can be triggered right after the comment was posted by the user. In contrast, the *pull principle* does not require programming on the data source's side but on the side of the platform. While this may be beneficial in case of closed-source software, where the website/weblog of the tenant cannot be adapted to implement the platform's API, the downside is that new comments are only periodically imported by the platform (e.g., every couple of minutes) and, hence, are not immediately available for processing.

The platform itself requires a database system for storing and processing user comments, where both `mongoDB` and `MySQL` are used. An additional Redis server is used for internal event processing. The web application's front-end uses modern technologies, namely HTML5 and CSS3 and is built using the web framework React. For the middleware (or application layer) we selected the Python-based framework `flask`[1]. The decision to go for a Python-based backend originates in the primarily Python-based analytics component. Hence, going for

[1] https://www.palletsprojects.com/p/flask/.

a full Python stack is supposed to reduce integration issues while simultaneously increasing the maintainability.

For the implementation of different classification algorithms and machine learning models, plenty of open-source tools and libraries are available. For development purposes, `jupyter notebook` is available. We use `fastText` to convert our textual input data into machine-processable word embeddings, which can be further utilized in machine learning models, e.g., by using `scikit-learn`. The libraries `Keras` and `TensorFlow` are included to add the capability to create deep-learning models in addition to the traditional ML models offered by `sklearn`. As an additional benefit, these specialist libraries do not only support the traditional CPU-based execution but the use of GPUs and TPUs which have been found more potent for such classification tasks.

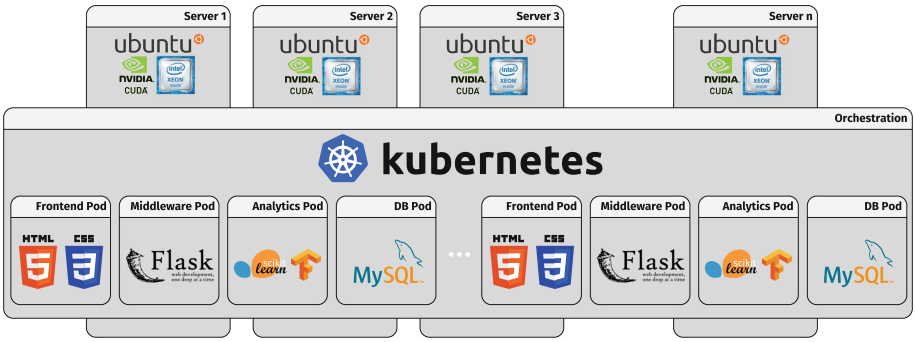

Fig. 3. IT architecture of an integrated platform

Based on the conceptualization of an integrated platform providing AaaS (cf. Fig. 2), we derive a suitable IT architecture. Since the platform supports different tenants, new data sources may be added or deleted at any time. Similarly, the amount of users as well as the amount of comments these users are posting may change over time. Consequently, the platform needs to be scalable to accomodate the changing computational load. To account for this, we have to implement our AaaS platform as virtualizable microservices. Nowadays, containerization is the most common virtualization, since it is light-weight and easy to deploy. Here, every microservice runs isolated in a separate container using a containerization engine like `Docker` or `containerd`. For orchestration of containers, i.e., the management of containers running and scaling the amount of running containers to the actual computation demand, usually a separate tool is used. We have chosen `Kubernetes`[2] for this purpose, since it is both open-source and industry-stable. While we exclusively discussed the software-stack so far, some of the included tools set limitations regarding the usable hardware, as, e.g., `tensorflow` can only compute on GPUs with `CUDA`, restricting our servers

[2] https://kubernetes.io/.

to use nvidia chips[3]. As the base operating system we chose Ubuntu, which is currently the most common Linux distribution for web services [58]. `Kubernetes` integrates all physical servers into a single cluster, where services can be executed as "pods" (technically containers). Figure 3 visualizes all involved components. The frontend pod serves the website's front-end, i.e., the HTML, CSS files etc. to the client's browser. The middleware pod does the processing of web requests in the back-end and maps all HTTP requests to an actual service. The analytics pod performs background tasks related to data processing like, for instance, the training of machine learning models with new data. Lastly, the DB pod serves and manages the database instance for data storing. All pods can be replicated, i.e, they can be executed multiple times, which allows the platform to scale to the actual computation demand.

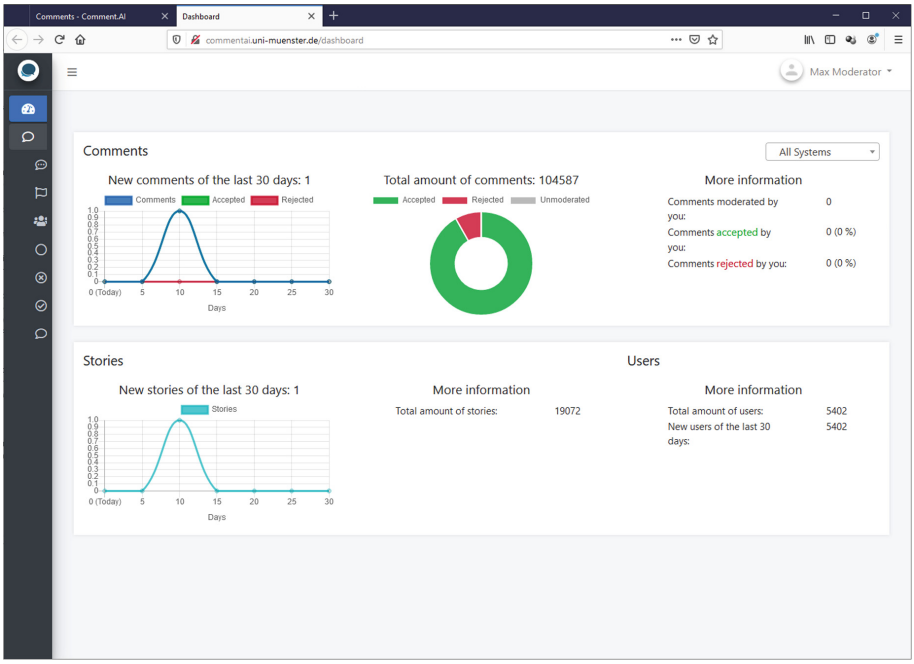

Fig. 4. Dashboard of the moderation platform

Currently, we have developed a prototypical implementation of the designed integrated platform according to the principles mentioned above. Figure 4 shows a screenshot of that implementation. In this example, we have imported user comments from two different sources: A Wordpress blog and the proprietary

[3] The decision for Intel CPUs is acknowledging Intel's leading market position for server processors.

CMS of a large German newspaper. The dashboard shows the amount of comments that were composed recently and visualizes the share of rejected comments among the amount of total comments. Furthermore, it provides an overview of newly created stories/articles as well as the development of users participating in the discussions. Combining this diverse information, the dashboard should cater to the needs of community managers as well as their superior (social media) managers.

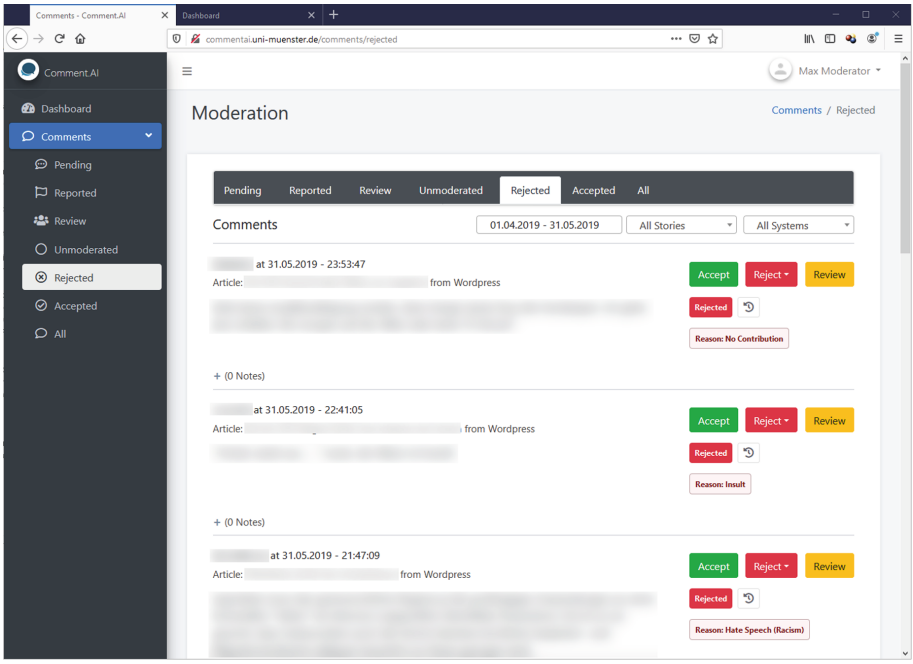

Fig. 5. User interface for comment moderation

Moderators (e.g., social media managers) can approve or reject comments as shown in Fig. 5. Comments are pre-moderated by algorithmic means which is reflected in the different queues depicted in the horizontal tab bar above the list of comments. Here, moderators can quickly jump into different queues to focus on comments which need attention. When comments are rejected, one rejection reason can be selected out of a list of pre-defined reasons. These pre-defined reasons are subject to specification by the site administrator. Besides approving or rejecting comments, moderators can also forward comments to other moderators by requesting a review. This supports discussing comments with colleagues in cases where decisions are hard to take. An exemplary, simplified moderation process with the AaaS platform is depicted in Fig. 6.

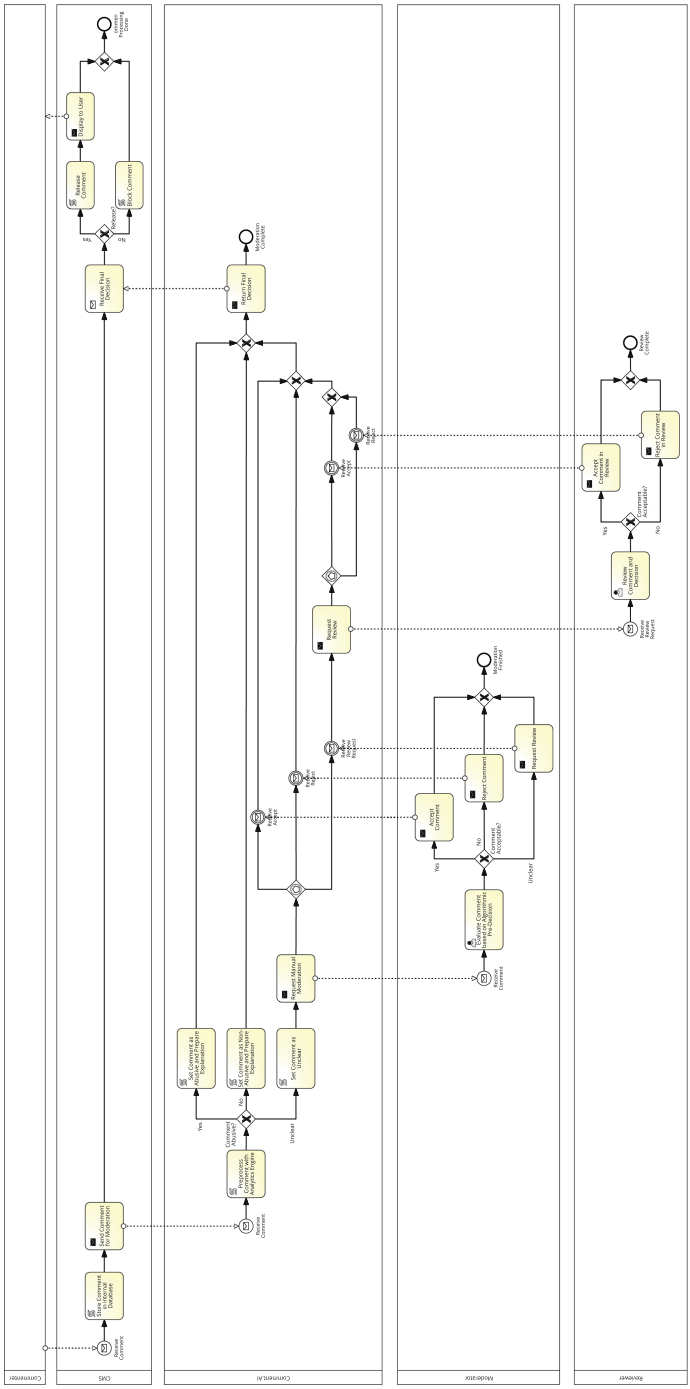

Fig. 6. Exemplary moderation process (simplified)

5 Conclusion and Outlook

We pointed out that there is an industry need for tool support in user-generated comment moderation. The concept of (semi-) automatic comment moderation provides a possible solution for this problem. Even though research on abusive language detection has gained much focus and several researchers have worked on data collection, data annotation and building machine learning models, existing tools still have a long way to go. Especially in the context of SMEs, these technologies are not yet used in practice. We have identified four different challenges that hinder the adoption in practice. To overcome these challenges, our goal was to develop an AaaS platform, which provides comment AI services to newspaper and online community providers of different sizes.

Within our initial study we had analyzed recent literature on online comment moderation systems and had developed a business model for an AaaS platform (cf. [6]). In the paper at hand, we have taken the business model one step further towards implementation, as we have conceptualized the platform in a manner to integrate data from different media organizations. Additionally, we have described an IT architecture suitable for operating such a platform and developed a prototypical implementation. Both the platform itself as well as the IT architecture behind it are based on modern open-source technologies and enable flexible scaling on demand. Finally, by following the `Kubernetes` approach of operating a comment processing platform, our system is able to provide analytics as a service.

The next step in our research agenda is to deploy and test our AaaS platform in practice. To achieve this, we aim to implement the required API endpoints within the local comment moderation systems of our project partners (several newspapers of various size throughout Germany). Once implemented, these field studies will enable us to continuously improve our AaaS platform as well as the provided service. Additionally, insights into the required transparency of the system and the accompanying trust in and acceptance of the system can be evaluated from a first hand perspective for the first time.

Acknowledgements. The research leading to these results received funding from the federal state of North Rhine-Westphalia and the European Regional Development Fund (EFRE.NRW 2014–2020), Project: MODERAT! (No. CM-2-2-036a).

References

1. van Aken, B., Risch, J., Krestel, R., Löser, A.: Challenges for toxic comment classification: an in-depth error analysis. In: Proceedings of the Second Workshop on Abusive Language Online, ALW2, Brussels, Belgium, pp. 33–42 (2018)
2. Badjatiya, P., Gupta, S., Gupta, M., Varma, V.: Deep learning for hate speech detection in tweet. In: Proceedings of the 26th International Conference on World Wide Web Companion, WWW 2017, Companion, Perth, Australia, pp. 759–760 (2017)
3. Bilton, R.: Why some publishers are killing their comment sections (2014). https://digiday.com/media/comments-sections/

4. Boberg, S., Schatto-Eckrodt, T., Frischlich, L., Quandt, T.: The moral gatekeeper? Moderation and deletion of user-generated content in a leading news forum. Media Commun. **6**(4), 58–69 (2018)
5. Brunk, J., Mattern, J., Riehle, D.M.: Effect of transparency and trust on acceptance of automatic online comment moderation systems. In: Proceedings of the 21st IEEE Conference on Business, Informatics, Moscow, Russia, pp. 429–435 (2019)
6. Brunk, J., Niemann, M., Riehle, D.M.: Can analytics as a service save the online discussion culture? - the case of comment moderation in the media industry. In: Proceedings of the 21st IEEE Conference on Business Informatics, CBI 2019, Moscow, Russia, pp. 472–481 (2019)
7. Burnap, P., Williams, M.L.: Cyber hate speech on Twitter: an application of machine classification and statistical modeling for policy and decision making. Policy Internet **7**(2), 223–242 (2015)
8. Chatzakou, D., Kourtellis, N., Blackburn, J., De Cristofaro, E., Stringhini, G., Vakali, A.: Mean birds: detecting aggression and bullying on Twitter. In: Proceedings of the 2017 ACM Web Science Conference, WebSci 2017, Troy, New York, USA, pp. 13–22 (2017)
9. Chen, H., Mckeever, S., Delany, S.J.: Harnessing the power of text mining for the detection of abusive content in social media. In: Angelov, P., Gegov, A., Jayne, C., Shen, Q. (eds.) Advances in Computational Intelligence Systems. AISC, vol. 513, pp. 187–205. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-46562-3_12
10. Chen, Y., Zhou, Y., Zhu, S., Xu, H.: Detecting offensive language in social media to protect adolescent online safety. In: Proceedings of the 2012 ASE/IEEE International Conference on Social Computing, 2012 ASE/IEEE International Conference on Privacy, Security, Risk Trust, SOCIALCOM-PASSAT 2012, Amsterdam, Netherlands, pp. 71–80 (2012)
11. Cheng, J.: Report: 80 percent of blogs contain "offensive" content (2007). https://arstechnica.com/information-technology/2007/04/report-80-percent-of-blogs-contain-offensive-content/
12. Cramer, H., Wielinga, B., Ramlal, S., Evers, V., Rutledge, L., Stash, N.: The effects of transparency on perceived and actual competence of a content-based recommender. In: Proceedings of the Semantic Web User Interaction: Workshop CHI 2008 Exploring HCI Challenges, SWUI 2008, Florence, Italy, pp. 1–10 (2008)
13. Davidson, T., Warmsley, D., Macy, M., Weber, I.: Automated hate speech detection and the problem of offensive language. In: Proceedings of the Eleventh International Conference on Web Social Media, ICWSM 2017, Montreal, Canada, pp. 512–515 (2017)
14. Diakopoulos, N.: Picking the NYT picks: editorial criteria and automation in the curation of online news comments. #ISOJ, Off. Res. ISOJ J. **5**(1), 147–166 (2015)
15. Dinakar, K., Reichart, R., Lieberman, H.: Modeling the detection of textual cyberbullying. In: Social Mobile Web, Paper from 2011 ICWSM Workshop, ICWSM 2011, Barcelona, Spain, pp. 11–17 (2011)
16. Djuric, N., Zhou, J., Morris, R., Grbovic, M., Radosavljevic, V., Bhamidipati, N.: Hate speech detection with comment embeddings. In: Proceedings of the 24th International Conference on World Wide Web, WWW 2015 Companion, Florence, Italy, pp. 29–30 (2015)
17. Etim, B.: The Most Popular Reader Comments on the Times (2015). https://www.nytimes.com/2015/11/23/insider/the-most-popular-reader-comments-on-the-times.html

18. Feurer, M., Klein, A., Eggensperger, K., Springenberg, J.T., Blum, M., Hutter, F.: Efficient and robust automated machine learning. In: Proceedings of the 28th International Conference on Neural Information Processing Systems, NIPS 2015, Montreal, Canada, pp. 2755–2763 (2015)
19. Feurer, M., Klein, A., Eggensperger, K., Springenberg, J.T., Blum, M., Hutter, F.: Auto-sklearn: efficient and robust automated machine learning. In: Hutter, F., Kotthoff, L., Vanschoren, J. (eds.) Automated Machine Learning. TSSCML, pp. 113–134. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-05318-5_6
20. Fišer, D., Erjavec, T., Ljubešić, N.: Legal framework, dataset and annotation schema for socially unacceptable online discourse practices in Slovene. In: Waseem, Z., Chung, W.H.K., Hovy, D., Tetreault, J. (eds.) Proceedings of the First Workshop on Abusive Language Online, ALW1, Vancouver, Canada, pp. 46–51 (2017)
21. Fleischmann, K.R., Wallace, W.A.: A covenant with transparency. Commun. ACM **48**(5), 93–97 (2005)
22. Fortuna, P., Nunes, S.: A survey on automatic detection of hate speech in text. ACM Comput. Surv. **51**(4), 1–30 (2018). https://doi.org/10.1145/3232676
23. Gardiner, B., Mansfield, M., Anderson, I., Holder, J., Louter, D., Ulmanu, M.: The dark side of Guardian comments (2016). https://www.theguardian.com/technology/2016/apr/12/the-dark-side-of-guardian-comments
24. Gefen, D., Karahanna, E., Straub, D.W.: Trust and TAM in online shopping: an integrated model. MIS Q. **27**(1), 51–90 (2003)
25. Gelber, K.: Differentiating hate speech: a systemic discrimination approach. Crit. Rev. Int. Soc. Polit. Philos. 1–22 (2019)
26. Goodman, B., Flaxman, S.: European union regulations on algorithmic decision-making and a "Right to Explanation". AI Mag. **38**(3), 50 (2017)
27. Gregor, S., Benbasat, I.: Explanations from intelligent systems: theoretical foundations and implications for practice. MIS Q. **23**(4), 497–530 (1999)
28. Hine, G.E., et al.: Kek, cucks, and god emperor trump: a measurement study of 4chan's politically incorrect forum and its effects on the web. In: Proceedings of the 11th International Conference Web Social Media, ICWSM 2017, Montral, Canada, pp. 92–101 (2017)
29. Howe, J.: The rise of crowdsourcing. Wired Mag. (2006)
30. Hutter, F., Kotthoff, L., Vanschoren, J. (eds.): Automated Machine Learning: Methods, Systems, Challenges. Springer, Heidelberg (2018, in press). http://automl.org/book
31. Köffer, S., Riehle, D.M., Höhenberger, S., Becker, J.: Discussing the value of automatic hate speech detection in online debates. In: Tagungsband Multikonferenz Wirtschaftsinformatik 2018. MKWI 2018, Lüneburg, Germany (2018)
32. Kolhatkar, V., Taboada, M.: Constructive language in news comments. In: Proceedings of the First Workshop on Abusive Language Online, ALW1, Vancouver, Canada, pp. 11–17 (2017)
33. Lee, Y., Yoon, S., Jung, K.: Comparative studies of detecting abusive language on Twitter. In: Proceedings of the Second Workshop on Abusive Language Online, ALW2, Brussels, Belgium, pp. 101–106 (2018)
34. Lewis, S.C., Holton, A.E., Coddington, M.: Reciprocal journalism: a concept of mutual exchange between journalists and audiences. J. Pract. **8**(2), 229–241 (2014)
35. Lukyanenko, R., Parsons, J., Wiersma, Y., Wachinger, G., Huber, B., Meldt, R.: Representing crowd knowledge: guidelines for conceptual modeling of user-generated content. J. Assoc. Inf. Syst. **18**(4), 297–339 (2017)

36. Mansfield, M.: How we analysed 70m comments on the Guardian website (2016). https://www.theguardian.com/technology/2016/apr/12/how-we-analysed-70m-comments-guardian-website
37. Mathur, P., Sawhney, R., Ayyar, M., Shah, R.R.: Did you offend me? Classification of offensive Tweets in Hinglish language. In: Proceedings of the Second Workshop on Abusive Language Online, ALW2, Brussels, Belgium, pp. 138–148 (2018)
38. McKnight, D.H., Choudhury, V., Kacmar, C.: The impact of initial consumer trust on intentions to transact with a web site: a trust building model. J. Strateg. Inf. Syst. **11**(3–4), 297–323 (2002)
39. Mehdad, Y., Tetreault, J.: Do characters abuse more than words? In: Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue, SIGDIAL 2016, Los Angeles, CA, USA, pp. 299–303 (2016)
40. Niemann, M., Riehle, D.M., Brunk, J., Becker, J.: What is abusive language? Integrating different views on abusive language for machine learning. In: Grimme, C., Preuss, M., Takes, F.W., Waldherr, A. (eds.) MISDOOM 2019. LNCS, vol. 12021, pp. 59–73. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-39627-5_6
41. Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., Chang, Y.: Abusive language detection in online user content. In: Proceedings of the 25th International Conference on World Wide Web, WWW 2016, Montreal, Canada, pp. 145–153 (2016)
42. Osterwalder, A., Pigneur, Y.: Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers. Wiley, Hoboken (2010)
43. Owotoki, P., Mayer-Lindenberg, F.: Transparency of computational intelligence models. In: Bramer, M., Coenen, F., Tuson, A. (eds.) SGAI 2006, pp. 387–392. Springer, London (2007). https://doi.org/10.1007/978-1-84628-663-6_29
44. Papacharissi, Z.: Democracy online: civility, politeness, and the democratic potential of online political discussion groups. New Media Soc. **6**(2), 259–283 (2004)
45. Park, J.H., Fung, P.: One-step and two-step classification for abusive language detection on Twitter. In: Proceedings of the First Workshop on Abusive Language Online, ALW1, Vancouver, Canada, pp. 41–45 (2017)
46. Pavlopoulos, J., Malakasiotis, P., Androutsopoulos, I.: Deep learning for user comment moderation. In: Proceedings of the First Workshop on Abusive Language Online, ALW1, Vancouver, Canada, pp. 25–35 (2017)
47. Plöchinger, S.: Über den Hass (2016). http://ploechinger.tumblr.com/post/140370770262/%C3%BCber-den-hass
48. Pöyhtäri, R.: Limits of hate speech and freedom of speech on moderated news websites in Finland, Sweden, the Netherlands and the UK. Annales–Series historia et sociologia izhaja štirikrat letno **24**(3), 513–524 (2014)
49. Reynolds, K., Kontostathis, A., Edwards, L.: Using machine learning to detect cyberbullying. In: Proceedings of the 10th International Conference on Machine Learning and Applications and Workshops, ICMLA 2011, Honolulu, Hawaii, USA, pp. 241–244 (2011)
50. Sahlgren, M., Isbister, T., Olsson, F.: Learning representations for detecting abusive language. In: Proceedings of the Second Workshop on Abusive Language Online, ALW2, Brussels, Belgium, pp. 115–123 (2018)
51. Samek, W., Wiegand, T., Müller, K.R.: Explainable artificial intelligence: understanding, visualizing and interpreting deep learning models. ITU J. ICT Discov. **1**(1), 39–48 (2017)
52. Schmidt, A., Wiegand, M.: A survey on hate speech detection using natural language processing. In: Ku, L.W., Li, C.T. (eds.) Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media, SocialNLP 2017, Valencia, Spain, pp. 1–10 (2017)

53. Serrà, J., Leontiadis, I., Spathis, D., Stringhini, G., Blackburn, J.: Class-based prediction errors to categorize text with out-of-vocabulary words. In: Proceedings of the First Workshop on Abusive Language Online, ALW1, Vancouver, Canada, pp. 36–40 (2017)
54. Sinha, R., Swearingen, K.: The role of transparency in recommender systems. In: Extended Abstracts on Human Factors in Computing Systems, CHI 2002, Minneapolis, MN, USA, pp. 830–831 (2002)
55. Sood, S.O., Antin, J., Churchill, E.F.: Using crowdsourcing to improve profanity detection. In: AAAI Spring Symposium Series, Palo Alto, CA, USA, pp. 69–74 (2012)
56. Švec, A., Pikuliak, M., Šimko, M., Bieliková, M.: Improving moderation of online discussions via interpretable neural models. In: Proceedings of the Second Workshop on Abusive Language Online, ALW2, Brussels, Belgium, pp. 60–65 (2018)
57. The Coral Project Community (2016). https://community.coralproject.net/t/shutting-down-onsite-comments-a-comprehensive-list-of-all-news-organisations/347
58. W3Techs: Usage Statistics and Market Share of Linux for Websites (2020). https://w3techs.com/technologies/details/os-linux
59. Wang, C.: Interpreting neural network hate speech classifiers. In: Proceedings of the Second Workshop on Abusive Language Online, ALW2, Brussels, Belgium, pp. 86–92 (2018)
60. Wulczyn, E., Thain, N., Dixon, L.: Ex Machina. In: Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, pp. 1391–1399 (2017)
61. Yin, D., Xue, Z., Hong, L., Davison, B.D., Kontostathis, A., Edwards, L.: Detection of harassment on web 2.0. In: Proceedings of the Content Analysis in the WEB, CAW 2.0, Madrid, Spain, pp. 1–7 (2009)