Software Engineering Case Study Analysis on Automated Content Moderation

Christopher Morcom

CS 487

Computer Science Department

Illinois Institute of Technology

Chicago, Illinois 60616

morcom01@gmail.com

***Abstract – This paper reports on a case study that automates content moderation to an extent, describing it from a software engineering perspective that outlines specific challenges, concerns, and best practices of the system’s development.***

*Index Terms – Software (SW) Engineering, Moderator, Semi-Automation*

**[1] Introduction**

The internet is a massive place containing petabytes of data that has been reviewed by man or machine, and sometimes both. As the rate of data being posted online drastically increases, we find an increasing need to effectively moderate user-generated content. This study talks about [1] from a software engineer’s perspective.

This report is organized as follows: Section 2 summarizes the results of the case and whether is was quantifiably “successful” from a SW Engineering perspective. Section 3 describes the specific challenges noted by their group during each phase of development. We conclude Section 3 in an overall context and how it relates to SW Engineering before moving to Section 4. The next section talks about areas of concern and “best practices for automation development. Section 5 discusses one of the tools used for automation development. Finally, we state our overall findings and opinions in or conclusions in section 6.

**[2] Case Study Summary**

With issues in the volume, velocity, and variety of data being shared over social media, the need for automating data processing emerges. The study [1] talks about how current machine learning systems process data with a “human-in-the-loop” approach which is where algorithms will occasionally consult humans for feedback and correction. The issue with this is that humans are often presented data out of context by the system, making the current systems that are data-driven redundant since context of identified information is lost.

The study mentions the “human-***is***-the-loop” approach to “semi-automate a manual content moderation workflow where human moderators take a dominant role.” The study analyzes content generated about natural disasters, conflicts, and benign data for testing. Rather than the former method, where human moderators must gather context manually, the study’s prototype uses machine learning to seek the context and meaning of data and “provide human analysts with suggestions regarding the relevance and categorization of collected data.

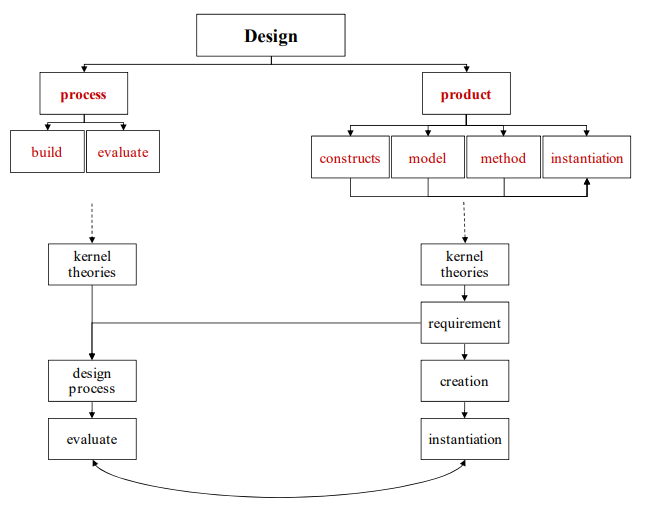
During testing, the results found that although the system accomplished in providing the user relevant information, with relatively low precision since the amount of information returned is too high, the system needs more focus on categorizing information by priority, where more relevant information is displayed first. Additionally, most organizations tend to favor frequently used sources of information, which results in an issue of building trust and need for this new system.

Since social media is mostly a subjective process, quality control is also a concern before data is sent to users for review. For instance, information about a natural disaster would need to have a different context from a conflict. However, on the positive side, the prototype successfully accomplishes its goal with “improved moderation quality and higher flexibility” while being extremely compatible with users and how their work is completed.

**[3] Challenges Throughout Development**

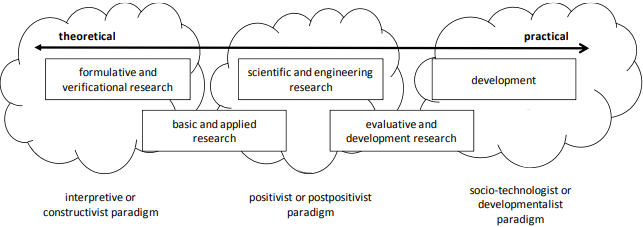
This section will break down the difficulties noted by the authors during each stage of development.

For the research stage, the writers used the Design Science Research paradigm *(fig. 1)* which is commonly used for Information Systems (IS) in solving relevant problems, prototyping, and contributing to the knowledge of socio-technical artifacts [2].



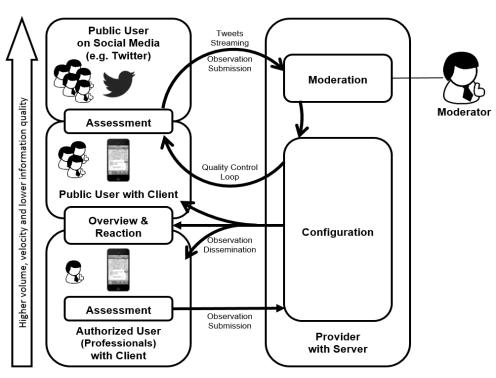
***Figure 1:* DSR Framework Diagram**

Furthermore, the study uses IS research paradigms to move a theoretical knowledge base to practical development *(fig. 2)*. In contrast to the standard model, the writers use data-driven research to substantiate this development. Overall, the only real challenge here is data validity and quantity, which may be seen in later stages – a common issue in this stage in terms of SW engineering.



***Figure 2:* Perception ofResearch Objectives and Methods**

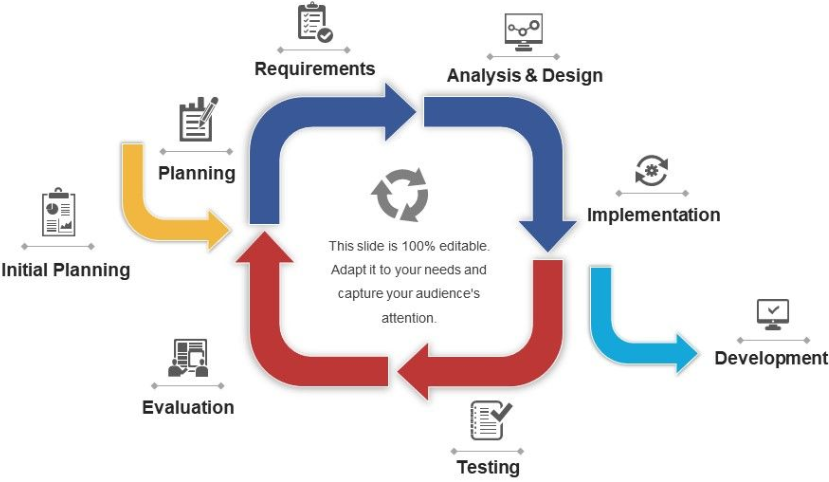
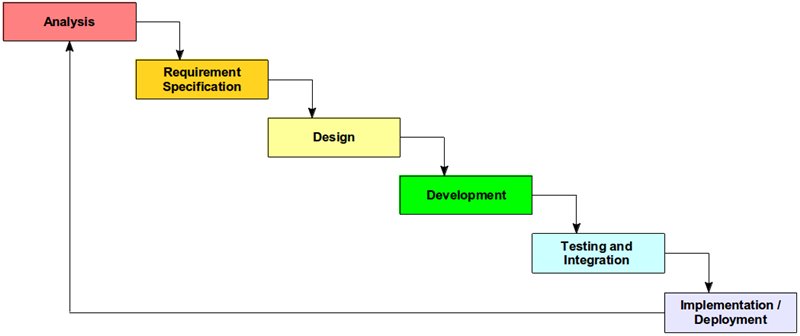
During requirements specification, the writers focus heavily on workflow specification. Using previous works on automatic, hybrid, and manual workflow, the writers were unable to find a way to define requirements to fully automate their system. The writers faced a huge challenge in defining requirements that would not bottleneck the system on delivery to the human moderator, while maintaining the purity of the data. This led them to define a hybrid system, that is based on the manual workflow of the GDACSmobile system they analyzed *(fig 3)* [1]. The study uses four defined requirements for the new moderation workflow of their system. In summary, the largest issue here was defining specific requirements for the system that allow finding viable candidates for tools that help in creating the system.



***Figure 1:* GDACSmobile Information Flow Model**

Little is known about the design and development stages; however, the study reflects the challenges in each through a technical standpoint. The limitations of the software, at a fault of not enough scope defined in the requirements were that the system has no measure of trustworthiness, it returns too much data to its human users, with low precision, and the data is not sorted based on relevance/context, nor is a requirement for this defined. In terms of SW Engineering, this is a common issue where the requirements need to be revisited due to a faulty or not-good-enough design.

Testing and Integration of the system was extensive, as the writers had a predefined set of data that this system would be trained and tested on (natural disasters). Other than the challenges outlined in the paragraph above, there were no additional challenges



***Figure 3:* Waterfall Model (Left)[3] and Iterative Process Model (Right)[4]**

defined. However, in the larger context of SW Engineering, on challenge may be that the test data is not wholly encompassing of all test cases. It is also hard to define test cases for systems of social context.

Finally, the Implementation and deployment (maintenance) stage of SW development was less technical. In the study, the writers talk about probable ways to deploy this system but were turned down by larger social media enterprises. The study says it is because the system is not “trusted” to be effective by potential users. In the broad context of SW Engineering, this is equivalent to the user not being satisfied with the system and the project needing another iteration before evaluation.

Overall, the system seems to follow the Waterfall Methodology, while simultaneously running into common issues of software engineering, causing some backtracks in the model to the previous step, which later was a driving factor in requiring the project to go into another iteration of development.

**[4] Areas of Concern and “Best Practices” for Software Development**

Areas of Concern:

* Time and Cost
* Good Human-Computer Interaction (HCI)
* Ease of use
* Customer Preferences
* Future Feasibility

Best Practices:

* Periodic customer feedback
* Defensive designing
* Good, in-depth iterations
* Using Validated Test Data and Tools/Resources
* Good Code reusability/integration into other systems

**[5] Tools for Automation Development**

Mention GDACSmobile and how it works with automation development in terms of SW Engineering

**[6] Conclusions**

To be written

**[7] References**

[1] <https://pdfs.semanticscholar.org/2223/f7245f7c310db2c4a24e6e4603c85936d460.pdf>

[2] <https://pdfs.semanticscholar.org/7c31/a4019ab19db0123c90ba72250bc99d80ace5.pdf>

[3] <https://sysgears.com/articles/agile-vs-waterfall-development-model/>

[4] <https://www.slideteam.net/iterative-process-diagram.html>

[5] Dummy Source for any more sources