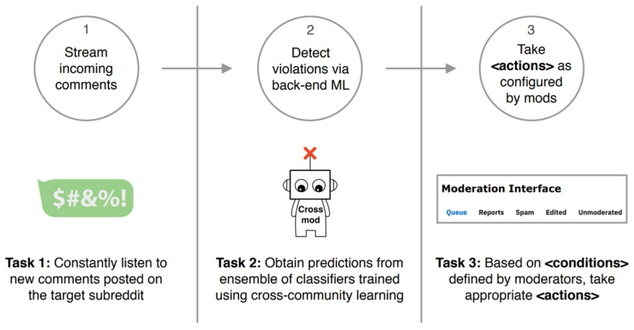
Crossmod - A Real-time AI-backed Sociotechnical Moderation System for Reddit

Weiji Li*, Aditya Shylesh*, Dr. Eshwar Chandrasekharan

Introduction:

Crossmod is a real-time **AI-backed** sociotechnical moderation system for Reddit. It uses ML models to detect violated and toxic comments on Reddit in real time. Crossmod will stream incoming comments on some subreddits, detect violations via back-end ML models automatically and take actions according to manual configuration. It is built to extend the capabilities of moderators, while also fitting into their existing workflows.

As a research project, Crossmod is focuses on **Human-computer Interaction** and **Artificial Intelligence.** Crossmod tries to improve moderation by **detecting undesirable content which is currently undetected by existing automated moderation infrastructure**. It was initially designed as a research system, as described in the paper authored by our mentor Eshwar Chandrasekharan — ***Crossmod: A Cross-Community Learning-based System to Assist Reddit Moderators***[1]. Over the past two semesters, our team collaborated to refactor the codebase for Crossmod transforming it from a research system, repackaging it as an open-source API service, deploying it on a cloud platform, and analysing data collected from the deployment.



Methods:

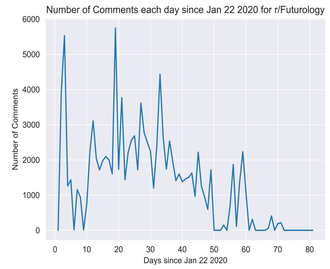
Crossmod is primarily written in **Python**. It is built with the Python Flask web framework and the team aims to create an idiomatic, coherent and maintainable codebase (which is **open-source**, and publicly available on GitHub, at github.com/ceshwar/crossmod).

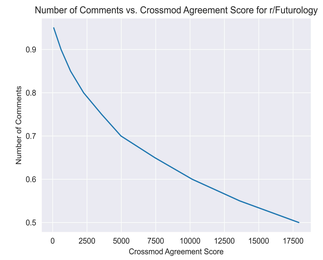
Crossmod's primary goal is to reduce mean moderation time, which is the period of time between when an undesirable comment is posted in a subreddit and when it is removed by moderators. Crossmod achieves this goal by reporting undesirable comments to human moderators on a subreddit as soon as they are posted. The requirements of the human moderators were the main concerns for the team when designing the system. Crossmod is architected as a multi-component web system: it consists of a public-facing API service (to be used by moderators) that communicates with a machine learning backend. The machine learning component is comprised of an ensemble of 108 Fasttext classifiers that are used to assign "would remove" or "would not remove" labels to comments that are ingested real-time from a subreddit.

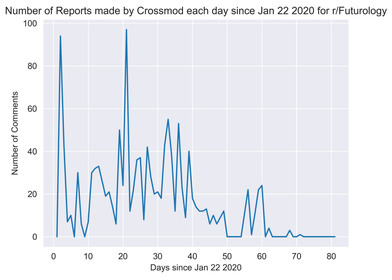
It can be configured using an intuitive *If This Then That* (IFTTT) format where moderators just create chains of simple conditional statements to trigger moderation actions, providing moderators a rich set of options they can use to control and configure the system to meet their subreddit-specific needs.

Data analysis:

Crossmod has been listening in on comments from **r/Futurology** for around 6 months at this point and our team has been interested in several metrics such as the average rate of comments posted in the subreddit each day, the total number of comments with a given agreement score (graphs show below) and the number of comments flagged by Crossmod and human moderators. These metrics help us decide what parameters are important when configuring Crossmod to minimize the average response time of r/Futurology moderators. For example, we used metrics such as daily number of reports by Crossmod, the number of comments posted on the subreddit each day and the total number of comments for a given "classification score" (i.e. number of classifiers that would remove) to determine how to make Crossmod more useful to moderators.







Development

Initially, Crossmod was a **local script** running on a laptop. It implemented the basic functionality described in the research paper. Our team has made many improvements to this, and has developed a much more **feature complete web system**.

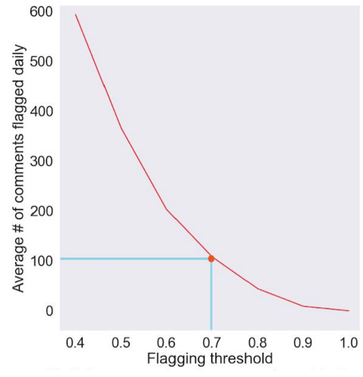
- Migrated from the local script to an AWS web server; this was done to make use of cloud compute and storage to provide greater availability and up-time for the system.
- Added a database management system and designed a schema to store previously seen comments, human moderator actions, settings and any other auxiliary data required.
- Refactored codebase from a single script into a more capable and robust web system
- Packaged our classification back-end system in an easy-to-use API to help moderators and other developers design more tools that interface with Crossmod.
- Containerized application using Docker to simplify the initial setup workflow and also to enable faster deployment and scaling.
- Added a background queueing system in the codebase to simplify periodic tasks such as metric generation and querying Reddit for comment status.
- Created a more user-friendly web interface for configuring various parameters used to monitor subreddits and managing access to the system
- Wrote comprehensive setup tutorials and documentation to simplify the process of setting up and running Crossmod for new users.
- Implemented dynamic retraining based on the data we collected; Provided interface for users to train new models based their configuration(i.e. time period, parameters)

Result:

Since its conception in our mentor's research paper, **Crossmod** has grown in scope, both in its application as a moderation assistance system and as a software engineering project. Crossmod has grown from a script running the machine learning classifiers running on a laptop to a web system with a publicly accessible API and on-demand, real-time moderation capabilities.

Our research team has built functionality to listen in on multiple subreddits that can be easily configured through a web interface, created a website that can be used to learn about Crossmod's functionality and documented all our open-source code that is also publicly accessible. We have also gained a lot of insight into how Reddit comments are posted and moderated by studying the data we collect about the comments posted on the subreddits we listen in on, which helps tailor our moderation assistance according to real-time feedback from human moderators. For example, we used our data to determine that the rate of reports produced by Crossmod was most useful to moderators when around 70% of classifiers regarded a given comment as "to be removed " (figure below).

Average number of comments flagged by Crossmod on a daily-basis different flagging thresholds



Conclusion:

Our research was mainly deploying the idea from the paper into a usable full-featured real-time system. Although we started our project by merely reporting toxic comments on Reddit, we have at this point built out what is nearly a product service: a real-time moderation assistance service capable of handling the moderation needs of multiple subreddits. There is still a lot left to be done for Crossmod as there are many aspects of the system that we believe require improvement or further experimentation. From building better machine-learning models in the back-end, to collecting and determining more metrics important for efficient moderation we hope we can continue making Crossmod an even more useful tool to assist moderators.

Reference:

1. Eshwar Chandrasekharan, Chaitrali Gandhi, Matthew Wortley Mustelier, and Eric Gilbert. 2019. Crossmod: A Cross-Community Learning-based System to Assist Reddit Moderators. Proc. ACM Hum.-Comput. Interact. 3, CSCW, Article 174 (November 2019), 30 pages. https://doi.org/10.1145/3359276