Introduction

Many goals in NLP focus on analyzing a chunk of text, and making some sort of prediction or identification from it. In the CS742 course, we were given two articles [GK10] and [ana09] to read, and come up with interesting research proposals.

[ana09] acknowledges the failure of traditional moderation methods in ensuring productive conversation on forum posts and threads. The article explores changes to traditional moderation systems, and puts forth the idea of evaluating conversations in threads as an indicator of engagement within the online community, instead of just looking at comments.

While several interesting research questions were proposed, due to deadline constraints, we will only look at the following problem.

Research problem

The core question follows: *Is it possible to predict how a conversation thread will turn out, based on the first few conversations between users, using a two comment dyad or more?*

Here, we will first look at seeing if it is possible to predict the number of comments a particular thread receives based on the first few posts, as well as how productive a thread could be.

Our analysis will be done using data from BC3's blog corpus, in particular the Slashdot blog conversations.

Proposed techniques

We looked at several papers given by Professor Lee, notably [BKLDNM13] and

As we are using data from Slashdot conversations, the techniques in these papers may not be directly applicable, but we will see if we can build on the ideas.

For example, [BKLDNM13] looks at data from Facebook and Wikipedia pages, but one of the core metrics the paper uses is the "links" between users[1]. We cannot necessarily know if any two posters on Slashdot have a link with each other.

[1] For Facebook, a link exists between two users who are Facebook friends, and for Wikipedia, a link exists between two editors if one posts or comments on another editor's talk page.

We will start from scratch, doing exploratory data analysis, and look at several metrics such as:

1. Sentiments in text, by using `textblob` in Python. While it is true that there are moderator labels on the posts, the majority of posts are unclassified, as the following output from Python shows.

   ```
   {'Troll': 14932, 'Funny': 40672, 'None': 464104, 'Flamebait': 7456,
   'Redundant': 4792, 'Offtopic': 11384, 'Informativ': 40188, 'Interestin':
   50168, 'Insightful': 73864}
   ```

2. Number of posts per user.
3. Length of post by words, sentences, and characters.
4. Time between posts.

amongst others.

Processing and selection of data

While it might seem tempting to focus on a subset of data which has mostly been classified by a moderator, it appears that for all threads, the dominating post is classified as `None`.

Preliminary results

What we learnt

Roles played

References

[ana09] anaesthetica. Attacked from within. http://aiweb.techfak.uni-bielefeld.de/content/bworld-robot-control-software/, Mar 2009.

[BKLDNM13] Lars Backstrom, Jon Kleinberg, Lillian Lee, and Cristian Danescu-Niculescu-Mizil. Characterizing and curating conversation threads: Expansion, focus, volume, re-entry. In Proceedings of WSDM, pages 13–22, 2013.

[GK10] Eric Gilbert and Karrie Karahalios. Understanding deja reviewers. In Kori Inkpen Quinn, Carl Gutwin, and John C. Tang, editors, CSCW, 2010.