PREPRINT of: Squire, M. and R. Gazda (2015). FLOSS as a source for profanity and insults: Collecting the data. In Proceedings 48th Hawaii International Conference on System Sciences. IEEE. Kauai, HI, USA. January 8, 2015.

FLOSS as a Source for Profanity and Insults: Collecting the Data

Megan Squire
Elon University
msquire@elon.edu

Rebecca Gazda
Elon University
rgazda@elon.edu

Abstract

An important task in machine learning and natural language processing is to learn to recognize different types of human speech, including humor, sarcasm, insults, and profanity. In this paper we describe our method to produce test and training data sets to assist in this task. Our test data sets are taken from the domain of free, libre, and open source software (FLOSS) development communities. We describe our process in constructing helper sets of relevant data, such as profanity lists, lists of insults, and lists of projects with their codes of conduct. Contributions of this paper are to describe the background literature on computer-aided methods of recognizing insulting or profane speech, to describe the parameters of data sets that are useful in this work, and to outline how FLOSS communities are such a rich source of insulting or profane speech data. We then describe our data sets in detail, including how we created these data sets, and provide some initial guidelines for usage.

1. Introduction

In July 2013 a controversy about dialogue and speech style erupted on the 20-year old Linux Kernel Mailing List (LKML). Sarah Sharp, a kernel developer and employee of Intel, accused Linus Torvalds, the creator of Linux and project leader, of being verbally abusive to other kernel developers. After Torvalds asked one of his next-in-command to be more critical when people submit bad patches [1], Sharp told Torvalds he was "...one of the worst offenders when it comes to verbally abusing people and publicly tearing their emotions apart" [2]. Torvalds defended his right to his speaking style: "I can't just say 'please don't do that,' because people won't listen. I say 'On the internet, nobody can hear you being subtle', and I mean it" [3]. This thread on the LKML mailing list took a few days to be resolved, and captured a lot of media attention in the meantime [4][5].

In the resulting discussion of the incident, many interesting related issues were raised by participants and onlookers: was Torvalds insulting people, or just their code? Is there a difference? Is this really verbal abuse or just impolite? Are verbal abuse and profanity related? Do these types of messages represent an unemotional, meritocratic "code is all we care about" attitude, or do insults and profanity mask a heightened emotional content or aggression? Is this type of speech the norm in FLOSS development, or is the LKML different? Was Sharp being overly sensitive? Was Sharp's sensitivity gender-based? Was Torvalds' defense of his rant gender-based? (Sharp identifies as female, Torvalds as male.) Would having a Code of Conduct for participation affect the style of communication for that project?

This paper does not attempt to answer all of these fascinating and important questions. Instead, the motive behind this work is to use the momentum of this incident (and dozens of others like it in the history of FLOSS development [6]) to learn something new. The verbal culture of FLOSS development can become another source of valuable raw material for detecting insults and profanity, an important task for natural language processing (NLP) and machine learning (ML), and an interesting area of study on its own.

Because most FLOSS is developed using transparent, archived communication media (usually email mailing lists and IRC chat), its internal workings are convenient to analyze, and a body of literature has built up over time for doing so (i.e. the survey in [7]). This paper builds upon that history of empirically studying FLOSS developer communications so as to understand how this type of software is made. In this paper we create data sets specifically focused on the profanity and insults present in FLOSS developers' speech, and then show how to use these to train natural language classifiers.

The rest of this paper is organized as follows. In Section 2, we give background on the task of insult detection and profanity detection in NLP and ML. We also describe what related data sets have been used already for this task, and by whom. Section 3 describes the data collection methodology and the data sets we built. We also describe some existing useful data sets, and show how those can be paired with ours. In Section 4 lists some remaining questions and ideas for future work in this area.

2. Background on detecting profanity and insults

The detection of negative or malicious content in written communication is a challenging problem. As electronic communities and social media sites become more numerous and ubiquitous, understanding issues such as cyberbullying [8] and cross-cultural and intercultural differences [9] in online communication becomes more important. At the same time, online communication artifacts increase in volume, velocity, and variety, making the detection of harmful or annoying content more difficult. Because of inherent differences in detecting profanity and insults, we will describe each of these efforts separately. We also want to immediately distinguish profanity and insult detection from spam detection, which is a related but different problem. Whereas spam is annoying, it is generally not considered a dialogue between two or more people, nor does it typically take place in real time (as IRC chat does, for example). Some of the detection techniques may be the same, but their practical application to profanity and insult detection will be very different.

Early attempts at spam detection worked by searching known word lists, known email sender lists, and the like in order to compute a probability that a given message was spam. Later attempts use Bayesian classification to train the system to recognize a new message as spam, given the learned characteristics of previously tagged spam. Current attempts to block profanity are generally designed around word list systems. Given a list of "bad" words, regular expressions are used to match a given word to the bad word lists. Lists vary in their length and whether they can be added to. Allowing users to add words to a bad word list increases both the likelihood of false positives (someone updates the list and now the formerly harmless 'foo' is considered a bad word) but also increases the likelihood of proper list inclusion (slang and profanity tend to be obscured in online communication, and variants to words evolve quickly). Examples of bad word lists include [10][11][12].

Existing NLP and ML approaches to profanity detection in online communities include [13][14][15]. Some of these (e.g. [13][14]) rely on word lists, either created by the researchers themselves, or using a combination of downloadable lists. Others (e.g. [15]) take a human coding and classification approach and do not rely on word lists at all - other than those encapsulated in the brains of the human classifiers. Other works studying profanity in various domains (e.g. [16][17]) have also relied on word lists but were simply counting instances and not attempting to predict whether a new case is also profane.

Insulting speech detection is also an interesting task for NLP. It is more difficult than profanity detection because insults may include profanity, but may not. The sentences in which insults occur may take a variety of verbal forms: interjections, accusations, put-downs, double entendres, sarcasm. There are also subjective and contextual components to insult detection. Kaggle, the data science competition site, offered a recent challenge on insult detection [18]. The insults used in the competition were taken from generic social media sites and were usually not more than a single sentence or phrase. There was no context given for the insults other than what was in those sample sentences. Some research in non-automated insult detection [19] has used word lists and directed the human participants to read the word and imagine someone is using that word about them, for example saying "You are bossy" or "You are nice", with mixed results. Other, automated attempts at insult detection (e.g. [20][21][22][23][24]), rely on the use of seed lists of putdowns and profanities, combined with hand-generated rules for semantics such as looking for phrases that start with "you" or denote extreme subjectivity.

Detecting insults may be similar to recognizing other types of highly variable and contextualized speech, for example metaphors, sarcasm, humor, and double entendres. The NLP and ML approach has been used with some success here as well. For this paper, we are particularly interested in the work on "that's what she said" (a double entendre that denotes sexual content in the preceding line) since that can be used as a marker to build a training set for classification as was done in [25] and [26], and since the issue of possible sexism, and "brogrammer culture", or "lad's culture" in FLOSS has been raised by critics as well (see [27] for a good summary).

3. Data collection

Based on the demonstrated utility of data sets for both profanity and insult detection, described in Section 2, and mindful of the controversy surrounding the allegedly-abusive speech of the LKML, we decided to collect some data to assist other researchers who may wish to study this issue further. Our initial effort at data collection and describing the profanity and insult situation within FLOSS is described in this section. It consists of a collection of simple measurements and data sets, intended to start the conversation or inspire others to continue or improve the work.

3.1. Codes of Conduct

Table 1 shows a list of popular FLOSS projects and whether they have a Code of Conduct for participation. Typically a Code of Conduct will include a list of the types of behavior that are not acceptable, specific information about where to report a violation, and specific guidelines for how the Code of Conduct will be enforced [28]. If we are interested in profanity and insults, whether the community has a Code of Conduct seems relevant, since it is one indicator for what the stated community norms are in that project.

Table 1. Selected FLOSS Projects with Codes of Conduct

Project Name	Link to Code of Conduct
Apache	http://wiki.apache.org/incubator/CodeOfConduct (depends on subproject)
Debian	https://www.debian.org/code_of_conduct
Django	https://www.djangoproject.com/conduct/
Drupal	https://drupal.org/dcoc
Fedora	http://fedoraproject.org/code-of-conduct
Gnome	https://wiki.gnome.org/action/show/Foundation/CodeOfConduct
Joomla!	http://www.joomla.org/about-joomla/the-project/code-of-conduct.html
KDE	http://www.kde.org/code-of-conduct/
MariaDB	https://mariadb.org/en/community/ (uses Ubuntu CoC)
Mozilla	http://www.mozilla.org/en-US/about/governance/policies/participation/
Mozilla Rust	https://github.com/mozilla/rust/wiki/Note-development-policy
Nginx	http://ngx.readthedocs.org/en/latest/topics/community/irc.html
Node.js	http://confcodeofconduct.com/
Open Stack	http://www.openstack.org/legal/community-code-of-conduct/
OSI	http://opensource.org/codeofconduct
Puppet	http://docs.puppetlabs.com/community/community_guidelines.html
Python	https://www.python.org/psf/codeofconduct/
Sugar	http://wiki.sugarlabs.org/go/Sugar_Labs/Legal/Code_of_Conduct
Ubuntu	http://www.ubuntu.com/about/about-ubuntu/conduct
Wordpress	http://en.forums.wordpress.com/topic/wordpresscom-forums-code-of-conduct (forums)

We are particularly interested in Codes of Conduct for online community participation (mailing list, forum, IRC chat). We are less interested in whether that project has a Code of Conduct for its conferences, since (a) those are often put in place by external conference committees or agencies, and (b) conferences are often not the sole domain of that project; other projects might be involved, and (c) we are primarily interested in online communication artifacts, but conferences are largely a face-to-face phenomenon.

There are a few notable FLOSS projects which do not have a Code of Conduct at all. Although there are several Linux distributions which *do* have a CoC (Debian, Fedora, Ubuntu), the Linux kernel itself is not listed. Wired reported Torvalds' opinion of Codes of Conduct as follows "...venting of frustrations and anger is actually necessary, and trying to come up with some 'code of conduct' that says that people should be 'respectful' and 'polite' is just so much crap and bullshit." [3]

3.2. Profanity detection

To start the discussion on whether and how to use FLOSS to build data sets for machine learning profanity classifiers, we provide some basic descriptive statistics about profanity on FLOSS mailing lists and IRC channels.

3.2.1. Profanity word list. In order to keep the tables short and manageable, we begin with the method presented in [16] and [17], which calculate profanity of a medium based on the presence of the categories of words. They begin with the "seven dirty words" that were the foundation of the lawsuit FCC v. Pacifica Foundation (438 U.S. 726, 1978). The original words that started the lawsuit were: shit, piss, fuck, cunt, cocksucker, motherfucker, and tits. Using a 2005 sample, [17] found that U.S. prime time television (broadcast and cable, combined) aired the "seven dirty" words a total of 559 times. (That works out to about twice per hour. Additional categories of words were also part of their study, including excretory words, sexual-content words, mild profanity, etc. bringing the total to 3500 words, or 12.5 per hour.)

3.2.2. Profanity on LKML. We used the word list from 3.2.1 to explore whether there was even enough profanity on the LKML to make an interesting data set. Table 2 (next page) shows that Linus Torvalds does use the word 'shit' (and 'bullshit') more than any other person on the LKML. The "highest/next-highest" column is for the person who exhibits the next-highest use of the word after Torvalds (or the highest uses, in the case of 'piss' and 'fuck'). Note that we had to drop four of the words for lack of use on the LKML. But

these counts in Table 2 do include variants, such as 'shitty' and 'pissed'.

Table 3 indicates that Linus Torvalds also does use some mild profanity such as 'crap', 'hell', and 'damn' more than any other person on the list, but this does not necessarily hold true when we take percentages into account. There are other mild profanities referenced in [17] that were not used in significant enough numbers to be included, for example, 'bitch', 'cock', 'slut', 'bastard'.

Table 2. Rates for the "three dirty words" on the LKML, with author

	Torvalds	**Highest/Next-Highest**	**LKML, all**
total messages sent	21,049	-	1.9m
shit/bullshit	179 (0.8%)	58 out of 18367 (0.3%)	3607 (0.2%)
piss	19 (>0.1%)	33 out of 1771 (1.8%)	1328 (0.1%)
fuck	14 (>0.1%)	32 out of 5544 (0.6%)	801 (>0.1%)

Limitations to our method were that we did *not* search for obscured words (e.g. f*ck, sh*t), and we did *not* count multiple words in a single message separately. We limited our words to only English, and only the words shown. We did *not* disambiguate profanity in quoted replies versus original utterances, but we *did* count profanities in Subject: headers.

How do these rates compare to other FLOSS projects? One interesting finding that we did not put in the table was that if we consider the entire collection of nearly 80 million email messages indexed by MarkMail.org for 8800 different (mostly-FLOSS) projects, there is no single individual who uses the words "crap" or "shit" in email more often than Linus Torvalds. This is true for raw counts as well as on a percentage basis (profanity per email sent).

How do these rates compare to other social media? Other profanity studies show that cursing rates in daily life are about 0.5% of all words spoken [29]. On Internet chatrooms, research shows about 3% of the chat entries contain profanity [30], or about 40 per hour (recall that television rates were about 12.5 per hour). The researchers in [13] found that around 7% of all tweets on Twitter contain at least one profanity. Since email messages are substantially longer than tweets, and since our counts are on a per-message basis, not a per-word or even per-sentence basis, we can not claim that the rates of profanity on FLOSS email mailing lists are anywhere close to prime-time television, Twitter, teen chat rooms, or daily American life. Thus we conclude that FLOSS emails are probably not an ideal source for corpuses of generic profanity, though they may be an interesting source for studying a type of workplace profanity.

Table 3. Rates of common "mild profanity" on the LKML, with author

	Torvalds	**Highest/Next-Highest**	**LKML, all**
total messages sent	21,049	-	1.9m
crap	1070 (5.1%)	446 out of 30253 (1.5%)	14,626 (0.74%)
hell	816 (3.9%)	473 out of 5544 (8.5%)	11,409 (0.58%)
damn	749 (3.6%)	240 out of 5544 (4.3%)	8107 (0.41%)
ass/arse	76 (0.4%)	181 out of 18367 (1%)	3156 (0.16%)

3.2.3. Profanity in FLOSS project IRC chat. Next we investigated the language usage on FLOSS projects that use IRC chat channels to communicate. IRC chat is a type of synchronous chat, most similar to the type of "chat room" mentioned in [30], and with line lengths more similar to Twitter. We limited our study to IRC chat channels that were officially recommended by a given FLOSS project for support or discussion. We confirmed that each channel was recommended by the project, and that there were links and/or joining instructions located somewhere on the project's official support page. We also limited our study to channels which had downloadable logs available. (We did not run any IRC log bots ourselves.)

Table 4 (next page) shows six IRC chat channels for four different FLOSS projects. For this table, we

only included profanities that appeared in more than 1% of messages. Even on IRC, where users commonly have pseudonyms and can change their identity at will, the only rates above 1% (lines with profanity) were Ubuntu ("crap") and Django ("shit"). (Wordpress and OpenStack have separate developer and user channels. We were curious if the level of profanity differed between them. The Ubuntu project has hundreds of channels; the channel shown is the main one.)

Table 4. Common profanity rates on selected project IRC channels

	total lines	shit	crap
wp-dev	436k	239	377
wp	2.6m	1952	4476
openstack-dev	334k	89	266
openstack	518k	231	355
ubuntu	27.8m	20k	31k (1%)
django	1.4m	1806 (1%)	1286

In 1999, the LKML had a discussion of its reputation for having too much profanity in the kernel source code. Linus Torvalds wrote,

> So don't worry. People have sometimes worried that it is "unprofessional" to use profanity, but if you think professionals don't swear you've either been living in a monestary [sic] or playing golf your whole life ;) [31]

Despite this laissez-faire attitude from the project leader, we find the level of conversational profanity in the LKML to be very low. We find even lower levels in the IRC chats of similar projects. Even common (mundane?) profane words are not very ubiquitous, and are not highly varied or creative in their spelling or usage. There is little attempt to obscure profane words, making their detection unchallenging from a ML classification perspective. In sum, profanity data sets and good testing scenarios will be very difficult to get using FLOSS projects, and not terribly useful when we get them. Even so, we have a few additional ideas for further study in the area of profanity, which we outline in "Future Work" (Section 4).

3.3. Insult detection

In the original discussion between Sharp and Torvalds ([1],[2]) the main point of contention was not around profanity, but rather around the use of insults, sometimes profane and sometimes not, and whether Torvalds, as the leader of the project, was unfairly using his position of power to verbally dominate and degrade his subordinates. A critical issue is that many (if not most) of the participants in the July 2013 LKML discussion about 'verbal abuse' issue seemed to draw a distinction between insulting someone personally and insulting someone's code. Even the profanity we investigated in this paper (Section 3.2) was often directed at code, in the form of an insult: "your code is shit", "I don't want to see obvious and shitty crap like this", and so on. Thus, the part of insult detection that we decided to focus on for this work is to distinguish between code-based insults versus personal insults, so as to be able to determine if there were actually insults directed at people that were NOT also code-related. (We do not attempt to address whether code-based insults are verbally aggressive or not [32][33][34] or could be ultimately harmful to a professional reputation built on open source contributions [35].)

To that end, we have created a list of insulting sentences gleaned from the LKML postings by Linus Torvalds from 1995-2014. To create this list, we have read the entirety of these postings, then created a data set of the insulting sentence or phrase. We give the time and date of the message and its link in the MarkMail database, as well as the entire sentence or sentence cluster in which the insult occurs. We have added an encoding for whether we believe the insult to refer to code, personality, or both. We have donated the data set to the FLOSSmole project [36] so that anyone can query the data (MySQL database), export it, or add additional columns to indicate alternate codings for code/personality/both. That way the researcher can use her own human classifiers if she wishes and store the results. Finally, Table 5 (next page) shows examples of each type of insult.

Some of the markers for Linus-style insults include the use of the phrase 'Grr' (representing a growl), and certain words with negative sentiment like 'ugly', 'horrible', 'idiot', 'stupid', 'retarded', 'insane'. So far our analysis shows that purely personal insults and purely code insults are relatively rare. He usually insults the person and their code at the same time, which makes sense because this is a work related email list, so the primary topic of conversation is source code and its production. The vast majority (>80%) of insults are code-related or activity-related, but also highly personal, including the pronoun 'you', or a first name of the person being insulted.

The intention of this exercise is twofold: to document incidents (if any) in which there were personal insults on LKML, to learn the difference between code insults and personal insults (if any), and to create an insult list that could be used to train a Linus-style insult classifier.

Table 5. Samples of Linus Torvalds insults and whether they refer to code, personality-based insults, or both

Type	Insult
Code	"the patch really is ugly, and already adds random stuff to map the vvar/hpet pages into user memory, using absolutely disgusting code." "Exposing it at all is a disgrace. making it 'default y' is doubly so."
Personal	"Why are you making up these completely invalid arguments? Because you are making them up." "Whee. Third time is the charm. I didn't know my email address was *that* hard to type in correctly. Usually it's the "torvalds" that trips people up, but you had some issues with "foundation", didn't you ;)"
Both	"And I'd really like to avoid adding hacky code to the kernel because of Kay's continued bad behavior, so I hope this works." "This patch was clearly never tested anywhere. Why was it sent to me? Grr. Consider yourself cursed at. Saatana." "And you ignored the real issue: special-casing idle is *stupid*. It's more complicated, and gives fewer cases where it helps. It's simply fundamentally stupid and wrong."

3.4. Gender-based insults

In the next two sections we have elected to specifically collect data for three types of insults that are somewhat gender-related. The first is the "that's what she said" (TWSS) sexual double entendre, the second is using mothers and grandmothers as a stand in for a mild personal insult (e.g. "your mom" jokes, also called "maternal insults"), and the third is using women (especially grandmothers and older women) to represent an un-intelligent or un-sophisticated person ("Even Grandma can use the software!"). We are interested in whether these types of comments appear commonly in FLOSS communication.

Our reason for being interested in the prevalence of these types of comments in FLOSS stem from the fact that it is a primarily male enterprise, with estimates of female participation hovering anywhere between 1% [37] and 13% [38] in surveys, and because some other works have claimed that these types of jokes are sexist in nature [39][40] or are to be expected in male-dominated online environments [41].

3.4.1. Double Entendre Detection via TWSS. Following on the work in [25] and [26], we constructed a data set that includes TWSS straight lines and punch lines for several online chat and email environments. The data set includes the following columns: project name, communication channel (irc/email), date, speaker of the setup line, the setup line, speaker of the TWSS punch line, the TWSS punch line itself. An example set of rows looks like Table 6. (We have removed speaker names from this data for formatting reasons.)

Table 6. Sample rows from double entendre detection data set using TWSS as a marker

Project	Media	Date	Setup	TWSS
Django	irc	6/25/2011	one must go down, in order to come up	That's what she said.
Django	irc	9/14/2011	the length issue is the problem	<name>: that's what she said
Django	irc	11/19/2011	oh yeah oh yeah ? well you aint much help here buddy	that's what SHE said!

So far, we have collected approximately 250 example of TWSS, along with their straight line markers. We exclude any row for which we can not be sure that the TWSS is being used to mark a double entendre (e.g. non sequiturs). As with the previous data sets, this data set has also been donated to the FLOSSmole project.

Prior work in using TWSS to identify double entendres [25] [26] used approximately 2000 thousand rows for training and another 200 for testing, all taken from the twssstories.com web site. So our corpus could be valuable in adding more records, and in a workplace/software engineering context.

3.4.2. Maternal insults. The venerable "mom joke" is a staple of predominantly male insult culture [42][43]. We constructed a data set that includes "your mom" dialogue in a similar fashion to the TWSS set above. Table 7 shows some sample rows from this data set. Again, for space reasons, we do not show the speaker names.

Table 7. Sample rows from maternal insults data set, using "your mom" as a marker

Proj.	Media	Date	Setup	Mom joke
WP	irc	7/11/2009	he said someone met their wife on WordPress	I met **your mom** on wordpress. ahem,
WP	irc	7/17/2009	What's a good local testing program then?	**your mom**
WP	irc	7/18/2009	not appropriate	neither is **your mom**

As of this writing, we have collected approximately 300 maternal insults and their setup lines (markers). We exclude any row for which we can not be sure that the maternal reference is actually intended to be an insult. As with the previous data sets, this data set has also been donated to the FLOSSmole project.

3.4.3. Old female relatives. The so-called "Grandma Test" [44] or the "Aunt Tillie" test on the LKML [45] (and occasionally, the girlfriend test [46]) refers to ensuring that your technical design is easy enough for an unsophisticated user. This is a form of gender-based condescension. Table 8 (next column) shows some sample rows from our data set of conversational examples where the Grandma/Tillie test is being used. There are three examples from IRC and three from mailing lists in this table.

So far in this data set, we have collected approximately 500 grandmother/girlfriend/Tillie lines from three different FLOSS projects. We exclude any row for which we can not be sure that the female reference is actually intended to be a reference to technical unsophistication. As with the previous data sets, this data set has also been donated to the FLOSSmole project.

4. Future work

In this section we provide some ideas for future work that can use these data sets. First, we must point out the implications of language use and socialization in online communities [47]. This research shows that as a person becomes a core member of a community his or her language changes to match the rest of the core. Can we identify the roles (core, periphery) for each member of the community and understand how their language changes over time as they move roles?

Table 8. Sample rows from Grandma data set, using "grandma" and "Aunt Tillie" as markers

Project	Media	Date	Line
ubuntu	irc	12/10/2004	If Linux were to give all that easy to configure and handle than even my **grandma** can use it without not knowing what a computer is then linux will be infeccted
ubuntu	irc	11/18/2004	Hoodster: you'll figure it out - it was designed for your **grandma** ;)
ubuntu	irc	1/17/2005	there's no beating ubuntu install though, excellent... and the layout of the gui is nice... all that's left is to improve hardware detection a little bit and ubuntu just may become a word **grandma** knows
linux-kernel	ML	1/14/2002	If it screws up, and **Aunt Tillie** shelled out for support (which of course, she did being the 'needing support' type)
linus-kernal	ML	1/14/2002	Yes, and yes. **Aunt Tillie** is running Linux because someone installed a distribution for her.
linux-kernel	ML	1/14/2002	We (yes, we) should make sure **Aunt Tillie** doesn't ever

			have to build a kernel, ever.

It also seems obvious to do some detection of language pattern changes on communities with and without a Code of Conduct, also before and after the CoC was adopted. Similarly, a valuable addition to the data sets (especially the 'twss' and 'your mom' sets) would be a flag for how the community reacted to the joke. Possible encodings for a reaction could include: ignoring it, admonishing the behavior, laughing or positive response, etc.

The LKML thread in [31] references profanity in the kernel source code (including source code comments and log messages). In this work we did not look at any source code whatsoever, only communication artifacts surrounding the production and support of the FLOSS. There is some suggestion in previous messages on LKML (including [31]) that profanity in the *actual* source code is somehow more undesirable than profanity in the communication around *making and supporting* the source code.

Finally, we are curious how this data compares to other workplace communities. There is very little literature on workplace profanity and even less on written insults in the workplace, though there is a growing body of literature on generic workplace bullying and verbal aggression. Most of this takes place in face-to-face environments, however, or in classroom environments. Perhaps these data sets will be useful in future studies in this area.

5. Conclusion

This paper reports on an attempt to construct data sets of insulting and profane language found on some common FLOSS projects. The intention is that the data sets can be used to train natural language classifiers (or the like), and are specific to a workplace-like, software-oriented domain. While the attempt to construct a profanity data set was not very successful (the quantity and variability of profanity was low), we have constructed what we believe to be useful data sets for verbal insults, double entendres around "that's what she said" jokes, examples of maternal insults and their setup lines, and examples of gender-based condescension. All data sets have been made publicly-available on FLOSSmole for anyone to use or improve.

6. References

[1] Torvalds, L. 2013. Re: [00/19] 3.10.1-stable review. On *Linux Kernel Mailing List*. July 12. Available at: http://markmail.org/message/mmvlphelw5gea4lw Last accessed June 1, 2014.

[2] Sharp, S.. 2013. Re: [00/19] 3.10.1-stable review. On *Linux Kernel Mailing List*. July 15. Available at: http://markmail.org/message/zuuvxeukn54jtdpn. Last accessed June 1, 2014.

[3] McMillan, R. 2013. Linus Torvalds defends his god-given right to offend you. *Wired*. July 16. Available at: http://www.wired.com/2013/07/linus-torvalds-right-to-offend/ Last accessed June 1, 2014.

[4] Brodkin, J. 2013. Linus Torvalds defends his right to shame Linux kernel developers. ArsTechnica. July 16. Available at: http://arstechnica.com/information-technology/2013/07/linus-torvalds-defends-his-right-to-shame-linux-kernel-developers/ Last accessed June 1, 2014.

[5] Bort, J. 2013. Intel programmer Sarah Sharp wants Linux creator Linus Torvalds to knock off the 'verbal abuse'. *Business Insider.* July 16. Available at: http://www.businessinsider.com/geekess-calls-linus-torvalds-out-for-rants-2013-7 Last accessed June 1, 2014.

[6] Geek Feminism Wiki. n.d. Timeline of incidents. Available at: http://geekfeminism.wikia.com/wiki/Timeline_of_incidents Last accessed June 1, 2014.

[7] Squire, M. 2012. How the FLOSS research community uses email archives. *International Journal of Open Source Software and Processes, 4(1).* 37-59.

[8] Hinduja, S. & Patchin, J.W. 2008. Cyberbullying: An exploratory analysis of factors related to offending and victimization. *Deviant Behavior*, 29(2).

[9] Kim, K.J. & C.J. Bonk. 2002. Cross-cultural comparisons of online communication. *Journal of Computer-Mediated Communication*, 8(1).

[10] List of Bad Words. n.d. Available at: http://www.noswearing.com/dictionary Last accessed April 23, 2014.

[11] Daniels, M. n.d. Profanity List. Available at: http://www.mdaniels.com/profanity/ Last accessed April 23, 2014.

[12] Dubs, J. 2011. Google's official list of bad words. Available at: http://fffff.at/googles-official-list-of-bad-words/ Last accessed May 15, 2014.

[13] Wang, W., L. Chen, K. Thirunarayan, A.P. Sheth. 2014. Cursing in English on Twitter. In *Proceedings of Computer Supported Cooperative Work and Social Computing (CSCW 2014).* ACM. 415-425.

[14] Xiang, G., B. Fan, L. Wang, J.I. Hong, & C.P. Rose. 2012. Detecting offensive tweets via topical feature discovery over a large scale twitter corpus. In *Proceedings of Conference on Information and Knowledge Management (CIKM'12).* Oct 29-Nov 2. ACM. 1980-1984.

[15] Sood, S.O., J. Antin, & E.F. Churchill. 2012. Profanity use in online communities. In *Proceedings of Computer-Human Interaction (CHI'12)*. ACM. 1481-1490.

[16] Ivory, J.D., D. Williams, N. Martins, M. Consalvo. 2009. Good clean fun? A content analysis of profanity in video games and its prevalence across game systems and ratings. *CyberPsychology and Behavior*, 12(4). 457-460.

[17] Kaye, B.K., & B.S. Sapolsky. 2009. Taboo or not taboo? That is the question: Offensive language on prime-time broadcast and cable programming. *Journal of Broadcasting & Electronic Media, 53(1).* 1-16.

[18] Kaggle. 2012. Detecting insults in social commentary. Competition. Available at: https://www.kaggle.com/c/detecting-insults-in-social-commentary Last accessed June 7, 2014.

[19] Wellsby, M., P.D. Siakaluk, P.M. Pexman, & W.J. Owen. 2010. Some insults are easier to detect: The embodied insult detection effect. *Frontiers in Psychology*, 2010(1):198. DOI: http://dx.doi.org/10.3389/fpsyg.2010.00198

[20] Mahmud, A., M.Z. Ahmed, & M. Khan. 2008. Detecting flames and insults in text. In *Proceedings of the 6th International Conference on Natural Language Processing.*

[21] Xu, Z. & S. Zhu. 2010. Filtering offensive language in online communities using grammatical relations. In *Proceedings of the 7th Collaboration, Electronic Messaging, Anti-Abuse and Spam Conference*.

[22] Razavi, A.H., D. Inkpen, S. Uritsky, & S. Matwin. 2010. Offensive language detection using multi-level classification. In *Proceedings of the 23rd Canadian Conference on Artificial Intelligence.* 16-27.

[23] Chen, Y., Y. Zhou, S. Zhu, & H. Xu. 2012. Detecting offensive language in social media to protect adolescent online safety. In *Privacy, Security, Risk and Trust (PASSAT) at 2012 International Conference on Social Computing (SocialCom)*, 71-80.

[24] Spertus, E. 1997. Smokey: Automatic recognition of hostile messages. In *Proceedings of the 9th Conference on Innovative Applications of Artificial Intelligence*. 1058-1065.

[25] Kiddon, C. & Y. Brun. 2011. That's what she said: Double entendre identification. In *Proceedings of the 49th Association for Computational Linguistics (short papers)*. 89-94.

[26] VandenBos, B. 2011. Pre-trained That's-What-She-Said (TWSS) classifier in Ruby. Available at: https://github.com/bvandenbos/twss

[27] Reagle, J. 2012. "Free as in sexist?" Free culture and the gender gap. *First Monday*, Dec. ISSN 13960466. Available at: http://firstmonday.org/ojs/index.php/fm/article/view/4291/3381. Last accessed: June 14, 2014.

[28] Geek Feminism Wiki. n.d. Code of Conduct. Available at: http://geekfeminism.wikia.com/wiki/Code_of_conduct

[29] Mehl, M.R., & J.W. Pennebaker. 2003. "The sounds of social life: a psychometric analysis of students' daily social environments and natural conversations." *Journal of Personality & Social Psychology* 84(4). 857-870.

[30] Subrahmanyam, K., D. Smahel, & P. Greenfield. 2006. Connecting developmental constructions to the internet: Identity presentation and sexual exploration in online teen chat rooms. *Developmental Psychology, 42(3)*. 395-406.

[31] Torvalds, L. 1999. Profanity in the Linux kernel?!?!? On *Linux Kernel Mailing List*. June 8. Available at: markmail.org/message/t27yki4pnxbdp2rw Last accessed June 1, 2014.

[32] Jay, T.B. 2000. *Why we curse*. Philadelphia: John Benjamins.

[33] Jay, T.B. 2009. The utility and ubiquity of taboo words. *Perspectives on Psychological Science*, 4. 153–161.

[34] Jay, T.B. 2009. Do offensive words harm people?. *Psychology, Public Policy, and Law* 15(2). 81-101.

[35] Lerner, J., & J. Tirole. 2002. Some simple economics of open source. *Journal of Industrial Economics, 50(2*). 197-234.

[36] Howison, J., M. Conklin, & K. Crowston. 2006. FLOSSmole: A collaborative repository for FLOSS research data and analyses. *International Journal of Information Technology and Web Engineering (IJITWE)* 1(3)). 17-26.

[37] Ghosh, R.A. 2005. Understanding free software developers: Findings from the FLOSS study. *Perspectives on free and open source software*. 23-46.

[38] Arjona-Reina, L., G. Robles, & S. Dueñas. (2014). *The FLOSS2013 Free/Libre/Open Source Survey*. January 2014. Available on-line: http://floss2013.libresoft.es

[39] Phipps, A., & I. Young. 2012. That's what she said: women students' experience of 'lad culture' in higher education. Available at: http://www.academia.edu/3623232/Thats_what_she_said_women_students_experiences_of_lad_culture_in_higher_education Last accessed June 1, 2014.

[40] Geek Feminism Wiki. n.d. So Simple Your Mother Could do it. Available at: http://geekfeminism.wikia.com/wiki/So_simple,_your_mother_could_do_it. Last accessed June 1, 2014

[41] Dickey, Michele D. 2011. World of Warcraft and the impact of game culture and play in an undergraduate game design course." *Computers & Education* 56(1).200-209.

[42] Ayoub, M.R. & S.A. Barnett. 1965. Ritualized verbal insult in white high school culture. *The Journal of American Folklore, 78(310)*. 337-344.

[43] Conway, A. 1995. You're ugly, your dick is small, and everybody fucks your mother. *Maledicta: The International Journal of Verbal Aggression: 1990-1995*,11. 34.

[44] Brockmeier, J. 2007. It's time to retire the mom test. Sept. 8. Available at: http://archive09.linux.com/feature/118863. Last accessed June 1, 2014.

[45] Aunt Tillie Test. n.d. *Jargon File v4.4.7.* Available at: http://catb.org/~esr/jargon/html/A/Aunt-Tillie.html. Last accessed June 1, 2014.

[46] Content Consumer. 2008. The great Ubuntu-girlfriend experiment. April 27. Available at: http://contentconsumer.wordpress.com/2008/04/27/is-ubuntu-useable-enough-for-my-girlfriend/ Last accessed June 1, 2014.

[47] Nguyen, D. & C.P. Rose. 2011. Language use as a reflection of socialization in online communities. In *Proceedings of the Workshop on Language in Social Media (LSM2011)*. June 23. ACL. 76-85.