Offensiveness Detection In Reddit Corpora

Charlie Dektar
cdektar@stanford.edu

Jonathan Engel
jengel@stanford.edu

Aaron Zweig
azweig@stanford.edu

Abstract

Our project was to create an algorithm that could determine whether a given comment on a website is offensive. We re-purposed the logistic regression algorithm for sentiment analysis to run on a data set obtained from various parts of Reddit that was hand-labeled for offensiveness. We also implemented specific features designed to detect offensive content better than a simple Bayesian analysis. Our best version of the algorithm reached an F1 score of .852 across all data. Overall, grammatical data appeared to be much less useful in detecting offensiveness than identifying context and the presence of specific unigrams and bigrams.

1 Introduction

Offensive posts have been an inescapable feature of online message boards, websites with comment sections, and other such public online mediums for years. Users posting with or without the intent to offend can exclude others from conversations, reduce traffic on websites, and even induce psychological harm on those that read their postings. Given the enormous volume of texts on the web, the best way to flag offensive texts and remove them from coloring online discussions seems to be an automatic classifier using some sort of NLP. Our goal is to create such a classifier with the hopes of contributing to knowledge about the creation of such classifiers and thereby help chip at pulling down the mass of offensive comments online. While there already exist various implementations of such automatic classifiers of offensive texts, as discussed below, each implementation has its own shortcomings.

One component of the problem of classification of offensive texts is the ambiguity of the term "offensive." Each person has his or her own definition of what is offensive. One person may be offended by a word such as "hell" while another would find profanity-laden racist epithets to be perfectly acceptable. We settled on a working definition of "offensiveness" that is "anything that prevents others from continuing in the conversation," a definition that is further discussed in a section below. With this definition both direct insults and ethnicity/sexuality-wide slurs are classified as offensive, but run-of-the-mill profanity is generally not.

Given that everyone has their own sense of what is offensive, and this varies from message board to message board (ex: a children's educational website generally has a much lower threshold for offensiveness than a subreddit about video games), context is also a part of the problem of classifying offensive posts. If one's goal were to create a universal offensive text classifier it would have to take the context of the post into account in order to effectively classify a wide range of texts to the standards of a wide range of readers.

In addition, we did not find any useful publicly accessible datasets of labeled offensive and inoffensive texts[1], so part of the problem of training a classifier was creating and labeling dataset to use.

[1]While some exist, all of those that we found were extremely old, dating from the AOL/MySpace era of the internet. As such, these corpora often incorporated slang and jargon so foreign from conventional spoken English that they were not useful for our current purposes.

2 Previous Solutions

The literature on classifiers of offensive texts contains various implementations of classifier algorithms. Some of these classifiers used purely statistical methods, classifying offensive texts much like sentiment analysis classifiers. Other classifiers used the syntactical context of words and phrases to create features. We called the latter sort of classifier "grammatical" models and the former "statistical models."

One of the first such solutions, representative of early thinking on offensive text classification, was "Smokey", a statistical classifier made by Ellen Spertus (1997). The Smokey model converts each sentence into a feature vector with 47 elements and does a simple calculation using that vector to determine whether or not it is classified as abusive. The rules for each feature of the vector features are straightforward, ranging from more syntactic rules such as "Imperative sentence containing 'take'" to rules having to do with the present of semantic content, like "Contains a smiley face, such as ':-)' or ':)'". Smokey uses hand-labeled data for training and testing.

Another statistical model (Xiang et al. 2012) manages to greatly increase the size of their training data by developing an unsupervised model. They boast a training set of about 680 million tweets, and a test set of about 16 million tweets, which they categorize through bootstrapping. Beginning with a small set of inappropriate words, they can identity vulgar users, and mine their tweets to broaden their inappropriate word set. Their features include the output of LDA, and binary classification whether a tweet contains a word from their set or not. At that point, they follow a similar procedure as Reynolds and Kontostathis, namely the requisite machine learning algorithms and compare their performances. They achieve an F1 score of .810, and prove that the inclusion of their binary classification gives a non-trivial benefit over vanilla LDA.

A model that attempted to make progress on the issue of context is the subject of Yin et al. (2012). The concept behind the contextual features is that many posts with personal pronouns (often part of offensive comments) and offensive language that might match the profile of harassing language are not harassment at all. Internet commenters mock one another in a friendly fashion and have heated debates that would not be classified as harassment by Yin et al.'s standards. One contextual feature used is if the post is very different from surrounding posts, for example a spewing of vulgar language in a thread that otherwise had no vulgar language. The first post in the thread is of particular relevance to Yin et al. because it often sets the topic of discussion, so they weight the comparison of a potentially harassing post against that first post highly. In addition, Yin et al. found that offensive posts often occur in clusters, and compute the clustering of harassing messages accordingly.

Finally, it may be worth noting that a lot of the literature on the topic of offensiveness detection takes a very absolutist view on removing offensiveness. Some papers, such as Xu, Zhu 2010 had the explicit goal of eliminating offensive content from comments while retaining other non-offensive content in the comment. In their model, an offensive word is flagged for removal, at which point it is determined whether it is a key element of the phrase's syntactic structure. If not, it alone is removed. Otherwise, the entire phrase is removed, and possibly additional phrases that were determined to be dependent on the offensive phrase. Ultimately, the goal is to eliminate all offensive content in the message, leaving only a semantically inoffensive message that can stand on its own. To accomplish this goal, Xu and Zhu use part-of-speech and typed dependency tagging to build a relation tree for a sentence, and then after identifying individual offensive words, the algorithm deletes the offensive words and all parts of the sentence that directly depend upon them. This algorithm proves to have an accuracy of 90.94%, with sources of error stemming primarily from incorrect part-of-speech tagging. While their algorithm would be useful in contexts where it is important to filter out all offensive content, it is less useful for identifying offensive language in an attempt to engage with it, and its lack of real-world knowledge to determine offensiveness

makes it less than optimal as part of an AI's language parser.

3 Proposed Solution

3.1 Definition of "Offensive"

In order to effectively label data for classification, it was critical that we come up with a single, comprehensive, definition of offensiveness. Past literature has tended to use an extremely broad definition of offensiveness, roughly correlating to "things a parent would not want their ten-year-old child to see". While this can be an effective metric for some purposes, for our project we wanted to create an algorithm that could distinguish between comments that are simply crude or use coarse language, and comments that truly offend people. Given that online communities and comment-sections are often known for their tendency to use profanity in both offensive and non-offensive contexts, we wanted to build an algorithm that more closely reflected the judgment of an adult native speaker of English.

We ultimately chose to define "offensive" as "containing content such that it discourages further involvement in the conversation by either an entire existential group or someone previously participating in the conversation". "Existential group", for the purposes of our paper, is differentiated from "functional group"; where an existential group is defined as a group whose membership is non-voluntary and intrinsic to the member (for example, sexual or ethnic identity), a functional group is for our purposes defined as a group whose membership is predicated upon some sort of voluntary decision or action (such as "journalists", or "people who do X"). This is an important distinction, because we want to allow some justified insulting content as "non-offensive": for instance, if there is an online thread about a corruption scandal in government, "Fuck those politicians" should be considered a valid and non-offensive statement (Similarly, commenters should be allowed to express distaste of people who voluntarily engage in certain behaviors, like "people who leave their high beams on", for which they can be criticized). For existential groups, however, as a result of their non-voluntary membership, any generalizations can be seen as unjustified and hurtful, and thus we term such comments offensive.

Our definition of offensiveness would allow comments containing general expletives to be marked as "not-offensive" (such as "Fuck, good point"). However, we have flagged as offensive any content containing slurs (defined as in the Internet Encyclopedia of Philosophy entry on Pejorative Language), with the exception of some metalinguistic discussions of slurs (for instance, the sentence "Kike is a racial slur against Jews" would not have been labeled as offensive in our evaluation). Furthermore, we stipulated that a comment is also offensive if it directly attacks someone previously participating in the conversation. Independent from broad categories of offensiveness such as slurs, we felt that it was important to also categorize the denigration of a specific person partaking in a conversation as "offensive", as this behavior effectively makes it impossible for such a person to continue participating in the conversation. This can be contrasted from functional groups, as those groups are not present to take offense and the comments would not be seen as offensive to the general public.

Despite this definition, which we feel is both comprehensive and relatively unambiguous, we had some trouble classifying certain comments, particularly politically-charged ones. Depending on the speaker and audience in question, words such as "communist" or "fascist" could be seen as slurs or neutral descriptions of people or policies. We tried our best to classify these based on any available context in the comment, but these remained a source of confusion for which we did not find a generalizable solution. Additionally, we also had trouble classifying comments which, while grossly misinformative and thus possibly alienating, are not strictly insulting, such as comments like this one:

"Feminists seem to be under the impression that they can just demand free Wage Gap money and businesses will make it appear from nowhere"

While there is nothing directly insulting in this comment, its dismissal of problems facing women could be seen as offensive. On the other hand, one could easily fall into the trap of labeling every truthism accepted by an opposing political party as "offensive", which would undermine the goals of our research. Ultimately, we tried to evaluate whether such comments stemmed from an active dismissal of another party's concerns (which we considered offensive), or whether it arose from a genuine, different interpretation of objective data (which would be inoffensive).

It is also worth acknowledging that, as a rather small group of male university students that come from similar socioeconomic backgrounds and hold roughly similar political views, our opinions of which specific passages are offensive are inherently subjective, and a possible source of error. While we have tried our best to be objective, we acknowledge that our labeling could be inherently flawed, and ideally a practical implementation of our algorithm would have our data-labeling be vetted by a much more diverse panel. Furthermore, we have been obtaining our data from a rather small set of communities on Reddit, which are known for particular types of sociopolitical thought that some may find inherently offensive. As such, our source of offensive data is rather homogeneous, and thus while we can trust our algorithm's results on further Reddit data as being generally trustworthy, we would not assume that given our training data we have created a valid model for detecting offensive content in general.

3.2 Collecting Data

We collected our data from the website Reddit, drawing data from four subreddits in particular: /r/funny, /r/worldnews, /r/kotakuinaction, and /r/blackcrimematters. Reddit as a whole is known for its often offensive content, and /r/funny (being one of the "default" subreddits that everyone who visits the site is shown unless they actively opt-out) has content that is roughly representative of the website as a whole. /r/worldnews, while ostensibly a site for general world news, is heavily populated by groups originating from white supremacist websites like Stormfront and /pol/. /r/kotakuinaction is a haven for the alt-right movement Gamergate (which tasks itself in fighting against feminist and "social justice" influence online), and /r/blackcrimematters bills itself as a community for the discussion of "the news that #BlackLivesMatter chooses to ignore", and is openly a haven for white supremacists.

For each of these websites, we pulled the text of roughly 900 of the most recent comments on the website, and had two readers hand-label every comment for offensiveness, with offensive comments being labeled "2" and non-offensive comments labeled "1". Disagreements between the two readers were resolved by discussion (oftentimes one of the two readers picked up on an additional offensive subtext to a comment that, when pointed out, made the classification of the comment straightforward). Further disagreements were resolved by having a third reader independently assign a tie-breaking vote based on their assessment of the comment's offensiveness.

The comments, together with their offensiveness scores, were then collected into one document to be read by the algorithm.

3.3 Our Algorithm

Equipped with this large, labeled dataset, it becomes possible to train an effective model. The algorithm first vectorized the comments into "one-hot" matrices. We apply our chosen features, which are explained and justified in the following section, in order to obtain pairs of feature vectors, and offensiveness labels.

Because we are grading on a binary scale of offensiveness rather than a numeric one, we utilize logistic regression in order to train an optimal model based on the chosen features. A number of alternative models for binary classification exist, which were originally considered. For example, neural networks are especially qualified for binary classification where the choice of features is non-obvious. However, neural nets generally require substantially larger corpora in order to avoid overfitting, and given that all of our data was labeled and double-checked by hand, we opted instead for a simpler but less demanding regression model.

The process of determing optimal features required extensive testing, and it was necessary to find a means to assess the model's success

without overfitting. An evaluation set was determined through the following system. Before every iteration of the algorithm, the entire corpus was randomly partitioned in a 70-30 split. Then the model was trained on the larger portion and evaluated on the smaller portion in order to collect unbiased accuracy and F1 scores.

3.3.1 Noteworthy Features

The most innovative component of our algorithm appears in the determined features. Meaningfully, the features that best captured "offensiveness" were determined through repeated trials and experiments with the data. The most successful model utilized the following five:

- Bigrams of comment words
- Trigrams of comment words
- Inclusion of the "*" character
- The subreddit of origin
- Whether a pronoun and a slur/swear/curse appear in the comment, separated by a distance of four or fewer words

The last three of these features demand further explanation. The "*" character as a feature relates specifically to Reddit's formatting; any comment text within asterisks appear in italics on the internet forum, which empirically correlates with sarcasm and aggravated emphasis. By the same logic, an earlier attempt was made to include a feature detecting words in all caps, but the presence of acronyms and the use of capitals to denote surprise rather than offense made that feature less significant to the model.

Including the subreddit of origin made a great impact on the model's performance. Observably different subreddit have dramatically different proporitions of offensive to non-offensive comments, which allows the model to alter its baseline expectation for each one. Notably, the inclusion of this feature does preclude the model from being applied directly to a new subreddit without any previously labeled examples. However, due to the slang and rude language that is unique to individual subreddits, being able to apply a model without any previous training does not appear to be a realistic goal.

The final feature required a small amount of preprocessing. We compiled a list of common pronouns, and three larger lists of swears, slurs, and crass language. The division of these lists was based on the following principles. Slurs included unambiguously offensive language, which might be considered insulting simply by their presence, even without context. Swears included mainly curse words, those that were deemed derogatory not because of their literal meaning but through frequent use as insults. Crass words included language that could be completely benign when used for their literal, biological meaning, but in other situations could be repurposed for offense.

The justification for this feature was that, in general, dirty words are only offensive when used in reference to another person. As discussed above, some such language can appear in a non-offensive context, and correspondingly attempts to include unigram features for the number of appearances of each slur, swear, and crass word weakened the model. Rather than investing in the expensive calculation of parse trees in order to determine which entity a slur refers to, we used relative closeness between pronouns and curses as a proxy for when those curses were directly aimed at those persons. A distance of four was determined through trial and error to be an optimal measure of "closeness".

4 Results

Fig.1

Class	Precision	Recall	F1
non-offensive	.924	.880	.901
offensive	.606	.719	.658
total	.859	.847	.852

The most effective version of our algorithm (see Fig. 1) had an F1 score of .901 for non-offensive comments and .658 for offensive data, for a total F1 score of .852 over a held-out test set size of about 1200 (from a training set of roughly 2400 data points), with roughly a fifth of those comments labeled as offensive. The weakest statistic for our offensive data set was our relatively low precision score of .606.

We also ran another version of the program that incorporated part-of-speech tagging in our feature selection using the NLTK part-of-speech tagger. We saved these features as additional bigrams. For example, "you pig", would be saved as (["you", "pig"], ["PN", "pig"], ["you", "N"]). The results for this additional implementation were slightly inferior to those of our normal implementation (the F1 score dropped to .843), and the additional steps of running the part-of-speech tagger and saving the additional features slowed down our algorithm considerably. We further modified our NLTK model to also take into account whether words in the bigram were in our offensive word lists. Such words would have their part-of-speech denotation marked for offensiveness. For example, a feature set for "you *slur*" would include (["you", "N*"]), with the star denoting that the unnamed noun in the bigram was an offensive word. This algorithm had the following results:
Fig 2.

Class	Precision	Recall	F1
non-offensive	.927	.869	.897
offensive	.555	.704	.621
total	.856	.838	.845

With a total F1 score of .845, our implementation with NLTK included was worse than our grammar-free implementation on every measurement except for non-offensive precision, and took considerably longer to run.

5 Discussion

5.1 Analysis of Errors

A better understanding of our model comes from directly studying the misclassified comments. In particular, based on the very high F1 on non-offensive examples and the lower F1 on offensive examples, the most elucidating misclassifications are when the model marks an offensive comment as non-offensive. One such example alludes to a relatively straightforward improvement to the model's features:

"the butthurt is strong in this one."

This example fails to be recognized as offensive mainly due to the absence of "butthurt" in our crass words list. A potential alteration might be to identity any crass substrings within comments, although this tactic would still fail to recognize common misspellings such as "buthurt". Furthermore, we're faced with issues in our final feature that determines the reference point of insulting language. For example, a sentence "the butthurt isn't strong in this one" would not necessarily evoke offensive, but even with recognition of "butthurt" as crass and its closeness to the pronoun "one", our current features are unable to deduce how "isn't" alters the meaning of this sentence. A snippet of a misclassified comment appears below:

"**No it's not ibutter go suck a cock**"

Even without seeing the rest of this comment, it is clear the model failed in classifying it as non-offensive. However, "cock" appears in our list of crass words which are generally less immediately offensive than the swears and slurs. Furthermore, the word "suck" doesn't appear on any of these lists, and there is no external feature to determine why this sentence wouldn't fall into the category of anatomical discussion rather than offense.

Additionally, the feature to determine whether this curse references anyone doesn't fire. This is because the sentence contains no subject pronouns, as it is imperative with an implicit 2nd person subject. Correcting this error would likely require greater grammatical analysis to recognize implicit pronouns and manually insert the missing pronouns into the comment.

5.2 Discrepancies with Previous Literature

It is not entirely clear to us why our grammar-added implementation performed more poorly than our more basic algorithm, especially given that most of the existing literature indicated that part-of-speech tagging was quite useful in offensiveness classification. It may be worth noting that our algorithm seeks to extract a more nuanced definition of offensiveness as compared to previous work; it is entirely possible that for our definition of offensiveness there are few grammatical differences between offensive comments

and non-offensive comments, and that the distinction of offensiveness lies largely in the realm of context analysis and the usage of particular words, as opposed to grammatical classes of words. Based on these discouraging and slow results, we did not attempt to implement syntax-tree based dependency analysis on our data, as we calculated that if our grammatical bigrams (a very simple approximation of syntactic relatedness) only made our algorithm's accuracy worse and made our runtime extremely slow, an improvement as a result of an even more in-depth grammatical analysis would be unlikely to return superior results, and would likely make the algorithm almost unworkably slow.

5.3 The Importance of Conext

It is also interesting to observe that the single largest accuracy jump in the iterations of our algorithm was when we implemented a feature corresponding to the subreddit from which each comment was sourced. While this may at first glance seem somewhat like cheating and contrary to the spirit of this offensiveness detection experiment, identifying the source of information is critical in human heuristic analyses of offensiveness. If a native speaker of English were to see a sentence like

"He's just introducing new people to his culture."

Out of context, the speaker may judge this to be completely inoffensive. If the speaker were told that this sentence came from a peer-reviewed sociology paper, they would likely maintain the view that the sentence is inoffensive. However, if told that this sentence was found in the xenophobic subreddit /r/BlackCrimeMatters (which is where this sample is actually from), the speaker would likely interpret this sentence very differently, perceiving it as a sarcastic insult against some subset of foreigners in general.

5.4 Future Work

As with virtually all machine learning projects, our algorithm would benefit from being able to obtain more data. So far, this has been a laborious process, as all of our training data must be labeled for offensiveness by hand. We noticed, however, that in some of the more politically insular communities such as /r/kotakuinaction and /r/blackcrimematters, commonly known offensive slurs often appeared alongside much more coded language that upon review also seemed to be slurs, or at least slur-like: for example, the word "cuck"[2] appeared to be used in a derogatory fashion alongside other more canonical English insults and extreme xenophobia, as well as generally espousing radical right-wing ideologies. Our encounters with heavily coded language lead us to believe that we may be able to extract further features of ultra-right extremist speech using a bootstrapping method. This could be useful both because these groups are often directly associated with offensive speech and online harassment, although it may also be a useful tool in its own right for tracking political discourse.

Other paths for further research could include adding support for other languages. Our current algorithm has trained exclusively on English-language data[3], although adding support for other languages would be as simple as just adding data in those languages (we currently have no estimate as to how similar the syntax of offensiveness is across languages).

Finally, while we decided that it would not be productive to examine the implementation of grammatical parsing trees in our analysis of comment structure and offensiveness, given other studies' general success with these methods, they may still prove worthwhile given more time and resources, and it would be

[2] Upon review, "cuck" appears to be slang for cuckolded, and seems to directly refer to a particular xenophobic fear that "undesirable foreigners" are copulating with local (read: white) women, and that this is the fault of feeble, easily manipulated men of left-leaning political inclinations. As a result, it is used as an insult against a target's manliness (while also carrying a racist presupposition connected to its xenophobic origins), and in many ways appears to have similar semantic content to pejorative uses of the slur "faggot", which has itself fallen out of favor in recent years

[3] Although due to the prevalence of some loanwords, particularly in video game-related communities, we did find some comments that contained Latinized Russian swears, like "cyka blyat" (Cyrillic: Сука блять, literally "bitch whore", it is used in Russian similarly to "Fuck you" in English)

foolhardy to rule these methods out simply because of our cursory forays into that field over the course of the project.

6 Conclusion

Our investigation appears to have found that the evaluation of offensiveness based on grammatical features is inconclusive. While there may be grammatic similarities between very broad forms of insulting language, it does not appear that there are any grammatic differences at the level of nuance which we chose to use in our definition of offensiveness. Our results indicate that context is truly the most important factor in determining offensiveness (at least after one has accounted for the presence or absence of coarse language). While this seems at first glance to be of little use, it allows insight into ways that we could improve how artificial intelligence programs that learn from reading natural language inputs choose to assimilate information. An example that readily comes to mind is that of Microsoft's Tay AI. Shortly after Tay's introduction to the internet, it was sabotaged by malicious users, largely stemming from hostile online communities like 4chan, that taught Tay to use racial slurs and spout radical conspiracy theories. When we consider how humans filter out information, however, at first glance much less importance is given to the content of the information than to where it is coming from: the average person would be much more likely to believe and accept a statement as true if it came from The New York Times or Time than if the exact same statement came from a fringe newsletter or the ramblings of a total stranger. We believe that this intuition is consistent with our findings in this project: computer algorithms must take in data about their information sources and weigh inputs accordingly, as opposed to blindly assimilating data.

Our offensiveness detector is relatively effective at identifying offensive comments, and thus could be implemented to curtail the prevalence of offensive content on public websites without strangling general discourse. Our recommendation would be to implement this algorithm as a first-line-of-defense filter, with comments deemed "offensive" being sent to a human moderator for approval. Additionally, our algorithm could work as a reasonable filter for Tay-like AI, preventing the AI from assimilating content which would alienate much of the general public.

7 References

1. Internet Encyclopedia of Philosophy, Pejorative Language.

2. Spertus, E. Smokey, Automatic recognition of hostile messages. Microsoft Research, MIT AI Lab, and University of Washington. 1997.

3. Xiang, G. B. Fan, L. Wang, J. I. Hong, and C. P. Rose, Detecting Offensive Tweets via Topical Feature Discovery over a Large Scale Twitter Corpus. Carnegie Mellon.

4. Z. Xu, S. Zhu, Filtering Offensive Language in Online Communities using Grammatical Relations. Pennsylvania State University. 2010

5. Yin, D. Z. Xue, L. Hong, B. Davidson, A. Kontostathis, and L. Edwards, Detection of Harassment on Web 2.0. Lehigh University, Ursinus College.