Toxic Comment Detection in Online Discussions

Julian Risch and Ralf Krestel

Abstract Comment sections of online news platforms are an essential space to express opinions and discuss political topics. In contrast to other online posts, news discussions are related to particular news articles, comments refer to each other, and individual conversations emerge. However, the misuse by spammers, haters, and trolls makes costly content moderation necessary. Sentiment analysis can not only support moderation but also help to understand the dynamics of online discussions. A subtask of content moderation is the identification of toxic comments. To this end, we describe the concept of toxicity and characterize its subclasses. Further, we present various deep learning approaches, including datasets and architectures, tailored to sentiment analysis in online discussions. One way to make these approaches more comprehensible and trustworthy is fine-grained instead of binary comment classification. On the downside, more classes require more training data. Therefore, we propose to augment training data by using transfer learning. We discuss real-world applications, such as semi-automated comment moderation and troll detection. Finally, we outline future challenges and current limitations in the light of most recent research publications.

Key words: Deep Learning; Natural Language Processing; User-generated Content; Toxic Comment Classification; Hate Speech Detection;

Julian Risch (corresponding author)
Hasso Plattner Institute, University of Potsdam, Prof.-Dr.-Helmert-Str. 2–3, 14482 Potsdam, Germany e-mail: julian.risch@hpi.de phone: +49 331 5509 272

Ralf Krestel
Hasso Plattner Institute, University of Potsdam, Prof.-Dr.-Helmert-Str. 2–3, 14482 Potsdam, Germany e-mail: ralf.krestel@hpi.de

1 Online Discussions and Toxic Comments

Posting comments in online discussions has become an important way to exercise one's right to freedom of expression in the web. This essential right is however under attack: malicious users hinder otherwise respectful discussions with their toxic comments. A *toxic* comment is defined as a rude, disrespectful, or unreasonable comment that is likely to make other users leave a discussion. A subtask of sentiment analysis is toxic comment classification. In the following, we introduce a fine-grained classification scheme for toxic comments and motivate the task of detecting toxic comments in online discussions.

1.1 News Platforms and Other Online Discussions Forums

Social media, blogs, and online news platforms nowadays allow any web user to share his or her opinion on arbitrary content with a broad audience. The media business and journalists adapted to this development by introducing comment sections on their news platforms. With more and more political campaigning or even agitation being distributed over the Internet, serious and safe platforms to discuss political topics and news in general are increasingly important. Readers' and writers' motivations for the usage of news comments have been subject to research [15]. Writers' motivations are very heterogeneous and range from expressing an opinion, asking questions, and correcting factual errors, to misinformation with the intent to see the reaction of the community. According to a survey among U.S. American news commenters [51], the majority (56 percent) wants to express an emotion or opinion. This reason is followed by wanting to add information (38 percent), to correct inaccuracies or misinformation (35 percent) or to take part in the debate (31 percent).[1]

Toxic comments are a problem for these platforms. First, they lower the number of users who engage in discussions and consequently, the number of visitors to their platform. As a result, an exchange of diverse opinions becomes impossible. With subscription models and ads as a way to earn money, a lower number of visitors means losing money. Second, legal reasons might require the platforms to deploy countermeasures against hate speech and to delete such content or not publish it at all. For example, in Germany, platform providers are obliged by the Network Enforcement Act[2] to delete "obviously illegal" content within 24 hours of being notified. Some comments might be legal but still prohibited by the terms of use or discussion guidelines by the

[1] Multiple reasons could be selected.

[2] https://germanlawarchive.iuscomp.org/?p=1245

platform. To exemplify reasons for comment deletion, we summarize nine rules that comprise the discussion guidelines by a German news platform.[3] A team of moderators enforces these rules. Most rules are not platform-specific but are rather part of the "Netiquette" — the etiquette on the Internet.

1. **Insults** are not allowed. Criticize the content of the article and not its author!
2. **Discrimination and defamation** are not allowed.
3. **Non-verifiable allegations and suspicions** that are not supported by any credible arguments or sources will be removed.
4. **Advertising and other commercial content** should not be part of comments.
5. **Personal data** of others may not be published.
6. **Copyright** must be respected. Never post more than short excerpts when quoting third party content.
7. **Quotations** must be labeled as such and must reference its source.
8. **Links** may be posted but may be removed if the linked content violates our rules.

These rules and our interest in understanding what makes a particular comment toxic motivates the creation of a classification scheme for toxic comments. Also, it helps to distinguish what toxic comment detection focuses on (e.g., insults, discrimination, defamation) and what it does not (e.g., advertising, personal data, copyright). Such a scheme is defined in the next section.

1.2 Classes of Toxicity

Toxicity comes in many different forms and shapes. For this reason, a classification scheme for toxic comments has evolved, which is inspired by annotations provided in different datasets as described in Section 2.1. Research on a classification scheme for toxic comments is a connection between computer science on the one hand and media and communication studies on the other hand. Waseem et al. proposed a two-dimensional scheme of abusive language with two dimensions "generalized/directed" and "explicit/implicit" [55]. "Directed" means a comment addresses an individual, while "generalized" means it addresses a group. "Explicit" means, for example, outspoken name-calling, while "implicit" means, for example, sarcasm or other ways of obfuscation. Other terms for this dimension are overtly and covertly abusive.

Still, researchers have not reached a consensus about what constitutes harassment online and the lack of a precise definition complicates annotation [21]. Waseem et al. compare annotations by laymen (users of a crowdsource platform) and by experts ("theoretical and applied knowledge of hate

[3] https://www.zeit.de/administratives/2010-03/netiquette/seite-2

speech") [54]. They find that models trained on expert annotations significantly outperform models trained on laymen annotations.

In the following, we discuss one such classification scheme consisting of five different toxicity classes. We show examples for the different classes of toxic comments for illustration.[4]

1.2.1 Obscene Language/Profanity

Example: "That guideline is bullshit and should be ignored.". The first class considers swear or curse words. In the example, the single word "bullshit" comprises the toxicity of this comment. Typical for this class, there is no need to take into account the full comment if at least one profane word has been found. For this reason, simple blacklists of profane words can be used for detection. To counter blacklists, malicious users often use variations or misspellings of such words.

1.2.2 Insults

Example: "Do you know you come across as a giant prick?". While the previous class of comments does not include statements about individuals or groups, the class "insults" does. "Insults" contain rude or offensive statements that concern an individual or a group. In the example, the comment directly addresses another user, which is common but not necessary.

1.2.3 Threats

Example: "I will arrange to have your life terminated.". In online discussions, a common threat is to have another user's account closed. Severely toxic comments are threats against the life of another user or the user's family. Statements that announce or advocate for inflicting punishment, pain, injury, or damage on oneself or others fall into this class.

[4] Warning: The remainder of this chapter contains comment examples that may be considered profane, vulgar, or offensive. These comments do not reflect the views of the authors and exclusively serve to explain linguistic patterns. The following examples stem from a dataset of annotated Wikipedia article page comments and user page comments [58], which is publicly available under Wikipedia's CC-SA-3.0 (https://creativecommons.org/licenses/by-sa/3.0/).

1.2.4 Hate Speech/Identity Hate

Example: “Mate, sound like you are jewish. Gayness is in the air”. In contrast to insults, identity hate aims exclusively at groups defined by religion, sexual orientation, ethnicity, gender, or other social identifiers. Negative attributes are ascribed to the group as if these attributes were universally valid. For example, racist, homophobic, and misogynistic comments fall into the category of identity hate.

1.2.5 Otherwise Toxic

Example: “Bye! Don’t look, come or think of coming back!”. Comments that do not fall into one of the previous four classes but are likely to make other users leave a discussion are considered “toxic” without further specification. Trolling, for example, by posting off-topic comments to disturb the discussion falls into this class. Similarly, an online discussion filled with spam messages would quickly become abandoned by users. Therefore, spam falls into this class, although spam detection is not the focus of toxic comment detection.

The listed classes are not mutually exclusive. Comment classification problems are sometimes modeled as multi-class classification and sometimes as multi-label classification. Multi-class means that different labels are mutually exclusive, e.g., a comment can be an insult or a threat but not both at the same time. In contrast, multi-label means that a comment can have multiple labels at the same time. Multi-label classification better mirrors real-world applications, because a comment can, for example, be both an insult and a threat at the same time. In research, this problem is often slightly simplified by assuming analyzed classes are mutually exclusive. We will discuss research datasets later, e.g., Table 3 gives an overview of datasets used in related work.

2 Deep Learning for Toxic Comment Classification

Deep learning for sentiment analysis and in particular toxic comment classification is mainly based on two pillars: large datasets and complex neural networks. This section summarizes available datasets and explains neural network architectures used for learning from this data.

2.1 Comment Datasets for Supervised Learning

Online comments are publicly available and every day the number of data samples increases. For example, in 2018, 500 million tweets have been posted on Twitter per day.[5] However, without labeling this data, it can only be used for unsupervised learning, such as clustering or dimensionality reduction. Semi-supervised and supervised learning approaches require labeled data. Examples of labels are the before-mentioned classes of toxicity. In a rather costly process, human annotators check for each and every comment whether it fits into one of the pre-defined classes. Because of the inherent ambiguity of natural language, annotators might not always agree on the label. Further, a comment might be perceived abusive in one context but not abusive in a different context. Different annotation guidelines, low annotator agreement, and overall low quality of annotations are one of the current research challenges in toxic comment classification [1].

Another issue is repeatability. Comments are publicly available, but typically, researchers are not allowed to distribute datasets that they annotated. This is because both — the original author of an online comment and the platform provider — hold rights of the data. Alternatively, researchers can distribute their annotations alongside the web scrapers that they used to collect online comments. However, it is impossible to rebuild the exact same dataset from scratch by scraping the original web pages again. In the meantime, comments are added, edited, or deleted entirely. It has been proposed to address this issue by measuring the extent of data changes with fingerprinting techniques [44]. The idea of *partial* data repeatability is to use fingerprints to identify unchanged subsets of the data and repeat experiments only on these subsets. This novel idea has not (yet) prevailed and therefore, today's research on toxic comment classification focuses on a small set of publicly available datasets: the "Yahoo News Annotated Comments Corpus" (522k unlabeled and 10k labeled comments) [33], the "One Million Posts Corpus" (1M unlabeled and 12k labeled comments) [49], and a collection of Wikipedia discussion pages (100k human-labeled and 63M machine-labeled comments) [58]. Wulczyn et al. also publish their annotation guidelines. Thereby, other researchers can understand and potentially reproduce the annotation process. Further, publishing annotation guidelines and annotated data is necessary to allow other researchers to verify/falsify the findings.

The annotation process is crucial for unbiased training datasets and a necessity for training unbiased models. Collecting a large number of labeled, toxic comments is complicated for several reasons. First, moderators edit or delete toxic comments. Moderation might happen shortly after publication so that the comment is shown to the public only for a short time frame. Only in this short time frame, the comment can be collected by a web scraper.

[5] https://www.omnicoreagency.com/twitter-statistics/

Class	# of occurrences
Clean	201,081
Toxic	21,384
Obscene	12,140
Insult	11,304
Identity Hate	2,117
Severe Toxic	1,962
Threat	689

Class	# of occurrences
Offensive	19,190
Clean	4,163
Hate	1,430

Table 1 Statistics of the datasets by Wulczyn et al. (left) [58] and Davidson et al. (right) [12] show that both datasets are highly imbalanced.

Alternatively, moderation takes place before publication, when web scrapers cannot obtain the comment.

Nevertheless, web scrapers use pre-defined lists of abusive language to find large numbers of toxic comments. This approach introduces a bias: toxic comments that do not match with the pre-defined list will not be included in the dataset. Although this bias is unintended, datasets with such bias are still valuable for research, simply because there is a lack of alternatives. One such dataset comprises 25k labeled tweets that have been collected by searching the Twitter API for tweets that contain words and phrases from a hate speech lexicon [12]. Overall, most related work analyzes datasets extracted from Twitter and makes the tweet IDs publicly available to support the re-creation of the dataset for repeatability [12, 18, 36, 54, 56].

Another challenge is the inherent class imbalance of available datasets. Table 1 lists statistics for two of these datasets. The class distribution of the dataset by Wulczyn et al. [58] is strongly imbalanced with a bias to "clean" comments, whereas the dataset by Davidson et al. [12] is strongly imbalanced with a bias to "offensive" comments. These class distributions are not representative of the underlying data in general. In fact, most comment platforms contain only a tiny percentage of toxic comments. Since these datasets are collected with a focus on toxic comments, they are biased in a significant way. This needs to be taken into account when deploying deep neural models trained on these datasets in real-world scenarios.

2.2 Neural Network Architectures

Large datasets of toxic comments allow training complex neural networks with millions of parameters. Word embeddings are the basis of neural networks when working with text data in general and also in the specific context of toxic comment classification. They translate each word to a vector of typically 50 to 300 floating-point numbers and thus serve as the input layer. As opposed to sparse, one-hot encoded vectors, these dense vectors can capture and represent word similarity by cosine similarity of the vectors. Beyond simple distance measurements, arithmetics with words can be performed as presented with the Word2Vec model [30]. The similar approaches GloVe [39] and FastText [9] provide alternative ways to calculate word embeddings. FastText is particularly suited for toxic comments because it uses subword embeddings. The advantage of subword embeddings is that they overcome the out-of-vocabulary problem. Toxic comments often use obfuscation, for example "Son of a B****", "***k them!!!!" but also misspelled words, which are common in online discussions. Fast-paced interaction, small virtual keyboards on smartphones, and the lack of editing/correction tools reinforce this problem. Word2Vec and GloVe fail to find a good representation of these words at test time because these words never occurred at training time. These words are out-of-vocabulary. In contrast, FastText uses known subwords of the unknown word to come up with a useful representation. The ability to cope with unknown words is the reason why previous findings [34] on the inferiority of word embeddings in comparison to word n-grams have become outdated.

Similar to other text classification tasks, neural networks for toxic comment classification use recurrent neural network (RNN) layers, such as long short-term memory (LSTM) [23] or gated recurrent unit (GRU) [11] layers. Standard neuronal networks suffer from the vanishing gradient problem. Back-propagation through time might cause the gradients used for the weight updates to become vanishingly small with the increasing number of time steps. With gradients close to zero, no updates are made to the weights of the neural network and thus there is no training process. LSTM and GRU layers overcome the vanishing gradient problem with the help of gates. Each cell's state is conveyed to the next cell and gates control changes to these states. Long-range dependencies can be conveyed for an arbitrary number of time steps if the gates block changes to the states for the respective cells. An extension to standard LSTM and GRU layers are bi-directional LSTM or GRU layers, which process the sequence of words in correct and reverse order.

All recurrent layers, regardless whether it is a simple RNN, LSTM or GRU layer, can either return the last output in the output sequence or the full sequence. If the last output in the sequence is returned, it serves as a representation of the full input comment. However, the outputs of each step in the sequence can be used as an alternative. So-called pooling layers can combine this sequence of outputs. Pooling in neural networks is typically used to reduce an input with many values to an output of fewer values. In neural networks for computer vision, pooling is widespread because it makes the output translation-invariant. Pooling on the word level can also make neural networks in natural language processing translation-invariant so that the exact position in a sequence of words is irrelevant. For toxic comment classification, both average-pooling and max-pooling are common with a focus on the latter. An intuitive explanation for the use of max-pooling over average-pooling is the following. If a small part of a comment is toxic, max-pooling will focus on the most toxic part and finally result in classifying the comment as toxic. In contrast, with average-pooling, the larger non-toxic part overrules the small toxic part of the comment and thus the comment is finally classified as non-toxic. The definition of toxicity classes typically assumes that there is no way to make up a toxic part of a comment by appeasing with other statements. Therefore, max-pooling is more suited than average-pooling for toxic comment classification. As an extension to max-pooling, k-max-pooling outputs not only the largest activation but also the second largest (up to k-largest). It has been shown to further improve classification accuracy in some scenarios [45].

An alternative to pooling after the recurrent layer is an attention layer. Graves has originally introduced the attention mechanism for neural networks in 2013 with an application to handwriting synthesis [20]. It was quickly followed by an application to image classification [31] and neural machine translation to align words in translations [11]. It has been successfully applied also to toxic comment classification [37]. The attention mechanism is basically a weighted combination of all outputs from the preceding recurrent layer. The model can thereby put more emphasis on selected words (or outputs of the recurrent layer) that are decisive for the classification. In semi-automated moderation scenarios, attention can be imagined as a spotlight that highlights abusive or otherwise suspicious words. The final dense layer handles the classification output. For multi-label classification, the dense layer uses a sigmoid activation and for multi-class classification problems, it uses a softmax activation.

Due to relatively small amounts of training data, overfitting can be an issue. Dropout is a countermeasure against this issue. It does only alter the training process and has no influence on validation or testing. The different kinds of dropouts used in neural networks for toxic comment classification are not task-specific:

1. **Standard dropout** randomly selects neurons and blocks their incoming and outgoing connections. The neuron is therefore ignored during forward and backward propagation.
2. **Spatial dropout** aims to block not only the connections of single neurons but of correlated groups of neurons. For example, if a single value of a 300-dimensional word embedding is dropped, it can be estimated based on the other 299 values. To prevent this, the full embedding vector with its 300 values is dropped at once.
3. **Recurrent dropout** is a special kind of dropout that is used in recurrent neural networks. It affects the updates of recurrent cell states.

Table 2 lists published approaches for toxic comment detection with deep learning. It provides an overview of used model architectures, embeddings, and evaluation metrics. For example, for the particular task of hate speech classification (three classes: sexist, racist or neither), Badjatiya et al. iden-

Table 2 Overview on neural network architectures used in related work

Study	Model	Embeddings	Metric
[16]	-	paragraph2vec	roc-auc
[7]	CNN/LSTM/FastText	GloVe, FastText	p,r,f1
[49]	LSTM	Word2Vec	p,r,f1
[38]	GRU	Word2Vec	roc-auc
[37]	CNN/GRU/RNN+Att	Word2Vec	roc-auc,spearman
[58]	muli-layer perceptron	-	roc-auc,spearman
[18]	CNN	Word2Vec	p,r,f1
[45]	GRU	FastText	f1
[43]	LSTM	FastText	f1
[60]	CNN+GRU	Word2Vec	f1
[46]	-	Word2Vec	p,r,f1
[41]	LSTM	-	p,r,f1
[1]	CNN/LSTM/GRU/RNN+Att	GloVe, FastText	p,r,f1,roc-auc

Table 3 Overview on datasets used in related work

Study	# Annotated Comments	Available	classes
[16]	950k Yahoo finance	no	hate-speech,other
[7]	16k Twitter	yes	sexist,racist,neither
[49]	12k news	yes	8 classes[a]
[38]	1.5m news	yes	accepted,rejected
[37]	1.5m news, 115k Wikipedia	yes	reject,accept/personal attack,other
[58]	100k Wikipedia	yes	personal attack,other
[18]	6.7k Twitter	yes	racism,sexism[b]
[45]	30k Facebook	yes	overtly,covertly aggressive,neither
[43]	5k Twitter/Facebook	yes	profanity,insult,abuse,neither
[60]	2.5k Twitter	no	hate,non-hate
[46]	3m news	no	accepted,rejected
[41]	16k Twitter	yes	sexist,racist,neither
[1]	25k Twitter, 220k Wikipedia	yes	offense,hate,neither/7 classes[c]

[a] negative sentiment, positive sentiment, off-topic, inappropriate, discriminating, feedback, personal stories, argumentative

[b] multi-label

[c] toxic, obscene, insult, identity hate, severe toxic, threat, neither (multi-label)

tify a combination of LSTM and gradient boosted decision trees as the best model [7]. Their neural network approaches outperform their various baseline methods (tf-idf or BOW and SVM classifier; char n-gram and logistic regression). Comparing convolutional neural networks (CNNs) and recurrent neural networks (RNNs), there is no clear favorite in Table 2. Both network architectures are of comparable popularity because they achieve comparable performance. However, the training of CNNs is, in general, faster than the training of RNNs because it can be better parallelized. Djuric et al. [16] use comment embeddings based on paragraph2vec [30] and refrain from using both CNNs and RNNs.

Table 2 also shows that several different metrics are used for evaluation. Because the datasets are imbalanced, accuracy is not used but precision, recall, and (weighted) macro- (or micro-) f1-score. Weighted f1-score focuses on the classification of the minority class by emphasizing the respective penalty for misclassification. Further roc-auc and Spearman correlation are used, which we explain in more detail in the following. Spearman's rank correlation coefficient is used to compare ground truth annotations with the model predictions. To this end, the correlation between the fraction of annotators voting in favor of toxic for a particular comment and the probability for the class toxic as predicted by the model is calculated. The receiver-operating characteristics area under the curve (roc-auc) is used to measure how good a model is at distinguishing between two classes, e.g., toxic and non-toxic comments. For that purpose, the majority class label in the set of annotations is considered the ground truth and is compared to the predicted probability.

3 From Binary to Fine-Grained Classification

In real-world applications, toxic comment classification is used to support a decision-making process: Does a particular comment need moderation or can it be published right away? This problem is a binary classification problem, which oversimplifies the different nuances in language and abstracts from the classification scheme that we described earlier. A more fine-grained classification, on the other hand, gives insights on why a comment is not suitable for publication. This can help the moderators in making a final decision but also the benevolent offender to avoid infringement of comment rules in the future. Therefore different classes of toxicity, such as insult, threat, obscene language, profane words, hate speech, etc. have to be distinguished. With this fine-grained classification, it is also possible to distinguish between merely bad comments and criminal offenses. The following explains why fine-grained comment classification is a much harder task than binary comment classification. Further, we discuss two related topics: *transfer learning* to deal with limited training data, and *explanations* to help moderators to understand and trust neural network predictions.

3.1 Why is it a Hard Problem?

Binary classification is already difficult. Nobata et al. list several reasons why abusive language detection is a difficult task [34]. For example, simple detection approaches can be fooled by users who obfuscate and conceal the true meaning of their comments intentionally. Another difficulty is the use of stylistic devices in online discussions such as *irony* to express sarcasm or quoting possible problematic content. Further, language is not static: new words are introduced, other words change their meaning, and there is an ever-shifting fine line of what is barely considered legitimate to state and what not. This flexible and ever *changing language* requires a detection approach to adapt over time, for example, to neologisms. It is also unclear what classification scheme to use and how to precisely distinguish classes from each other. As a consequence of this uncertainty, researchers have come up with various annotation guidelines and resulting datasets use different labels, as seen earlier (e.g., in Table 3).

If we now switch to more fine-grained labels, we face two additional problems:

1. Reduced available training data per class
2. Increased difficulty for annotation

With a fine-grained classification, the number of available samples per class gets lower. It is a major challenge to collect enough samples per class without introducing a problematic bias to the sampling from a basic population of comments. The class imbalance complicates training neural networks and therefore countermeasures become necessary. Downsampling and upsampling alter the dataset so that there is an equal number of samples from every class. To achieve this, unnecessary samples of the majority class can be discarded or samples of the minority class can be sampled repeatedly. Another technique is to use a weighted loss function, which influences the training process: penalties for errors in the minority class are made higher than in the majority class. Another idea is the synthetic minority over-sampling technique (SMOTE) [10], which has already been used to augment a dataset of aggressive comments [43]. For both SMOTE and class weights, similar gains in increased f1-score have been reported [43].

It is essential to keep the number of trainable parameters and thus the model's capacity as small as possible if training data is limited. While GRU units have only two gates, LSTM units have three gates. GRU units are preferable because of their smaller number of parameters. The aim to keep the number of parameters small also explains the popularity of pooling layers, because they do not contain any trainable parameters. The alternative of using dense layers to combine the outputs of recurrent layers increases the number of parameters. Depending on the network architecture, multiple layers can also share their weights and thereby reduce the number of parameters. Last but not least, weight regularization can be used to limit the value range of parameters.

The second problem relates to the increased effort to annotate the training data. The inter-annotator agreement is already relatively low for binary labels when looking at all but the most obvious examples. Moreover, it gets even lower with more fine-grained classes. The boundaries between those classes are often fuzzy and the meaning of sentences depends on context, cultural background, and many more influencing factors. An insult for one person could be regarded as a legitimate utterance by another. The inherent vagueness of language makes the annotation process even for domain experts, such as forum moderators, extremely difficult. This means the focus on training data generation lies on quality, not on quantity. The flip side of this is that there is not much high quality annotated data available. One way to cope with the limited availability of annotated data besides adapting the network architecture as mentioned earlier is to make the most of the available data, e.g., by using transfer learning.

3.2 Transfer Learning

For English-language texts, large amounts of training data are available. However, for less common languages, training data is sparse and sometimes no labeled data is available at all. One way to cope with this problem is to machine-translate an English-language dataset to another language. If the machine-translation is of good quality, the annotations of the English-language comments also apply to the translated comments. For offensive language detection on German-language comments, 150,000 labeled, English comments were machine-translated to German and then used as training data [43].

In a similar way, datasets for the English language can also be augmented. The idea is to make use of slight variations in language introduced by translating a comment from, for example, English to German and then back to English. The following comments exemplify this idea (example by Risch et al. [45]):

- Original comment: “Happy Diwali.!!let’s wish the next one year health, wealth n growth to our Indian economy.”
- Comment translated to German and then back to English: “Happy Diwali, let us wish the next year health, prosperity and growth of our Indian economy.”

The word *wealth* is substituted by *prosperity*, the short form *let’s* is substituted by *let us*, and *n* is correctly extended to *and*. The augmentation by machine-translation increases the variety of words and phrases in the dataset and it also normalizes colloquial expressions. A dataset that has been augmented with this approach is available online[6].

Another idea to overcome the problems of small amounts of training data is to pre-train a neural network on different data or for a different task first. Afterward, only the last layer or several last layers of the network are fine-tuned on the actual, potentially much smaller dataset. During the fine-tuning parameters on all other layers are fixed, because these layers are assumed to have learned a generic representation of comments on the larger dataset. Only

[6] https://hpi.de/naumann/projects/repeatability/text-mining.html

task-specific parameters are trained during fine-tuning. For example, this approach has been successfully used to first pre-train on 150,000 comments with coarse-grained labels and to afterward fine-tune on 5,000 comments with fine-grained labels [43].

In the paper titled “Attention Is All You Need” Vaswani et al. propose a novel attention mechanism called transformer [52]. This attention mechanism has laid the groundwork for the following progress in pre-training deep neural networks on large text corpora and transferring these models easily to a variety of tasks. With ELMo, a technique to learn contextualized word embeddings has been proposed [40]. The key idea is that a word can be represented with different embeddings depending on its surrounding words in a particular sentence. Technically, the approach is to train bidirectional LSTMs to solve a language modeling task. With ULMFiT, a fine-tuning method called “discriminative fine-tuning” has been introduced, which allows to transfer and apply pre-trained models to a variety of tasks [24]. BERT overcomes the limitation of all previous models that input needs to be processed sequentially left-to-right or right-to-left [13].

With fine-grained classification for toxic comment detection, we can not only distinguish comments that are allowed to be published online from comments that should be deleted by moderators. The fine-grained classes can also provide a first explanation of why a comment is deleted. For example, it could be deleted because it contains an insult or a threat to the news article author. Similarly, a hate speech comment could be fine-grained classified by the target group of the attack, e.g., a particular religious or ethnic group. Such explanations for classification results increase trust in the machine-learned model. The following section goes into more detail and shines a light on explanations of neural networks for toxic comment classification.

3.3 Explanations

Explanations play an essential role in real-world recommender and classification systems. Users trust recommendations and algorithmic decisions much more if they provide an explanation as well. One example are the “other customers also bought” recommendations in e-commerce applications. By explaining why a particular product was recommended, the recommendations are considered better and more trustworthy.

In the context of user comment classification, explanations are also very much needed to establish trust in the (semi-)automatic moderation process. If no reason is provided why a user’s comment was deleted or not published in the first place, this user might get the feeling of being censored or her opinion otherwise oppressed. Therefore, a fine-grained classification is inevitable. Even if results for binary classification (“delete or not delete”) are slightly better compared to fine-grained classification results (“deleted because of x”), the latter is preferred. Explaining to users why their comment was deleted does not only help to dispel worries about censorship but also to keep the users engaged on the platform. In addition, they get educated about the way the comment sections are supposed to be used in this particular community ("Netiquette").

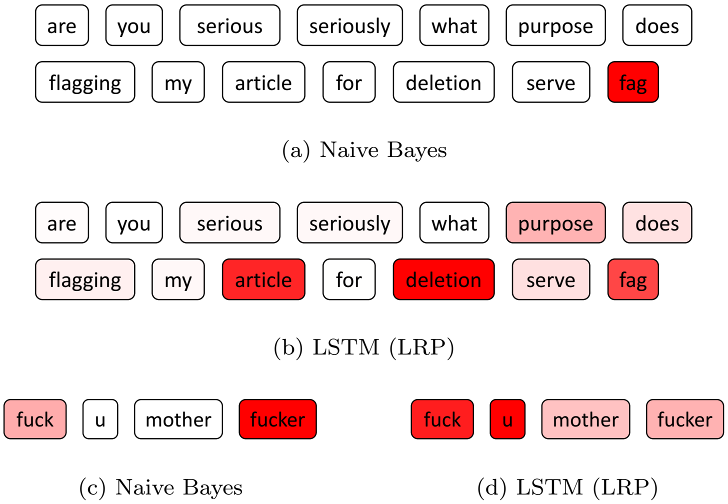
A fine-grained classification of deleted content helps to broadly categorize an offending comment but does not explain why a comment was classified into a particular class. To this end, explanations of the machine learning algorithm are needed. There is a large volume of research concerned with explaining deep learning results. For text classification, it is necessary to point towards the phrases or words that make a comment off-topic, toxic, or insulting. These kinds of explanations are beneficial to monitor the algorithm and identify problems early on. If a comment was classified as insulting because of a very common, neutral word, it can mean that the algorithm needed to be recalibrated or retrained to make comprehensible decisions.

Naive Bayes can serve as a baseline approach for explanations because it is simple and yet gives some insights. For each word in the vocabulary, we calculate the probability that a comment containing this word is classified as toxic. The naive assumption of word independence is inherent to this approach, which means word correlations are not taken into account. As a consequence, the same word is assigned the same probabilities across all comments.

Another approach, *Layer-wise relevance propagation* (LRP), has been first proposed to explain image classifications by neural networks [6]. More recently, LRP has been successfully applied to natural language processing and to sentiment analysis in particular [4, 5]. Figure 1 shows heatmaps for two example comments based on naive Bayes probabilities and LRP relevance scores for an LSTM-based neural network.[7] For naive Bayes, red boxes indicate a high conditional probability that given the occurrence of the word the comment is toxic. For LRP, red boxes indicate the relevance score in favor of the class "toxic".

The naive Bayes approach highlights only a small number of words as decisive for the classification. This problem is known as *over-localization* and has been reported as a problem also for other explanation approaches [53]. The LRP visualization reveals that the LSTM correctly identifies word pairs that refer to each other, such as "article deletion" and "fuck u". In contrast, for the naive Bayes approach "fuck" and "u" are independent words and therefore "u" is not highlighted. Figure 2 shows heatmaps for an exemplary toxic comment based on four different techniques. The comparison includes a naive Bayes approach, an LSTM-based network visualized with LRP, and a CNN visualized with LRP and pattern attribution [25]. Again, red boxes indicate probability or relevance score in favor of the class "toxic", while blue boxes indicate the opposite class "not toxic".

[7] The visualizations are based on a tool called "innvestigate" by Alber et al. [2]: https://github.com/albermax/innvestigate


are you serious seriously what purpose does
flagging my article for deletion serve fag
are you serious seriously what purpose does
flagging my article for deletion serve fag
fuck u mother fucker
fuck u mother fucker

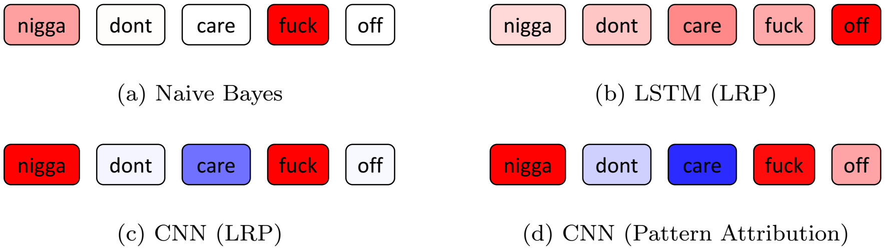
(a) Naive Bayes

(b) LSTM (LRP)

(c) Naive Bayes

(d) LSTM (LRP)

Fig. 1 Heatmaps highlight the most decisive words for the classification with a naive Bayes approach and an LSTM-based network.



(a) Naive Bayes

(b) LSTM (LRP)

(c) CNN (LRP)

(d) CNN (Pattern Attribution)

Fig. 2 Heatmaps highlight the most decisive words for the classification with a naive Bayes approach, an LSTM, and two CNNs.

4 Real-World Applications

Overwhelmed by the recent shift from a few written letters to the editor to online discussions with dozens of participants on a 24/7 basis, news platforms are drowning in vast numbers of comments. On the one hand, moderation is necessary to ensure respectful online discussions and to prevent misuse by spammers, haters, and trolls. On the other hand, moderation is also very expensive in terms of time, money, and working power. As a consequence, many online news platforms have discontinued their comment sections. Different lines of machine learning research aim to support online platforms in keeping their discussion sections open. This section covers a selection.

4.1 Semi-Automated Comment Moderation

For example, semi-automated comment moderation can support human moderators but does not completely replace them [46]. A machine learning model is trained on a binary classification task: A set of presumably appropriate comments that can be published without further assessment and a set of presumably inappropriate comments that need to be presented to a human moderator for assessment. Today's industrial applications so far refrain from using deep learning models for comment moderation due to the lack of explainability. Such black-box models do not fulfill the requirement of comprehensible classification results. Moderators and readers both want to understand the reasons behind a classification decision. Future improvements in explaining the decisions of deep neural networks are needed to apply them for comment moderation. Until then, the industry will fall back to less complex models, such as logistic regression models. These models can explain which features make a comment inappropriate in a specific context [46].

Ambroselli et al. propose a logistic regression model based on article metadata, linguistic, and topical features to predict the number of comments that an article will receive [3]. Based on these predictions, news directors can balance the distribution of highly controversial topics across a day. Thereby readers are enabled to engage in more discussions and the moderation workload is distributed evenly. Further, guiding the attention of moderators towards potentially disrespectful discussions facilitates efficient moderation. There are several studies of implemented systems that support the moderation of online discussions [3,46,48]. These discussions can also be mined to predict the popularity of news stories [47], to measure how controversial a comment is [19] or to rank comments by persuasiveness [57]. Figure 3 shows how the fraction of moderated comments varies over time. Interestingly, the peeks correlate with breaking news events.

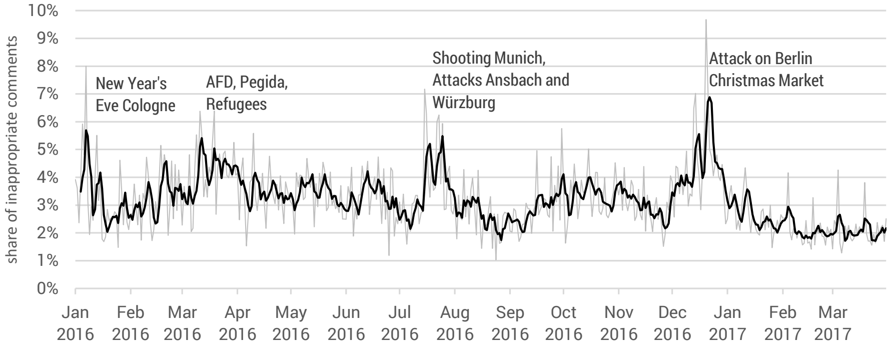


Fig. 3 The share of inappropriate comments (light gray) aggregated with a 4-day centered moving average (black) stands out at the date of specific news events.

4.2 Troll Detection

We consider malicious users of comment sections, as users who post comments with a motivation to disturb otherwise respectful discussions. In contrast to toxic comment classification, the focus is on users who attract negative attention with multiple misbehaviors. Research on malicious users in online discussions distinguishes trolls and sockpuppets [28]. Trolls characterizes that they try to disturb on-topic discussions with provoking or off-topic utterances. Hardaker defines trolls as users "...whose real intentions are to cause disruption and/or to trigger or exacerbate conflict for the purposes of their own amusement." [22].

Sockpuppets are multiple user accounts that are under the control of the same person. The latter can have multiple reasons and is not per se a problem for a discussion — although the platform's terms of use typically forbid it. For example, users who access a platform from multiple different devices might use multiple user accounts to protect their privacy and prevent tracking across their devices. Some users who forgot their account password create a new account. If they, later on, remember their old account's password, they sometimes continue to use both accounts. However, there are also malicious intents, such as to upvote own comments or argue in favor of own comments and create the impression of consensus if there is not. If there actually is a broad consensus, malicious users can use multiple accounts to create the impression of strong dissent and controversial discussions with divisive comments.

There is a publicly available dataset of 3 million tweets by almost 3,000 Twitter troll accounts[8]. These accounts are considered trolls because of their connection to a Russian organization named Internet Research Agency (IRA). IRA is a defendant in an indictment filed by the U.S. Justice Department in February 2018. The organization is characterized as a "troll factory" and is accused of having interfered with the U.S. presidential election in 2016 in a way that is prohibited by U.S. law. Fake profiles posing as U.S. activists allegedly tried to influence the election systematically. Linvill and Warren defined five different classes of IRA-associated Twitter accounts:[9]

1. **Right Trolls** support Donald Trump and other Republicans, while attacking Democrats.
2. **Left Trolls** support Bernie Sanders and criticize, for example, Hillary Clinton with divisive tweets. They also discuss socially liberal topics.

[8] https://about.twitter.com/en_us/values/elections-integrity.html#data

[9] Their article was originally published on the Resource Centre on Media Freedom in Europe according to the terms of Creative Commons: Attribution-NonCommercial 4.0 International (CC BY-NC 4.0). https://www.rcmediafreedom.eu/Publications/Academic-sources/Troll-Factories-The-Internet-Research-Agency-and-State-Sponsored-Agenda-Building, https://creativecommons.org/licenses/by-nc/4.0/

3. **News Feeds** post local, regional, and U.S. news and links to news platforms.
4. **Hashtag Gamers** post their tweets in context of a particular hashtag. By choosing popular hashtags they maximize the visibility of their tweets.
5. **Fearmongers** spread fear and panic by posting hoaxes.

Galán-García et al. trained a machine learning model to detect troll profiles in Twitter [17]. Their publication focuses on real-world applications and they prove that current models are already good enough to be beneficial for selected tasks. However, the next section deals with an error analysis for state-of-the-art models and identifies their weaknesses. We outline different directions for further research based on this analysis.

5 Current Limitations and Future Trends

Common challenges for toxic comment classification among different datasets comprise out-of-vocabulary words, long-range dependencies, and multi-word phrases [1]. To cope with these challenges, sub-word embeddings, GRUs and LSTMs, and phrase mining techniques have been developed. A detailed error analysis by van Aken et al. for an ensemble of several state-of-the-art approaches [12, 34, 34, 42, 50, 59, 60] reveals open challenges [1]. We discuss this analysis and its implications in the following.

5.1 Misclassification of Comments

Based on the analysis by van Aken et al. we discuss six common causes for misclassification [1]. We distinguish causes for false positives (non-toxic comments that are misclassified as toxic) and false negatives (toxic comments that are misclassified as non-toxic). The following examples are Wikipedia talk page and user page comments [58]. This dataset was also used in the Kaggle Challenge on Toxic Comment Classification[10].

5.1.1 Toxicity Without Swear Words

Toxicity can be conveyed without mentioning swear words. The toxic meaning is only revealed with the help of context knowledge and understanding the full sentence, as exemplified by the toxic comment: “she looks like a horse”. The word “horse” is not insulting in general. To understand the toxicity of the comment, a model needs to understand that “she” refers to a person and

[10] https://kaggle.com/c/jigsaw-toxic-comment-classification-challenge/

that “looking like a horse” is generally considered insulting if directed to a person. However, this insult is not revealed by looking at the words of the sentence independently.

In contrast to these false negatives, there are false positives that contain toxic words, although they are overall non-toxic. If a user posts a self-referencing comment, human annotators rarely consider these comments toxic, for example: “Oh, I feel like such an asshole now. Sorry, bud.”. However, the learned model focuses on the mentioned swear words, which triggers the misclassification. Taking into account a full sentence and getting its meaning still remains a challenge for deep learning approaches.

5.1.2 Quotations, References, Metaphors, and Comparisons

A problem is that state-of-the-art models are not able to take into account the context of a comment, which includes other comments in the discussion. On the one hand, examples of false positives are otherwise non-toxic comments that cite toxic comments. Because of the toxic citation, the overall comment can be misclassified as toxic. Example: “I deleted the Jews are dumb comment.”

On the other hand, an example of false negatives is the comment: “Who are you a sockpuppet for?”. The word sockpuppet is not toxic in itself. However, the accusation that another user is a sockpuppet attacks the user without addressing his or her comment itself. In Paul Graham’s hierarchy of disagreement, which lists types of arguments in a disagreement, this is the second-lowest type of argument called “Ad Hominem”.[11]

5.1.3 Sarcasm, Irony, and Rhetorical Questions

Sarcasm, irony, and rhetorical questions have in common that the meaning of the comment is different from its literal meaning. This disguise can cause false negatives in the classification. While they are not the focus of this book chapter, we at least give examples for this reported problem for toxic comment detection [34, 42]. Example comment: “hope you’re proud of yourself. Another milestone in idiocy.”. If the first sentence in this example is taken literally, there is nothing toxic about the comment. However, the user who posted the comment actually means the opposite, which is revealed by the second sentence. Other examples are rhetorical questions, which do not ask for real answers. Example: “have you no brain?!?!”. This comment is an insult because it alleges another user to act without thinking. Rhetorical questions in toxic comments often contain subtle accusations, which current approaches hardly detect.

[11] http://www.paulgraham.com/disagree.html

5.1.4 Mislabeled Comments

The annotation of toxic comments is a challenging task for several reasons. Annotation guidelines cannot consider each and every edge case. For example, a comment that criticizes and therefore cites a toxic comment is not necessarily toxic itself. Example: “No matter how upset you may be there is never a reason to refer to another editor as ‘an idiot’ ”. State-of-the-art approaches classify this comment as not toxic, although it is labeled as toxic. We argue that this comment is actually not toxic. Thus, this false negative is not a misclassification by the current models but rather a mislabeling by the annotators.

Similar to false negatives, there are false positives caused by wrong annotations. Ill-prepared annotators, unclear task definition, and the inherent ambiguity of language may cause a minority of comments in training, validation, and test dataset to be annotated wrongly. Example: “IF YOU LOOK THIS UP UR A DUMB RUSSIAN”.

5.1.5 Idiosyncratic and Rare Words

Intentionally obfuscated words, typos, slang, abbreviations, and neologisms are a particular challenge in toxic comment datasets. If there are not enough samples with these words in the training data, the learned representations (e.g., word embeddings) may not account for the true meaning of a word. Thus, wrong representations may cause misclassification. Example: “fucc nicca yu pose to be pullin up”. Similarly, the classification of the comment: “WTF man. Dan Whyte is Scottish” depends on the understanding of the term “WTF”. The amount of slang used is platform-specific. For this reason, misclassification due to rare words is twice as high for tweets than for Wikipedia talk page comments [1].

5.2 Research Directions

What is the opposite of toxic comments? High quality, engaging comments! Finding them automatically is a growing research field [14, 26, 27, 32, 35]. A possible application is to automatically choose editor picks, which are comments highlighted by the editors of a news platform. State-of-the-art work involves supervised machine-learning approaches in order to classify comments. However, all these approaches require large annotated datasets (30k annotated comments [27]), which are costly to obtain. Lampe and Resnick study whether a similar task can be accomplished by a large team of human moderators [29]. On the website Slashdot the moderators need to distinguish high- and low-quality comments in online conversations.

A different direction is to improve classification by taking into account the context of a comment. Instead of using a single comment as input, the full discussion and other context, such as the news article or user history can be used. A motivation for this additional input is the way that humans read online comments. Because of the web page layout of social networks and news platforms and the chronological order of comments, early comments receive the most attention. To read later comments, users typically need to click through dozens of subpages. For this reason, research assumes that the first few comments play a special role in setting the tone of further discussion as respectful or disrespectful [3, 8].

Dealing with biased training data is another research challenge common to many supervised machine learning approaches. One reason why this problem occurs is that the sampling of the training data is biased. For example, an annotated comment training set might include only comments from discussions of the politics section, not including comments from other sections, such as sports. This distribution might not mirror the distribution in the test set. A second type of bias is due to prejudices and stereotypes inherent to the data. A representative sample would contain this bias, although we might want to prevent our model from learning it. The research question of how to reduce bias in trained models is also addressed by a data science challenge and the corresponding dataset[12] by Kaggle and Jigsaw.

Another challenge, especially for deployed systems, is the explainability of classification decisions. This is also true for other deep learning models and not unique to comment classification. For comment moderation, explanations are not just nice to have but play an essential part in the process. As discussed earlier, explaining the automatic deletion of a comment is crucial in the context of freedom of expression. Besides, no news outlet wants to be perceived as censoring undesired opinions. Finding good, convincing explanations is therefore essential for successful comment moderation.

Good explanations are essential in semi-automated comment moderation tools to help the moderators to make the right decision. For fully automated systems, explanations are even more critical. Moreover, with the growing number of comments on platforms without moderation, such as Facebook or Twitter, more automatic systems are needed. Finding a balance between censorship and protecting individuals and groups on the web will be challenging. However, this challenge is not only a technical but also a societal and political one, with not less than democracy on the line.

[12] https://www.kaggle.com/c/jigsaw-unintended-bias-in-toxicity-classification

6 Conclusions

In this chapter, we discussed sentiment analysis for toxic comment detection. One motivation for this task is the overwhelming number of comments posted online, which needs moderation to remain engaging, respectful, and informative. Real-world applications, such as semi-automated comment moderation, can benefit from research on toxic comment detection. We defined and discussed fine-grained classification schemes for toxicity to support further progress in this field and we gave an overview of publicly available datasets and state-of-the-art neural network architectures. Toxic comment detection was also set into context with the most recent research on transfer learning and on explaining neural networks. Finally, we outlined current challenges and future directions for research in this field.

Acknowledgements We thank Betty van Aken, Carl Ambroselli, Samuele Garda, Lasse Kohlmeyer, Niklas Köhnecke, Eva Krebs, Alexander Riese, Andreas Loos, Alexander Löser, and Felix Naumann for valuable input, either through implementation or discussion of the topic.

References

1. van Aken, B., Risch, J., Krestel, R., Löser, A.: Challenges for toxic comment classification: An in-depth error analysis. In: Proceedings of the Workshop on Abusive Language Online (ALW@EMNLP), pp. 33–42 (2018)
2. Alber, M., Lapuschkin, S., Seegerer, P., Hägele, M., Schütt, K.T., Montavon, G., Samek, W., Müller, K.R., Dähne, S., Kindermans, P.J.: innvestigate neural networks! arXiv preprint arXiv:1808.04260 (2018)
3. Ambroselli, C., Risch, J., Krestel, R., Loos, A.: Prediction for the newsroom: Which articles will get the most comments? In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL), pp. 193–199. ACL (2018)
4. Arras, L., Horn, F., Montavon, G., Müller, K.R., Samek, W.: Explaining predictions of non-linear classifiers in nlp. In: Proceedings of the Workshop on Representation Learning for NLP, pp. 1–7. Association for Computational Linguistics (2016)
5. Arras, L., Montavon, G., Müller, K.R., Samek, W.: Explaining recurrent neural network predictions in sentiment analysis. In: Proceedings of the Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, pp. 159–168. Association for Computational Linguistics, Copenhagen, Denmark (2017)
6. Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.R., Samek, W.: On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS one **10**(7) (2015)
7. Badjatiya, P., Gupta, S., Gupta, M., Varma, V.: Deep learning for hate speech detection in tweets. In: Proceedings of the International Conference on World Wide Web (WWW), pp. 759–760. International World Wide Web Conferences Steering Committee (2017)

8. Berry, G., Taylor, S.J.: Discussion quality diffuses in the digital public square. In: Proceedings of the International Conference on World Wide Web (WWW), pp. 1371–1380. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland (2017)
9. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics (TACL) **5**(1), 135–146 (2017)
10. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: synthetic minority over-sampling technique. Journal of artificial intelligence research **16**(1), 321–357 (2002)
11. Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using rnn encoder–decoder for statistical machine translation. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1724–1734. Association for Computational Linguistics (2014)
12. Davidson, T., Warmsley, D., Macy, M., Weber, I.: Automated hate speech detection and the problem of offensive language. In: Proceedings of the International Conference on Web and Social Media (ICWSM), pp. 512–515 (2017)
13. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
14. Diakopoulos, N.: Picking the nyt picks: Editorial criteria and automation in the curation of online news comments. International Symposium on Online Journalism (ISOJ) **6**(1), 147–166 (2015)
15. Diakopoulos, N., Naaman, M.: Towards quality discourse in online news comments. In: Proceedings of the Conference on Computer Supported Cooperative Work (CSCW), pp. 133–142. ACM (2011)
16. Djuric, N., Zhou, J., Morris, R., Grbovic, M., Radosavljevic, V., Bhamidipati, N.: Hate speech detection with comment embeddings. In: Proceedings of the International Conference on World Wide Web (WWW), pp. 29–30. International World Wide Web Conferences Steering Committee (2015)
17. Galán-García, P., Puerta, J.G.d.l., Gómez, C.L., Santos, I., Bringas, P.G.: Supervised machine learning for the detection of troll profiles in twitter social network: Application to a real case of cyberbullying. Logic Journal of the IGPL **24**(1), 42–53 (2016)
18. Gambäck, B., Sikdar, U.K.: Using convolutional neural networks to classify hate-speech. In: Proceedings of the Workshop on Abusive Language Online (ALW@ACL), pp. 85–90 (2017)
19. Gómez, V., Kaltenbrunner, A., López, V.: Statistical analysis of the social network and discussion threads in slashdot. In: Proceedings of the International Conference on World Wide Web (WWW), pp. 645–654. ACM (2008)
20. Graves, A.: Generating sequences with recurrent neural networks. arXiv preprint arXiv:1308.0850 (2013)
21. Guberman, J., Schmitz, C., Hemphill, L.: Quantifying toxicity and verbal violence on twitter. In: Proceedings of the Conference on Computer Supported Cooperative Work (CSCW), pp. 277–280. ACM, New York, NY, USA (2016)
22. Hardaker, C.: Trolling in asynchronous computer-mediated communication: From user discussions to academic definitions. Journal of Politeness Research. Language, Behaviour, Culture **6**, 215–242 (2010)
23. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation **9**(8), 1735–1780 (1997)
24. Howard, J., Ruder, S.: Universal language model fine-tuning for text classification. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL), pp. 328–339. Association for Computational Linguistics, Melbourne, Australia (2018)

25. Kindermans, P.J., Schütt, K.T., Alber, M., Müller, K.R., Erhan, D., Kim, B., Dähne, S.: Learning how to explain neural networks: PatternNet and PatternAttribution. arXiv preprint arXiv:1705.05598 (2017)
26. Kolhatkar, V., Taboada, M.: Constructive language in news comments. In: Proceedings of the Workshop on Abusive Language Online (ALW@ACL), pp. 11–17 (2017)
27. Kolhatkar, V., Taboada, M.: Using new york times picks to identify constructive comments. In: Proceedings of the Workshop: Natural Language Processing meets Journalism@EMNLP, pp. 100–105 (2017)
28. Kumar, S., Shah, N.: False information on web and social media: A survey. arXiv preprint arXiv:1804.08559 (2018)
29. Lampe, C., Resnick, P.: Slash (dot) and burn: distributed moderation in a large online conversation space. In: Proceedings of the Conference on Human Factors in Computing Systems (CHI), pp. 543–550. ACM (2004)
30. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems (NIPS), pp. 3111–3119 (2013)
31. Mnih, V., Heess, N., Graves, A., Kavukcuoglu, K.: Recurrent models of visual attention. In: Advances in Neural Information Processing Systems (NIPS), pp. 2204–2212 (2014)
32. Napoles, C., Pappu, A., Tetreault, J.R.: Automatically identifying good conversations online (yes, they do exist!). In: Proceedings of the International Conference on Web and Social Media (ICWSM), pp. 628–631 (2017)
33. Napoles, C., Tetreault, J., Pappu, A., Rosato, E., Provenzale, B.: Finding good conversations online: The yahoo news annotated comments corpus. In: Proceedings of the Linguistic Annotation Workshop (LAW), pp. 13–23 (2017)
34. Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., Chang, Y.: Abusive language detection in online user content. In: Proceedings of the International Conference on World Wide Web (WWW), pp. 145–153. International World Wide Web Conferences Steering Committee (2016)
35. Park, D., Sachar, S., Diakopoulos, N., Elmqvist, N.: Supporting comment moderators in identifying high quality online news comments. In: Proceedings of the Conference on Human Factors in Computing Systems (CHI), pp. 1114–1125. ACM (2016)
36. Park, J.H., Fung, P.: One-step and two-step classification for abusive language detection on twitter. In: Proceedings of the Workshop on Abusive Language Online (ALW@ACL), pp. 41–45. Association for Computational Linguistics, Vancouver, BC, Canada (2017)
37. Pavlopoulos, J., Malakasiotis, P., Androutsopoulos, I.: Deeper attention to abusive user content moderation. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1125–1135. Association for Computational Linguistics, Copenhagen, Denmark (2017)
38. Pavlopoulos, J., Malakasiotis, P., Bakagianni, J., Androutsopoulos, I.: Improved abusive comment moderation with user embeddings. In: Proceedings of the Workshop on Natural Language Processing meets Journalism (co-located with EMNLP), pp. 51–55. Association for Computational Linguistics, Copenhagen, Denmark (2017)
39. Pennington, J., Socher, R., Manning, C.: Glove: Global vectors for word representation. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543 (2014)
40. Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer, L.: Deep contextualized word representations. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL), pp. 2227–2237. Association for Computational Linguistics, New Orleans, Louisiana (2018)

41. Pitsilis, G.K., Ramampiaro, H., Langseth, H.: Effective hate-speech detection in twitter data using recurrent neural networks. Applied Intelligence **48**(12), 4730–4742 (2018)
42. Qian, J., ElSherief, M., Belding-Royer, E.M., Wang, W.Y.: Leveraging intra-user and inter-user representation learning for automated hate speech detection. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL), pp. 118–123 (2018)
43. Risch, J., Krebs, E., Löser, A., Riese, A., Krestel, R.: Fine-grained classification of offensive language. In: Proceedings of GermEval (co-located with KONVENS), pp. 38–44 (2018)
44. Risch, J., Krestel, R.: Measuring and facilitating data repeatability in web science. Datenbank-Spektrum DOI 10.1007/s13222-019-00316-9
45. Risch, J., Krestel, R.: Aggression identification using deep learning and data augmentation. In: Proceedings of the Workshop on Trolling, Aggression and Cyberbullying (TRAC@COLING), pp. 150–158 (2018)
46. Risch, J., Krestel, R.: Delete or not delete? semi-automatic comment moderation for the newsroom. In: Proceedings of the Workshop on Trolling, Aggression and Cyberbullying (TRAC@COLING), pp. 166–176 (2018)
47. Rizos, G., Papadopoulos, S., Kompatsiaris, Y.: Predicting news popularity by mining online discussions. In: Proc. of the Int. Conf. on World Wide Web Companion (WWW), pp. 737–742. International World Wide Web Conferences Steering Committee (2016)
48. Schabus, D., Skowron, M.: Academic-industrial perspective on the development and deployment of a moderation system for a newspaper website. In: Proceedings of the Language Resources and Evaluation Conference (LREC), pp. 1602–1605 (2018)
49. Schabus, D., Skowron, M., Trapp, M.: One million posts: A data set of german online discussions. In: Proceedings of the International Conference on Research and Development in Information Retrieval (SIGIR), pp. 1241–1244 (2017)
50. Schmidt, A., Wiegand, M.: A survey on hate speech detection using natural language processing. In: Proceedings of the International Workshop on Natural Language Processing for Social Media, pp. 1–10 (2017)
51. Stroud, N.J., Van Duyn, E., Peacock, C.: News commenters and news comment readers. Engaging News Project pp. 1–21 (2016)
52. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems (NIPS), pp. 5998–6008 (2017)
53. Wang, C.: Interpreting neural network hate speech classifiers. In: Proceedings of the Workshop on Abusive Language Online (ALW@EMNLP), pp. 86–92. Association for Computational Linguistics, Brussels, Belgium (2018)
54. Waseem, Z.: Are you a racist or am i seeing things? annotator influence on hate speech detection on twitter. In: Proceedings of the Workshop on NLP and Computational Social Science, pp. 138–142. Association for Computational Linguistics, Austin, Texas (2016)
55. Waseem, Z., Davidson, T., Warmsley, D., Weber, I.: Understanding abuse: A typology of abusive language detection subtasks. In: Proceedings of the Workshop on Abusive Language Online (ALW@ACL), pp. 78–84. Association for Computational Linguistics, Vancouver, BC, Canada (2017)
56. Waseem, Z., Hovy, D.: Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In: Proceedings of the Student Research Workshop@NAACL, pp. 88–93. Association for Computational Linguistics, San Diego, California (2016)
57. Wei, Z., Liu, Y., Li, Y.: Is this post persuasive? ranking argumentative comments in online forum. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL), vol. 2, pp. 195–200 (2016)

58. Wulczyn, E., Thain, N., Dixon, L.: Ex machina: Personal attacks seen at scale. In: Proceedings of the International Conference on World Wide Web (WWW), pp. 1391–1399. International World Wide Web Conferences Steering Committee (2017)
59. Zhang, Z., Luo, L.: Hate speech detection: A solved problem? the challenging case of long tail on twitter. Semantic Web Journal pp. 1–21 (2018)
60. Zhang, Z., Robinson, D., Tepper, J.: Detecting hate speech on twitter using a convolution-gru based deep neural network. In: European Semantic Web Conference, pp. 745–760. Springer (2018)