Challenges of Automatically Detecting Offensive Language Online: Participation Paper for the Germeval Shared Task 2018 (HaUA)

Tom De Smedt
University of Antwerp
Computational Linguistics Research Group
Experimental Media Research Group
tom.desmedt@uantwerpen.be

Sylvia Jaki
University of Hildesheim
Department of Translation and
Specialized Communication
jakisy@uni-hildesheim.de

Abstract

This paper presents our submission (HaUA) for Germeval Shared Task 1 (Binary Classification) on the identification of offensive language. With feature selection and features such as character n-grams, offensive word lexicons, and sentiment polarity, our SVM classifier is able to distinguish between offensive and non-offensive German-language tweets with an in-domain F_1 score of 88.9%. In this paper, we report our methodology and discuss machine learning problems such as imbalance, overfitting, and the interpretability of machine learning algorithms. In the discussion section, we also briefly go beyond the technical perspectives and argue for a thorough discussion of the dilemma between internet security and freedom of speech, and what kind of language we are actually predicting with such algorithms.

1 Introduction

The new German Netzwerkdurchsetzungsgesetz law (NetzDG) allows for the removal of illegal content posted on social media platforms, where *illegal* pertains to one of 21 elements of offense according to the German Strafgesetzbuch. Recent reports expose several points of interest (Brühl and von Au, 2018). Firstly, the most common reasons for suspension on Twitter are incitement to hatred (§130), insults (§185), unconstitutional symbols (§86a), incitement to crime (§111), and pornography (§184). Secondly, only a fraction of the reported content has been blocked (11%, or 28,645 out of 264,828 tweets). Thirdly, primarily relating to Facebook, the decision-making is not transparent, with various reported cases of under- and overblocking. As a result, it is not surprising that many people feel that the current situation, in which for-profit IT companies independently decide what should be removed, is undesirable.

The recent surge of workshops on offensive language such as this year's Shared Task, and the large number of participants, reveals a commitment of the linguistics community to collaborate towards a safer internet, by providing algorithms that can help to detect abusive content online. In this workshop, comparing approaches, methods, and opinions will foster advances in the long run, which may be useful to German policy makers and human-rights organizations to counter online polarization and the proliferation of hate.

In our contribution, we have paid attention to the ethical consequences of releasing AI in the wild. We can offer a model that is not perfect, but interpretable. In section 2, we will discuss a brief analysis of the training data. In section 3, we will discuss the (unknown) test data and how we have approximated it by in-domain and cross-domain evaluation. We will then describe our algorithm in section 4, and zoom in on the model's features in section 5 and methods for feature selection in section 6. After the technical report, we briefly discuss some implications of our approach and challenges that, as of yet, cannot be solved with automatic NLP techniques alone in section 7.

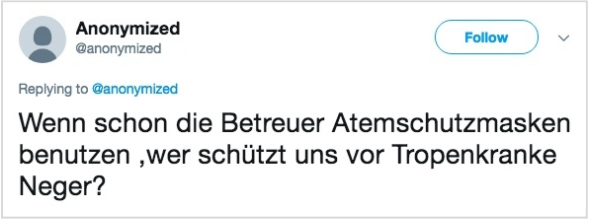

Figure 1a: Example OFFENSE tweet.

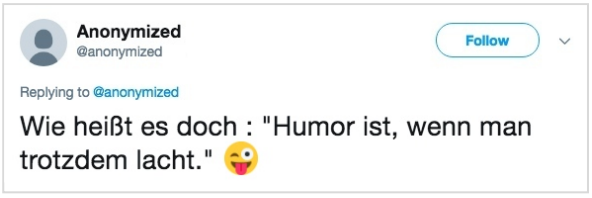

Figure 1b: Example OTHER tweet.

2 Training Data

The training data for the Shared Task consists of 5,009 manually annotated German tweets, each about 70-210 characters long, of which 1,688 are labeled OFFENSE (33.7% or about 1/3) and 3,321 are labeled OTHER (66.3% or about 2/3). Tweets labeled OFFENSE use offensive language (Fig 1).

2.1 Data Distribution

The training data is imbalanced (1:2 ratio), which reflects reality – assuming most Twitter users will not post offensive tweets – but which can also be problematic, since classifiers tend to be "overwhelmed" by the majority class (Chawla et al., 2004). Solutions for imbalanced data that are reported to be effective include undersampling, i.e., discarding training examples of the majority class until the data is balanced, oversampling, e.g., training on examples of the minority class multiple times, and feature selection, removing ambiguous features to increase the separability of the classes. We tested with both undersampling and oversampling as well as feature selection, where oversampling + feature selection seems to work best in our case (~ +5% F_1 score).

2.2 Data Entropy

Ideally, a given machine learning algorithm will automatically discover features in the training data that can be used to predict whether unknown tweets are OFFENSE or OTHER. Such features might be words like *Scheiße* that are statistically biased, i.e., occurring more often in offensive tweets. To get a sense of the biased words in the data, we used the chi-squared test ($p \leq 0.01$; see also Liu and Motoda, 2007) with word counts per class to expose them. The results[1] are in line with what we observed in previous work on German far-right propaganda (Jaki and De Smedt, 2018) and jihadist extremism (De Smedt et al., 2018).

[1] https://docs.google.com/spreadsheets/d/1Q3fLs4mfjWEWYJtv26ddUd8jk_1Tz2svk-94LxtONwA

Broadly, offensive tweets seem to be marked by:

- **defamation** (often political opponents), e.g., *Gutmensch*, *Nazi*, *Volksverräterin* (do-gooder, fascist, traitor of the people),
- **dehumanization** (refugees), e.g., *Abschaum*, *Pack*, *Schmarotzer* (scum, rabble, parasites),
- **stereotyping**, e.g., *Kanakenstadt*, *Museldiebe* (Turk town, Muslim thieves),
- **racism**, e.g., *Nafris*, *Neger* (North African repeat offenders, niggers),
- **profanity**, e.g., *Arsch*, *Dreck*, *Scheiße* (ass, crap, shit),
- **negativity**, e.g., *dumm*, *Gelaber*, *kotzen*, 🤢 (dumb, drivel, to vomit),
- **capitalization**, e.g., *DEUTSCH*, *ISLAM*, *LINKS* (German, Islam, left),
- **propaganda** and **fake news** posted by known user profiles (see Netzpolitik, 2017).

About 50% of the most biased nouns exposed by the chi-squared test occur in our automatically generated list of offensive words, which is an important feature in our model (see section 5).

3 Test Data

The Shared Task 1 entails a test dataset of 3,532 German tweets for which we have to accurately predict either OFFENSE or OTHER.

3.1 In-domain Evaluation

Various statistical techniques exist to predict how well our trained classifier is going to perform. Most notably, *k*-fold cross-validation partitions the training data into *k* training / test subsets and reports the average recall and precision, where recall is an estimation of *how many* offensive tweets are found, and precision is an estimation of how many reported offensive tweets *are really* offensive (henceforth called the IN evaluation). For example, a classifier with 75% recall finds 3/4 of offensive tweets (1/4 goes by undetected). A classifier with 75% precision mislabels 1/4 of "normal" tweets as offensive.

The main drawback of this approach is that it only reports in-domain performance, it assumes that unknown tweets on which the classifier will eventually be applied will have features identical to those in the training data, which may be false.

3.2 Cross-domain Evaluation

Domain adaptation refers to a machine learning problem where a classifier seems to perform well on its training data (in-domain performance) but not on related data (out-of-domain performance). To test the scalability of our classifier, we cut 500 tweets (~10%) from the training data as a holdout testing set, for which we know the class labels (henceforth called the OUT evaluation). Since we do not know the distribution of the class labels in the *actual* test data, we did three runs with the holdout set having respectively a 1:1 (250/250), 1:2 (150/300), and 1:4 (100/400) ratio of OFFENSE/OTHER tweets.

We also used a manually annotated subset of Jaki and De Smedt (2018) for testing (henceforth called the CROSS evaluation). This set consists of 800 German right-wing extremist tweets with offensive language + 1,600 other German tweets. The 1:2 ratio means that a classifier that always predicts OTHER (the majority class) would score F_1 44% on this data. We can use this as a baseline for our classifier (see also Table 2 & 3). In other words, it must score at least F_1 45% to have any predictive value.

4 Algorithm

We used the LIBSVM machine learning algorithm (Chang and Lin, 2011) in the Pattern toolkit for Python (De Smedt and Daelemans, 2012) to train our classifier, and Pattern helper functions.

4.1 Interpretability

No doubt, the most recent multi-layered neural networks ("Deep Learning") will achieve better results, especially in combination with word embeddings. The downside of deep neural nets is that their decision-making might be difficult to interpret (Lipton, 2016). This is problematic once such systems are applied in the wild: as of yet, there is still ongoing debate as to what exactly constitutes offensive language / hate speech, and laws such as NetzDG tend to be vague (Human Rights Watch, 2018). Introducing "black box" AI systems to the decision-making may be morally questionable and may jeopardize the freedom of expression (see section 7), particularly in light of the new privacy protection regulations in the EU (GDPR; European Commission, 2018).

By comparison, classic machine learning algorithms such as *k*-NN, decision trees, and linear SVMs are often more interpretable. In fact, in our tests a lexicon of offensive words with confidence scores (e.g., *autoritär* = 0.5) is only about 3% less accurate and might also be useful, e.g., offensive words can be visually highlighted for human moderators.

5 Features

The LIBSVM algorithm expects its input to be vectorized, where each tweet is represented as a vector of feature/weight pairs. The features could be the words that appear in the tweet and the weights could be word count. In our case, we use lexical features such as character trigrams, e.g., *Scheiß* → { *Sch*, *che*, *hei*, *eiß* }, and binary weights, i.e., a feature is present or not. Character n-grams efficiently capture spelling errors, word endings, function words, emoticons, and so on. For example, *Scheiß* and *Scheiss* have multiple matching trigrams (*Sch*, *che*, *hei*).

An overview of the features we used:

- each tweet is lowercased: *Dreck* → *dreck*,
- `C1`, character 1-grams, e.g., *d*, *r*, *e*, *c*, *k*,
- `C3`, character 3-grams, e.g., *dre*, *rec*, *eck*,
- `C5`, character 5-grams, e.g., *dreck*,
- `W1`, word 1-grams, e.g., *dreckiger*,
- `W2`, word 2-grams, e.g., *dreckiger neger*,
- `W3`, word 3-grams, e.g., *neger dürfen bleiben*,
- `UP`, if tweet has +40% uppercase characters,
- `!!`, if tweet has 2+ exclamation marks,
- `O?`, if tweet has an offensive word,
- `O+`, if tweet has 2+ offensive words,
- `O%`, if tweet has *autoritär* (for example) then a feature `O%50` will be present,
- `:(`, if tweet has a negative polarity.

Offensive words are those words that occur in our automatically generated lexicon of 1,750 words and their confidence scores. To populate the lexicon, we started with 50 high-precision seed words to which we assigned a score (e.g., *Abfall* = 0.50, *Arsch* = 0.75, *Gesindel* = 1.00) and then queried the German Twitter Embeddings (Ruppenhofer, 2018) to find semantically similar words (Mikolov et al., 2013).

For each seed word, we then took the 100 most similar words (*Gesindel* → 81% *Dreckspack*), propagated the seed score (1.00 x 0.81 = 0.81), and then assigned new words to one of five bins (`0.00 | 0.25 | 0.50 | 0.75 | 1.00`; e.g., *Dreckspack* = 0.75, *Schnurrbart* = 0.25).

Sentiment analysis (Pang and Lee, 2008) refers to the task of automatically detecting the polarity (positive or negative tone) of a text. Polarity was predicted using a Perceptron classifier trained on German tweets containing emoji from the POLLY corpus (De Smedt and Jaki, 2018). The model is about 85% accurate. For example, *sehr schöner Urlaub!* (very nice holiday!) is labeled positive while *islamgeile Propaganda* (Islam-loving propaganda) is labeled negative.

Using this set of features, the LIBSVM algorithm trained on the given data (1:2 OFFENSE/OTHER) yields recall 75.8% and precision 78.7% with in-domain 10-fold cross-validation.

Table 1 provides an overview of performance (i.e., F_1 score = mean of recall and precision) for different combinations of features. Interestingly, offensive words and shape features are nearly as predictive (**`O`** + **`UP`** + **`!!`** = 74.5%) as all features combined (77.2%). However, the best results are achieved by applying feature selection (FSEL, i.e., removing noisy features), which raises F_1 score from 77.2% to 88.9% (1 mistake per 10 tweets).

6 Feature Selection

Using this set of features, the trained model (after holdout) has about 250K features in total. Each tweet has about 350 features. To improve the performance for imbalanced data, we computed the posterior probability of each feature (e.g., *der* = 50% OFFENSE vs 50% OTHER, and *Dreckspack* = 100% OFFENSE vs 0% OTHER) and removed the most ambiguous ones with probabilities between 25% and 75% until each vector has at most 100 features. This removes about 50K features in total: 90% of **`C1`** (e.g., @ is too noisy), 50% of **`C3`**, 25% of **`W1`** (e.g., *skeptisch* is too noisy), 10% of **`W2`** (e.g., *und mit*), and so on.

6.1 Model Overfitting

This raises the F_1 score by about 10% for the IN evaluation, from 77.2% to 88.9% (recall 87.3% and precision 90.6%). We can remove even more features, eventually training a model that has 99% in-domain performance, but which also has no features left to fit out-of-domain data. This is known as overfitting (Hawkins, 2004). To assess whether we might be overfitting our classifier, we tested on the OUT and CROSS sets. In general, our feature selection method raises F_1 score by about 2% on the OUT set (with varying OFFENSE/OTHER distributions) and by 6% on the CROSS set (see Table 2 for an overview). Removing more features lowers the F_1 score on both sets.

6.2 Model Oversampling

We also experimented with undersampling and oversampling to boost performance. For given training data of ~1,500 OFFENSE + 3,000 OTHER tweets, we either removed 1,500 OTHER tweets (= undersampled 1500/1500) or trained OFFENSE tweets twice (= oversampled 3000/3000).

Table 2 provides an overview of performance (F_1) for the imbalanced and balanced classifiers, with or without feature selection (100 vs 350), on the in-domain (IN) and cross-domain tests (OUT set of 500 tweets, CROSS set of 800/1600 political tweets). Oversampling combined with feature selection works well if there are less OFFENSE than OTHER tweets. With a 1:4 ratio the F_1 score is about 76% on the OUT set, and about 70% on the CROSS set with a 1:2 ratio, which is above the 44% majority class baseline.

Table 3 provides an error analysis with recall and precision by class, as measured on the OUT 1:4 (100/400) test set, which we think is the most representative of real-life. Not surprisingly, most classification errors occur in the OFFENSE class. The best AUC score (Area Under Curve) is 0.83 for the oversampled model with feature selection.

This is the classifier that we submitted for the Shared Task 1 (`HaUA-coarse`).

7 Discussion

Our tests with domain adaptation highlight the importance of clearly defining what exactly we are detecting. To illustrate this: There is overlap between the task's training data and the CROSS data we used. Looking at retweeted usernames, both sets appear to draw from the same sources, but where the CROSS data focuses on politically motivated hate speech grounded in racism, the task's data focuses on disrespect and contempt of individuals and groups (who are not necessarily refugees or political factions). The difference is

subtle, and there is some overlap in performance, however it is not a perfect fit. The divergence is in part due to different views of what constitutes offensive language online. Profanity like *Scheiße* is unacceptable by Ruppenhofer et al. (2018: 2) while the CROSS data focuses more exclusively on ideologically disparaging language.

This stresses the need to discuss how we will operationalize regulations on a linguistic level. Which "bad content" should AI be detecting? Do we train systems according to society's norms of what is inappropriate, or legal definitions? This means that the challenge is not purely linguistic but also societal and political (cf. Ruppenhofer et al., 2018: 4). What language can we ethically and legally justify to remove from the internet?

There is little doubt that content classified as illegal by the German Strafgesetzbuch should be removed, justifying the need for AI tools. People who criticize NetzDG claim that it infringes on freedom of speech, which is anchored in German Grundgesetz, but they forget that these freedoms are also limited by StGB. Apart from such cases, there is admittedly a grey area between offensive language and freedom of speech. For example, what is the line between an expressed opinion of a foreign culture and incitement to hatred? To avoid the shadow of censorship, policy makers should not be satisfied with the current legal situation, but strive to continue the discussion about the boundaries of freedom of speech and the measures to take against offensive behavior on social media.

Acknowledgements

The authors wish to thank "Schmutzi" for giving insight into profanity, slurs and slang language in online social media.

References

Jannis Brühl and Caspar van Au. 2018. Was das NetzDG mit Deutschland macht. *Süddeutsche Zeitung*. https://www.sueddeutsche.de/digital/bilanz-was-das-netzdg-mit-deutschland-macht-1.4072480

Nitesh Chawla, Nathalie Japkowicz, and Aleksander Kotcz. 2004. Special issue on learning from imbalanced data sets. *ACM SIGKDD*, 6(1):1–6.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: a library for support vector machines. *ACM TIST*, 2(3), 27.

Tom De Smedt and Walter Daelemans. 2012. Pattern for Python. *JMLR*, 13:2063–2067.

Tom De Smedt, Guy De Pauw, and Pieter Van Ostaeyen. 2018. Automatic detection of online jihadist hate speech. *CLiPS CTRS*, 7:1–30.

Tom De Smedt and Sylvia Jaki. 2018. The Polly corpus: online political debate in Germany. In *Proceedings of CMC and Social Media Corpora*.

Douglas M. Hawkins. 2004. The problem of overfitting. *ACS JCIM*, 44(1):1–12.

Sylvia Jaki and Tom De Smedt. 2018, submitted. Right-wing German hate speech on Twitter: analysis and automatic detection.

European Commission. 2018. Data protection. https://ec.europa.eu/info/law/law-topic/data-protection_en

Zachary C. Lipton. 2018. The mythos of model interpretability. *Queue*, 16(3).

Human Rights Watch. 2018. Germany: flawed social media law. https://www.hrw.org/news/2018/02/14/germany-flawed-social-media-law

Huan Liu and Hiroshi Motoda (eds.). 2007. *Computational methods of feature selection*. Chapman & Hall/CRC, Boca Raton.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, 2(1/2):1–135.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. arXiv:1301.3781

Netzpolitik. 2017. Datenrecherche: offizielle AfD-Accounts retweeten Neonazi-Kanal auf Twitter. https://netzpolitik.org/2017/datenrecherche-offizielle-afd-accounts-retweeten-neonazi-kanal-auf-twitter/

Josef Ruppenhofer. 2018. German Twitter embeddings. http://www.cl.uni-heidelberg.de/english/research/downloads/resource_pages/GermanTwitterEmbeddings/GermanTwitterEmbeddings_data.shtml

Josef Ruppenhofer, Melanie Siegel, and Michael Wiegand. 2018. Guidelines for IGGSA Shared Task on the Identification of Offensive Language. http://www.coli.uni-saarland.de/~miwieg/Germeval/guidelines-iggsa-shared.pdf

C1	C3	C5	W1	W2	W3	UP	!!	O?	O+	O%	:(	FSEL	F1
✓	-	-	-	-	-	-	-	-	-	-	-	-	59.1%
✓	✓	-	-	-	-	-	-	-	-	-	-	-	68.4%
✓	✓	✓	-	-	-	-	-	-	-	-	-	-	72.5%
✓	✓	✓	✓	-	-	-	-	-	-	-	-	-	73.3%
✓	✓	✓	✓	✓	-	-	-	-	-	-	-	-	73.1%
✓	✓	✓	✓	✓	✓	-	-	-	-	-	-	-	73.4%
✓	✓	✓	✓	✓	✓	✓	-	-	-	-	-	-	73.8%
✓	✓	✓	✓	✓	✓	✓	✓	-	-	-	-	-	74.0%
✓	✓	✓	✓	✓	✓	✓	✓	✓	-	-	-	-	74.9%
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-	-	-	76.2%
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-	-	77.2%
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-	✓	**88.9%**
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-	76.6%
-	-	-	-	-	-	✓	✓	✓	✓	✓	-	-	74.5%

Table 1: Performance (F1) for 10-fold cv on 1,500 OFFENSE + 3,000 OTHER tweets represented as character n-grams (C), word n-grams (W), offensive words (O), and after feature selection (FSEL).

Model		# features	IN 10-fold cv	OUT			CROSS
				1:1	1:2	1:4	1:2
imbalanced	1500/3000	350	77%	72%	72%	70%	63%
balanced	1500/1500	350	76%	77%	73%	69%	64%
balanced	3000/3000	350	91%	72%	71%	70%	64%
imbalanced	1500/3000	100	89%	72%	73%	73%	70%
balanced	1500/1500	100	88%	77%	75%	71%	70%
balanced	3000/3000	100	96%	73%	75%	**76%**	70%
baseline	-	-	-	33%	40%	44%	44%

Table 2: Performance (F1) for balanced/imbalanced classifiers using 10-fold cv (IN), on holdout sets with different OFFENSE/OTHER distributions (OUT), and on a set labeled by other authors (CROSS).

Model		# features	OUT 1:4				AUC
			OFFENSE		OTHER		
			P	R	P	R	
imbalanced	1500/3000	350	49%	57%	89%	85%	0.77
balanced	1500/1500	350	42%	69%	90%	76%	0.74
balanced	3000/3000	350	49%	57%	89%	85%	0.77
imbalanced	1500/3000	100	57%	56%	89%	89%	0.80
balanced	1500/1500	100	45%	71%	92%	78%	0.76
balanced	3000/3000	100	62%	60%	90%	91%	**0.83**
baseline	-	-	0%	0%	80%	100%	0.50

Table 3. Precision and Recall by class label and AUC score for balanced/imbalanced classifiers, as measured on the holdout set with 1:4 ratio of 100 OFFENSE + 400 OTHER tweets.