Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 3498–3508
Marseille, 11–16 May 2020

Offensive Language and Hate Speech Detection for Danish

Gudbjartur Ingi Sigurbergsson, Leon Derczynski
IT University of Copenhagen
Denmark
ld@itu.dk

Abstract

The presence of offensive language on social media platforms and the implications this poses is becoming a major concern in modern society. Given the enormous amount of content created every day, automatic methods are required to detect and deal with this type of content. Until now, most of the research has focused on solving the problem for the English language, while the problem is multilingual. We construct a Danish dataset DKHATE containing user-generated comments from various social media platforms, and to our knowledge, the first of its kind, annotated for various types and target of offensive language. We develop four automatic classification systems, each designed to work for both the English and the Danish language. In the detection of offensive language in English, the best performing system achieves a macro averaged F1-score of 0.74, and the best performing system for Danish achieves a macro averaged F1-score of 0.70. In the detection of whether or not an offensive post is targeted, the best performing system for English achieves a macro averaged F1-score of 0.62, while the best performing system for Danish achieves a macro averaged F1-score of 0.73. Finally, in the detection of the target type in a targeted offensive post, the best performing system for English achieves a macro averaged F1-score of 0.56, and the best performing system for Danish achieves a macro averaged F1-score of 0.63. Our work for both the English and the Danish language captures the type and targets of offensive language, and present automatic methods for detecting different kinds of offensive language such as hate speech and cyberbullying.

Keywords: abusive language, hate speech detection, Danish

1. Introduction

Offensive language in user-generated content on online platforms and its implications has been gaining attention over the last couple of years. This interest is sparked by the fact that many of the online social media platforms have come under scrutiny on how this type of content should be detected and dealt with. It is, however, far from trivial to deal with this type of language directly due to the gigantic amount of user-generated content created every day. For this reason, automatic methods are required, using natural language processing (NLP) and machine learning techniques. The task of finding these poses a pressing and formidable challenge (Vidgen et al., 2019).

Given the fact that the research on offensive language detection has to a large extent been focused on the English language, we set out to explore the design of models that can successfully be used for both English and Danish. To accomplish this, an appropriate dataset must be constructed, annotated with the guidelines described in Zampieri et al. (2019a). We furthermore set out to analyze relevant linguistic features by analyzing the patterns that prove hard to detect.

2. Background

Offensive language varies greatly, ranging from simple profanity to much more severe types of language. One of the more troublesome types of language is hate speech and the presence of hate speech on social media platforms has been shown to be in correlation with hate crimes in real life settings (Müller and Schwarz, 2018). It can be quite hard to distinguish between generally offensive language and hate speech as few universal definitions exist (Davidson et al., 2017). There does, however, seem to be a general consensus that hate speech can be defined as language that targets a group with the intent to be harmful or to cause social chaos. This targeting is usually done on the basis of some characteristics such as race, color, ethnicity, gender, sexual orientation, nationality or religion (Schmidt and Wiegand, 2017). Offensive language, on the other hand, is a more general category containing any type of profanity or insult. Hate speech can, therefore, be classified as a subset of offensive language. (Zampieri et al., 2019a) propose guidelines for classifying offensive language as well as the type and the target of offensive language. These guidelines capture the characteristics of generally offensive language, hate speech and other types of targeted offensive language such as cyberbullying. However, despite offensive language detection being a burgeoning field, no dataset yet exists for Danish (Kirkedal et al., 2019) despite this phenomenon being present and readily detectable in this language (Derczynski et al., 2019).

Many offensive and harmful language detection sub-tasks have been considered, ranging from detection of general offensive language, to focused tasks such as detecting hate speech (Davidson et al., 2017) and cyberbullying (Van Hee et al., 2015b).

A key aspect in the research of automatic classification methods for language of any kind is having substantial amount of high quality data that reflects the goal of the task at hand, and that also contains a decent amount of samples belonging to each of the classes being considered. To approach this problem as a supervised classification task the data needs to be annotated according to a well-defined annotation schema that clearly reflects the problem statement. The quality of the data is of vital importance, since low quality data is unlikely to provide meaningful results.

2.1. Cyberbullying

This is commonly defined as targeted insults or threats against an individual (Zampieri et al., 2019a). Three factors are mentioned as indicators of cyberbullying: in-

tent to cause harm, repetitiveness, and an imbalance of power (Van Hee et al., 2015b). This type of behaviour most commonly occurs among children and teenagers. Cyberbullying acts are prohibited by law in several countries, as well as many US states (Gregorie, 2001).

Van Hee et al. (2015a) focus on classifying cyberbullying events in Dutch. They define cyberbullying as textual content published online by an individual that is aggressive or hurtful toward a victim. Their annotation consists of two steps. In the first step, a three-point harmfulness score is assigned to each post as well as a category denoting the authors role (i.e. harasser, victim, or bystander). In the second step a more refined categorization is applied, by annotating the posts using the following labels: *Threat/Blackmail*, *Insult*, *Curse/Exclusion*, *Defamation*, *Sexual Talk*, *Defense*, and *Encouragement to the harasser*.

2.2. Hate Speech

As discussed in Section 1., hate speech is generally defined as language that is targeted towards a group, with the intend to be harmful or cause social chaos. Hate speech is prohibited by law in many countries, although the definitions may vary. Article 20 of the *International Covenant on Civil and Political Rights (ICCPR)* states that *"Any advocacy of national, racial or religious hatred that constitutes incitement to discrimination, hostility or violence shall be prohibited by law"* (Joseph and Castan, 2013). In Denmark, hate speech is prohibited by law, and is formally defined as public statements where a group is threatened, insulted, or degraded on the basis of characteristics such as nationality, ethnicity, religion, or sexual orientation (Straffeloven, 2011). Hate speech is generally prohibited by law in the European Union, where it is defined as public incitement to violence or hatred directed against a group predicated on characteristics such as race, religion, and national or ethnic origin (EU, 2008). Hate speech is, however, not prohibited by law in the United States. This is due to the fact that hate speech is protected by the freedom of speech act in the *First Amendment of the U.S. Constitution* (Banks, 2010).

(Davidson et al., 2017) focus is on classifying hate speech by distinguishing between general offensive language and hate speech. They define hate speech as *"language that is used to express hatred towards a targeted group or is intended to be derogatory, to humiliate, or to insult the members of the group"*. They argue that the high use of profanity on social media makes it vitally important to be able to effectively distinguish between generally offensive language and the more severe hate speech. The dataset is constructed by gathering data from Twitter, using a hate speech lexicon to query the data with crowdsourced annotations.

2.3. Contradicting definitions

It becomes clear that one of the key challenges in doing meaningful research on the topic are the differences in both the annotation schemas and the definitions used, since it makes it difficult to effectively compare results to existing work, as pointed out by several authors (Nobata et al., 2016; Schmidt and Wiegand, 2017; Waseem et al., 2017; Zampieri et al., 2019a). These issues become clear when comparing the work of Van Hee et al. (2015b), where racist and sexist remarks are classified as a subset of *insults*, to the work of Nobata et al. (2016), where similar remarks are split into two categories; *hate speech* and *derogatory language*. Another clear example of conflicting definitions becomes visible when comparing (Waseem and Hovy, 2016), where *hate speech* is considered without any consideration of overlaps with the more general type of offensive language, to (Davidson et al., 2017) where a clear distinction is made between the two, by classifying posts as either *Hate speech*, *Offensive* or *Neither*. This lack of consensus led (Waseem et al., 2017) to propose annotation guidelines and introduce a typology. (Zampieri et al., 2019b) argue that these guidelines do not effectively capture both the type and target of the offensive language.

3. Dataset

In this section we give a comprehensive overview of the structure of the task and describe the dataset provided in Zampieri et al. (2019a). Our work adopts this framing of the offensive language phenomenon.

3.1. Classification Structure

Offensive content is broken into three sub-tasks to be able to effectively identify both the type and the target of the offensive posts. These three sub-tasks are chosen with the objective of being able to capture different types of abusive language (Section 2.).

Sub-task A - Offensive language identification In sub-task A the goal is to classify posts as either offensive or not. Offensive posts include insults and threats as well as any form of untargeted profanity (Zampieri et al., 2019b). Each sample is annotated with one of the following labels:

- Not Offensive (NOT) In English this could be a post such as *#TheNunMovie was just as scary as I thought it would be. Clearly the critics don't think she is terrifyingly creepy. I like how it ties in with #TheConjuring series*. In Danish this could be a post such as: *Kim Larsen var god, men hans død blev alt for hyped* ("Kim Larsen was good, but his death was all too hyped").
- Offensive (OFF) In English this could be a post such as *USER is a #pervert himself!*. In Danish this could be a post such as *Kalle er faggot.*. ("Kalle is a faggot..").

Sub-task B - Automatic categorization of offensive language types In sub-task B the goal is to classify the type of offensive language by determining if the offensive language is targeted or not. Targeted offensive language contains insults and threats to an individual, group, or others (Zampieri et al., 2019b). Untargeted posts contain general profanity while not clearly targeting anyone. Only posts labeled as offensive (OFF) in sub-task A are considered in this task. Each sample is annotated with one of the following labels:

- Targeted Insult (TIN) In English this could be a post such as *@USER Please ban this cheating scum*. In Danish this could be e.g. *Hun skal da selv have 99 år, den smatso* ("She should get 99 years herself, the [untranslatable word for disgusting woman]").

- Untargeted (UNT) In English this could be a post such as *2 weeks of resp done and I still don't know shit my ass still on vacation mode*. In Danish this could e.g. *Dumme svin...* ("Stupid pig...").

Sub-task C - Target identification In sub-task C the goal is to classify the target of the offensive language. Only posts labeled as targeted insults (TIN) in sub-task B are considered in this task (Zampieri et al., 2019b). Samples are annotated with one of the following:

- Individual (IND): Posts targeting a named or unnamed person that is part of the conversation. In English this could be a post such as *@USER Is a FRAUD Female @USER group paid for and organized by @USER*. In Danish this could be a post such as *USER du er sku da syg i hoved* ("@USER you are god damn sick in the head"). These examples further demonstrate that this category captures the characteristics of cyberbullying, as it is defined in Section 2..
- Group (GRP): Posts targeting a group of people based on ethnicity, gender or sexual orientation, political affiliation, religious belief, or other characteristics. In English this could be a post such as *#Antifa are mentally unstable cowards, pretending to be relevant*. In Danish this could be e.g. *Åh nej! Svensk lorteret!* ("Oh no! The Swedish shit dish!"). These examples clearly show that this category captures the characteristics of hate speech as it is defined in Section 2..
- Other (OTH): The target of the offensive language does not fit the criteria of either of the previous two categories. (Zampieri et al., 2019b). In English this could be a post such as *And these entertainment agencies just gonna have to be an ass about it.*. In Danish this could be a post such as *Netto er jo et tempel over lort* ("Netto is just a temple of shit").

One of the main concerns in the collection of data for offensive language detection is to find rich sources of user-generated content that represent each class in the annotation schema to some extent. We considered three social media platforms given their popularity with Danish speakers: *Twitter*, *Facebook*, and *Reddit*.

Twitter Twitter has been used extensively as a source of user-generated content and it was the first source considered in our initial data collection phase. The platform provides excellent interface for developers making it easy to gather substantial amounts of data with limited efforts. However, Twitter was not a suitable source of data for our task. This is due to the fact that *Twitter* has limited usage in Denmark, resulting in low quality data with many classes of interest unrepresented.

Facebook We next considered *Facebook*, and the public page for the Danish media company *Ekstra Bladet*. We looked at user-generated comments on articles posted by Ekstra Bladet, and initial analysis of these comments showed great promise as they have a high degree of variation. The user behaviour on the page and the language used ranges from neutral language to very aggressive, where some users pour out sexist, racist and generally hateful language. We faced obstacles when collecting data from Facebook, due to the fact that Facebook recently made the decision to shut down all access to public pages through their developer interface. This makes computational data collection approaches for research impossible. We faced restrictions on scraping public pages with Facebook, and turned to manual collection of randomly selected user-generated comments from Ekstra Bladet's public page, yielding 800 comments of sufficient quality.

Reddit Given that language classification tasks in general require substantial amounts of data, our exploration for suitable sources continued and our search next led us to *Reddit*. We scraped Reddit, collecting the top 500 posts from the Danish sub-reddits *r/DANMAG* and *r/Denmark*, as well as the user comments contained within each post.

3.2. Danish Hate Speech Lexicon

In efforts to maximize the number of user-generated comments from Reddit belonging to the classes of interest in our final dataset we published a survey on Reddit,[1] asking Danish speaking users to suggest offensive, sexist, and racist terms. We published a survey on Reddit asking Danish speaking users to suggest offensive, sexist, and racist terms for a lexicon. Language and user behaviour varies between platforms, so the goal is to capture platform-specific terms. This gave 113 offensive and hateful terms which were used to find offensive comments. It is crucial to note that this is a platform-specific lexicon with narrow demographic and temporal focus, and so neither exhaustive nor conclusive. That is, there are many constructions of abusive language not captured by the list – and also that context is all-important with abusive language, and so many of these terms many not be intrinsically offensive either; so seeing a word on the list is no guarantee of an offensive comment. Rather, they serve as seed terms for searching for potentially offensive comments. This is required because the majority of utterances are not abusive, and so if one is to efficiently annotated abusive language, one needs a tool to narrow the search space to more likely candidates.

3.3. Corpus collection

This Danish lexicon was used to filter the social media comments to find potentially-offensive comments. The remainder of comments in the corpus were shuffled and a subset of this corpus was then used to fill the remainder of the final dataset, ensuring that the data would have significant coverage beyond just terms found in the lexicon. The resulting dataset contains 3600 user-generated comments, 800 from *Ekstra Bladet on Facebook*, 1400 from *r/DANMAG* and 1400 from *r/Denmark*. A detailed breakdown of the final dataset is presented in Section 3.6.. The full lexicon from our survey can be found in Appendix A.

3.4. Privacy Concerns

In light of the *General Data Protection Regulations in Europe (GDPR)* and the increased concern for online pri-

[1] https://www.reddit.com/r/Denmark/comments/9ozhdc/hate_racist_sexist_etc_terms_in_danish/

Data Source	# Comments	% of all
Facebook - Ekstra Bladet	800	22.2
Reddit; r/Denmark w off. term	200	5.6
Reddit; r/Denmark, no off. term	1,200	33.3
Reddit; r/DANMAG w off. term	32	0.9
Reddit; r/DANMAG	1,368	38.0

Table 1: Distribution of samples by source in our final dataset. 'w off. terms" indicates that samples were retrieved using the Danish hate speech lexicon terms.

vacy, we applied some necessary pre-processing steps on our dataset to ensure the privacy of the authors of the comments that were used. Personally identifying content (such as the names of individuals, not including celebrity names) was removed. This was handled by replacing each name of an individual (i.e. author or subject) with *@USER*, as presented in both (Zampieri et al., 2019a) and (Davidson et al., 2017). All comments containing any sensitive information were removed. We classify sensitive information as any information that can be used to uniquely identify someone by the following characteristics; racial or ethnic origin, political opinions, religious or philosophical beliefs, trade union membership, genetic data, and bio-metric data.

3.5. Annotation Procedure

We base our annotation procedure on the guidelines and schemas presented in Zampieri et al. (2019a), discussed in detail in Section 3.1.. As a warm-up procedure, the first 100 posts were annotated by two annotators and the results compared. This exercise was used to refine the understanding of the task at hand and to discuss the mismatches in these annotations for each sub-task.

We used a *Jaccard index* (Hamers et al., 1989) to assess the similarity of our annotations. In sub-task A the Jaccard index of these initial 100 posts was 41.9%, 39.1% for sub-task B, and 42.8% for sub-task C. Analysis of these results and the posts that we disagreed on indicated disagreement was mainly caused by:

1. Guesswork of the context where the post itself was too vague to make a decisive decision on whether it was offensive or not without more context. An example of this is a post such as *Skal de hjælpes hjem, næ nej de skal sendes hjem* ("Do they need to be helped home, no they need to be sent home"), where one might conclude, given the current political climate, that this is an offensive post targeted at immigrants. The context is, however, lacking so we cannot make a decisive decision. This post should, therefore, be labeled as non-offensive, since the post does not contain any profanity or a clearly stated group.

2. Failure to label posts containing some kind of profanity as offensive (typically when the posts themselves were not aggressive, harmful, or hateful). An example could be a post like *@USER sgu da ikke hans skyld at hun ikke han finde ud af at koge fucking pasta* ("@USER god it's not his fault that she can't even work out how to boil fucking pasta"), where the post itself is rather mild, but the presence of *fucking* makes this an offensive post according to our definitions.

Task A	Task B	Task C	Train	Test	Total
OFF	TIN	IND	77	18	95
OFF	TIN	OTH	30	6	36
OFF	TIN	GRP	98	23	121
OFF	UNT		147	42	189
NOT			2,527	632	3,159
ALL			2,879	721	3,600

Table 2: The distribution of labels in the annotated Danish dataset for both the train and test set.

In light of these findings our guidelines were refined so that no post should be labeled as offensive by interpreting any context that is not directly visible in the post itself and that any post containing any form of profanity should automatically be labeled as offensive. These stricter guidelines made the annotation procedure considerably easier while ensuring consistency.

3.6. Final Dataset

Table 1 show distribution of samples by sources in our final dataset, DKHATE. Although a useful tool, using the hate speech lexicon as a filter only resulted in 232 comments. The remaining comments from Reddit were then randomly sampled from the remaining corpus.

The fully annotated dataset was split into a train and test set, while maintaining the distribution of labels from the original dataset. The training set contains 80% of the samples, and the test set contains 20%. Table 2 presents the distribution of samples by label for both the train and test set. The dataset is skewed, with around 88% of the posts labeled as not offensive (NOT). This is typical of abusive user-generated content on online platforms, and any automatic detection system needs be able to handle imbalanced data in order to be truly effective.

4. Features

One of the most important factors to consider when it comes to automatic classification tasks is the feature representation. This section discusses various representations used in the abusive language detection literature.

Top-level features In Schmidt and Wiegand (2017) information comes from top-level features such as bag-of-words, uni-grams and more complex n-grams, and the literature certainly supports this. In their work on cyberbullying detection, (Van Hee et al., 2015a) use word n-grams, character n-grams, and bag-of-words. They report uni-gram bag-of-word features as most predictive, followed by character tri-gram bag-of-words. Later work finds character n-grams are the most helpful features (Nobata et al., 2016), underlying the need for the modeling of un-normalized text. these simple top-level feature approaches are good but not without their limitations, since they often have high recall but lead to high rate of false positives (Davidson et al., 2017). This is due to the fact that the presence of certain terms can easily lead to misclassification when using these types of features. Many words, however, do not clearly indicate which category the text sample belongs to, e.g. the word *gay* can be used in both neutral and offensive contexts.

Linguistic Features (Nobata et al., 2016) use a number of linguistic features, including the length of samples, average word lengths, number of periods and question marks, number of capitalized letters, number of URLs, number of polite words, number of unknown words (based on an English dictionary), and number of insults and hate speech words. Although these features have not proven to be valuable alone, they have been shown to be a good addition to the overall feature space (Nobata et al., 2016).

Word Representations The top-level features discussed so far yield decent performance in general language classification tasks. This is however limited when it comes offensive language detection since the goal is to classify small samples of *noisy* text. Top-level features often require predictive words to occur in both the training set and the test sets, as discussed in Schmidt and Wiegand (2017), but unseen terms prevail in noisy text (Derczynski et al., 2013). For this reason, word generalization is required. (Nobata et al., 2016) explore three uses of embedding-derived features for abusive language detection. First, they explore pre-trained embeddings derived from a large corpus of news samples. Secondly, they use *word2vec* (Mikolov et al., 2013) to generate word embeddings using their own corpus of text samples. We use both approaches. Both the pre-trained and word2vec models represent each word as a 200 dimensional distributed real number vector. Lastly, they develop a *comment2vec* model Le and Mikolov (2014). Their results show that the comment2vec and the word2vec models provide the most predictive features (Nobata et al., 2016). Badjatiya et al. (2017) experiment with pre-trained *GloVe* embeddings (Pennington et al., 2014), learned *FastText* embeddings (Mikolov et al., 2018), and randomly initialized learned embeddings; interestingly, the randomly initialized embeddings slightly outperform the others (Badjatiya et al., 2017).

Sentiment Scores Sentiment scores are a common feature in systems dealing with offensive and hateful speech. We experiment with these by including these scores as features in some models. To compute these sentiment score features our systems use two libraries: *VADER* (Hutto and Gilbert, 2014) and *AFINN* (Nielsen, 2011). Our models use the *compound* attribute, giving a normalized sum of sentiment scores over all words in the sample. This ranges from -1 (extremely negative) to $+1$ (extremely positive).

Reading Ease As well as some of the top-level features mentioned so far, we also use *Flesch-Kincaid Grade Level* and *Flesch Reading Ease scores*. The Flesch-Kincaid Grade Level is a metric assessing the level of reading ability required to easily understand a sample of text.

Pre-trained Embeddings The pre-trained FastText (Mikolov et al., 2018) embeddings are trained on data from the *Common Crawl* project and Wikipedia, in 157 languages (including English and Danish). FastText also provides models that can be used to predict word embeddings for *out-of-vocabulary (OOV)* words. This is a major advantage since challenges arise when using pre-trained word embeddings depending on how often words in the data are missing from the pre-trained corpus.

Randomly-Initialized Learned Embeddings Some of our models use randomly initialized embeddings that are updated during training. In this case, the matrix for the *embedding layer* is initialized using a uniform distribution.

5. Models

We introduce a variety of models in our work to compare different approaches to the task at hand. First of all, we introduce naive baselines that simply classify each sample as one of the categories of interest (based on (Zampieri et al., 2019a)). Next, we introduce a logistic regression model based on the work of (Davidson et al., 2017), using the same set of features as introduced there. Finally, we introduce three deep learning models: Learned-BiLSTM, Fast-BiLSTM, and AUX-Fast-BiLSTM. The logistic regression model is built using Scikit Learn (Pedregosa et al., 2011) and the deep learning models are built using Keras (Chollet and others, 2015). The following sections describe these model architectures in detail, the algorithms they are based on, and the features they use.

Baselines Following the work of (Zampieri et al., 2019a), we create simple baseline prediction models that simply classify all samples as the class containing the largest amount of samples. This allows us to investigate the properties and distribution of the samples in the datasets, and to evaluate how well our classifiers are performing. The baseline models are the following:

- Sub-Task A: All NOT for both languages.
- Sub-Task B: All TIN for both languages.
- Sub-Task C: All IND for English and All GRP for Danish.

Logistic Regression One of our model architecture uses a *Logistic Regression* as the classification algorithm. Logistic regression predicts the probability of events using a continuous function. In the case of a categorical domain, the domain of this function also needs to map continuous variables to discrete categorical values. Here a logistic regression is computed by applying the sigmoid function to the linear regression. Here, y is the dependent variable, $X_1, \ldots, X_n$ are the explanatory variables, and $\beta_0, \ldots, \beta_n$ are the constants we are trying to estimate.

$$y = \beta 0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n X_n \quad (1)$$

$$p = \frac{1}{1 + e^{-y}} \quad (2)$$

$$p = (1 + e^{-(\beta 0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n X_n)})^{-1} \quad (3)$$

Logistic Regression Classifier We base one of our models on (Davidson et al., 2017), where the objective is to distinguish between neutral, offensive and hateful language.

Learned-BiLSTM Classifier The *Learned-BiLSTM model* consists of four parts; a randomly initialized embedding layer, a bi-directional long short memory (BiLSTM) layer, a fully connected hidden layer, and a fully connected output layer. The BiLSTM layer consists of two parts; a forward and a backward LSTM, each of size 20. This

Model	Data	$F1_{macro}$
All NOT	-	0.419
Logistic Regression	OLID	0.724
Learned-BiLSTM $\varepsilon = 10$	OLID	0.707
Fast-BiLSTM $\varepsilon = 100$	**OLID**	**0.735**
AUX-Fast-BiLSTM $\varepsilon = 10$	OLID	0.692
Logistic Regression	OLID+HSAOFL	0.728
Learned-BiLSTM $\varepsilon = 10$	OLID+HSAOFL	0.704
Fast-BiLSTM $\varepsilon = 100$	OLID+HSAOFL	0.688
AUX-Fast-BiLSTM $\varepsilon = 20$	OLID+HSAOFL	0.712

Table 3: Results from sub-task A in English. ε=epochs.

Model	Data	Macro F1
All NOT	-	0.467
Logistic Regression	**DA**	**0.699**
Learned-BiLSTM (10 Epochs)	DA	0.658
Fast-BiLSTM (100 Epochs)	DA	0.630
AUX-Fast-BiLSTM (50 Epochs)	DA	0.675

Table 4: Results from sub-task A in Danish.

vector is then used as input to the fully connected hidden layer, which contains 16 hidden units. The output is a single node for sub-tasks A and B and 3 nodes in sub-task C. The activation function used in the LSTM layers is *tanh* and *ReLU* is used in the hidden layer. For sub-tasks A and B, the activation function for the output layer is *Sigmoid*, and *Softmax* is used for sub-task C. Loss is calculated using *Binary Crossentropy*.

Fast-BiLSTM Classifier The *Fast-BiLSTM* model is built using the same layers and the same set of hyper-parameters as the *Learned-BiLSTM* model. With this the embedding layer is initialized with the FastText embeddings. These embeddings stay fixed and are not updated during the training of the model.

AUX-Fast-BiLSTM Classifier To experiment with a wider combination of features, we extend the Fast-BiLSTM model to *AUX-Fast-BiLSTM*, which accepts auxiliary features, namely: sentiment scores, n-grams weighted by their TF-IDF scores, n-gram POS-tags, counters for the number of characters, count of: syllables; words; Twitter hashtags; URLs; Twitter mentions; and re-tweets, and Flesch-Kincaid reading ease and grade level.

Hyper-Parameter Tuning We perform *Grid Search Cross Validation* to determine the optimal dropout amount, the batch size, the optimizer and the learning rate. The best set of hyper-parameters for all of our models are the following: batch size of 128, Adam (Kingma and Ba, 2014) as the optimization algorithm with a learning rate of 0.001, and a dropout rate of 0.2 between all layers. To tackle imbalance in our dataset we use class weights, giving class a weight equal to the reciprocal of the number of samples it contains.

6. Results and Analysis

For each sub-task (A, B, and C, Section 3.1.) we present results for all methods in each language.

6.1. Task A - Offensive language identification:

English For English (Table 3) Fast-BiLSTM performs best, with $\varepsilon = 100$, using the OLID dataset (Zampieri et al., 2019a). The model achieves a macro averaged F1-score of 0.735, comparable to BiLSTM-based methods in OffensEval (Zampieri et al., 2019b).

	Metric	Fast BiLSTM EN	Log.Reg. DA
NOT	Recall	0.835	0.913
	Precision	0.859	0.929
	F1	0.847	0.921
OFF	Recall	0.646	0.506
	Precision	0.603	0.450
	F1	0.624	0.476

Table 5: Recall (R), precision (P), and F1 score by class for our best performing models in sub-task A.

Figure 1: The train and validation loss curve for the Fast-BiLSTM classifier for sub-task A and the English language.

Additional training data from HSAOFL (Davidson et al., 2017) does not consistently improve results. For the models using word embeddings results are worse with additional training data. On the other hand, for models that use a range of additional features (Logistic Regression and AUX-Fast-BiLSTM), the additional training data helps.

Danish Results are in Table 4. Logistic Regression works best with an F1-score of 0.699. This is the second best performing model for English, though the best performing model for English (Fast-BiLSTM) is worst for Danish.

Best results are given in Table 5. The low scores for Danish compared to English may be explained by the low amount of data in the Danish dataset. The Danish training set contains 2,879 samples (Table 2) while the English training set contains 13,240 samples. Further, in the English dataset around 33% of the samples are labeled offensive while in the Danish set this rate is only at around 12%. The effect that this under represented class has on the Danish classification task can be seen in more detail in Table 5. Differences in both recall and precision by category for the Danish language are larger than for English, indicating that imbalance in or the relative size of the Danish set may have affected the results.

Model	Data	Macro F1
All TIN	-	0.470
Logistic Regression	OLID	0.593
Learned-BiLSTM (60 Epochs)	**OLID**	**0.619**
Fast-BiLSTM (10 Epochs)	OLID	0.567
AUX-Fast-BiLSTM (50 Epochs)	OLID	0.595

Table 6: Results from sub-task B in English.

Model	Data	Macro F1
All TIN	-	0.346
Logistic Regression	DA	0.594
Learned-BiLSTM (40 Epochs)	DA	0.643
Fast-BiLSTM (100 Epochs)	DA	0.681
AUX-Fast-BiLSTM (100 Epochs)	**DA**	**0.729**

Table 7: Results from sub-task B in Danish.

	Metric	L.-BiLSTM EN	AUX-Fast-BiLSTM DA
UNT	Recall	0.370	0.690
	Prec.	0.303	0.725
	F1	0.333	0.707
TIN	Recall	0.892	0.766
	Prec.	0.918	0.735
	F1	0.905	0.750

Table 8: Recall (R), precision (P), and F1 score by class for our best performing models in sub-task B.

6.2. Task B - Offensive language category

English In Table 6 the results are presented for sub-task B on English. The Learned-BiLSTM model trained for 60 epochs performs the best, with macro F1-score of 0.619. Recall and precision scores are lower for UNT than TIN (Table 5). One reason is skew in the data, with only around 14% of the posts labeled as UNT. The pre-trained embedding model, Fast-BiLSTM, performs the worst, with a macro averaged F1-score of 0.567. This indicates this approach is not good for detecting subtle differences in offensive samples in skewed data, while more complex feature models perform better.

Danish Table 7 presents the results for sub-task B and the Danish language. The best performing system is the AUX-Fast-BiLSTM model (Section 5.) trained for 100 epochs, which obtains an impressive macro F1-score of 0.729. This suggests that models that only rely on pre-trained word embeddings may not be optimal for this task. This is be considered alongside the indication in Section 3.6. that relying on lexicon-based selection also performs poorly.
The limiting factor seems to be recall for the UNT category (Table 8). As mentioned in Section 2., the best performing system for sub-task B in OffensEval was a rule-based system, suggesting that more refined features (e.g. lexica) may improve performance on this task. The better performance of models for Danish over English can most likely be explained by the fact that the training set used for Danish is more balanced, with around 42% of the posts labeled as UNT.

6.3. Task C - Target identification

English The results for sub-task C and the English language are presented in Table 9. The best performing system is the Learned-BiLSTM model (Section 5.) trained for 10 epochs, obtaining a macro averaged F1-score of 0.557. This is an improvement over the models introduced in Zampieri et al. (2019a), where the BiLSTM based model achieves a macro F1-score of 0.470.
The main limitations of our model seems to be in the classification of OTH samples, as seen in Table 11. This may be explained by the imbalance in the training data. It is interesting to see that this imbalance does not effect the GRP category as much, which only constitutes about 28% of the training samples. One cause for the differences in these, is the fact that the definitions of the OTH category are vague, capturing all samples that do not belong to the previous two.

Model	Data	Macro F1
All IND	-	0.213
Logistic Regression	OLID	0.458
Learned-BiLSTM (10 Epochs)	**OLID**	**0.557**
Fast-BiLSTM (50 Epochs)	OLID	0.516
AUX-Fast-BiLSTM (40 Epochs)	OLID	0.536

Table 9: Results for sub-task C in English.

Model	Data	Macro F1
All GRP	-	0.219
Logistic Regression	DA	0.438
Learned-BiLSTM (100 Epochs)	**DA**	**0.629**
Fast-BiLSTM (60 epochs)	DA	0.579
AUX-Fast-BiLSTM (100 Epochs)	DA	0.401

Table 10: Results from sub-task C in Danish.

	Metric	L.-BiLSTM EN	L.-BiLSTM DA
IND	Recall	0.670	0.556
	Prec.	0.770	0.667
	F1	0.717	0.606
GRP	Recall	0.667	0.696
	Prec.	0.634	0.640
	F1	0.650	0.667
OTH	Recall	0.343	0.667
	Prec.	0.273	0.571
	F1	0.304	0.615

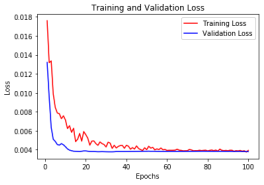
Table 11: Recall (R), precision (P), and F1 score by class for best performing models in sub-task C.

Danish Table 10 presents the results for sub-task C and the Danish language. The best performing system is the same as in English, the Learned-BiLSTM model (Section 5.), trained for 100 epochs, obtaining a macro averaged F1-score of 0.629. Given that this is the same model as the one that performed the best for English, this further indicates that task specific embeddings are helpful for more refined classification tasks.
The models using additional features (Logistic Regression and AUX-Fast-BiLSTM) perform the worst. This indicates that additional features are not beneficial for this refined sub-task in Danish. A low number of samples are used in this sub-task. Imbalance has as much effect for Danish as for English (Table 11). Only about 14% of the samples are labeled as OTH in the data (Table 2). However, recall and precision are closer than they are for English.

7. Analysis

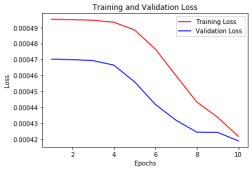
We evaluated best-performing models based on their misclassifications. To accomplish this, we compute the TF-IDF scores for a range of n-grams, taking top scoring n-grams in each category and examining resulting trends. We also perform some manual analysis of these misclassified samples. The goal of this process is to try to get a clear idea of the areas our classifiers are lacking in.

(a) Learned-BiLSTM (en)

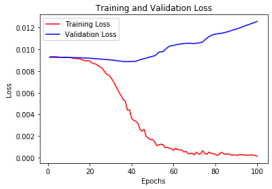


(b) AUX-Fast-BiLSTM (da)

Figure 2: Train and validation loss in sub-task B for each language.



(a) Learned-BiLSTM (en)



(b) Learned-BiLSTM (da)

Figure 3: Train and validation loss in sub-task C for each language.

7.1. Task A - Offensive language identification

The classifier struggles to identify obfuscated offensive terms. This includes words that are concatenated together, such as *barrrysoetorobullshit*. The classifier also seems to associate *she* with offensiveness, and samples containing *she* are misclassified as offensive in several samples while *he* is less often associated with offensive language.

In several cases the classifier spuriously labels profanity-bearing content as offensive. Posts such as *Are you fucking serious?* and *Fuck I cried in this scene* are labeled non-offensive in the test set, but according to annotation guidelines should be classified as offensive.

The best classifier tends classify longer sequences as offensive. The mean character length of misclassified offensive samples is 204.7, while the mean character length of the samples misclassified not offensive is 107.9. This may be due to the inclusive nature of sub-task A's definition, so more words increase the likelihood of > 0 profane words.

Figure 1 shows train and validation loss curves, with divergence around epoch 40 indicating some overfitting.

The classifier suffers from the same limitations as the classifier for English when it comes to obfuscated words, misclassifying samples such as *Hahhaaha lær det biiiiiaaaatch* ("Haha learn it bitch") as non-offensive. It also seems to associate the occurrence of the word *svensken* ("the Swedes") with offensive language, and quite a few samples containing that word are misclassified as offensive. This can be explained by the fact that offensive language towards Swedes is common in the training data, resulting in this association. From this, we can conclude that the classifier relies too much on the presence of individual keywords, ignoring the context of these keywords.

7.2. Task B - Offensive language category

Obfuscation prevails in sub-task B. Our classifier misses indicators of targeted insults such as *WalkAwayFromAllDemocrats*. It seems to rely too highly on the presence of profanity, misclassifying samples containing terms such as *bitch, fuck, shit, etc.* as targeted insults.

The issue of the data quality is also concerning in this sub-task, as we discover samples containing clear targeted insults such as *HillaryForPrison* being labeled as untargeted in the test set. This fairly typical use mode of social media (Derczynski et al., 2013) needs some extra effort to handle, as might be afforded by tokenization at the level of hashtags (Maynard and Greenwood, 2014) or subwords (Sennrich et al., 2016).

Figure 2 (a) shows the model fits aggressively after just 10 epochs while the training loss approaches zero. A possible reason for this aggressive over-fitting is inclusion of embeddings in updates for the Learned-BiLSTM, giving a tight fit when coupled with dataset size. Figure 2 (b) shows a validation loss constantly lower than training loss, possibly due to small validation set size.

Our Danish classifier also seems to be missing obfuscated words such as *kidsarefuckingstupid* in the classification of targeted insults. It relies to some extent to heavily on the presence of profanity such as *pikfjæs, lorte* ("dickface", "shit" ADJ) and *fucking*, and misclassifies untargeted posts containing these keywords as targeted insults.

7.3. Task C - Target identification

Misclassification based on obfuscated terms as discussed earlier also seems to be an issue for sub-task C. This problem of obfuscated terms could be tackled by introducing character-level features such as character level n-grams.

Figure 3 shows training and validation loss curves for each language. There are no indications of early overfitting for English. The classifier Danish behaves similarly to the Learned-BiLSTM classifier for English and sub-task B (Section 7.2.), where validation loss escalated. The Learned-BiLSTM seems therefore prone to overfitting in this context-sensitive, high variable data task. One solution might be to decrease the size of the embedding layer, downgrading the number of parameters that can be tuned during training, giving a denser representation space.

8. Conclusion

Offensive language on online social media platforms is harmful. Due to the scale of content on these platforms, automatic methods are required to detect this content. Much of the prior research on the topic has focused on solving the problem for English. We explored English and Danish abusive language detection and categorization, finding that sharing learnings across languages and platforms leads to good models for the task, capturing a broad range of language from the so-called "hygge-racisme" (Black, 2018) to targeted attacks.

The resources and classifiers are available from the authors under CC-BY license, pending use in a shared task (OffensEval 2020). A data statement (Bender and Friedman, 2018) is included in the appendix. Extended results and analysis are given in Sigurbergsson (2019).

Acknowledgments

We would like to thank Digitalt Ansvar for helpful conversations in the formation of this research, and Pushshift.io for the availability of Reddit archive data.

Bibliographical References

Badjatiya, P., Gupta, S., Gupta, M., and Varma, V. (2017). Deep learning for hate speech detection in tweets. In Proceedings of the 26th International Conference on World Wide Web Companion, pages 759–760.

Banks, J. (2010). Regulating hate speech online. *International Review of Law, Computers & Technology*, 24(3):233–239.

Bender, E. M. and Friedman, B. (2018). Data statements for natural language processing: Toward mitigating system bias and enabling better science. *Transactions of the Association for Computational Linguistics*, 6:587–604.

Black, M. B. (2018). 'Hygge racism':" noget som man nok bruger mere end man tænker over". A qualitative study of well-intentioned racism. Master's thesis, Sociologiska Institutionen, Lund Universitet.

Chollet, F. et al. (2015). Keras. `https://keras.io`.

Davidson, T., Warmsley, D., Macy, M., and Weber, I. (2017). Automated hate speech detection and the problem of offensive language. In Eleventh International AAAI Conference on Web and Social Media.

Derczynski, L., Maynard, D., Aswani, N., and Bontcheva, K. (2013). Microblog-genre noise and impact on semantic annotation accuracy. In Proceedings of the 24th ACM Conference on Hypertext, pages 21–30.

Derczynski, L., Albert-Lindqvist, T. O., Bendsen, M. V., Inie, N., Petersen, J. E., and Petersen, V. D. (2019). Kvinder nedgøres oftere end mænd i politiske debatter på sociale medier. *TjekDet / Mandag Morgen*.

EU. (2008). European Union Council Framework Decision 2008/913/JHA. `https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=LEGISSUM%3Al33178`. Accessed: 2019-05-29.

Gregorie, T. M. (2001). Cyberstalking: Dangers on the information superhighway. *National Center for Victims of crime*.

Hamers, L., Hemeryck, Y., Herweyers, G., Janssen, M., Keters, H., Rousseau, R., and Vanhoutte, A. (1989). Similarity measures in scientometric research: The Jaccard index versus Salton's cosine formula. *Information Processing and Management*, 25(3):315–18.

Hutto, C. J. and Gilbert, E. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. In Proceedings of the International Conference on Weblogs and Social Media (ICWSM). AAAI.

Joseph, S. and Castan, M. (2013). The international covenant on civil and political rights: cases, materials, and commentary. Oxford University Press.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kirkedal, A., Plank, B., Derczynski, L., and Schluter, N. (2019). The Lacunae of Danish Natural Language Processing. In Proceedings of the Nordic Conference on Computational Linguistics (NODALIDA).

Le, Q. and Mikolov, T. (2014). Distributed representations of sentences and documents. In Proceedings of the International Conference on Machine Learning, pages 1188–1196.

Maynard, D. G. and Greenwood, M. A. (2014). Who cares about sarcastic tweets? Investigating the impact of sarcasm on sentiment analysis. In Proceedings of the International Conference on Language Resources and Evaluation (LREC). ELRA.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Grave, E., Bojanowski, P., Puhrsch, C., and Joulin, A. (2018). Advances in pre-training distributed word representations. In Proceedings of the International Conference on Language Resources and Evaluation (LREC).

Müller, K. and Schwarz, C. (2018). Fanning the flames of hate: Social media and hate crime. *Available at SSRN 3082972*.

Nielsen, F. Å. (2011). A new anew: Evaluation of a word list for sentiment analysis in microblogs. *arXiv preprint arXiv:1103.2903*.

Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., and Chang, Y. (2016). Abusive language detection in online user content. In Proceedings of the 25th international conference on the World Wide Web, pages 145–153. International World Wide Web Conferences Steering Committee.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12(Oct):2825–2830.

Pennington, J., Socher, R., and Manning, C. (2014). GloVe: Global Vectors for word representation. In Proceedings of the 2014 conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1532–1543.

Schmidt, A. and Wiegand, M. (2017). A survey on hate speech detection using natural language processing. In Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media, pages 1–10.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural machine translation of rare words with subword units. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pages 1715–1725.

Sigurbergsson, G. I. (2019). Offensive and hate speech detection. Master's thesis, IT University of Copenhagen.

Straffeloven. (2011). Straffeloven § 266 b. `https://danskelove.dk/straffeloven/266b`. Accessed: 2019-05-29.

Van Hee, C., Lefever, E., Verhoeven, B., Mennes, J., Desmet, B., De Pauw, G., Daelemans, W., and Hoste, V. (2015a). Detection and fine-grained classification of cyberbullying events. In Proceedings of the international conference Recent Advances in Natural Language Processing, pages 672–680.

Van Hee, C., Verhoeven, B., Lefever, E., De Pauw, G., Hoste, V., and Daelemans, W. (2015b). Guidelines for the fine-grained analysis of cyberbullying. Technical report, Language and Translation Technology Team, Ghent University.

Vidgen, B., Harris, A., Nguyen, D., Tromble, R., Hale, S., and Margetts, H. (2019). Challenges and frontiers in abusive content detection. In Proceedings of the Third Workshop on Abusive Language Online, pages 80–93.

Waseem, Z. and Hovy, D. (2016). Hateful symbols or hateful people? Predictive features for hate speech detection on twitter. In Proceedings of the NAACL student research workshop, pages 88–93.

Waseem, Z., Davidson, T., Warmsley, D., and Weber, I. (2017). Understanding abuse: A typology of abusive language detection subtasks. In Proceedings of the First Workshop on Abusive Language Online, pages 78–84, Vancouver, BC, Canada, August. Association for Computational Linguistics.

Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N., and Kumar, R. (2019a). Predicting the type and target of offensive posts in social media. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 1415–1420, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N., and Kumar, R. (2019b). SemEval-2019 Task 6: Identifying and Categorizing Offensive Language in Social Media (OffensEval). In Proceedings of SemEval.

A Danish Hate Speech Lexicon (Reddit)

- bekvem
- bessefar
- bondejokke
- bondeknold
- bonderøv
- bondsk
- bæskubber
- bøssekarl
- establishment
- fisefornem
- fjeldabe
- fjæs
- floskelmager
- flæbe
- fløddebolle
- frøæder
- gnom
- hadsprædikant
- hedenskab
- hjemmefødning
- kopist
- kraftidiot
- krigsliderlig
- kvasiintellektuel
- kvindagtig
- lebbe
- lort
- ludder
- middelalderlig
- møgunge
- nigger
- offergøre
- offergørelse
- papmor
- partout
- perker
- pigebarn
- pigefnidder
- plapre
- plasticmor
- røvhul
- skaffedyr
- skrælling
- slipsedyr
- snerpe
- snotdum
- snotunge
- spastiker
- stikker
- støjbergsk
- svans
- symbolpolitik
- torsk
- tude
- tyskertøs
- vatpik
- Amatører
- bidesild
- bløddyr
- bollefjæs
- fedtefyre
- hundehoveder
- fnatmider
- fæhoveder
- grødbønder
- hængerøve
- ignoranter
- jammerkommoder
- karklud
- klamhuggere
- klodsmajor
- lusepustere
- narrehatte
- pattebørn
- pjalt
- pjok
- pudseklud
- skidespræller
- skvadderhoveder
- skvat
- skvatpissere
- slapsvanse
- snotklatte
- elendige socialdemokrater
- Sindssyge
- kvindemenneske
- svabrefjams
- Hestepære
- kolort
- kolibriafføring
- myggefjols
- kældernisse
- buskerusker
- hårtygger
- våben
- ledningsfletter
- højreradikal
- højreekstremist
- fremmedfjendsk
- nynazist
- kartoffel
- ny bruger
- kvindehader
- hvid
- privilegeret
- heteronormativ
- undertrykker
- krænker
- kristen
- muslimer
- multikultur
- nazist
- sort

B Data statement

Curation rationale Examples of offensive and non-offensive language, in Danish

Language variety Danish, BCP-47: `da-DK`

Speaker demographic

- Danish Reddit and Facebook users
- Age: Unknown – mixed.
- Gender: Unknown – mixed.
- Race/ethnicity: Unknown – mixed.
- Native language: Unknown; Danish speakers.
- Socioeconomic status: Unknown – mixed.
- Different speakers represented: Unknown; upper bound is the number of posts.
- Presence of disordered speech: Some presences.

Annotator demographic

- Age: 25-40.
- Gender: male.
- Race/ethnicity: white northern European.
- Native language: Icelandic, English.
- Socioeconomic status: higher education student / university faculty.

Speech situation Discussions held in public on the Reddit or Facebook platform.

Text characteristics Danish colloquial web speech.

Provenance Originally taken from Reddit, Ekstra Bladet, and Facebook, 2018; details given in Section 3..