*Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020)*, pages 5079–5087
Marseille, 11–16 May 2020
© European Language Resources Association (ELRA), licensed under CC-BY-NC

# Offensive Language Detection Using Brown Clustering

**Zuoyu Tian, Sandra Kübler**
Indiana University
zuoytian@indiana.edu, skuebler@indiana.edu

**Abstract**

In this study, we investigate the use of Brown clustering for offensive language detection. Brown clustering has been shown to be of little use when the task involves distinguishing word polarity in sentiment analysis tasks. In contrast to previous work, we train Brown clusters separately on positive and negative sentiment data, but then combine the information into a single complex feature per word. This way of representing words results in stable improvements in offensive language detection, when used as the only features or in combination with words or character $n$-grams. Brown clusters add important information, even when combined with words or character $n$-grams or with standard word embeddings in a convolutional neural network. However, we also found different trends between the two offensive language data sets we used.

**Keywords:** Brown clustering, offensive language detection, machine learning

## 1. Introduction

Over the last two decades, the increase in the use of social media has led to an increase in offensive language. Social media often allow anonymous access, and users increasingly use this anonymity to publicize aggressive or offensive attitudes. Because of the vast volume of data produced on social media platforms every day, we need automated methods that can detect hate speech without restricting people's right to freedom of expression.

Research on offensive language detection has mostly concentrated on using machine learning algorithms with a set of shallow lexical features such as word or character $n$-grams (e.g., (Warner and Hirschberg, 2012; Malmasi and Zampieri, 2018)). One of the challenges that these approaches face is data sparsity since such shallow features do not generalize well: We may not have seen the exact word that characterizes a tweet as hate speech, but we may have seen similar or related words in training. This problem has been approached by using word embeddings. However, most of these approaches use pre-trained models, trained on large sets of English language data (Peters et al., 2018; Devlin et al., 2018; Mikolov et al., 2018). Such models currently provide state of the art models (Badjatiya et al., 2017; Liu et al., 2019; Zhu et al., 2019; Mishra et al., 2019). Since these embeddings were not created from data sets specifically targeting the detection of hate speech, it is not clear if they actually generalize over hate speech related words, or if they generalize over specific domains in which hate speech is common. The Waseem data set (Waseem and Hovy, 2016), for example, was sampled based on specific topics that had created large amounts of hate speech, such as football or an Australian cooking show. Since the approaches using word embeddings often require more data and compute power than most people have access to, it is impossible or at least difficult to train them on more specific data.

We approach the problem of data sparsity in offensive language detection from a slightly different angle: We use Brown clustering (Brown et al., 1992; Liang, 2005) to create generalized word representations. Brown clusters, while not necessarily competitive to embeddings in a range of tasks, have the advantage that they can be trained on smaller data sets, and training times are faster than training times for neural networks. Thus, we can investigate whether it makes sense to have more specialized word representations, which provide a good balance between generalizing over the different forms of a word and losing a distinction between offensive and non-offensive uses. More generally, we investigate the following questions:

1. When creating Brown clusters, it is not clear which data set will be the most useful: Is it more important to have specialized data or a larger data set? The data sets for hate speech detection tend to be fairly small, but are the most relevant for the problem. Using data sets from sentiment analysis provides a larger data set from a problem that shares many characteristics with offensive language detection. The largest data set would be a general collection of tweets, but this data set will be out of domain.

2. The next question concerns the problem that Brown clustering has a tendency to cluster words of opposite sentiment together. E.g., the words 'good' and 'bad' tend to occur in similar contexts, which means that they are clustered in the same cluster. We investigate whether we can avoid this behavior by clustering tweets with positive and negative sentiment separately, and how to optimally combine these separate clusters into features.

3. Our last question is concerned with whether the features from Brown clusters represent similar information to features already in use. I.e., we combine our Brown clusters with other features (character and word $n$-grams) and with neural network architectures. If the results improve when adding Brown clusters, we can conclude that clusters add new, relevant information.

The paper is structured as follows: We present an overview of related work in section 2. In section 3., we introduce the data sets. Section 4. describes the experimental setup, and section 5. discusses the results. We finish with a conclusion and future work in section 6.

## 2. Related Work

Research in offensive language detection often focuses on the problem of finding appropriate features that provide enough information to detect offensive language. Studies show that word and character $n$-grams are the most common surface features used in these tasks, and also the most successful ones (Warner and Hirschberg, 2012; Waseem and Hovy, 2016; Malmasi and Zampieri, 2018). Malmasi and Zampieri (2018) found character 4-grams to outperform other features, such as word $n$-gram and Brown clusters. Waseem and Hovy (2016) report that their best results are based on a model built with character $n$-gram and gender features.

More recent work shows that neural network models using pre-trained word representations significantly advanced the state of the art in offensive language detection: Kumar et al. (2018) perform a benchmark analysis of the first shared task on aggression identification and show that half of the top 15 systems use neural network models. Besides traditional neural network models, transformer-based language models like Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018) achieve state of the art performance in the SemEval 2019 shared task: OffensEval (Zampieri et al., 2019b). Seven out of the 10 highest performing systems adopt BERT models with variations in the parameters for offensive language identification tasks (Zampieri et al., 2019b).

Warner and Hirschberg (2012) first introduce Brown cluster features in hate speech identification tasks. They build SVM classifiers with template-based features including $n$-grams, POS tags, and Brown clusters. However, they obtain the best results using only unigram features. Wester et al. (2016) examines the performance of various types of linguistic features in threat detection, a task with many similarities to hate speech detection. They also utilize Brown cluster features as one of their semantic features.

These previous studies indicate that classifiers trained on simple Brown clusters cannot outperform those trained on surface features. Socher et al. (2011) argue that the failure of Brown clusters in sentiment analysis tasks is because "they do not capture sentiment information (good and bad are usually in the same cluster) and cannot be modified via backpropagation." However, we also find that most of the studies using Brown clusters in the sentiment analysis tasks only use existing Brown clusters trained on the general genre, which can cause an insensitivity in capturing sentiment information.

## 3. Data

### 3.1. Offensive Language Detection Data Sets

For consistency with previous work, we use two data sets to examine the performance of domain-specific Brown cluster features. The first data set is the Offensive Language Identification Dataset (OLID), which has been used in the SemEval 2019 Task 6 (Zampieri et al., 2019a). This data set is built by crowd-sourcing, with at least two annotators per tweet, and contains annotations on three levels. For the current work, we only use the annotations for offensive language detection. There are 13 240 tweet instances in the training data, and each instance has been labeled as either 'offensive' (OFF) or 'not offensive' (NOT). We use the designated test set for evaluation; it contains 860 tweet instances.

| | Waseem and Hovy (2016) | | | OLID | |
|---|---|---|---|---|---|
| | Racism | Sexism | None | Offensive | None |
| Train | 1 719 | 2 741 | 9 683 | 4 400 | 8 840 |
| Test | 191 | 305 | 1 076 | 240 | 620 |
| **Total** | 1 910 | 3 046 | 10 759 | 4 640 | 9 460 |
| **Total %** | 12% | 19% | 69% | 33% | 67% |

Table 1: Distribution of class labels in the two data sets

The second data set is the English twitter hate speech data set created by Waseem and Hovy (2016), to which we will refer as the Waseem data set. The original corpus includes approximately 16 000 tweets, however, we could only retrieve 15 715 tweets, the rest were unavailable or deleted. This data set was manually annotated using specific criteria with three different labels: 'racism', 'sexism' and 'none', where the latter label refers to non-hate speech. For the evaluation, we extracted every tenth tweet (1 572 tweets) to create the test set. The remaining 90% (14 143 tweets) of the data set are used as training data.

Table 1 shows the distribution of labels in the two data sets.

### 3.2. Data for Brown Clustering

The Brown clustering algorithm (Brown et al., 1992) groups the words of a corpus into clusters of related or similar words using bigram mutual information to determine the similarity between words. Brown clustering requires a predefined number of clusters. The approach has been shown to produce clusters of semantically similar words.

In order to use Brown clusters for offensive language detection, we use one existing set of clusters but also generate our own from a range of different data sets:

The existing set of clusters, henceforth called "general", was created by Owoputi et al. (2013). This set contains 1000 clusters and was trained on 56 million English tweets from 560 days (837 million tokens). All words with a frequency of less than 40 are ignored. This is our largest data set, but it is also out of domain in the sense that it covers a much wider range of tweets than our offensive language data sets. Consequently, our hypothesis is that the clusters will be less than optimal for separating offensive from non-offensive words.

The second largest data set is the Sentiment 140 Tweet Corpus (Go et al., 2009). It contains 1.6 million Twitter messages with automatically labeled positive and negative sentiment labels. The corpus is balanced, half of the tweets are positive, the other half are negative. Our assumption is that most of the offensive tweets should contain clear sentiment preferences, which means that offensive tweets can mostly be regarded as a subset of the negative tweets. We use the "positive" set of tweets and the "negative" one to create separate Brown clusters.

The smallest, but most specific corpus consists of offensive tweets. For this corpus, we collected all tweets labeled as offensive in the training data from the following four data sets: abusive speech from Davidson et al. (2017), hate

| | general | positive | negative | abusive |
|---|---|---|---|---|
| good | 64 | 34 | 44 | 2 |
| bad | 16 | 127 | 41 | 3 |

Table 2: Size of 'good' and 'bad' clusters across the data sets

speech from Founta et al. (2018) and Waseem and Hovy (2016), and offensive tweets from Zampieri et al. (2019a). This resulted in 42 956 tweets with 641 521 words; this will henceforth be called "abusive".

### 3.3. A Quick Look at Brown Clusters

Before we start investigating the use of Brown clusters experimentally, we decided to have a look at the size of the different Brown clusters. We also had a closer look at the clusters to see if more specialized Brown clusters, trained separately on positive and negative sentiment data or on offensive data, can model the differences between positive and negative sentiment better. If we see differences in the clusters, we can assume that we do have a basis for assuming our specialized Brown clusters are more discriminating than a general model that has been criticized by Socher et al. (2011) for not being able to capture differences in sentiment.

For both the size differences and the content differences, we focused on the two words 'good' and 'bad'. Table 2 shows the size of these clusters based on the different corpora, based on 1 500 clusters for the positive, negative, and abusive data set, and based on 1 000 clusters for the general set (the only size available). The numbers show some unexpected trends: The 'good'/'bad' clusters based on the abusive data are extremely small, and thus not very useful. However, it is more puzzling that the 'bad' cluster from the positive sentiment data is the largest cluster while the two clusters are of similar size in the negative sentiment data. Additionally, the 'good' cluster is considerably larger than the bad' cluster in the general data. It is also worth noting that the cluster sizes from the general data are not necessarily larger than the clusters from the positive or negative sentiment data, even though the former are trained 56 million tweets.

Table 3 shows a sample of words in the 'good' and 'bad' clusters. For all but the abusive clusters, we can see that words are often successfully clustered with their sentiment relations, for example, 'good' and its different types of typos. This contradicts the assumption that Brown clusters cannot detect sentiment. The clusters based on the large, general data are the most stable ones, they mostly contain spelling variants of our focus words. The clusters based on either positive or negative sentiment data seems less clean, containing a good sized proportion of unrelated words, such as 'values', 'ex-geek', '#canada', or 'i'm-awake-no-really'. However, we also notice that some of the clusters contain words of the opposite sentiment. For example, 'god-awful' is in the 'good' cluster based on positive sentiment data, 'good/' in the 'bad' cluster based on general data, and 'guhd' in the 'bad' cluster based on negative sentiment data. Some of these differences are certainly due to the fact that the training size of the positive and negative Brown clusters is significantly smaller than the general one. Domain-specific Brown clusters are trained on 800 000 tweets. In comparison, the original Brown clusters are trained on 56 million tweets.

Given the sizable differences in cluster sizes of the 'good' and 'bad' clusters, we also had a look at the cluster sizes overall. Table 4 shows the minimum, maximum, and average size of all our clusters. The numbers show that unsurprisingly, the general corpus results in the highest average size of clusters, even though they ignore any words with a frequency of less than 40. The positive and negative sentiment clusters reach a decent average size, despite being trained on only a fraction of the general data set size. The abusive clusters, in contrast, tend to be very small, thus limiting their capability for generalizing over individual words.

## 4. Experimental Setup

### 4.1. Brown Clustering

We use the Brown clustering algorithm by Liang (2005) to generate the word clusters. All tweets have been tokenized and converted to lower case, to decrease the number of unique words, but we do not use a frequency cutoff, i.e., all words are included in the clusters, independent of their frequency. We examined the performance of 500, 1000, 1500 and 2000 clusters in each experimental setting, but we only report the best results.

After creating the Brown clusters, we directly replace each word in a tweet with its Brown cluster ID. Any word that is not represented in the Brown clusters is assigned an ID for unknown words.

When we create separate Brown clusters for positive and negative data, we need to decide how to represent this information. We experiment with two different representations: 1) For each word, we combine the IDs from the positive and the negative cluster into a single, complex feature by concatenating the IDs (.e.g, 011110111010_000111100). 2) We represent each word as two features, one for the positive and one for the negative cluster ID.

### 4.2. Machine Learning

We performed initial experiments with standard machine learning algorithms that have been used in offensive language detection: SVM, naive Bayes, and logistic regression. We found that logistic regression classifiers give the best performance and are the most stable across different settings. Since all classifiers show similar trends in all the experiments, we only report the results using logistic regression. We utilized the Logistic Regression implementation in Scikit-Learn (Pedregosa et al., 2011). We performed a grid search to determine the optimal settings, which correspond to the default settings.

### 4.3. Neural Network Models

We used the Keras library (Abadi et al., 2016) to build the models. Batch size was set to 32, the drop out rate is between 0.3 and 0.5, optimized by model, and we trained 3 epochs for each model.

**Preprocessing:** We utilized two types of features for generating the embeddings weights. For surface features, we

| good | | | | bad | | | |
|---|---|---|---|---|---|---|---|
| general | positive | negative | abusive | general | positive | negative | abusive |
| good | good | good | good | bad | bad | bad | bad |
| gud | gud | ggod | cemetery | baa[*]d | baad | baa[*]d | funnel |
| gd | gd | g0od | | shitty | shabby | baadd[*] | celebration |
| gooo[*]d | gooo[*]d | goooooddd | | bad/good | dismal | fvck | |
| good[*] | g00d | g00d | | good/bad | fussed | chickenshit | |
| gewd | gurd | ood | | good/ | troubling | aaaaaaaaaaw | |
| g0od | v.good | wikid | | tashard | guhd | biggggg | |
| ghud | god-awful | raaahhhh | | creigh | not-as-bad | #twin | |
| goid | peely | sweat-filled | | crumby | i'm-awake-no-really | lateline | |
| goos | scarry | shizzexx | | | ex-geek | quicky | |
| goodnight/good | valued | morrissey's | | | fullbright | intricate | |
| godo | hardworker | lyt | | | @roddyjdotcom | beyonceee | |
| | dystopian | hwaaaa | | | hilarrious | #canada | |
| | redskin | goodl | | | #underrated | hearbreaking | |
| | nicki's | excyted | | | sanny | sleep-whine | |

Table 3: Brown cluster for the words 'good' and 'bad' in the different tweet corpora

| | general | positive | negative | abusive |
|---|---|---|---|---|
| min. | 40 | 1 | 1 | 1 |
| max. | 15 501 | 2 837 | 1794 | 62 |
| average | 216.9 | 160.6 | 142.2 | 11.5 |

Table 4: Size of clusters across the data sets

used the one-hot encoder to process the texts and generated a 300 dimension embeddings vector. The vocabulary size is set to 10 000. Then we padded sentences to the length of 30 zero vectors. For Brown cluster features, sentences are represented as described in section 4.1., then we adopted the same preprocessing pipelines of surface features to generate the Brown cluster embeddings.

**CNN model:** Word embeddings or Brown cluster embeddings are fed to a CNN layer. We used 3 filter sizes (between 3 and 5) with 100 filters in each size. Then all outputs are concatenated after a max-pooling layer. The concatenated outputs are followed by a 64 dimension dense layer with relu activation. The final layer is a dense layer with sigmoid for binary classification or softmax activation for multi-classification.

**CNN-merge model:** We concatenated the outputs of the 64 dimension hidden layers from word and Brown cluster embeddings in the above models, creating a 128-dimensional vector. It is followed by a 64 dimension hidden layers with relu activation and the output layer.

**BiLSTM model:** We fed word embeddings or Brown cluster embeddings to a 64 dimension bidirectional LSTM layer with relu activation. The final layer is a dense layer with softmax or sigmoid activation (see above).

**BiLSTM-merge model:** We concatenated the outputs of the 64 dimension hidden layers of the Brown cluster embeddings BiLSTM model and that of the word embeddings BiLSTM model. This merged model consists of 128 dimension hidden layers, which are then fed to 64 dimension hidden layers with relu activation, then to the output layer.

### 4.4. Features

Brown cluster features are represented as discussed in Section 4, but, besides the normal tokenization, we also experimented with word piece tokenization. The latter method of tokenization has been used as basic tokens for recent transformer models (Devlin et al., 2018). Bodapati et al. (2019) show that using word piece as features can boost the performance of abusive language detection. We used the BERT tokenizer to process the data and generated the word piece input for Brown clustering.

Besides adopting Brown clusters as features, we also examined the performance of character $n$-grams and word $n$-grams for comparison. Character $n$-grams are widely used in traditional machine learning methods and outperform other surface features in many tasks (Waseem and Hovy, 2016; Malmasi and Zampieri, 2018) that show high tolerance to spelling errors and variations in tweets (Schmidt and Wiegand, 2017). Word $n$-gram also performed well in previous studies, for example, the study by Warner and Hirschberg (2012) indicates that unigrams outperformed other combinations of word or character $n$-gram in detecting abusive language.

After running preliminary experiments, we decided to adopt unigram Brown cluster IDs (minimal 2 occurrences), word unigrams (minimal 2 occurrences), and character 1-4 grams (minimal 2 occurrences) to build the bag of words models. When we merge different types of features including words and $n$-grams, we use feature selection on the combined feature set, which has been shown to be effective in sentiment classification tasks (Kübler et al., 2018). Our best results are based on feature selection with mutual information.

### 4.5. Evaluation

Since abusive language is the minority class in a highly skewed data set, we report macro-averaged precision, recall, and F1 for each class as well as accuracy and F1 across all classes (as calculated by Scikit-Learn).

| | | Acc | $F_1$ |
|---|---|---|---|
| OLID | fine-tuned BERT model (Liu et al., 2019) | 86.28 | 82.86 |
| | SVM baseline (Zampieri et al., 2019a) | - | 69 |
| Waseem | SVM baseline | - | 64.58 |
| | GCN+LR model(Mishra et al., 2019) | - | 85.42 |

Table 5: Baselines and state of the art for the OLID and Waseem data sets

**OLID**

| Method | # cl | Offensive | | | Not offensive | | | Overall | |
|---|---|---|---|---|---|---|---|---|---|
| | | P | R | $F_1$ | P | R | $F_1$ | Acc | $F_1$ |
| $Brown_{general}$ | 1000 | 70.07 | 42.92 | 53.23 | 80.79 | 92.90 | 86.42 | 78.95 | 69.82 |
| $Brown_{sentiment}$ | 1500 | 60.15 | 33.33 | 42.89 | 77.99 | 91.45 | 84.18 | 75.23 | 63.54 |
| $Brown_{offensive}$ | 2000 | 59.57 | 35.00 | 44.00 | 78.30 | 90.81 | 84.09 | 75.23 | 64.09 |

**Waseem data set**

| Method | # cl | Racism | | | Sexism | | | Not offensive | | | Overall | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | P | R | $F_1$ | P | R | $F_1$ | P | R | $F_1$ | Acc | $F_1$ |
| $Brown_{general}$ | 1000 | 69.93 | 56.02 | 62.21 | 75.11 | 55.41 | 63.77 | 81.83 | 90.80 | 86.08 | 79.71 | 70.69 |
| $Brown_{sentiment}$ | 1500 | 71.52 | 61.78 | 66.29 | 77.53 | 57.70 | 66.17 | 83.14 | 91.17 | 86.97 | 81.11 | 73.14 |
| $Brown_{offensive}$ | 2000 | 74.23 | 63.35 | 68.36 | 70.54 | 51.80 | 59.74 | 82.03 | 90.33 | 85.98 | 79.58 | 71.36 |

Table 6: Result for Brown cluster models based on different data sets: general, sentiment, and offensive tweets.

## 5. Results

Before we discuss our results per research question (see section 1.), we provide an overview of baseline and state of the art systems for abusive language detection on the two data sets that we used, to provide a frame of reference. However, note that our goal is not to improve the state of the art. We are interested in a deeper investigation of Brown clustering features, mostly as an alternative to deep learning approaches, especially when training data are sparse.

### 5.1. Baselines and State of the Art

The two data sets used here have been widely used previously. Table 5 presents the baseline and state of the art results.

For OLID, the results are from SemEval 2019 task 6 (Zampieri et al., 2019b). The best result is based on a fine-tuned BERT model (Liu et al., 2019). The baseline model is a linear SVM model with word unigrams (Zampieri et al., 2019a). For the Waseem data set, the baseline reported by Waseem and Hovy (2016) is a logistic regression model using word $n$-grams. Mishra et al. (2019) achieved the current state of the art results using a combined model of graph convolutional neural network and logistic regression. However, note that these results should not be directly compared with ours since the corpus versions differ, based on when the tweets were retrieved. Additionally, there is no predefined split into training and test data, which means that the data splits are most likely different.

### 5.2. Data Size versus Specialization

In this section, we investigate which type of data is the most useful for creating Brown clusters: either a large scale, but general data set, or a smaller sentiment data set, or an even smaller data set specific for hate speech detection. All of these experiments use logistic regression. The setting based on sentiment data refers to Brown clusters trained on the full sentiment set, i.e., we combined the positive and negative sentiment sets and trained the Brown clusters on the complete set.

The results of these experiments are shown in Table 6. These results show that for OLID, we reach the highest macro-averaged F-score when using Brown clusters based on the large, general data set (Owoputi et al., 2013), reaching 69.82 as opposed to 64.09 based on offensive Brown clusters. For the Waseem data set, however, the best results are based on the sentiment Brown clusters, with an F-score of 73.14 as opposed to 70.69 for the general Brown clusters. This means that the amount of data seems to be more important than having a match with regard to the task or genre.

When we look at the individual classes, we see the overall trend repeated for OLID: Using the general Brown clusters results in the highest values for both precision and recall for both classes. For the Waseem data set, the sentiment Brown clusters work best across precision and recall for the Sexism and the Not-offensive class. Surprisingly, for Racism, the results for the Brown clusters based on offensive tweets are considerably higher than the ones for the other clusters (F: 68.36 vs. 66.29 for the sentiment Brown clusters). We assume that this is due to different levels of overtly abusive language in the different classes. As a methodological side comment, this shows how important it is to use different data sets for such experiments.

### 5.3. Sentiment Specific Brown Clusters

Here, we investigate whether we obtain better results if we separate the sentiment data into two separate sets and train two separate sets of Brown clusters, one on the positive and one on the negative data only. The results of these experiments, using logistic regression are shown in Table 7.

**OLID**

| Method | # cl | Offensive | | | Not offensive | | | Overall | |
|---|---|---|---|---|---|---|---|---|---|
| | | P | R | $F_1$ | P | R | $F_1$ | Acc | $F_1$ |
| $Brown_{pos}$ | 2000 | 66.00 | 41.25 | 50.77 | 80.14 | 91.77 | 85.56 | 77.67 | 68.17 |
| $Brown_{neg}$ | 2000 | 65.81 | 42.50 | 51.65 | 80.43 | 91.45 | 85.58 | 77.79 | 68.62 |
| $Brown_{pos\text{-}wp}$ | 2000 | 63.49 | 50.00 | 55.94 | 82.12 | 88.87 | 85.36 | 78.02 | 70.65 |
| $Brown_{neg\text{-}wp}$ | 2000 | 58.59 | 48.33 | 52.97 | 81.27 | 86.77 | 83.93 | 76.05 | 68.45 |
| $Brown_{pos\_neg}$ | 1500 | 71.43 | 50.00 | 58.82 | 82.66 | 92.26 | 87.20 | 80.47 | 73.01 |
| $Brown_{pos+neg}$ | 2000 | 69.01 | 49.17 | 57.42 | 82.29 | 91.45 | 86.63 | 79.65 | 72.03 |
| $Brown_{pos\_neg\text{-}wp}$ | 2000 | 67.04 | 50.00 | 57.28 | 82.38 | 90.48 | 86.24 | 79.19 | 71.76 |
| $Brown_{pos\_wp+neg\_wp}$ | 2000 | 59.36 | 54.17 | 56.64 | 82.84 | 85.65 | 84.22 | 76.86 | 70.43 |
| $Brown_{pos\_neg\_general}$ | 1500 | 73.21 | 51.25 | 60.29 | 83.09 | 92.74 | 87.65 | **81.16** | **73.97** |
| $Brown_{pos\_neg\_off}$ | 1500 | 64.29 | 37.50 | 47.37 | 79.17 | 91.94 | 85.07 | 76.74 | 66.22 |

**Waseem data set**

| Method | # cl | Racism | | | Sexism | | | Not offensive | | | Overall | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | P | R | $F_1$ | P | R | $F_1$ | P | R | $F_1$ | Acc | $F_1$ |
| $Brown_{pos}$ | 2000 | 72.22 | 61.26 | 66.29 | 71.20 | 58.36 | 64.14 | 83.10 | 89.59 | 86.23 | 80.09 | 72.22 |
| $Brown_{neg}$ | 2000 | 71.35 | 63.87 | 67.40 | 76.72 | 58.36 | 66.29 | 83.49 | 90.71 | 86.95 | 81.17 | 73.55 |
| $Brown_{pos\text{-}wp}$ | 2000 | 70.33 | 67.02 | 68.63 | 76.28 | 63.28 | 69.18 | 85.22 | 90.06 | 87.57 | 82.06 | 75.13 |
| $Brown_{neg\text{-}wp}$ | 2000 | 73.74 | 69.11 | 71.35 | 75.82 | 60.66 | 67.40 | 84.77 | 90.52 | 87.55 | 82.12 | 75.43 |
| $Brown_{pos\_neg}$ | 2000 | 75.98 | 71.20 | 73.51 | 79.18 | 63.61 | 70.55 | 85.54 | 91.26 | 88.31 | 83.46 | 77.46 |
| $Brown_{pos+neg}$ | 1500 | 74.28 | 68.06 | 71.04 | 74.90 | 63.66 | 68.79 | 85.06 | 89.96 | 87.44 | 82.19 | 75.76 |
| $Brown_{pos\_neg\text{-}wp}$ | 1500 | 77.78 | 73.30 | 75.47 | 78.00 | 63.93 | 70.27 | 85.99 | 91.26 | 88.55 | 83.78 | 78.10 |
| $Brown_{pos\_wp+neg\_wp}$ | 2000 | 71.89 | 69.63 | 70.74 | 74.80 | 63.28 | 68.56 | 85.03 | 89.22 | 87.07 | 81.81 | 75.46 |
| $Brown_{pos\_neg\_general}$ | 2000 | 76.67 | 72.25 | 74.39 | 80.58 | 63.93 | 71.30 | 85.83 | 91.73 | 88.68 | **83.97** | 78.12 |
| $Brown_{pos\_neg\_off}$ | 2000 | 77.53 | 72.25 | 74.80 | 78.57 | 64.92 | 71.10 | 85.99 | 91.26 | 88.55 | 83.84 | **78.15** |

Table 7: Result for combining features from separate Brown clusters.

The first set of experiments uses only one type of Brown clusters, either the ones trained on positive sentiment or the ones trained on negative. For both data sets, a comparison of the results between using positive and negative Brown clusters shows little difference, with the negative clusters providing a small boost in performance.

When comparing the results with the complete sentiment model from Table 6, we see that for OLID, the separate models perform better: The positive setting reaches an overall F-score of 68.17, the negative one 68.62, while the general model reaches 63.54. For the Waseem data set, the negative model also outperforms the general model in terms of overall F-score, but the difference is less pronounced than for OLID: 73.55 vs. 73.14. However, the positive setting results in a lower F-score: 72.22. From these results, we can conclude that separating the sentiment information into separate clusters provides better clusters, even though they are trained on fewer data.

When we combine features from two Brown clusters, we can see a clear improvement: The overall F-score increases from 68.62 to 73.01 on OLID and from 73.55 to 77.46 (using standard tokenization) on the Waseem set. This tells us that separate clusters provide different types of information.

We can also answer the question whether it is better to combine the cluster IDs from the positive and negative Brown clusters into a single feature per word ($Brown_{pos\_neg}$) or keep them as two separate features ($Brown_{pos+neg}$): Combining the IDs into a single feature results in a better performance across both data sets and almost all evaluation metrics. The only exception is recall for Sexism in the Waseem data set, which is minimally higher for two separate features.

In the last part of Table 7, we investigate the effects of adding more Brown clustering features from the other two data sets, i.e., the general and the offensive data set, to the separate sentiment features. In this setting, we achieve our best results on both data sets. Adding Brown cluster features from the general domain clearly leads to improvement (from 73.01 to 73.97 F for OLID and from 77.46 to 78.12 F for the Waseem data set). We assume that this is because the general domain Brown clusters provide wider coverage of the words overall. However, there is a significant difference when adding Brown clusters from the offensive data: we reach a higher accuracy on the Waseem data, but a much lower accuracy on OLID. Thus the offensive Brown clusters may provide more specific information helping to separate sexist from racist abuse.

When we compare the results based on normal tokenization to using the word piece tokenization, we see that the latter tends to results in higher overall F-scores for the separate positive and negative Brown clusters, but for the combined models, the situation is less clear: The word piece tokenization provides lower result for OLID but higher results for the single-feature combination ($Brown_{pos\_neg\text{-}wp}$) on the Waseem data set, resulting in an increase in overall F of about 0.6 points. This is due to improved performance on the Racism class.

Overall, we conclude that Brown clustering profits from

**OLID**

| Method | # cl | Offensive | | | Not offensive | | | Overall | |
|---|---|---|---|---|---|---|---|---|---|
| | | P | R | $F_1$ | P | R | $F_1$ | Acc | $F_1$ |
| Char | - | 58.49 | 51.67 | 54.87 | 82.10 | 85.81 | 83.91 | 76.28 | 69.39 |
| Char+Brown | 1500 | 63.59 | 54.58 | 58.74 | 83.33 | 87.90 | 85.55 | 78.60 | 72.15 |
| Word | - | 70.81 | 47.50 | 56.86 | 81.97 | 92.42 | 86.88 | 79.88 | 71.87 |
| Word+Brown | 1500 | 78.18 | 53.75 | 63.70 | 84.03 | 94.19 | 88.82 | **82.91** | **76.26** |
| $CNN_{Word}$ | - | 63.77 | 55.00 | 59.06 | 83.46 | 87.90 | 85.62 | 78.72 | 72.34 |
| $CNN_{Brown+Word}$ | 1500 | 64.56 | 63.75 | 64.15 | 86.04 | 86.45 | 86.24 | 80.11 | 75.19 |
| $BiLSTM_{Word}$ | - | 62.09 | 54.58 | 58.09 | 83.20 | 87.10 | 85.11 | 78.02 | 71.59 |
| $BiLSTM_{Brown+Word}$ | 1500 | 61.01 | 55.42 | 58.08 | 83.33 | 86.29 | 84.79 | 77.67 | 71.43 |

**Waseem data set**

| Method | # cl | Racism | | | Sexism | | | Not offensive | | | Overall | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | P | R | $F_1$ | P | R | $F_1$ | P | R | $F_1$ | Acc | $F_1$ |
| Char | - | 73.26 | 71.73 | 72.49 | 70.59 | 66.89 | 68.69 | 86.13 | 87.73 | 86.92 | 81.74 | 76.03 |
| Char+Brown | 1500 | 74.43 | 68.59 | 71.39 | 73.54 | 70.16 | 71.81 | 86.70 | 89.03 | 87.85 | 82.89 | 77.02 |
| Word | - | 80.34 | 74.87 | 77.51 | 76.31 | 62.30 | 68.59 | 86.03 | 91.54 | 88.70 | 83.84 | 78.26 |
| Word+Brown | 1500 | 78.26 | 75.39 | 76.80 | 78.71 | 67.87 | 72.89 | 87.29 | 91.26 | 89.23 | **84.80** | **79.64** |
| $CNN_{Word}$ | - | 69.77 | 78.53 | 73.89 | 64.43 | 72.46 | 68.21 | 88.46 | 83.36 | 85.84 | 80.66 | 75.98 |
| $CNN_{Brown+Word}$ | 1500 | 64.45 | 86.39 | 73.83 | 65.55 | 76.72 | 70.69 | 90.51 | 80.67 | 85.31 | 80.60 | 76.61 |
| $BiLSTM_{Word}$ | - | 78.45 | 74.35 | 76.34 | 70.63 | 70.16 | 70.39 | 87.22 | 88.20 | 87.71 | 83.02 | 78.15 |
| $BiLSTM_{Brown+Word}$ | 1500 | 63.81 | 85.86 | 73.21 | 77.43 | 65.25 | 70.82 | 88.00 | 86.52 | 87.25 | 82.32 | 77.10 |

Table 8: Results for combining Brown clusters with other types of features and neural architectures.

having the data separated into positive and negative data, and then combining the information into a single feature afterwards. The other settings are only partially useful.

### 5.4. Adding Brown Clusters to Other Settings

Here, we investigate the question whether the features based on Brown clusters provide novel information, or if this information is already present in standard surface features such as words and character $n$-grams or in neural networks. Table 8 presents these results. Here, settings referring to Brown clusters refer to the best setting from the previous section, i.e., $Brown_{pos\_neg}$.

For logistic regression models using characters or word $n$-gram, adding Brown cluster features clearly enhances performance: On OLID, we gain about 5 points absolute in F-score when adding Brown clusters to words, reaching 76.26. Note that this is also about 2.3 points higher than the best results in Table 7. On the Waseem data set, the gain is smaller, we reach an F-score of 79.64, as compared to 78.26 on words only, and to 78.15 – the best result in Table 7.

For CNN models with pure word embeddings, adding Brown cluster features also aids performance in trends similar to the word features. For the BiLSTM model, in contrast, adding Brown cluster features leads to a deterioration of the results.

Overall, we can conclude that most models profit from the addition of Brown clusters, especially traditional machine learning models.

## 6. Conclusion and Future Work

We have investigated the use of Brown clusters as features in abusive language detection. Previous work using such features, such as (Malmasi and Zampieri, 2018), have born out the assumption by Socher et al. (2011) that Brown clusters cannot represent sentiment. Our work, in contrast, shows that when we train separate Brown clusters on positive and negative sentiment data, a machine learning approach based on logistic regression profits from such features. They add important information, even when combined with words or character $n$-grams or with standard word embeddings in a convolutional neural network. When balancing data size and genre specificity of the data used for training the Brown clusters, we found a good balance in using data automatically annotated for sentiment.

However, one of the most striking results is the amount of variance in results between the two data set for offensive language detection that we have used. We need to conclude that this calls for a deeper investigation into different data sets for offensive language detection, to better understand the differences between them. One first step towards a better understanding of the differences in English data sets has been made by Wiegand et al. (2019) with regard to sampling bias. Steimel et al. (2019) provide first insights into a multilingual setting. However, both are just first steps towards a better understanding of factors that influence the performance of automatic approaches to offensive language recognition.

## 7. Bibliographical References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. (2016). Tensorflow: A system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, pages 265–283, Savannah, GA.

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, Perth, Australia.

Bodapati, S., Gella, S., Bhattacharjee, K., and Al-Onaizan, Y. (2019). Neural word decomposition models for abusive language detection. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 135–145, Florence, Italy.

Brown, P. F., Desouza, P. V., Mercer, R. L., Pietra, V. J. D., and Lai, J. C. (1992). Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479.

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

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Founta, A. M., Djouvas, C., Chatzakou, D., Leontiadis, I., Blackburn, J., Stringhini, G., Vakali, A., Sirivianos, M., and Kourtellis, N. (2018). Large scale crowdsourcing and characterization of twitter abusive behavior. In *Proceedings of the Twelfth International AAAI Conference on Web and Social Media (ICWSM)*, Stanford, CA.

Go, A., Bhayani, R., and Huang, L. (2009). Twitter sentiment classification using distant supervision. CS224N project report, Stanford University.

Kübler, S., Liu, C., and Sayyed, Z. A. (2018). To use or not to use: Feature selection for sentiment analysis of highly imbalanced data. *Natural Language Engineering*, 24(1):3–37.

Kumar, R., Ojha, A. K., Malmasi, S., and Zampieri, M. (2018). Benchmarking aggression identification in social media. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC)*, pages 1–11, Santa Fe, NM.

Liang, P. (2005). *Semi-supervised Learning for Natural Language*. Ph.D. thesis, Massachusetts Institute of Technology.

Liu, P., Li, W., and Zou, L. (2019). NULI at SemEval-2019 task 6: Transfer learning for offensive language detection using bidirectional transformers. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 87–91, Minneapolis, MN.

Malmasi, S. and Zampieri, M. (2018). Challenges in discriminating profanity from hate speech. *Journal of Experimental & Theoretical Artificial Intelligence*, 30:1–16.

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)*, Miyazaki, Japan.

Mishra, P., Del Tredici, M., Yannakoudakis, H., and Shutova, E. (2019). Abusive language detection with graph convolutional networks. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2145–2150, New Orleans, LA.

Owoputi, O., O'Connor, B., Dyer, C., Gimpel, K., Schneider, N., and Smith, N. A. (2013). Improved part-of-speech tagging for online conversational text with word clusters. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 380–390, Atlanta, GA.

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*, 12:2825–2830.

Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2227–2237, New Orleans, LA.

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, Valencia, Spain.

Socher, R., Pennington, J., Huang, E. H., Ng, A. Y., and Manning, C. D. (2011). Semi-supervised recursive autoencoders for predicting sentiment distributions. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 151–161, Edinburgh, UK.

Steimel, K., Dakota, D., Chen, Y., and Kübler, S. (2019). Investigating multilingual abusive language detection: A cautionary tale. In *Proceedings of the Conference on Recent Advances in NLP (RANLP)*, pages 1151–1160, Varna, Bulgaria.

Warner, W. and Hirschberg, J. (2012). Detecting hate speech on the World Wide Web. In *Proceedings of the Second Workshop on Language in Social Media*, pages 19–26, Montreal, Canada.

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.

Wester, A., Øvrelid, L., Velldal, E., and Hammer, H. L. (2016). Threat detection in online discussions. In *Proceedings of the 7th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 66–71, San Diego, CA.

Wiegand, M., Ruppenhofer, J., and Kleinbauer, T. (2019). Detection of abusive language: The problem of biased datasets. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 602–608, Minneapolis, MN.

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 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Minneapolis, MN.

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 the 13th International Workshop on Semantic Evaluation*, pages 75–86, Minneapolis, MN.

Zhu, J., Tian, Z., and Kübler, S. (2019). UM-IU@LING at SemEval-2019 task 6: Identifying offensive tweets using BERT and SVMs. In *Proceedings of SemEval-2019: International Workshop on Semantic Evaluation*, pages 788–795, Minneapolis, MN.