NLP_Passau at SemEval-2020 Task 12: Multilingual Neural Network for Offensive Language Detection in English, Danish and Turkish

Omar Hussein, Hachem Sfar, Jelena Mitrović, Michael Granitzer
Faculty of Computer Science and Mathematics, University of Passau, Germany
{hussei05|sfar01}@ads.uni-passau.de
{jelena.mitrovic|michael.granitzer}@uni-passau.de

Abstract

This paper describes a neural network (NN) model that was used for participating in the OffensEval, Task 12 of the SemEval 2020 workshop. The aim of this task is to identify offensive speech in social media, specifically in tweets. The model we used, C-BiGRU, is composed of a Convolutional Neural Network (CNN) along with a bidirectional Recurrent Neural Network (RNN). A multidimensional numerical representation (embedding) for each of the words in the tweets, that were used by the model, was determined using fastText. This was utilized with a dataset of labeled tweets to train the model on detecting combinations of words that may convey an offensive meaning. The model was then used in the sub-task A of the English, Turkish and Danish competitions of the workshop, achieving F1 scores of 90.88%, 76.76% and 76.70%, respectively.

1 Introduction

With around 3.08 billion users in 2020, the number of social media users is continuing to grow. One of the most important features of social media is the presence of user generated content such as photos, videos or texts (Obar and Wildman, 2015). The reliance on user generated content, however, presents a significant challenge in moderating such content which has the potential to instigate violence within communities in real life, if it contains hate-speech (Laub, 2019). There has been growing interest from social media companies, fueled by the pressure from the government agencies, to detect and try to block such harmful content as in some cases it could lead the platform on which the content was distributed into paying significant fines (Deutsche-Welle, 2019). Furthermore, using computer algorithms to automatically detect any content that could contain hateful inclinations is a challenging task that has also attracted the attention of the scientific community as can be seen by various workshops, competitions and conferences in recent years: TRAC (Kumar et al., 2018), EVALITA hate-speech detection task (Cristina et al., 2018), GermEval (Wiegand et al., 2018), (Fersini et al., 2018) and HatEval (Basile et al., 2019).

The most prominent effort in this regard has been the OffensEval challenge (Zampieri et al., 2019), which sees its second edition this year as the 12th task of SemEval, Multilingual Offensive Language Identification in Social Media, or OffensEval 2 (Zampieri et al., 2020).The Offensive Language Identification Dataset or OLID which consists of a collection of 14,200 English tweets with three levels of annotation for each of them is provided. The first level of annotation identifies whether a tweet contains offensive language or not, which is the aim of Sub-task A. The second level determines the type of offensive language in the tweet which is the focus of Sub-task B, and lastly Sub-task C aims at determining the third annotation level which labels the intended target of the offensive speech in the tweet. In addition to English, four other languages (Arabic, Danish, Greek and Turkish) have datasets for Sub-task A.

In this paper, we describe the system that was used for our submissions to Sub-task A for 3 languages (Danish, English and Turkish). In the second section, we provide a literature overview of the various systems that were used for detecting offensive language in text. In the third section, a description of the pre-processing techniques, the experimental setup and tools that we used is given. The fourth section gives an overview of the word embeddings and the NN model that we used. The fifth section shows our results and their analysis. Lastly, we conclude the paper and recommend possible enhancments of our system.

Proceedings of the 14th International Workshop on Semantic Evaluation, pages 2090–2097
Barcelona, Spain (Online), December 12, 2020.

2 Background

Various systems have been created over the years to address the task of identifying the patterns of offensive or harmful language in text based data. A summary of various approaches to solving this problem is given by Radivchev and Nikolov (2019). The performance of a collection of models was additionally compared. A system designed by Zhang et al. (2015) utilized convolutional filters of a single dimension for extracting features from the embeddings of characters to categorize text. The authors del Arco et al. (2019) developed an SVM based system that integrated lexical features for categorizing text which they used to participate in OffensEval 1. An end to end Convulational Neural Network (CNN) with fine tuned fastText embeddings was used by Torres and Vaca (2019) which performed better than Linear Regression systems and other NN models. Rozental and Biton (2019) designed an NN model termed "Multiple Choice CNN" which is a type of a convolutional NN. They used the model in conjunction with their own novel contextual embedding to participate in Tasks 5 (Basile et al., 2019) and Task 6 of SemEval 2019. Merenda et al. (2018) used source driven representation to detect hate speech. Offensive tweets in German were classified using a unified model consisting of a CNN and an LSTM (dubbed C-LSTM) by Birkeneder et al. (2018), while Bai et al. (2018) used an ensemble model consisting of a Linear SVM, a CNN, and a Logistic Regressor as a meta-classifier.

We base our core model for participation in OffensEval 2 on the model we have used to participate in OffensEval 1. The difference between the previous system and the new one is in the implementation of different pre-processing techniques, and more importantly, the utilization of fastText instead of word2vec for getting the embedding of words. The previous system, C-BiGRU (Mitrović et al., 2019) has also shown its capability of reliably identifying offensive speech in German language text as illustrated by its participation in the GermEval workshop (Birkeneder et al., 2018), and now it has proved to be also reliable for Danish and Turkish as well.

3 Baseline Model

To evaluate the performance of our model, we built our first classifier by fine tuning a pre-trained BERT-Base Uncased Transformer (Devlin et al., 2018). The BERT model consists of 12 Transformer blocks, 12 self-attention heads, and 768 hidden dimension with a total of 110M parameters. It was trained on the Book Corpus (800M words) and the English Wikipedia (2,500M words). The model also includes a special classification embedding [CLS] at the beginning of every sentence, and this token in the final layer was extracted as the aggregate sequence representation for the current classification task. Then a linear layer of 768 dimensions was added on top of the model, using the [CLS] embeddings of the whole input sequence to predict a binary label. BERT tokenizes parts of words instead of tokenized words.

4 System Setup

In this section, a description of the data that was used for training the system is provided, along with the pre-processing techniques that were utilized.

4.1 Data

We mainly relied on the OLID 2019 dataset for the English language that was set up by (Zampieri et al., 2019) and which was provided for the first time in the SemEval 2019 workshop.

The English dataset consists of a collection of 14,100 tweets split between 13,240 for training and the rest for testing (Rosenthal et al., 2020). This set was annotated using a a three-level annotation system which matches the previously described sub-tasks of the SemEval workshop. We have only used the first level which concerns Sub-task-A that we participated in. This level of the dataset is split unevenly between 4,400 offensive and 8,840 non-offensive tweets.

The Danish and Turkish datasets were provided by the SemEval 2020 organizers and had only one annotation level that complied with Sub-task A. They were arranged similarly with the Danish dataset having a collection of 2961 tweets split into 384 offensive and 2,577 non-offensive tweets (Sigurbergsson and Derczynski, 2020), while the Turkish dataset consists of a total of 31,277 tweets, split into 6,046 offensive tweets and 25,231 non-offensive tweets (Çöltekin, 2020).

All datasets have imbalanced distributions which is why the F1 macro average score is the main metric we and the SemEval workshop rely on for evaluation. For the three languages, we have separated 10% of the data for testing and further split the remaining data into 90% and 10% for training and validation respectively. In addition, to avoid over-fitting, we used 10-fold cross validation on both the training and validation data.

4.2 Pre-processing

For the three languages, any HTML encoded characters in the tweet being pre-processed were swapped for their equivalents, or for token representations. In addition, any emojis, URLs and multiple spaces were removed. Then tokenization was performed using the nltk TweetTokenizer with all the tokens containing special characters such as ('\', '/', '&', '–') being split further.

Two unique pre-processing techniques were used for the English tweets. First, we ignored any tokens that are considered 'stop words'. Second, we handled text abbreviations (e.g. OMG as an abbreviation of Oh My God) that are very common in social media in general and in Twitter in particular due to the limited characters available for each tweet. This was done by checking if a token is in a list of common abbreviations and swapping it for the tokens of the equivalent expression.

5 System Overview

In this section, a description of the embedding utilized for the created tokens and of the C-BiGRU model is given.

5.1 Word Embedding

One of the main aspects that differentiates the system we used for OffensEval 2 apart from our system used for OffensEval 1 is the embedding used for the tokens. As opposed to our previous system that utilized word2vec embeddings for the tokens, this system utilizes fastText. The embeddings for both systems were obtained using a pre-trained model with word2vec using an embedding dimension of 400 and fastText using a smaller value of 300. Despite using a smaller embedding dimension, utilizing fastText instead of word2vec has improved the performance of the system. For each of the three languages, we use a pre-trained model with an embedding dimension of 300 to obtain pre-trained embeddings for the input tokens.

5.2 C-BiGRU

All tweets are pre-processed and tokenized before being uniformed into a size of 150 tokens through padding by removing any additional tokens or adding masking tokens until 150 is reached after which point the sequence can be used as an input to the C-BiGRU model.

5.2.1 Input Handling Layers

We create a dictionary out of all the tokens that appear more than once in the training data and use it to create the first layer of the C-BiGRU model which is the embedding layer. The layer is composed of a matrix of size $n * d$ where n is the total number of tokens handled and d is the size of the embedding for all of them. The handled tokens include the masking token, which will have an embedding vector of zeroes, and a special token for testing data tokens which were not in the dictionary. As recommended by (He et al., 2015), such special tokens will get a random embedding from a uniform distribution within the range $\left[-\sqrt{\frac{6}{dim}}, \sqrt{\frac{6}{dim}}\right]$. The output then passes through a dropout layer with a dropout rate of 0.2 so that overfitting is avoided.

5.2.2 Convolutional Layer

The next layer is the convolutional layer which utilizes 4 1D CNNs to extract internal features from the sequence of tokens. Still, each of the 4 CNNs has a different window size (the sizes are 2, 3, 4, 5) and they all produce output of the same length via padding. In addition, each of them performs 128 different convolutions on the token sequence and utilizes ReLu as an activation function. The resulting output is then concatenated into a feature map of a $150 * 512$ matrix before being passed onto the next layer.

5.2.3 Recurrent Layer

Capturing of long-term dependencies of input sequences is performed by the next layer which consists of a bidirectional GRU network (BiGRU). As one of the advanced RNNs, GRU, along with LSTM, overcomes the vanishing gradient problem by utilizing reset and update gates as part of its mechanism. Its gating mechanism, which is simpler than LSTM, is designed in a manner that allows it to have more persistent memory by simplifying the memorization of long-term dependencies, and it has been reported to outperform LSTM on smaller datasets (Chung et al., 2014). The layer consists of 2 GRU layers and one of them receives the output from the previous layer while the other one receives the same output but in its reversed form. The output of this layer is the concatenation of the hidden states of the 2 GRU layers producing a $150 * 128$ matrix as an output.

5.2.4 Final Dense Layers

The BiGRU output then passes through a global max pooling layer which condenses it into a single vector of 128 nodes that is then fully connected into a hidden layer with a size of 32 nodes. This is followed by another dropout layer (also with a rate of 0.2) before ending with a single output node which utilizes the sigmoid activation function.

During the training of the model, we use binary cross entropy as the error function and Adam optimizer (Kingma and Ba, 2014) is used for updating the network weights. A maximum of 5 epochs with a batch size of 32 is set up and early stopping is implemented. Figure [1] shows an overview of the architecture.

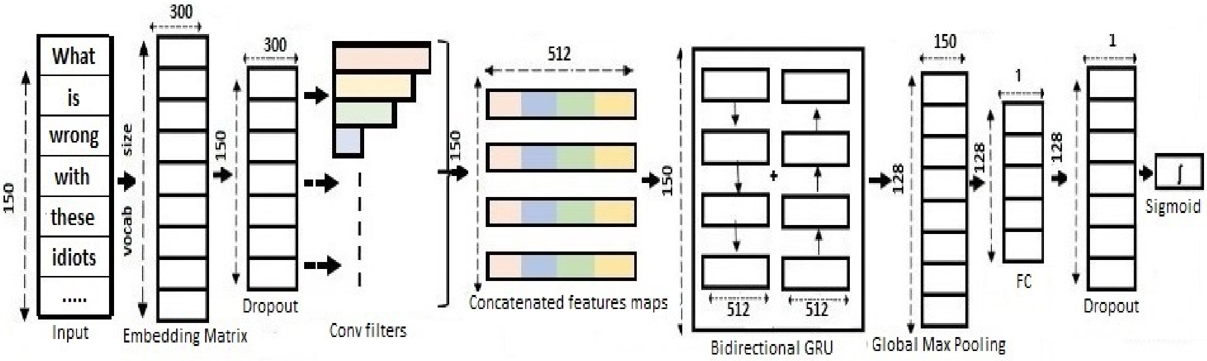


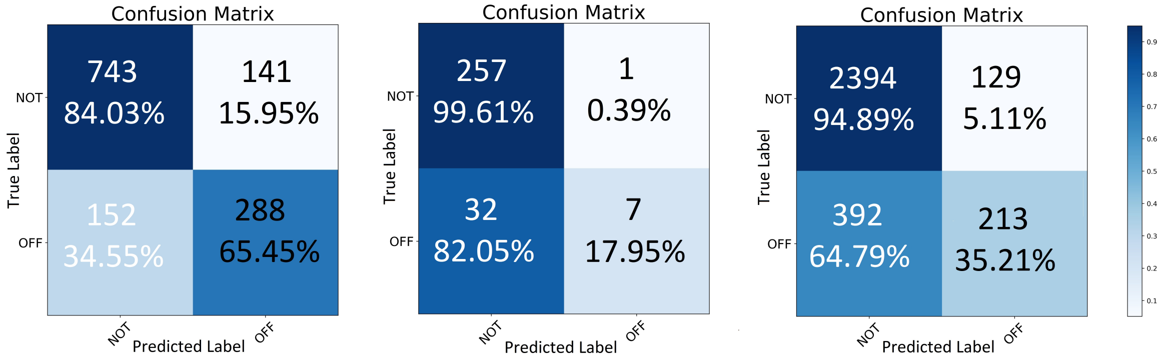
Figure 1: Architecture representation of C-BiGRU.

5.2.5 Hyperparameters

Hyperparameters such as the dropout values or the number of nodes in a layer were chosen based on previous similar work that we encountered during research and also based on our own experimentation and selecting the values that performed the best when validating the resultant model.

6 Results and Analysis

For each of the three languages, after training the C-BiGRU model, its first assessment was done using the validation set which is 10% of the data of each language. For Turkish, the model achieved an accuracy of 83.34%, a macro-average recall, precision and F1 score of 65%, 74% and 68% respectively. For Danish the accuracy was 88.89%, the macro-average recall, precision and F1 score were 59%, 88% and 62% respectively. Lastly, the English results were 77.87% accuracy and the macro-average of recall, precision and F1 score were all 75%. Figure [2] shows the resulting confusion matrices for each language.



English Danish Turkish

Figure 2: Confusion Matrices for the testing results of the 3 languages using C-BiGRU.

6.1 Submission Results

After training and testing the model for each of the three languages, it was used to label the data for OffensEval 2. The baseline BERT model achieved an F1 score of 90.36% for English, 76.2% for Danish, and 77.89% for Turkish. The C-BiGRU model achieved an F1 score of 90.88% for English, 76.7% for Danish, and 76.76% for Turkish. In conclusion, the C-BiGRU model achieved a slightly higher score for English and Danish, while the BERT model achieved a higher score in Turkish.

6.2 Analysis

The C-BiGRU model has shown its ability of differentiating between offensive and non-offensive tweets in the three languages with a considerable certainty. This is a follow-up of its previous performance in identifying offensive German tweets which proves its capability of handling 4 languages: Danish, English, German and Turkish. In addition, its F1 scores of the validation results are consistent with the submission results in terms of the order of the performance of the three languages with English having the best performance followed by Turkish and the least performance being for the Danish set.

6.2.1 Data

It can be observed that the model has a better performance in handling English tweets compared with Danish and Turkish. One reason that might explain the difference in performance between English and Danish handling is that English had a larger training dataset (14,100 for English vs. only 2961 for Danish), however the same cannot be said for the discrepancy between English and Turkish handling since the Turkish training data (31,277) is larger than the English one. One possible explanation is that, although larger, the Turkish data set was not diverse enough to provide a proper sampling that will keep the model generic after training; another reason could be the additional pre-processing techniques done to the English dataset.

6.2.2 Embedding

Another point that came to our attention during the experimentation is that fastText (Bojanowski et al., 2016) provides better pre-trained embedding than word2vec (Mikolov et al., 2013). We performed an experiment where we used the same C-BiGRU model, training and testing data as our control variables while utilizing different word embeddings. The system with the word2vec embedding achieved a macro-average F1 score of 65.95%, while the system with fastText achieved a score of 76%.

One possible reason for that could be the fact that fastText utilizes either Skipgram or CBOW (Continuous bag of words) mechanisms. It can still generate a vector representation for the meaning of tokens that have not appeared in its training corpus with which it was pre-trained. It performs this by adding the character n-gram of all the n-gram representations. Essentially each word is treated as a collection of its constituent n-grams. For example the embedding of "egypt" with the n = 3 is going to be the summation of the vectors of the following: “_eg”, “egy”, “gyp”, “ypt”and “pt_”. So even if its pre-trained model gets a new word, it might still be able to accurately represent its embedding by using the embedding of a known part of the word. This is superior to word2vec’s method of creating embeddings for each word as a singular atomic entity.

7 Conclusion and Future Work

In this paper, the system that we used to participate in OffensEval 2020 is shown and a detailed description is provided of the architecture of the C-BiGRU model and the pre-processing of the data that is utilized. The model achieves a macro-average F1 score of 90.882%, 76.76% and 76.70% for the English, Turkish and Danish languages respectively on labelling the OffensEval 2 dataset.

7.1 Data Enhancement

One limitation of the data is that all of the tweets are separate with no links between them. However, a real tweet can be a comment to another one which can help provide context that could help in determining whether the text in the tweet contains offensive language or not. Thus, we recommend working with data that takes into consideration the link between individual tweets as this provides the potential for an NN model to use the link between tweets as an additional factor when deciding whether they are offensive or not.

7.2 Handling of new tokens

One additional aspect that could be improved is the handling of input tokens that were not available during the training phase of the C-BiGRU model. Currently we utilize a vector of random distributions as embedding for such tokens. Therefore, a better alternative could be to use a matrix of embeddings for all possible n-grams that can be extracted from tokens. This ties in nicely with fastText functionality of providing n-gram embeddings which will increase the probability of finding a more suitable embedding for a new input token that has not appeared in the training phase.

7.3 Handling figurative language

Offensive language is very often implicit and rich with rhetorical figures. In future work, we will include the rhetorical features based on the work in (Mladenović et al., 2017) and (Mitrovic et al., 2020). The use of figurative language and its relationship with abusive/offensive language will be further explored and may help in creating a new dataset that can help to address messages with a strong abusive effect but weak surface forms, for example in the rhetorical figure litotes (Mitrović et al., 2017), e.g. “He is not the smartest pea in the pod”, or ”She is not the sharpest tool in the shed”. We will also work on a finer-grained difference between implicit and explicit offensive messages, following the methods that are based on the OLID dataset and envisaged in (Caselli et al., 2020).

7.4 Acknowledgement

SPONSORED BY THE

Federal Ministry
of Education
and Research

The project on which this report is based was funded by the German Federal Ministry of Education and Research (BMBF) under the funding code 01—S20049. The author is responsible for the content of this publication.

References

Xiaoyu Bai, Flavio Merenda, Claudia Zaghi, Tommaso Caselli, and Malvina Nissim. 2018. RuG at GermEval: Detecting Offensive Speech in German Social Media. In Josef Ruppenhofer, Melanie Siegel, and Michael Wiegand, editors, *Proceedings of the GermEval 2018 Workshop*, Wien, Austria. ÖAW Austrian Academy of Sciences.

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. 2019. Semeval-2019 task 5: Multilingual detection of hate speech against immigrants and women in twitter. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 54–63.

Bastian Birkeneder, Jelena Mitrović, Julia Niemeier, Leon Teubert, and Siegfried Handschuh. 2018. upInf - Offensive Language Detection in German Tweets. In Josef Ruppenhofer, Melanie Siegel, and Michael Wiegand, editors, *Proceedings of the GermEval 2018 Workshop*, Wien, Austria. ÖAW Austrian Academy of Sciences.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Tommaso Caselli, Valerio Basile, Jelena Mitrović, Inga Kartoziya, and Michael Granitzer. 2020. I Feel Offended, Don't Be Abusive! Implicit/Explicit Messages in Offensive and Abusive Language. In Nicoletta Calzolari (Conference chair), Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Koiti Hasida, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asuncion Moreno, Jan Odijk, Stelios Piperidis, and Takenobu Tokunaga, editors, *Proceedings of the Twelfth International Conference on Language Resources and Evaluation (LREC 2020)*, Marseille, France, May 11–16, 2020. European Language Resources Association (ELRA).

Çağrı Çöltekin. 2020. A Corpus of Turkish Offensive Language on Social Media. In *Proceedings of the 12th International Conference on Language Resources and Evaluation*. ELRA.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *arXiv preprint arXiv:1412.3555*.

Bosco Cristina, Dell'Orletta Felice, Fabio Poletto, Sanuguinetti Manuela, and Tesconi Maurizio. 2018. Overview of the EVALITA Hate Speech Detection (HaSpeeDe) Task. In Tommaso Caselli, Nicole Novielli, Viviana Patti, and Paolo Rosso, editors, *Proceedings of the 6th evaluation campaign of Natural Language Processing and Speech tools for Italian (EVALITA'18)*, Turin, Italy. CEUR.org.

Flor Miriam Plaza del Arco, M. Dolores Molina-Gonzalez, M. Teresa Martin-Valdivia, and L. Alfonso Urena-Lopez. 2019. Sinai at semeval-2019 task 6: Incorporating lexicon knowledge into svm learning to identify and categorize offensive language in social media. In *Proceedings of the 13th International Workshop on Semantic Evaluation*.

Deutsche-Welle. 2019. Eu hails social media crackdown on hate speech. *Deutsche Welle*.

Elisabetta Fersini, Debora Nozza, and Paolo Rosso. 2018. Overview of the evalita 2018 task on automatic misogyny identification (AMI). In Tommaso Caselli, Nicole Novielli, Viviana Patti, and Paolo Rosso, editors, *Proceedings of the Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2018) co-located with the Fifth Italian Conference on Computational Linguistics (CLiC-it 2018), Turin, Italy, December 12-13, 2018*, volume 2263 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *In Proceedings of the IEEE international conference on computer vision, pages 1026–1034*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Ritesh Kumar, Atul Kr. Ojha, Shervin Malmasi, and Marcos Zampieri. 2018. Benchmarking Aggression Identification in Social Media. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbulling (TRAC)*, Santa Fe, USA.

Zachary Laub. 2019. Hate speech on social media: Global comparisons. *Council on Foreign Relations*.

Flavio Merenda, Claudia Zaghi, Tommaso Caselli, and Malvina Nissim. 2018. Source-driven representations for hate speech detection. In *CLiC-it*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26*, pages 3111–3119.

Jelena Mitrović, Cliff O'Reilly, Miljana Mladenović, and Siegfried Handschuh. 2017. Ontological representations of rhetorical figures for argument mining. *Argument & Computation*, 8:267–287.

Jelena Mitrović, Bastian Birkeneder, and Michael Granitzer. 2019. nlpUP at SemEval-2019 task 6: A deep neural language model for offensive language detection. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 722–726, Minneapolis, Minnesota, USA, June. Association for Computational Linguistics.

Jelena Mitrovic, Cliff O'Reilly, Randy Allen Harris, and Michael Granitzer. 2020. Cognitive modeling in computational rhetoric: Litotes, containment and the unexcluded middle. In Ana Paula Rocha, Luc Steels, and H. Jaap van den Herik, editors, *Proceedings of the 12th International Conference on Agents and Artificial Intelligence, ICAART 2020, Volume 2, Valletta, Malta, February 22-24, 2020*, pages 806–813. SCITEPRESS.

Miljana Mladenović, Cvetana Krstev, Jelena Mitrović, and Ranka Stanković. 2017. Using lexical resources for irony and sarcasm classification. In *Proceedings of the 8th Balkan Conference in Informatics*, BCI '17, New York, NY, USA. Association for Computing Machinery.

Jonathan A. Obar and Steve Wildman. 2015. Social media definition and the governance challenge: An introduction to the special issue. In *Telecommunications Policy*. SSRN Electronic Journal.

Victor Radivchev and Alex Nikolov. 2019. Nikolov-radivchev at semeval-2019 task 6:offensive tweet classification with bert and ensembles. In *Proceedings of the 13th International Workshop on Semantic Evaluation*.

Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Marcos Zampieri, and Preslav Nakov. 2020. A Large-Scale Semi-Supervised Dataset for Offensive Language Identification. In *arxiv*.

Alon Rozental and Dadi Biton. 2019. Amobee at semeval-2019 tasks 5 and 6: Multiple choice cnn overcontextual embedding. In *Proceedings of the 13th International Workshop on Semantic Evaluation*.

Gudbjartur Ingi Sigurbergsson and Leon Derczynski. 2020. Offensive Language and Hate Speech Detection for Danish. In *Proceedings of the 12th Language Resources and Evaluation Conference*. ELRA.

Johnny Torres and Carmen Vaca. 2019. Jtml at semeval-2019 task 6:offensive tweets identification using convolutional neural networks. In *Proceedings of the 13th International Workshop on Semantic Evaluation*.

Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. 2018. Overview of the GermEval 2018 Shared Task on the Identification of Offensive Language. In *Proceedings of GermEval*.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. Predicting the Type and Target of Offensive Posts in Social Media. In *Proceedings of NAACL*.

Marcos Zampieri, Preslav Nakov, Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Hamdy Mubarak, Leon Derczynski, Zeses Pitenis, and Çağrı Çöltekin. 2020. SemEval-2020 Task 12: Multilingual Offensive Language Identification in Social Media (OffensEval 2020). In *Proceedings of SemEval*.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *NIPS'15: Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*.