Ensemble-based Semi-Supervised Learning for Hate Speech Detection

Safa Alsafari[a], Samira Sadaoui[a]
[a] Department of Computer Science, University of Regina, 3737 Wascana Pkwy, Regina, Canada

Abstract

Large and accurately labeled textual corpora are vital to developing efficient hate speech classifiers. This paper introduces an ensemble-based semi-supervised learning approach to leverage the availability of abundant social media content. Starting with a reliable hate speech dataset, we train and test diverse classifiers that are then used to label a corpus of one million tweets. Next, we investigate several strategies to select the most confident labels from the obtained pseudo labels. We assess these strategies by re-training all the classifiers with the seed dataset augmented with the trusted pseudo-labeled data. Finally, we demonstrate that our approach improves classification performance over supervised hate speech classification methods.

Keywords: Hate Speech Classification; Semi-Supervised Learning; Deep Learning; Pseudo Label Selection; Confidence Threshold.

Introduction

Past studies on hate speech detection have adopted supervised learning across all the examined languages, including English (Zampieri 2020; Zampieri et al. 2019) and Arabic (Alsafari, Sadaoui, and Mouhoub 2020c; Mubarak et al. 2020). These studies were conducted with small corpora because obtaining labeled data is costly and time-consuming. Indeed, manually annotating textual data requires experts who are native speakers of the spoken language. To address the problem of data scarcity, our work explores semi-supervised learning to take advantage of the tremendous amount of content on Twitter. Semi-supervised learning usually achieves higher accuracy than supervised learning and requires much less data annotation, saving human effort and time (Zhu and Goldberg 2020). Moreover, the majority of research conducted on semi-supervised learning has focused on standard machine learning.

The present paper aims to improve the performance of supervised hate speech classification models. To this end, we introduce an ensemble-based semi-supervised learning (ESSL) approach. To the best of our knowledge, only one study has adopted the semi-supervised learning paradigm for hate speech detection; this study was conducted for the English language and was based on co-training two classifiers(Rosenthal et al. 2020). We evaluate our approach's performance using a robust labeled Arabic dataset (publicly available) developed and tested by (Alsafari, Sadaoui, and Mouhoub 2020c) and a massive unlabeled dataset of one million tweets that we scraped from Twitter, cleaned, and normalized. As baseline models for the ESSL process, we develop three diverse hate speech classifiers: SVM with Word and Char Ngram, CNN, and BiLSTM; the latter two were both merged with W2V Skipgram Word embeddings. In this way, the classifiers create a heterogeneous representation of the textual data, producing a robust multi-view of the instances. We then utilize these classifiers to label the sizeable unlabeled Twitter corpus.

Nevertheless, the success of any semi-supervised classification method lies in selecting trusted data to avoid learning from unreliable data that may contain noise (van Engelen and Hoos 2019; Elshaar and Sadaoui 2020). Therefore, we propose several strategies to select the most confident predictions for both classes (Hate vs. Clean) by varying the confidence level of the predictions for each strategy; the strategies include data balancing, average voting, and majority voting. To demonstrate the ESSL approach's efficacy, we provide the predictive accuracy results after re-training the three classifiers on the newly produced datasets: the seed dataset augmented with the trusted pseudo-labeled datasets obtained by the different strategies (five in total). More precisely, we evaluate 18 hate speech classification models using the same testing dataset for a fair comparison.

We organize the paper as follows. Section 2 reviews recent research on semi-supervised learning approaches, including self-training and co-training. Section 3 presents the seed training and testing Arabic datasets as well as the large-scale unlabeled Twitter dataset. Section 4 describes the phases of our ESSL approach and the five data selection strategies. Section 5 carries out several experiments to validate our ESSL approach. Finally, Section 6 highlights the findings of our research.

Related Work

Recently, researchers have become interested in exploring pseudo-labeled data to address the scarcity problem of annotated data. Most existing methods employ standard machine learning. In this section, we review recent papers published in 2020. One study (Li et al. 2020) proposed an SSSL approach based on the "local cores" concept to tackle the adequacy and scarcity of labeled data, both of which are well-known problems in ML. The authors solved the problem of the insufficient initial labeled dataset by searching for the local cores in the unlabeled dataset. In the authors' method, local cores are used to show the data distribution, and their labels are predicted via co-training or active labeling. Then, the local cores are used to augment the labeled dataset. Next, two base classifiers using SVM and KNN are trained on the augmented labeled dataset. Using several UCI datasets, the authors showed that their proposed method is superior to several other self-labeled methods.

In the context of speech recognition, (Kahn, Lee, and Hannun 2020) devised an SSSL approach based on an encoder-decoder with attention. The authors' approach comprises several phases: 1) training a robust acoustic model on a small paired dataset, 2) fitting a language model with a large-scale text corpus to produce the pseudo-labels, 3) adopting two filtering techniques to remove noisy pseudo-labels, and 4) training an ensemble of acoustic models to augment pseudo-label diversity. Based on a paired and unpaired speech recognition corpus with clean and noisy settings, the experiments with single and ensemble models showed that the SSSL approach's performance was much better than a baseline model trained on only the paired dataset. In this work, ensembles of five and four models outperformed the single model with clean and noisy settings, respectively. Later,(Xu et al. 2020) explored the combination of the SSSL defined by (Kahn, Lee, and Hannun 2020) with unsupervised pre-training to take advantage of unlabeled audio data. The authors experimented with the combined approach using two benchmark datasets and attained the best performance in the literature. They concluded that the two approaches complement each other for speech recognition.

For image classification, (Nartey et al. 2020) proposed an "easy-to-hard" self-training approach based on CNN. The approach leverages unlabeled data by pseudo-labeling them and then adding the most confident examples to the labeled dataset. An image classifier was then trained using the expanded dataset. The most confident pseudo-labeled samples were selected based on a confidence threshold, and the authors experimented with three threshold settings: top 5%, top 10%, and top 20%. The proposed SSSL method obtained higher accuracy than fine-tuning over five image datasets (two standard and three coarse datasets). It also outperformed three supervised approaches on five out of the six datasets. According to these experiments, the best confidence threshold for pseudo-labeled selection is 10%.

Lastly, in the domain of hate speech detection, only (Rosenthal et al. 2020) investigated SSSL to benefit from the vast content of posts on Twitter. Based on democratic co-training, the authors developed a supervised dataset comprising over nine million English posts labeled via the SSSL process. The authors employed this type of co-training to train several models, including Bert, PMI, LSTM, and FastText, on a small labeled dataset. In the authors' method, the most confidently classified positive examples of the unlabeled dataset are retained for the next learning iteration. The confident data comprise the aggregation of the confidences obtained by the models. The new supervised dataset increased the predictive performance compared to the original dataset. The authors also thoroughly examined easy and hard examples.

Labeled and Unlabeled Corpora

In a recent study,(Alsafari, Sadaoui, and Mouhoub 2020c) built a robust hate speech corpus written in the two common Arabic languages: Modern Standard Arabic, which is understandable by all Arabic speakers, and the Gulf Arabic dialect, which is spoken in the countries of the Arabian Peninsula. The authors first queried the Twitter platform using four different searching strategies: keyword, hashtag, profile, and defensive methods. After data cleansing, they obtained a tally of 5,340 tweets annotated by rigorous labeling and normalization processes:

- Clean (3480 instances): Tweets that do not contain any offensive and hateful speech, such as profanity, insults, threats, and swear words.
- Hate (1860 instances): Tweets that attack or threaten individuals or groups based on their protected characteristics, including religion, race, gender, ethnicity, and nationality.

This Arabic corpus for hate speech classification was evaluated extensively using supervised deep learning algorithms combined with various text vectorization methods (Alsafari, Sadaoui, and Mouhoub 2020c; 2020a; 2020b). Therefore, we consider this corpus reliable and use it for the initial ESSL phase. For the experiments, we divide this corpus into 70% training data (3,738 texts) and 30% testing data (1,602 texts). We use the stratified splitting method so that both classes are well represented in the two subsets. In this paper, we call the training corpus the seed dataset. Next, using several prepositional Arabic keywords, we extract 5 million tweets randomly through the Twitter API. We then pre-process this corpus by removing (a) short tweets (less than three words), (b) redundant tweets, and (c) similar tweets that exceed a similarity threshold of 80% (using the Jaccard metric) to increase the dataset's lexical divergence.

One million reliable tweets remain in the dataset after the cleaning process. Next, we normalize the corpus using the same method employed for the seed dataset: we normalize numbers (by replacing them with "@number"), elongated words (by eliminating the repetition of three or more characters), hashtags (by deleting underscores and "#" symbol), and the three Arabic letters alef, alef maqsoura, and ta marbouta. Lastly, we delete non-Arabic characters, diacritics, punctuation, emojis, users' mentions, and stop words. Most tweets in this unlabeled corpus are short texts with less than 60 words and 300 characters.

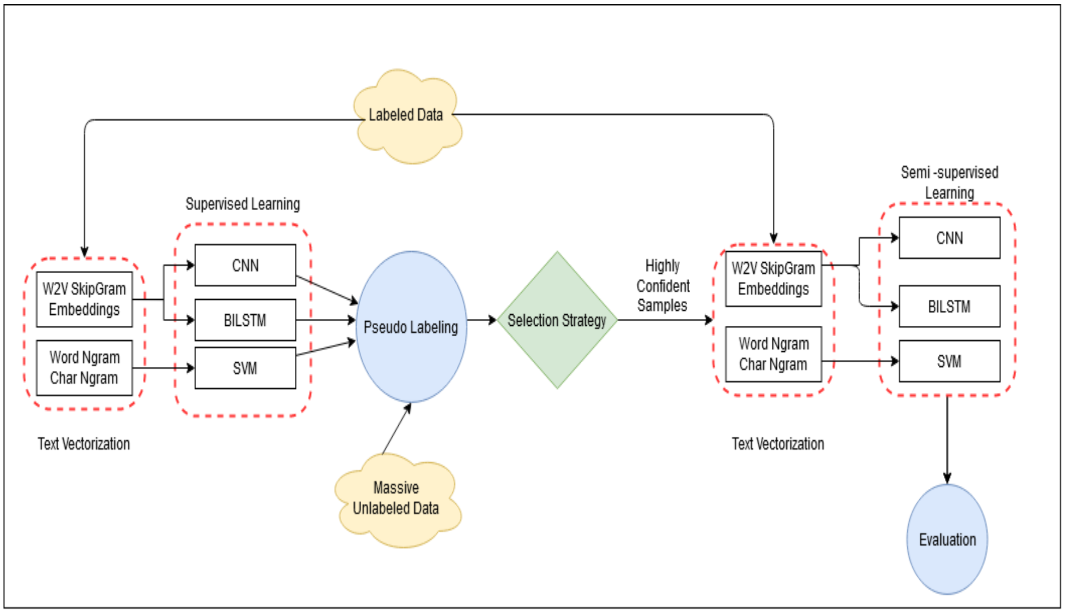


Figure 1: An Ensemble-based Semi-Supervised Learning of Hate Speech

Ensemble-based Semi-Supervised Learning

The present paper aims to increase the performance of supervised hate speech classification models. To this end, we propose an ensemble-based semi-supervised learning (ESSL) approach comprising four main stages. First, we train several baseline classifiers using the seed training corpus. Second, we employ the learned classifiers to label the unlabeled Twitter corpus. Third, we select highly confident pseudo-labeled data to re-train the base classifiers. In this phase, we experiment with different data selection strategies and several confidence thresholds. To improve their accuracy, we re-train the classifiers using both the seed and most trusted pseudo-labeled data. After each training, we evaluate all the learned classifiers with the same testing dataset for a fair comparison. Below, we describe the ESSL phases in more detail. Figure 1 illustrates the pipeline of our ensemble-based semi-supervised learning for hate speech detection.

A. Development of Diverse Baseline Classifiers

Using the seed dataset of 3,738 instances, we first train three heterogeneous classifiers based on standard and deep learning algorithms combined with different text vectorization techniques. More precisely, we train a support vector machine (SVM) merged with Word and Char Ngram, a convolutional neural network (CNN), and bidirectional long short-term memory (BiLSTM); the latter two are merged with Word2VeV Skip-gram word embeddings.

For the SVM model, the instances are first vectorized using 1-3 Word Ngram and 2-5 Char Ngram and then used to train the SVM algorithm using the grid search optimization method. For both the CNN and BiLSTM algorithms, the instances are vectorized through an embedding layer using pre-trained word embeddings. This layer is followed by a dropout layer with a rate of 0.2 for regularization. The next layer is either (a) a one-dimensional convolution in the CNN model that creates a feature map of the whole input or (b) a bidirectional LSTM in the BiLSTM model that processes the data sequentially, word by word. Consequently, the three models extract various features and create multiple views of every example. In this phase, we evaluate the accuracy of all the classifiers using the same testing dataset of 1,602 tweets.

B. Ensemble-based Pseudo-labels

At this stage, we employ the three classifiers to label the massive unlabeled dataset of 1 million tweets. To predict the labels, each classifier converts the instances into its feature map based on Ngram and Word embeddings. This heterogeneous representation of textual data helps to create a robust multi-view of the instances. At the end of this phase, each instance will have three pseudo-labels assigned by the baseline classifiers.

C. Selection of Confident Pseudo-abels

In semi-supervised learning, pseudo-labels are utilized to optimize supervised models. Nevertheless, for safe semi-supervised learning, it is of crucial importance to choose only trustworthy examples and to avoid using erroneously predicted examples that may mislead the learning process (Li and Liang 2019). Thus, we experiment with several data selection strategies by varying the prediction confidence level, including data balancing, majority voting, and average voting.

With the data balancing strategy, we choose the most confident pseudo-labeled examples by the three classifiers and by balancing the class distribution in the seed dataset, with a ratio of Clean to Hate equal to 1. In the majority voting strategy, we select the instances with the same label obtained by at least two classifiers with a confidence level above a threshold. To select only highly confident examples, we experiment with two thresholds: 0.999 and 0.99. The last strategy is average voting, where we first compute the average of the three probability scores for each instance and then select the instance with the average score above the confidence threshold.

D. Re-training of Classifiers

With the hope of increasing the baseline classifiers' predictive accuracy, we re-train all of them using the expanded training dataset: the confident pseudo-labeled dataset together with the seed dataset. However, first, we shuffle the whole new training dataset. For a fair performance comparison, we assess the newly trained models using the same testing dataset.

Experiments

We conduct several experiments to train the supervised SVM, CNN, and BiLSTM classifiers using the seed training dataset and the semi-supervised classifiers with the augmented datasets obtained through the five selection strategies. We train all the CNN and BiLSTM classifiers on a P100 Cloud GPU with a batch size of 32 and 50 epochs with early stopping criterion based on the validation loss.

Table 1 presents the pseudo-labeled datasets produced by the selection strategies as well as the new augmented datasets that we utilize to re-train all the classifiers. In this table, we rank the five strategies according to the size of the produced datasets, from smallest (4,974) to largest (66,374). From the one million tweets, data balancing produces the smallest confident dataset and majority voting the largest confident dataset.

Strategy	**Confidence Level**	**Pseudo Labels**	**Training Data**
Seed Only	NA	NA	3738
Data Balancing	0.999	1134	4974
Average Voting	0.999	6064	9886
Majority Voting	0.999	15405	19245
Average Voting	0.99	27233	31073
Majority Voting	0.99	62534	66374

Table 1: Training Data for each Selection Strategy and Confidence Level

Tables 2, 3 and 4 show the predictive accuracy of the 18 trained models evaluated on the same testing dataset. In our experiments, decreasing the pseudo-label selection strategy threshold from 0.999 to 0.99 for both majority and average voting does not harm the performance. On the contrary, the decrease improves the performance of all three classifiers. Thus, it is possible that reducing the threshold of the selection strategy has a limited effect when the confidence level is already relatively high.

Strategy	**SVM+Word/Char Ngram**		
	Precision	Recall	F-Macro
Seed Only (supervised)	85.79	87.32	86.46
Data Balancing CL=0.999	85.42	86.98	86.10
Average Voting CL=0.999	85.08	87.99	86.24
Majority Voting CL=0.999	84.79	88.33	86.14
Average Voting CL=0.99	84.32	87.48	85.55
Majority Voting CL=0.99	85.87	87.14	**86.44**

Table 2: Selection Strategies and Confidence Levels (CL) for SVM

Strategy	**CNN+Word2V SkipGram**		
	Precision	Recall	F-Macro
Seed Only	87.61	89.02	88.24
Data Balancing CL=0.999	87.53	89.16	88.32
Average Voting CL=0.999	88.47	90.68	89.41
Majority Voting CL=0.999	89.01	90.04	89.48
Average Voting CL=0.99	8967	90.26	89.95
Majority Voting CL=0.99	89.78	90.67	**90.20**

Table 3: Selection Strategies and Confidence Levels for CNN

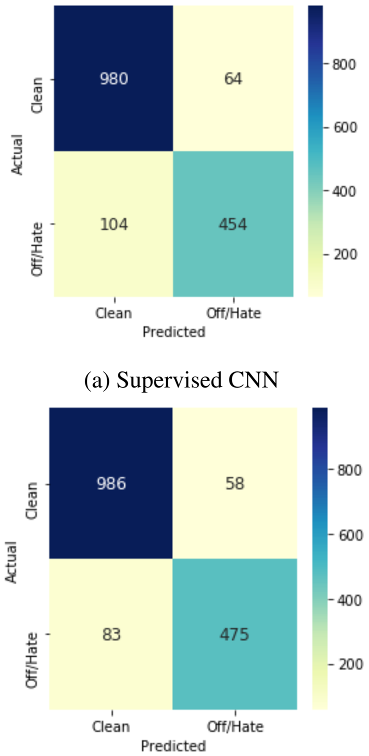
Discussion

For the initial training on the seed dataset, the three classifiers achieved high performance, and CNN slightly outperformed SVM and BiLSTM. The re-training performances for SVM suggest that standard learning algorithms can perform well—or even better—with far fewer training examples. After re-training the SVM classifier with the five weakly supervised datasets, the majority voting strategy with the confidence threshold of 0.99 provided the highest outcome, followed by average voting with a 0.999 confidence level.

Regarding the re-trained CNN classifiers, majority voting produced the highest accuracy of the three methods for both confidence thresholds; its accuracy was highest with the confidence level 0.99. The best model, trained with 62,534 pseudo-labeled samples, classified test samples as Clean or Hate with a precision of 89.67%, a recall of 90.675%, and an overall accuracy of 96.0%. To further verify and analyze the performance of this model, Figure 2 illustrates the classification confusion matrix compared with the confusion matrix of the supervised CNN model. As we can observe, the semi-supervised classification model correctly classified 21

Strategy	**BiLSTM+Word2V SkipGram**		
	Precision	Recall	F-Macro
Seed Only	86.45	88.33	87.25
Data Balancing CL=0.999	86.80	89.39	87.87
Average Voting CL=0.999	87.59	90.47	88.76
Majority Voting CL=0.999	88.62	90.76	89.30
Average Voting CL=0.99	89.55	90.48	**89.98**
Majority Voting CL=0.99	89.12	89.59	89.34

Table 4: Selection Strategies and Confidence Levels for BiLSTM



(a) Supervised CNN

(b) Semi-supervised CNN

Figure 2: Confusion Matrix for Supervised and best Semi-supervised CNN

additional hate samples and six clean samples, which is considered encouraging given the relatively small testing size. For the re-learned BiLSTM model, the majority and average voting approaches performed similarly with the confidence level of 0.99, with average voting slightly outperforming majority voting by 0.64%. Like CNN, semi-supervised BiLSTM proved effective by increasing the F-macro to 89.98%, with more than 2% improvement over the supervised classification model.

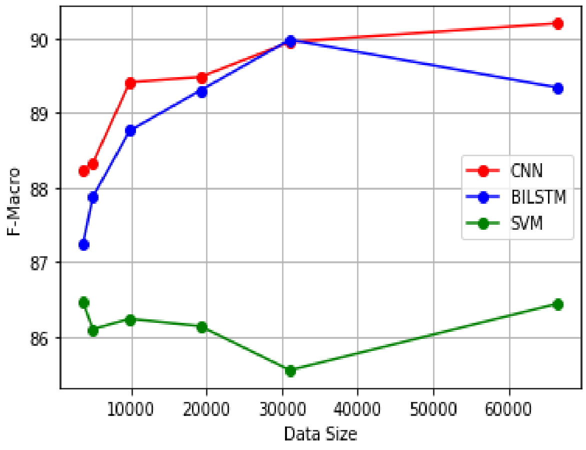


Figure 3: Plot of F-macro vs. Training Data Size for CNN, BiLSTM and SVM

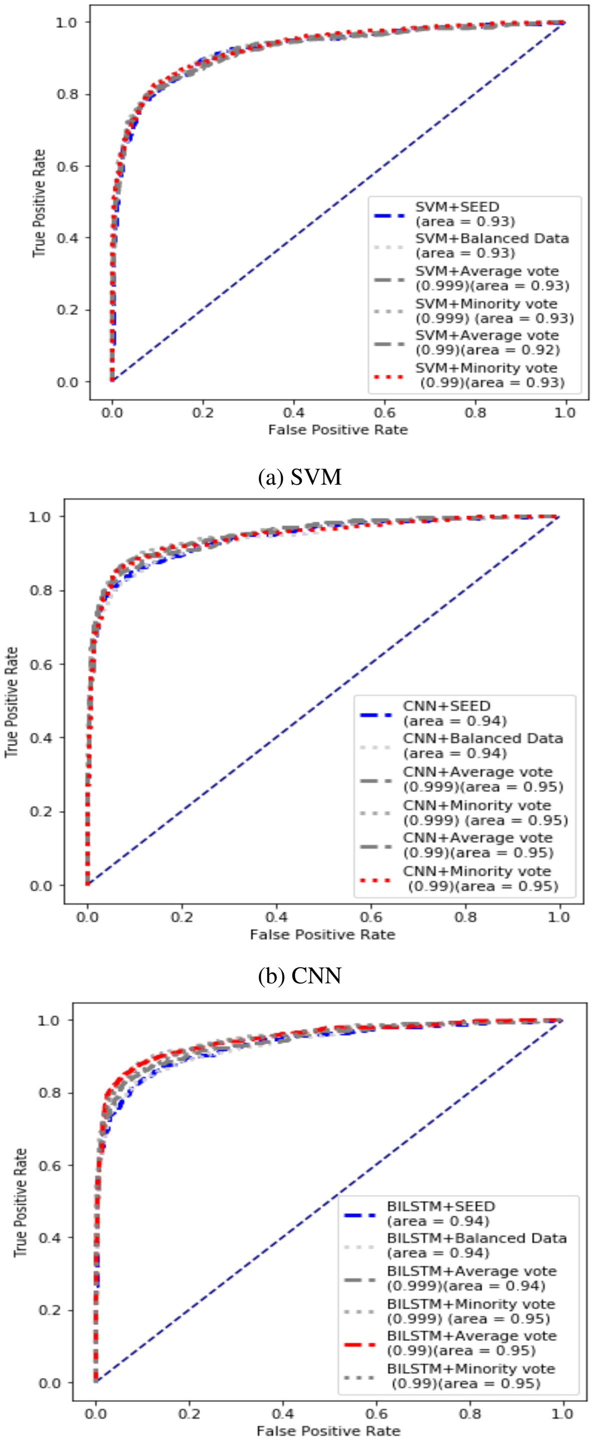
We also compare the models and strategies in terms of the ROC curve and the area under the curve (AUC), which measures the degree of separability (Figure 4). As we can see, the AUC value increased from 0.94 in supervised CNN and BiLSTM to 0.95 in semi-supervised models when using the average and majority voting strategies with the 0.99 confidence level. Overall, the F1 score of both CNN and BiLSTM increases when the training dataset increases (Figure 3). In conclusion, CNN is the best-performing model across all the classifiers, but there is no clear winner among the data selection strategies. Generally, the confidence threshold of 0.99 produced the best outcomes.

Conclusion

The present study fills a gap in hate speech detection since the use of semi-supervised learning is very minimal in this field. Our study introduces an ensemble-based semi-supervised learning approach to improve supervised hate speech classifiers' accuracy, benefiting from the abundant content available on social media platforms. Experimental results show that our ESSL approach improved the performance of deep neural-network models in terms of recall, precision and f-measure.

References

Alsafari, S.; Sadaoui, S.; and Mouhoub, M. 2020a. Deep learning ensembles for hate speech detection. In *2020 IEEE 32nd International Conference on Tools with Artificial Intel-*



(a) SVM

(b) CNN

(c) BiLSTM

Figure 4: ROC curve for Supervised and Semi-supervised Models

ligence (ICTAI), 526–531. 32th International Conference on Tools with Artificial Intelligence, ICTAI.

Alsafari, S.; Sadaoui, S.; and Mouhoub, M. 2020b. Effect of Word Embedding Models on Hate and Offensive Speech Detection. 1–9. ArXiv, CC BY 4.0.

Alsafari, S.; Sadaoui, S.; and Mouhoub, M. 2020c. Hate and Offensive Speech Detection on Arabic Social Media. *Online Social Networks and Media, Elsevier* 19:100096.

Elshaar, S., and Sadaoui, S. 2020. Semi-supervised classification of fraud data in commercial auctions. *Applied Artificial Intelligence, 34(1), Taylor & Francis* 47–63.

Kahn, J.; Lee, A.; and Hannun, A. 2020. Self-training for end-to-end speech recognition. In *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 7084–7088.

Li, Y.-F., and Liang, D.-M. 2019. Safe Semi-Supervised Learning: A Brief Introduction. *Frontier Computing Science, Springer-Verlag* 13(4):669–676.

Li, J.; Zhu, Q.; Wu, Q.; and Cheng, D. 2020. An effective framework based on local cores for self-labeled semi-supervised classification. *Knowledge-Based Systems* 197:105804.

Mubarak, H.; Rashed, A.; Darwish, K.; Samih, Y.; and Abdelali, A. 2020. Arabic Offensive Language on Twitter: Analysis and Experiments.

Nartey, O. T.; Yang, G.; Wu, J.; and Asare, S. K. 2020. Semi-supervised learning for fine-grained classification with self-training. *IEEE Access* 8:2109–2121.

Rosenthal, S.; Atanasova, P.; Karadzhov, G.; Zampieri, M.; and Nakov, P. 2020. A large-scale semi-supervised dataset for offensive language identification.

van Engelen, J. E., and Hoos, H. 2019. A survey on semi-supervised learning. *Machine Learning, Springer* 109:373–440.

Xu, Q.; Baevski, A.; Likhomanenko, T.; Tomasello, P.; Conneau, A.; Collobert, R.; Synnaeve, G.; and Auli, M. 2020. Self-training and pre-training are complementary for speech recognition.

Zampieri, M.; Malmasi, S.; Nakov, P.; Rosenthal, S.; Farra, N.; and Kumar, R. 2019. 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)*, 1415–1420.

Zampieri, M. e. a. 2020. SemEval-2020 Task 12: Multilingual Offensive Language Identification in Social Media (OffensEval 2020). In *Proceedings of SemEval*, 1425–1447.

Zhu, X., and Goldberg, A. 2020. *Introduction to Semi-Supervised Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning, edts: R. Brachman, J. Technion, F. Rossi and P. Stone.