# Hate Speech and Offensive Content Identification with Graph Convolutional Networks

Necva Bölücü[1], Pelin Canbay[2]

[1]Department of Computer Engineering, Hacettepe University, Ankara, Turkey
[2]Department of Computer Engineering, Sutcu Imam University, Kahramanmaras, Turkey

**Abstract**
Social media is a widespread platform and has a huge impact on society. There is a massive amount of data that plays an important role in expressing ideas, thoughts, emotions, etc. Identifying hate speech and offensive content on social media has gained attention recently. This is also the goal of the Hate Speech and Offensive Content Identification in Indo-European Languages (HASOC) 2021 Challenge in both English and Hindi languages. In this paper, we describe the system based on Graph Convolutional Networks (GCN) submitted by our team *HUNLP* for Subtask 1A and 1B. Our system has achieved a Macro F1-score of 82.15% for English Subtask 1A and ranked $2^{nd}$ in the leader-board. Moreover, our model has achieved 71.94% and 78.95% for Hindi and Marathi Subtask 1A on the official test set, respectively. Also, we have achieved Macro F1-score of 62.96% for English Subtask 1B.

**Keywords**
Social Media, Hate Speech, Graph Convolutional Network

## 1. Introduction

Recently, social media platforms such as Facebook, Twitter, and Instagram have gained attention, and users are creating various ways to express their opinions and thoughts. The use of social media has led to a huge volume of data with hateful and offensive content. Recent growing interest in Natural Language Processing (NLP) for identifying abusive and offensive content such as identification of abusive content [1, 2, 3], cyberbullying [4, 5, 6], hate speech [7, 8, 9], and offensive content [10, 11], have been observed.

The Hate Speech and Offensive Content Identification in Indo-European Languages (HASOC) [12] proposed identification of hate speech and offensive content task focusing on Indo-European languages in English and Hindi. The aim is to develop models to for identifying hate and offensive content on social media.

In this paper, we as HUNLP team have taken up the task and proposed a deep learning model based on Graph Convolutional Network (GCN) to identify hate and offensive content collected from Twitter by the HASOC data [12]. Previously, deep learning models such as LSTM [13], CNN [14], and pretrained models BERT [15], DistilBERT [16] have been applied for this task. The disadvantage of these models is ignoring word co-occurence in a corpus which carries non-

*FIRE 2021: Forum for Information Retrieval Evaluation, December 13-17, 2021, India*
necva@cs.hacettepe.edu.tr (N. Bölücü); pelincanbay@ksu.edu.tr (P. Canbay)
0000-0001-8121-3048 (N. Bölücü); 0000-0002-8067-3365 (P. Canbay)
© 2021 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
CEUR Workshop Proceedings (CEUR-WS.org)
CEUR-WS.org/Vol-3159/T1-4.pdf

**Table 1**
Statistics of HASOC 2021 Subtask 1 train dataset

| | | Subtask 1A | | Subtask 1B | | | |
|---|---|---|---|---|---|---|---|
| Language | Total # of Instances | NOT | HOF | HATE | OFFN | PRFN | NONE |
| English | 3843 | 1342 | 2501 | 683 | 622 | 1196 | 1342 |
| Hindi | 4594 | 3161 | 1433 | 566 | 654 | 213 | 3161 |
| Marathi | 1874 | 1205 | 669 | - | - | - | |

consecutive and long-distance semantics. To alleviate the disadvantage, GCN is proposed that contains rich relational structure and preserve global structure information in a graph. [17, 14]

The rest of this paper is organized as follows. Section 2 describes the task with the data on which the task was performed. Section 3 presents our method with preprocessing, and Section 4 presents the gives with details of our experimental setup. Finally, Section 5 summarizes our work.

## 2. Data

In this section, we briefly describe the tasks with the data proposed by the task organizers to train the model for the hate speech identification task.

The given dataset used on HASOC[1] in 2 languages, namely English and Hindi, consists of two Subtasks with a separate dataset for both Subtasks.

- **Subtask 1 [18, 19]:** is a classification problem consisting of two downstream tasks: Subtask 1A is a binary classification task to indicate whether the tweet is Hate and Offensive (HOF) or Non Hate-Offensive (NOT), and Subtask 1B is a three-classes classification task to classify tweets into three classes: HATE, Offensive (OFFN) or Profane (PRFN).
- **Subtask 2 [20]:** is the identification of conversational hate-speech in code-mixed languages.

Since we have dealt with Subtask 1, we give the details of the dataset for this subtask. The train dataset is provided in three different files for English, Hindi, and Marathi. The English and Hindi dataset files contain the fields _id, text, task_1 and task_2, where task_1 is the label of tweet post for Subtask 1A and task_2 is the label of tweet post for Subtask 1B. The Marathi dataset contains only the text_id, text, and task_1 fields because Marathi is not part of Subtask 1B. The training data statistics for Subtask 1 are presented in Table 1.

## 3. Methodology

The details of the preprocessing and the proposed model for Subtask 1 are given in the subsections.

[1]https://hasocfire.github.io/hasoc/2021/dataset.html Last visited: 14-10-2021.

**Table 2**
Output of ekphrasis library as pre-processing of tweets

| Language | State | Tweet |
|---|---|---|
| English | Original<br>Preprocessed | Oh they love this lol https://t.co/4kCudKSAk5k<br>oh they love this lol <url> |
| Hindi | Original<br>Preprocessed | @AskAnshul आसमानी किताब के नाजायज औलाद है।कूल<br><user> आसमानी किताब के नाजायज औलाद है।कूल |
| Marathi | Original<br>Preprocessed | @मराठ्यांनो कळालं का आता कोण तुमचा विचार करतो ते? ? ? ◌ूल<br>मराठ्यांनो कळालं का आता कोण तुमचा विचार करतो? <repeated> |

### 3.1. Preprocessing

Since the dataset we use consists of tweets in English and Hindi languages, we need to normalize the tweets before converting them into word embeddings.

Since the provided corpus is collected from Twitter, the tweets contain unstructured information like abbreviations, Twitter handles, punctuation marks, special characters, and more. Ekphrasis[2] library [21] is a tool designed to normalize text from social networks. It improves text through tokenization, normalization, segmentation, and spell correction by using word statistics extracted from a 2 corpus (English Wikipedia, Twitter - 330 million English tweets). The ekphrasis is used for preprocessing the corpus to improve the data quality and obtain the relevant information.

The preprocessing steps included in ekphrasis are:

- **Normalization:** To convert tweets into machine-understandable text, 8 normalizations are applied to the data: Normalizations of date, time, email, URL, currency, number, phone number, and username.
- **Annotations for emotions and emotion-causing features:** Social media users tend to express their emotions by using different styles. The normalization step includes normalization of hashtag, capitalization (all caps), elongated words, repeated characters, emphasis (included in asterisks), and censored words (censored abusive word).
- **Contractions unpacking:** Due to the character limits on Twitter, users tend to shorten text. Unpacking contractions is important to normalize the tweets (can't → can not).

The original and the preprocessed tweets by the ekphrasis library are given in Table 2.

### 3.2. Model Architecture

Graph Neural Networks are proposed as a paradigm-shifting method for solving NLP [22, 23] and Computer Vision [24, 25] tasks. Graph Convolutional Network (GCN) is a version of Graph Neural Networks that includes an additional convolutional layer.

The text classification task using GCN is the first study proposed by Yao et al. [14] in which a document on a graph is represented by GCN and the embedding vector of nodes is induced

[2]https://github.com/cbaziotis/ekphrasis. Last visited: 14-10-2021.

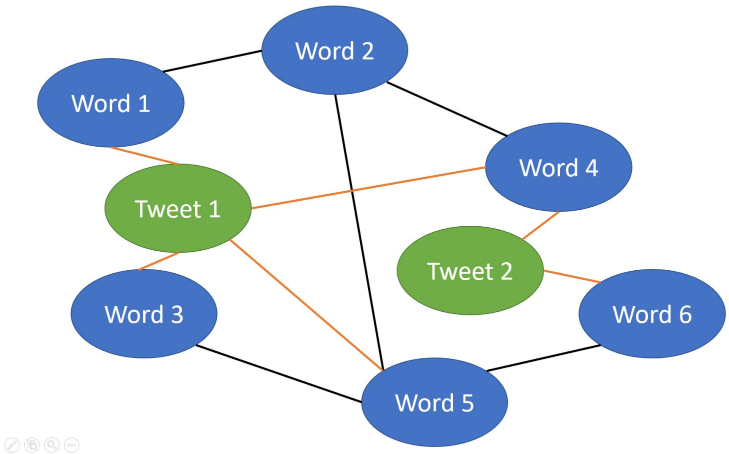


**Figure 1:** Graph structure

based on the properties of their neighborhoods. We adopt the study of Yao et al. [14] for the shared task. To convert the data into graph format, we follow the method of Yao et al. [14]. The graph $G = (N, E, W)$, where $N$ is the set of nodes, $E$ is the set of edges and $W : E \rightarrow R(R is the set of reals)$ is the function that assigns a weight each edge of the graph $G$. The details of the graph $G$ is given below:

- **Nodes (N):** Text GCN build a graph with word and tweet nodes. The number of nodes is a combination of word nodes (the number of unique words (vocabulary size)) and tweet nodes (number of tweets in the train file), defined as $|V|$
- **Edges (E):** To create edges between words, a sliding window is used. The intuition behind the sliding window corresponds to the Convolutional Neural Network filter. Each window acts as a a convolution filter of size $(1, n)$.
- **Weights (W):** $A$ is an adjacency matrix of the graph $G$ and its degree matrix is $D$, where $D_{ii} = \sum_j A_{ij}$. We use term frequency-inverse document frequency (TF-IDF) and point-wise mutual information (PMI) to form edges between word and tweet nodes and two word nodes, respectively. While PMI maps the word co-occurrence information, TF-IDF metric is statistical measure that evaluates how relevant a word is to a tweet in a collection of tweets.

The graph structure representation can be found in Figure 1.

The output of a one-layer GCN layer is computed as follows:

$$L^{(1)} = \rho(\tilde{A} \times W_0) \tag{1}$$

where $\rho$ is an activation function used in the model, $\tilde{A}$ is the normalized symmetric adjacency matrix, and $W_0$ is a weight matrix.

In the proposed model, we apply a simple two-layer GCN to the graph and feed the output of the node of the second layer into softmax classifier:

$$Z = softmax(\tilde{A}ReLU(\tilde{A} \times W_0)W_1) \tag{2}$$

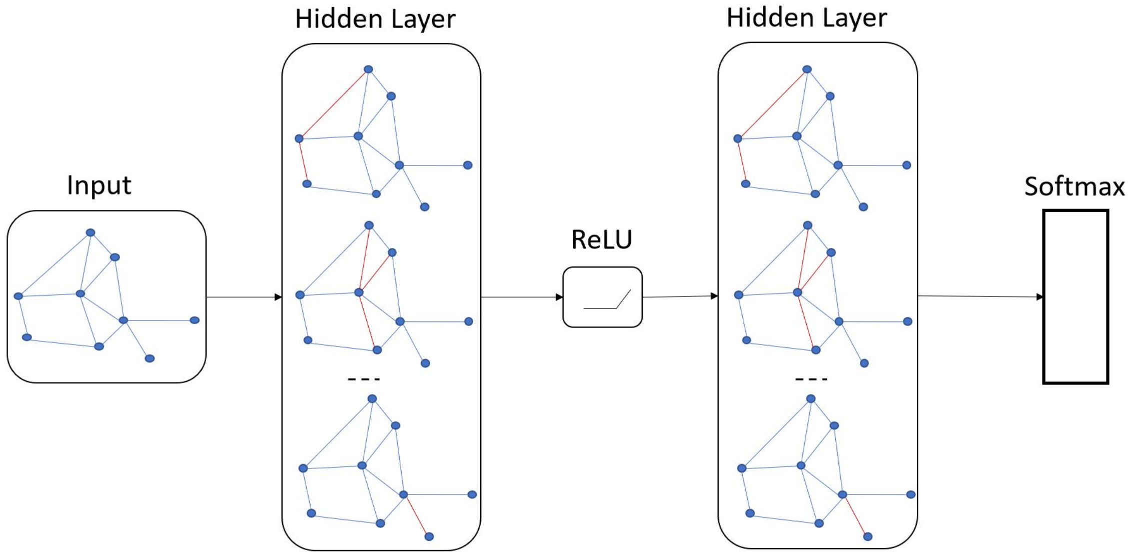


**Figure 2:** The general architecture of the proposed model

The Loss is calculated by using the cross-entropy for the task. The architecture of the proposed model is given in Figure 2.

## 4. Experiments & Results

In this section, we present the experimental settings and the obtained results on the test dataset in all languages for Subtask 1A and in English for Subtask 1B.

**Table 3**
Results on Test Dataset

| Task | Language | Macro F1 | Rank Obtained | 1st Ranked Team / Test Macro F1 |
|---|---|---|---|---|
| Subtask 1A | English | 0.8215 | 2 | NLP-CIC / 0.8305 |
| | Hindi | 0.7194 | 30 | t1 / 0.7825 |
| | Marathi | 0.7895 | 20 | WLV-RIT / 0.9144 |
| Subtask 1B | English | 0.6296 | 9 | NLP-CIC / 0.6657 |

**Settings** We split the train dataset into 80% train and 20% evaluation data to find the optimum hyperparameters. The model is built using Adam optimization [26]. The model was trained with parameters epochs = 200, learning rate = 0.02, dropout rate = 0.1, $L_2$ loss weight = 0 and consecutive epoch = 50. We used BERT [27], RoBERTa [28] and GloVe [29] word embeddings. Since the GloVe embeddings were trained specifically for Twitter (GloVe Twitter [3]), we chose to

[3] https://nlp.stanford.edu/projects/glove/ Last visited: 14-10-2021.

use the GloVe embeddings in the model for English. Since we couldn't find word embeddings trained for Twitter for Hindi and Marathi, we used multilingual BERT and RoBERTa for Hindi and Marathi and got the best results with BERT ( BERT multilingual base model (cased)[4]).

**Results** The best models obtained from the evaluation data were submitted by HASOC-2021 organizers in the competition for final evaluation. Table 3 shows the macro F1 score obtained by our best model with the names of $1^{st}$ ranker teams with their F1 macro scores for Subtask 1. The detailed results are also given in Table 4.

**Table 4**
Detailed Results on Test Dataset

| Task | Language | Macro F1 | Macro Precision | Macro Recall | Accuracy |
|---|---|---|---|---|---|
| Subtask 1A | English | 0.8215 | 0.8844 | 0.7669 | 79.24% |
| | Hindi | 0.7194 | 0.7258 | 0.7147 | 75.78% |
| | Marathi | 0.7895 | 0.7910 | 0.7881 | 81.44% |
| Subtask 1B | English | 0.6296 | 0.6305 | 0.6362 | 66.59% |

We assume that there are several reasons for the lower results for Hindi and Marathi. The first reason is word embeddings that are not trained on Twitter. It is clear that, the multilingual embeddings are not suitable for Twitter dataset in Hindi and Marathi. Another reason is the ekphrasis library that is proposed for English. For consistency, we have used it for Hindi and Marathi. However, the results show that it is not a good solution to normalize Hindi and Marathi dataset with ekphrasis.

# 5. Conclusion

In this paper, we presented the model of a graph convolutional network model on Subtask 1 of the shared task of hate speech and offensive content identification in English and Hindi languages. The results of the experimental study showed that using GCN model is very effective on hate speech and offensive content identification task. Compared to previous approaches, our model based on GCN is comparatively different for the shared task. We achieved rank 2, 30 and 20 for English, Hindi and Marathi in Subtask 1A and rank 9 for English in Subtask 1B respectively. In future work, we will further extend the experiments by combining datasets for the same Subtask to perform multilingual experiments.

# References


[1] C. Nobata, J. Tetreault, A. Thomas, Y. Mehdad, Y. Chang, Abusive language detection in online user content, in: Proceedings of the 25th international conference on world wide web, 2016, pp. 145–153.

[2] A. M. Founta, C. Djouvas, D. Chatzakou, I. Leontiadis, J. Blackburn, G. Stringhini, A. Vakali, M. Sirivianos, N. Kourtellis, Large scale crowdsourcing and characterization


[4]https://huggingface.co/bert-base-multilingual-cased. Last visited: 14-10-2021.

of twitter abusive behavior, in: Twelfth International AAAI Conference on Web and Social Media, 2018.

[3] P. Mishra, M. Del Tredici, H. Yannakoudakis, E. Shutova, Abusive language detection with graph convolutional networks, 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), 2019, pp. 2145–2150.

[4] C. Chelmis, D.-S. Zois, M. Yao, Mining patterns of cyberbullying on twitter, in: 2017 IEEE International Conference on Data Mining Workshops (ICDMW), IEEE, 2017, pp. 126–133.

[5] M. Yao, C. Chelmis, D.-S. Zois, Cyberbullying detection on instagram with optimal online feature selection, in: 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), IEEE, 2018, pp. 401–408.

[6] L. Cheng, J. Li, Y. N. Silva, D. L. Hall, H. Liu, Xbully: Cyberbullying detection within a multi-modal context, in: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, 2019, pp. 339–347.

[7] S. Malmasi, M. Zampieri, Detecting hate speech in social media, in: Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017, 2017, pp. 467–472.

[8] B. Mathew, A. Illendula, P. Saha, S. Sarkar, P. Goyal, A. Mukherjee, Temporal effects of unmoderated hate speech in gab, arXiv preprint arXiv:1909.10966 (2019).

[9] S. S. Aluru, B. Mathew, P. Saha, A. Mukherjee, Deep learning models for multilingual hate speech detection, arXiv preprint arXiv:2004.06465 (2020).

[10] M. Wiegand, M. Siegel, J. Ruppenhofer, Overview of the germeval 2018 shared task on the identification of offensive language (2018).

[11] M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, R. Kumar, 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), 2019, pp. 1415–1420.

[12] S. Modha, T. Mandl, G. K. Shahi, H. Madhu, S. Satapara, T. Ranasinghe, M. Zampieri, Overview of the HASOC Subtrack at FIRE 2021: Hate Speech and Offensive Content Identification in English and Indo-Aryan Languages and Conversational Hate Speech, in: FIRE 2021: Forum for Information Retrieval Evaluation, Virtual Event, 13th-17th December 2021, ACM, 2021.

[13] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Computation 9 (1997) 1735–1780.

[14] L. Yao, C. Mao, Y. Luo, Graph convolutional networks for text classification, in: Proceedings of the AAAI conference on artificial intelligence, volume 33, 2019, pp. 7370–7377.

[15] C. Sun, X. Qiu, Y. Xu, X. Huang, How to fine-tune bert for text classification?, in: China National Conference on Chinese Computational Linguistics, Springer, 2019, pp. 194–206.

[16] A. F. Adoma, N.-M. Henry, W. Chen, Comparative analyses of bert, roberta, distilbert, and xlnet for text-based emotion recognition, in: 2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), IEEE, 2020, pp. 117–121.

[17] H. Peng, J. Li, Y. He, Y. Liu, M. Bao, L. Wang, Y. Song, Q. Yang, Large-scale hierarchical text classification with recursively regularized deep graph-cnn, in: Proceedings of the

2018 world wide web conference, 2018, pp. 1063–1072.
[18] T. Mandl, S. Modha, G. K. Shahi, H. Madhu, S. Satapara, P. Majumder, J. Schäfer, T. Ranasinghe, M. Zampieri, D. Nandini, A. K. Jaiswal, Overview of the HASOC subtrack at FIRE 2021: Hate Speech and Offensive Content Identification in English and Indo-Aryan Languages, in: Working Notes of FIRE 2021 - Forum for Information Retrieval Evaluation, CEUR, 2021. URL: http://ceur-ws.org/.
[19] S. Gaikwad, T. Ranasinghe, M. Zampieri, C. M. Homan, Cross-lingual offensive language identification for low resource languages: The case of marathi, in: Proceedings of RANLP, 2021.
[20] S. Satapara, S. Modha, T. Mandl, H. Madhu, P. Majumder, Overview of the HASOC Subtrack at FIRE 2021: Conversational Hate Speech Detection in Code-mixed language, in: Working Notes of FIRE 2021 - Forum for Information Retrieval Evaluation, CEUR, 2021.
[21] C. Baziotis, N. Pelekis, C. Doulkeridis, Datastories at semeval-2017 task 4: Deep lstm with attention for message-level and topic-based sentiment analysis, in: Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), Association for Computational Linguistics, Vancouver, Canada, 2017, pp. 747–754.
[22] D. Wang, P. Liu, Y. Zheng, X. Qiu, X.-J. Huang, Heterogeneous graph neural networks for extractive document summarization, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020, pp. 6209–6219.
[23] W. Liao, B. Zeng, J. Liu, P. Wei, X. Cheng, W. Zhang, Multi-level graph neural network for text sentiment analysis, Computers & Electrical Engineering 92 (2021) 107096.
[24] Y. Shen, H. Li, S. Yi, D. Chen, X. Wang, Person re-identification with deep similarity-guided graph neural network, in: Proceedings of the European conference on computer vision (ECCV), 2018, pp. 486–504.
[25] W. Shi, R. Rajkumar, Point-gnn: Graph neural network for 3d object detection in a point cloud, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 1711–1719.
[26] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980 (2014).
[27] J. D. M.-W. C. Kenton, L. K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of NAACL-HLT, 2019, pp. 4171–4186.
[28] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, arXiv preprint arXiv:1907.11692 (2019).
[29] J. Pennington, R. Socher, C. D. Manning, Glove: Global vectors for word representation, in: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 2014, pp. 1532–1543.