TEXT CLASSIFICATION SURVEY

ABSTRACT

The purpose of this survey is to explore different techniques used in text classification, focusing specifically on the detection of offensive language and/or hate speech. To accomplish this, a variety of articles written between 2010 and 2023 concerning text classification methods were reviewed and summarized. The advantages and disadvantages of each approach were also examined. In addition, this paper addressed questions related to the HASOC (2019) Dataset.

INTRODUCTION

In today's digital age, where a vast volume of information is continuously generated and shared, the importance of classifying and organizing text has grown significantly. Text classification is the process of categorizing text documents into predefined classes or categories based on their content (Dhar et al., 2021). This technique not only helps in efficiently managing large volumes of data but also enables extracting valuable insights and making informed decisions. Text classification methods have a wide range of real-world applications, spanning from search engines delivering accurate search results to sentiment analysis offering valuable customer insights. Its impact is extensive, evident in tasks such as email filtering and spam detection, content categorization, personalized recommendations, and even legal document analysis.

Text classification can be executed through a rule-based (manual) approach or machine learning (automatic) methods (Antons et al., 2020). Manual text classification relies on human annotators to interpret the text's content and assign relevant categories. While this approach can yield satisfactory results, it is labor-intensive and costly. In contrast, automatic text classification employs machine learning, natural language processing (NLP), and other AI-driven techniques to automatically categorize text with greater speed, cost-effectiveness, and accuracy.

REVIEW OF GENERIC TEXT CLASSIFICATION METHOD

Researchers in the fields of machine learning, and text classifications have introduced multiple classification approaches. Some research papers from 2010 to 2023 0n supervised and unsupervised learning approaches for text classification methods were explored.

(Uguz ˇ , 2011) focused on enhancing text classification through the use of two-stage feature selection and feature extraction techniques. He proposed solving the problem of vast feature spaces in text classification by ranking the relevance of terms using information gain (IG). For dimension reduction, the top-ranked terms were subjected to the genetic algorithm (GA) and principal component analysis (PCA). The suggested model was tested using the k-nearest neighbor (KNN) and decision tree algorithms.

(Ghiassi et al., 2012) proposed a text classification method that does not require manual parameter settings or network architecture configuration called dynamic architecture for artificial neural networks (DAN2). Their algorithm performed exceptionally well compared to the leading algorithm as of then which were KNN and SVM.

(Song and Roth, 2014) proposed the use of Unsupervised Learning in text classification. They demonstrated the feasibility of assigning category labels to text documents without the need for labeled training data. Instead, understanding the labels can accurately perform the categorization process. They achieved precise classification without explicit supervision through the semantic similarity step and bootstrapping step.

(Shafiabady et al., 2016) proposed using Self Organizing Maps (SOM) and Correlation Coefficient (CorrCoef) for unsupervised clustering of data in the training phase. The resulting clusters were then used as labels to train Support Vector Machine (SVM) for text classification. They tested their proposed approach on standard text datasets, which demonstrated improved accuracy compared to training the SVM using expert knowledge.

(Kowsari et al., 2017) proposed a novel hierarchical document classification approach called HDLTex, which leverages multiple deep-learning techniques to achieve hierarchical classifications. Testing on a dataset of documents from the Web of Science revealed that combining RNN at the higher level with DNN or CNN at the lower level yielded higher accuracies compared to conventional methods like Naïve Bayes or SVM.

(Azam et al., 2018) evaluated data mining classifiers on a dataset with five categories using RapidMiner. Results showed that K-NN outperforms Naïve Bayes, with boosting and bagging increasing accuracy.

(Meng et al., 2020) developed a text classification method that used unlabeled data and label names to train models, without requiring many human-labeled documents. Their approach, LOTClass, achieved 90% accuracy on four benchmark datasets using pre-trained neural language models.

(David and Renjith, 2021) compared word embeddings in text classification using RNN and CNN. FastText was more accurate than GloVe but took longer to train. Word embeddings greatly affect text classification model performance. (Stammbach and Ash, 2021) proposed an unsupervised text classification method using Semantic Clustering and nearest neighbors. They utilized pre-trained language model vectors and weak learning signals from neighboring data points. Results showed their model outperformed other unsupervised techniques.

(Attieh and Tekli, 2023) proposed a new text classification framework that offers three lean classification approaches. They used a supervised weighting scheme based on the TF-ICF model which enhances text classification accuracy while requiring less computation time compared to other deep model alternatives.

CRITICAL DISCUSSION

Text classification methods have changed through the years, notably through the lenses of supervised and unsupervised learning approaches. Supervised learning approaches for text categorization are widely used and have shown amazing success. Traditional methods such as Support Vector Machine (SVM), Naive Bayes, K-Nearest Neighbors (KNN) and Decision Tree (DT) were popular at first (Aliwy and Ameer, 2017). These methods depended on well-crafted features, such as term frequency-inverse document frequency (TF-IDF), to represent text data (Dadgar et al., 2016). As the field progressed, researchers delved into using more advanced techniques and incorporating complex features. Word embeddings, which capture semantic links and contextual information, greatly improved classification tasks. The introduction of transfer learning has greatly impacted supervised learning in text classification, particularly with the introduction of pre-trained

language models such as BERT (Bidirectional Encoder Representations from Transformers)(Silva Barbon and Akabane, 2022). These models, which were trained on a large corpus of data, have shown remarkable performance in downstream tasks as they are able to capture both contextual and syntactic information. By fine-tuning these models for specific classification tasks, the amount of labeled data required is significantly reduced.

Text classification can be challenging when there is a shortage of labeled data. In such cases, unsupervised learning techniques, like clustering algorithms (such as K-means and hierarchical clustering), have proven to be vital. These techniques group documents based on their similarities, allowing them to be classified into clusters without predefined categories (Allahyari et al., 2017). However, the lack of label information can limit the interpretability and applicability of this approach. To address these limitations, Text classification has been enhanced by the emergence of topic modeling as a potent unsupervised learning method (Allahyari et al., 2017). Popular techniques like Latent Dirichlet Allocation (LDA) and Latent Semantic Analysis (LSA) have made it possible to identify latent topics within a corpus. This unsupervised approach aids in categorizing documents based on shared themes or topics by uncovering underlying themes (Rus et al., 2013).

Text classification has experienced significant advancements through the use of deep learning, which has become a dominant approach in recent years. Deep learning models, such as Convolutional Neural Networks (CNNs), and Recurrent Neural Networks (RNNs), automatically learn hierarchical representations from raw text data(Wang et al., 2021). This allows them to detect complex patterns and dependencies in the text, resulting in more precise and reliable classifications. The availability of large-scale labeled datasets, along with advancements in hardware and parallel computing, has enabled the training of deep learning models on massive amounts of data, which has further contributed to their success in text classification tasks. Although deep learning is the predominant method for text classification, it's important to recognize that other techniques, like traditional statistical method and unsupervised learning still hold value and can be helpful in certain situations. The best method to use depends on factors such as the size of the labeled dataset, available computational resources, interpretability needs, and the unique aspects of the text classification task.

REVIEW OF OFFENSIVE LANGUAGE DETECTION METHODS

Detecting offensive language can be challenging due to the informal and unstructured nature of its content. (Chen et al., 2012) presented the LSF architecture to detect offensive content and users. It analyzed pejoratives, obscenities, and writing styles for better predictions. LSF performs with high precision and processes sentences quickly, making it a useful social media tool .

(Kontostathis et al., 2013) proposed a two-pronged approach to detect cyberbullying. They used a Language Model of Bag-of-Words and Essential Dimensions of LSI (EDLSI) to analyze terms and identify bullying content. The 2013 murder of Drummer Lee Rigby in Woolwich, London, UK led to a study on the spread of online hate speech on Twitter.

(Burnap and Williams, 2015) developed a text classifier powered by machine learning that was trained on Twitter data to identify hateful responses based on ethnicity, religion, or race. The results were used to predict the spread of cyber hate on Twitter.

(Gambäck and Sikdar, 2017) proposed a system that used deep learning to classify Twitter hate speech into four categories. The system had a 78.3% F-score and used four different trained Convolutional Neural Network models, with the best model based on word2vec embeddings.

(Kiilu et al., 2018) found that a method using Naive Bayes classifier worked best in identifying hate speech on Twitter, achieving high precision, recall, and accuracy rates.

(Herwanto et al., 2019) proposed a hate speech classification model using CBOW and fastText. Pre-trained vectors from Wiki generally improved model performance.

(Dorris et al., 2020) proposed a defense system HateDefender that uses deep LSTM neural networks to detect hate speech and offensive language with high accuracy. It identifies harmful words and provides explanations for intervention in online incidents.

(Mazari et al., 2023) focused on multi-aspect hate speech detection using pre-trained Bidirectional Encoder Representations from Transformers (BERT) and Deep Learning (DL) models. Their approach classified text into multiple labels, including identity hate, threat, insult, obscene, toxic, and severely toxic.

CRITICAL DISCUSSON

Detecting hate and/or offensive language is a challenging task as it is becoming increasingly prevalent on various online platforms. One of the main difficulties in identifying hate speech is defining what exactly constitutes it. Legal and societal norms differ across countries and cultures, making it challenging to establish a universal definition. Some words or phrases may be harmless in one situation but offensive in another. The use of irony, sarcasm, or humor can make the detection process even more complicated. Moreover, hate speech can manifest in different languages and dialects, requiring multilingual and culturally sensitive detection models. To ensure effectiveness across various regions, training data must represent diverse linguistic and cultural contexts. However, Training data for hate speech identification might be skewed due to annotators' subjective judgments.

Hate speech evolves rapidly, adopting new terminologies and code words. Detection models must continuously adapt to identify emerging patterns of hate speech to remain effective over time. However, hate speech detection algorithms may produce false positives or false negatives. Striking the right balance between over-detection and under-detection is challenging. Overly strict filtering may suppress legitimate speech, while less stringent systems may fail to identify harmful content. Developing hate speech detection models raises ethical concerns, such as privacy and freedom of speech. It is a delicate task to balance protecting individuals from hate speech with respecting freedom of expression. The demand for explainable AI grows as hate speech detection methods get more advanced. To develop confidence and assure accountability, users and policymakers should understand how the models make their judgments. There has been a tremendous shift in the use of Traditional machine learning models towards deep learning in detecting hate and/or offensive language because of their ability to capture sequential patterns and contextual information. Pre-trained language models like BERT, GPT, RoBERTa, and Emsemble models have gained popularity in hate speech detection tasks as well.

DATASET CHARACTERISTICS OF HASOC

The Hate Speech and Offensive Content (HASOC) is a dataset in the field of hate speech and offensive language detection

1. The Dravidian Language Resource Center (DLRC) created and collected the HASOC dataset, which deals with Hate Speech and Offensive Content. The DLRC also organized a workshop in 2019 aimed at promoting research and development in hate speech detection with a focus on Hindi, German, and English. The dataset has been made publicly available for researchers to use. However, registration is required as it is password protected.

2. The HASOC Dataset has text data from Twitter and Facebook in English, Hindi, and German that focuses on hate speech and offensive language. The English dataset has 7005 data points, split into three subsets. Sub-task A identifies hate speech and offensive language in tweets. Sub-task B categorizes offensive posts into three groups: HATE, OFFN, and PRFN. Sub-task C classifies HOF posts from sub-task A into TIN and UNT.

3. The dataset was sampled from Twitter and partially from Facebook (Mandl et al., 2019). It is now used for research purposes.

4. HASOC dataset was produced to explore what other research work on offensive language identification had not indulged in, which is producing datasets in other languages aside from English. The HASOC dataset is available in Hindu and German.

5. HASOC Dataset was produced in 2019.

SUMMARY

This survey has covered various techniques for text classification, with a focus on detecting hate speech. Traditional machine learning methods, like Naive Bayes and SVM, have been commonly used and have shown decent results. However, deep learning techniques, such as CNN and RNN, have become more popular due to their ability to learn features from text automatically. Additionally, pre-processing techniques, like text cleaning and tokenization, are crucial for improving text classification performance. Despite these advancements, challenges still exist in detecting hate speech, including imbalanced data, evolving language, and understanding deep learning models. Overcoming these challenges will require further research and development to create effective methods for combating hate speech on online platforms.