

Attention based bidirectional GRU hybrid model for inappropriate content detection in Urdu Language



By

Ezzah Shoukat

Fall 2018-MS(IT)-18-000000295897

Supervisor

Dr Rabia Irfan

Department of Computing

A thesis submitted in partial fulfillment of the requirements for the degree of Masters
of Science in Information Technology (MS IT)

In

School of Electrical Engineering & Computer Science (SEECS) ,

National University of Sciences and Technology (NUST),

Islamabad, Pakistan.

(August 2022)

THESIS ACCEPTANCE CERTIFICATE

Certified that final copy of MS/MPhil thesis entitled "Attention based bidirectional GRU hybrid model for inappropriate content detection in Urdu language" written by EZZAH SHOUKAT, (Registration No 00000275897), of SEECs has been vetted by the undersigned, found complete in all respects as per NUST Statutes/Regulations, is free of plagiarism, errors and mistakes and is accepted as partial fulfillment for award of MS/M Phil degree. It is further certified that necessary amendments as pointed out by GEC members of the scholar have also been incorporated in the said thesis.

Signature: _____  _____

Name of Advisor: _____ Dr. Rabia Irfan _____

Date: _____ 10-Aug-2022 _____

HoD/Associate Dean: _____

Date: _____

Signature (Dean/Principal): _____

Date: _____

Approval

It is certified that the contents and form of the thesis entitled "Attention based bidirectional GRU hybrid model for inappropriate content detection in Urdu language" submitted by EZZAH SHOUKAT have been found satisfactory for the requirement of the degree

Advisor : Dr. Rabia Irfan

Signature:  _____

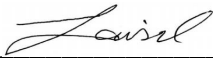
Date: 10-Aug-2022

Committee Member 1:Dr. Muhammad Ali Tahir

Signature:  _____

Date: 10-Aug-2022

Committee Member 2:Prof. Dr. Faisal Shafait

Signature:  _____

Date: 11-Aug-2022

Committee Member 3:Dr. Muneer Ahmad

Signature:  _____

Date: 11-Aug-2022

Dedication

This thesis is wholeheartedly dedicated to my beloved parents, who have been my source of strength and inspiration when I was at the verge of giving up. All their moral, emotional and financial support is the reason I am able to finish this work. They instilled in me the values that have made me who I am today.

To my elder siblings, who have always been there for me, and their words of advice helped me in solving the problems I faced while writing this study. They are my greatest support.


To my loving husband Muhammad Mughees, for his consistent support specially during the challenges of graduate school and life. And to my in-laws who encourage me to achieve my goals. I am appreciative of their unending support, love, and trust.

Last but not least, I want to dedicate this thesis to my wonderful friends Amna Noor and Ammarah Irum who have helped me in every way and have given me a lot of wonderful memories to hold dear. I would especially like to thank my friends Almas Shabbir and Faiza Qamar at NUST for their unwavering love and support.

Certificate of Originality

I hereby declare that this submission titled "Attention based bidirectional GRU hybrid model for inappropriate content detection in Urdu language" is my own work. To the best of my knowledge it contains no materials previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any degree or diploma at NUST SEecs or at any other educational institute, except where due acknowledgement has been made in the thesis. Any contribution made to the research by others, with whom I have worked at NUST SEecs or elsewhere, is explicitly acknowledged in the thesis. I also declare that the intellectual content of this thesis is the product of my own work, except for the assistance from others in the project's design and conception or in style, presentation and linguistics, which has been acknowledged. I also verified the originality of contents through plagiarism software.

Student Name: EZZAH SHOUKAT

Student Signature:  _____

Acknowledgments

I especially want to thank my honorable supervisors Dr. Rabia Irfan and Dr. Ali Tahir for their guidance and inspiration throughout the entire thesis process. They have been incredibly courteous and thoughtful towards me and my work. May Allah bestow upon them the finest in this life and the next.

I want to sincerely thank my committee members (Dr. Faisal Shafait, Dr. Muneer Ahmad) for their continuous support and guidance throughout the writing of my master's thesis.

Above all, to our God Allah (S.W.A), the Creator, the Sustainer of the Universe. Without His permission, absolutely nothing is possible. For He was the only one who blessed me, made opportunities for me, and paved the way for my achievements from the day I arrived at NUST until the day I left. I am thankful for His uncountable blessings.

Ezzah Shoukat

Contents

1	Introduction	1
1.1	Motivation	2
1.2	Granularity of Inappropriate Content	3
1.3	Automatic Detection of Inappropriate content	4
1.3.1	Inappropriate content in Urdu Language	5
1.4	Problem Statement	5
1.5	Research Contribution	6
1.6	Thesis Outline	7
2	Literature Review	8
2.1	Background	8
2.1.1	Language characteristics of Urdu Unicode Script and its challenges	9
2.1.2	Studies in Urdu Text Classification	11
2.2	Approaches to Inappropriate Content Detection	12
2.2.1	Machine Learning approaches	12
2.2.2	Deep Learning approaches	14
2.3	Comparative Analysis	17
3	Deep Neural Networks for Inappropriate Language Detection	20
3.1	Deep Learning Models	21

CONTENTS

3.1.1	Recurrent Neural Network (RNN):	22
3.1.2	Gated Recurrent Unit (GRU):	22
3.1.3	Long Short Term Memory (LSTM):	24
3.1.4	BiDirectional LSTM (BiLSTM):	25
3.1.5	Temporal Convolutional Network (TCN):	27
3.2	Word Embedding:	28
4	Design and Methodology	30
4.1	Dataset	30
4.1.1	Dataset collection	30
4.1.2	Dataset annotation and Statistics	31
4.2	Dataset Pre-Processing	34
4.3	Proposed Methodology	35
4.3.1	Bidirectional GRU	36
4.3.2	Attention	37
4.4	Experimental Setup	38
4.4.1	Sequence normalization	38
4.4.2	Layering of Proposed model	38
5	Results and Discussion	41
5.1	Evaluation Metrics	41
5.2	Experiments	42
5.3	Results comparison	44
5.4	Discussion on Proposed Model	46
6	Conclusion and Future Work	48
6.1	Synopsis	48
6.2	Challenges of the Research	49

CONTENTS

6.3	Limitations	50
6.4	Future Work	50
6.5	Applications	50

List of Abbreviations

Abbreviations

NLP	Natural Language Processing
ML	Machine Learning
DL	Deep Learning
NN	Neural Networks
SVM	Support Vector Machines
NB	Naive Bayes
K-NN	K-nearest Neighbours
CNN	Convolutional Neural Networks
LSTM	Long Short Term Memory
BiLSTM	Bidirectional Long Short Term Memory
GRU	Gated recurrent unit
BERT	Bidirectional Encoder Representations from Transformers

List of Figures

1.1	Classification basic workflow.	4
2.1	Decision Trees Example	13
2.2	A CNN and Bi-LSTM model approach.[24]	15
3.1	RNN Model	23
3.2	LSTM Model[17]	25
3.3	BiLSTM Model	26
3.4	Temporal Convolutional Network Architecture[19]	28
4.1	Sample of Dataset.	32
4.2	UrduInASmall Dataset Statistics Graph.	33
4.3	UrduInALarge Dataset Statistics Graph.	33
4.4	Attention based Bidirectional Gated Recurrent Unit BiGRU- A - Proposed model	35
5.1	Evaluation Metrics Comparison of both data sets with or without Word2vec	45
5.2	Accuracy Comparison of both data sets with or without Word2vec	46

List of Tables

2.1	Diatrics of Urdu Language.	9
2.2	Summary of research on Inappropriate language detection	19
4.1	Dataset categories.	31
4.2	Statistics of dataset.	32
4.3	Optimizing parameters used in both dataset.	40
5.1	Results Comparison of baseline Model with our proposed model on UrduInAsmall dataset without using Word2Vec layer.	43
5.2	Results Comparison of baseline Model with our proposed model on UrduInAsmall dataset with using Word2Vec layer.	43
5.3	Results Comparison of baseline Model with our proposed model on UrduInAlarge dataset without using Word2Vec layer.	43
5.4	Results Comparison of baseline Model with our proposed model on UrduInAlarge dataset with using Word2Vec layer.	44

Abstract

With the advancement in the scope of online discussion, the spread of toxic and inappropriate content on social networking sites has also increased. Several studies have been conducted in different languages. However, existing literature on inappropriate content detection lacks research in Urdu Unicode text language using deep learning techniques. Use of attention layer with deep learning model can help in handling the long-term dependencies and increase its efficiency. To explore the effect of attention layer, this study proposes an attention based Bidirectional GRU hybrid model for identification of Inappropriate content in Urdu Unicode text language. Four different baseline deep learning models LSTM, Bi-LSTM, GRU, and TCN are used to evaluate the performance of proposed model. The results of models are compared based on evaluation metrics, dataset size and impact of word embedding layer. The pre-trained Urdu word2vec embeddings are utilized for our case. Our proposed model BiGRU-A outperformed all other baseline models by yielding 84% accuracy without using pre-trained word2vec layer. From our experiments we have established that attention layer improves the efficiency of model and pre-trained word2vec embedding does not work well with inappropriate content dataset.

Keywords: *Deep Learning, Natural Language Processing, Text classification, Attention.*

CHAPTER 1

Introduction

This chapter explains the need of Inappropriate content detection originated, motivation for this research and what type of content can be referred as inappropriate. We will further discuss the automatic detection of inappropriate language specifically in Urdu. Derived problem statement and our research contributions will be discussed in detail. We will conclude with outlining the thesis in the end.

The exponential growth in social media users has evolved the communication technology and developed the Internet. This ease of access has made it possible for online social media platforms to play a big role in our everyday lives [43]. Social media networks like Twitter, Facebook and YouTube allows a wide diversity of people from all around the world. People belonging to different ages, culture, linguistics, ethnic and religious backgrounds have now access to these sites in their hands[16]. Usually, users find it more convenient to write and express their thoughts and reviews about products, movies, or articles in their local language rather than in English [36]. These social platforms are widely used around the globe for knowledge sharing, socializing, social media marketing, advertising communication and entertainment. At the same time, billions of people using social media platforms are prone to cyber-crimes such as bullying, threatening and scams. In addition, posting of controversial content without any check and balance can cause provocation, societal agitation, public rage, community opinion management and anarchy.

To address this matter, many social sites employ manual techniques where user reports the issue, and it is then manually reviewed by the customer support team.

But this method is highly dependent on the reviewer's speed, their ability to recognize the level of slang used, and multilingual content knowledge. Also, the manual processing takes a day or two at most and by then the intended damage is usually already done. Moreover, manual process has subjective and unclear boundaries of what words combination add up to be marked as offensive or not. This might lead to the misuse of manual process by silencing the minority groups and by suppressing the criticism raised against political parties, official policies, or religious beliefs. Thus, to counter ill use of social media, it is necessary to automatically detect, categorize, and clean the controversial content before it is posted online.

1.1 Motivation

A series of events popped up in recent years that has compelled the Government of Pakistan to take precautionary steps to avoid uncontrollable consequences [40]. These episodes consisted slandering of political parties and their leaders, famous media figures, hurting sentiments of religious minorities by bullying them on their beliefs, targeted harassment of women sharing their point of views, and snide remarks passing between the Indo-Pakistan natives due to the bitterness left after the independence war. Such actions give away the notions of nation's struggles with online hate speech dialogues and necessitates the immediate need for automated filtering system.

Manual identification of hate speech content is deemed inefficient due to the vast number of online users and the massive influx of internet content. Recent advancements have been made in the domain of Natural language processing (NLP) and most of the research is mainly conducted in resource rich languages like English. For all type of NLP tasks such as text translation, classification and sentiment analysis, Machine learning (ML) techniques have been the first choice of researchers in the recent years. Due to their impressive results and outstanding performances. Deep learning (DL) algorithms are now also being incorporated for the detection of inappropriate content from user's comments on social networking sites in different languages. Like Turkish and Arabic etc. Urdu is also resource scarce language with complex morphological structure, unique characters, and low

linguistic resources. It's a well-known fact that the hate-speech content varies with change in language. Thus, scarcity of language resources, small, labeled/unlabeled datasets are the reasons of limited research in this area [29].

1.2 Granularity of Inappropriate Content

Here, the term Inappropriate Content is used broadly to refer to any type of content that could be hurtful to the online user's community. Popular social networking sites such as Twitter, Facebook, or Instagram use words like Not Allowed Content or Inappropriate content to identify the malicious or violating content on their sites. Abusive language and profanities are a major part of the Inappropriate or violating content. But some popular social communicating sites don't mark it as violating content or rather as freedom of expression unless or until it poses real damage to a particular being or group of individuals. For example, suspected affiliation with terrorist organizations or other infringing media (audio, images, or videos). Due to the broad scope of inappropriate content, social networking platforms heavily rely on specific reports from users to remove content that violates their policies. Other website communities such as e-learning portals or educational university online forums have tougher rules about what constitutes as inappropriate language and they neither permit nor accept content from general social media sites.

In inappropriate vs appropriate content classification, inappropriate content will include all types of potential hurtful content like use of abuses, targeting and slang language. This type of content detection is mostly desired by national TV channels or online educational institution platforms. Inappropriate content is not only measured by the choice of words spoken but also by the context in which those words are used. For instance, it is acceptable to employ language that is exclusively directed at one gender to limit membership in a group that promotes health or well-being, such as breastfeeding support groups for women. By considering these factors, inappropriate content detection becomes more challenging and interesting.

1.3 Automatic Detection of Inappropriate content

With the development in data science domain, language understanding has advanced significantly. Many studies have been conducted using Machine and Deep learning algorithms to detect inappropriate content in resource rich languages. From basic ML models to simple recurrent neural networks and most recently advanced models with transformers are the approaches followed by researchers. Moreover, transfer learning is the new breakthrough in this field in which pre-trained models are fine-tuned to use for various other tasks. They not only decrease the development cycle time but also produce cutting-edge results.

In the basic classification workflow, we start by collecting data related to the domain of problem that needs to be solved. Then we inspect and thoroughly clean the data from any issues by pre-processing and fixing methods. In the next stage cleaned dataset is then loaded to adequate ML/DL framework. If issues were found during data loading, we return to the cleaning stage; otherwise, we begin to investigate the data using conventional statistical approaches. This cycle continues to assess and test performance measures of ML/DL models. In addition to building models, we also develop visualizations that aid in communicating and analyzing the results as well as generating insights about data. The outcomes are then released and made public with the community. The stages of the workflow for classification are shown in Figure 1.1. In supervised ML classification, there

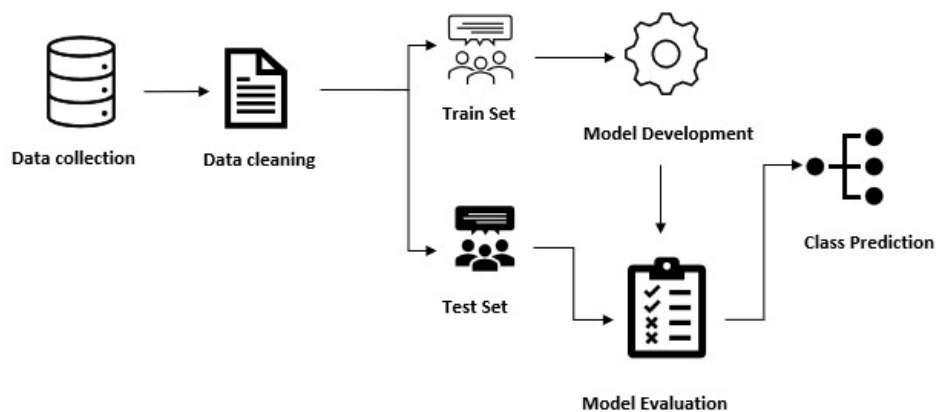


Figure 1.1: Classification basic workflow.

are two major steps. First is feature extraction and the second is classification. There are many techniques to extract features from data such as n-gram feature extraction [31]. Some other techniques like pattern matching [15] or lexicon based [10] approaches are also popular. Whereas in DL supervised classification, features are learned by the Neural networks and word vector word embedding models like Word2Vec are used for improved representation of text [50]. When compared to conventional ML models, the performance of Deep Learning (DL) models has significantly improved with a huge amount of data [37].

1.3.1 Inappropriate content in Urdu Language

The official language of Pakistan is English, but Urdu is recognized as the National language of the country. It is also widely spoken in many Indian provinces. It has not only unique writing script but also sophisticated lexicon structure. Because of its morphological composition, which begins on the right and moves to the left, Urdu is quite distinct from other languages. This is also why the Urdu script is not widely used hence a standard dataset or corpus is needed to carry out Urdu NLP tasks.

Identification of Inappropriate content in Urdu is just as significant as it is in any other language, since it helps non-Urdu speakers to grasp and understand the fundamental thoughts, feelings, and views of the writer behind text. On websites like Twitter, Facebook, and YouTube, many native Urdu speakers use the Urdu script to convey their feelings, ideas, and other sentiments. To comprehend the thoughts and emotions of native Urdu speakers, it is necessary to analyze text written in Urdu. Most of the existing literature studies in Urdu language are either focused on various other aspects of natural language processing such as sentiment analysis, news classification, gender identification etc. or only ML techniques and very few DL algorithms are explored for inappropriate content detection.

1.4 Problem Statement

Internet users can express, discuss, and share their opinions and ideas on a wide range of topics in a variety of online discussion forums like blogs, news portals, and

other social media platforms like Facebook, Twitter etc. It has frequently been noted on such forums that user conversations may easily derail and become improper, such as throwing insults or making nasty or disrespectful remarks to other people, certain communities, or groups. In a similar vein, it has been observed that certain search engines may send people inappropriate messages in return of a query. Thus, inappropriate content is gradually affecting the user experience and turning into a menace. Several native Urdu speakers communicate their sentiments, views, and thoughts on social media using the Urdu script. Hence, automatic filtering and identification of inappropriate content in Urdu language is equally important to help non-Urdu speakers in understanding the sentiment and to improve the quality of conversation.

1.5 Research Contribution

Although, in recent years many researchers have publicly shared their annotated Urdu corpus for future research workers. But, to find domain specific data is still a challenging task. Most of the annotated datasets are available either in Roman Urdu or its too small to apply advanced deep neural networks analyze their results. Furthermore, only ML baseline models are explored extensively on the available datasets. To address the gaps discussed till yet following are the key contributions of this research:

- A bigger inappropriate content corpus, which is obtained by combining two publicly available annotated dataset. First was available in Urdu Script. But second dataset is collected by converting from Roman Urdu to Urdu script.
- Bi-GRU with attention layer model for identification of inappropriate content is proposed.
- Result comparison of state-of-art DL models (LSTM, Bi-LSTM, GRU, TCN).
- Comparative Analysis of baseline models and proposed model with and without using pre-trained word embedding model word2vec for Urdu language.
- The impact of dataset size on the performance of DL models is studied.

1.6 Thesis Outline

The rest of the thesis is thoroughly categorized into chapters. We will present a complete literature review of the studies that have already addressed the issue of Inappropriate content detection, origin of Text classification and will briefly discuss ML/DL learning approaches utilized in this domain in Chapter 2. The baseline deep neural networks and word embedding utilized in this study discussed in Chapter 3 will further enhance the knowledge of their evolution, framework and applications. Chapter 4 illustrates the process of dataset collection, our proposed methodology structure and experimental setup. We will briefly describe the results obtained through our experimentation and evaluation metrics used for their evaluation in Chapter 5. In the last, Chapter 6 will conclude this thesis by summarizing the research conducted, challenges faced in this research, its limitations. We will also discuss future work recommendations and applications of automatic inappropriate content detection.

Literature Review

This chapter provides in-depth information on studies related to the identification of inappropriate language. It briefly recalls the origins and development of content identification. Additionally, it explores conventional and neural network approaches. Finally, we will provide a detailed comparative analysis of the work on offensive content detection in the Urdu language.

2.1 Background

Identifying inappropriate language on the internet has become one of the most prominent NLP applications [32]. Inappropriate speech detection in social media posts is not an easy task. A lot of people post comments in informal language on daily basis. Context of one sentence may differ from person to person as everybody has their own perspectives. Some words may be considered humorous by some and hateful by others [20]. It's difficult to differentiate this point.

In this domain, a lot of work has been done in English or many other languages. For Urdu language, Roman Urdu is most commonly used in social media post. It is Latin script of Urdu language in which Urdu words are written using English language alphabets. Since it is simpler to acquire data in Roman Urdu than in the Urdu usual Arabic script, the majority of studies focused on hate speech identification are on Roman Urdu. This makes it even more important to identify inappropriate speech in Roman Urdu and erase it in order to protect victims from online abuse.

Previous studies have explored ML baseline models and a few Neural Networks(NN) for the problem of offensive language detection in Urdu script. On large data sets, ML models struggle to perform well and usually learn just directed text features. When compared to standard ML models, the efficiency of DL models has significantly improved as the size of dataset increases [37].

2.1.1 Language characteristics of Urdu Unicode Script and its challenges

Pakistan’s official language, Urdu has more than 300 million speakers worldwide [5].The right-to-left writing system used for the Urdu is known as the Urdu alphabet. It is an adaptation of Perso-Arabic, a Persian script that itself is a descendant of Arabic. The Urdu language uses the calligraphic Nastaliq script to write its alphabet, which includes up to 58 characters and no unique letter cases. The basic characters of Urdu are 38 as written below:

ا, ب, پ, ت, ث, ج, چ, ح, خ, د, ڈ, ذ, ر, رڑ, ز, ژ, س, ش, ص, ض, ط, ظ, ع, غ, ف, ق, ک, گ, ل, م, ن, و, ہ, ہ, ی, ے

Some of the diatrics of urdu language are:

Pronunciation	Symbol
zabar(short a)	[َ]
zer (short i)	[ِ]
pesh (short u)	[ُ]
tashid (gemination)	[ّ]
jazam (vowel absence)	[ْ]
khari zabar (long a)	[َ]
do zabar (sound un)	[َ]

Table 2.1: Diatrics of Urdu Language.

Secondary Urdu alphabets are:

آ، ۓ، ء

It has the following distinctive qualities in addition to a complex morphological characteristics, which all together make it challenging to work with while doing

data processing tasks such as stopwords removal, stemming and text normalization.

- A lot of words in Urdu vocabulary are loaned from many other languages. Words from Persian, English, Turkish, Arabic and Sanskrit are blended with its own vocabulary to form whole Urdu dictionary. For example:

Arabic(صبر), Turkish(گل), Persian(دشمن) and from English(شیمپو).

- As discussed above, it has unique alphabet set. The writing style of Urdu is Nastaliq which is quite complex in itself.
- There are many words that are morphological variants. This means that numerous words share single root and have different meanings. For example: The root word (حفظ, hifz), means “to memorise”. Other words with same root are (حافظ, hafiz), means “Guardian or a male who memorise Holy Quran”, (حافظہ, hafiza) means “memory or female who memorise Holy Quran”.
- It is highly context sensitive. Some words have space between them they can not be written without spaces and reading them together completes its meaning. But due to space between words it does not define the word boundary. This creates issues in word segmentation process. For example:

(بے بس، براہ کرم) means (please, involuntarily) respectively.

- English language has the concept of word capitalization. The beginning of a sentence can be identified by capitalized letter. Urdu language has no such concept. For example:

[There are two rooms in this house., اس گھر میں دو کمرے ہیں.]

It can be identified in English sentence but there is no indication in Urdu sentence. Hence, proper nouns and start of sentence are unidentifiable [51].

- Case markers are regarded as Parts of Speech in Urdu. Case markers are lexically independent components that define sentence construction. Absence of these markers leads to issues like ambiguities in grammar. For example:

(شیر کی کہانی، کہانی شیر کی) Both have same meaning i.e (Lion’s story) but different order.

- Diatric marks alter the meaning of words written with same letters. The pronunciation of words with similar alphabets also changes. For example:
(اُل) means ‘all’ whereas (اَل) means ‘yesterday’.

2.1.2 Studies in Urdu Text Classification

Here we will discuss several noteworthy research pieces in Urdu Text classification using ML and DL techniques. Most of the ML approaches explored by multiple studies include support Vector Machines (SVM), Naive Bayes (NB), K-Nearest Neighbours (K-NN) and Decision Trees. The classification of News articles, Social Media posts and News headlines are explored frequently.

In [14] authors have explored SVM for the classification of urdu news headlines. To evaluate frequency, the collection of documents under inquiry was first normalised and then stemmed. In the last step stop words were eliminated. Inverse document frequency along with term frequency of words from selected corpus were also calculated by the authors. The authors in [13] examined five common methods of feature selection i.e Chi Statistics, Information Gain, Gain Ratio, Symmetrical Uncertain and oneR. They performed K-NN, Decision Trees and NB classifiers on two Urdu datasets. The analysis shows that the Information Gain approach improves the performance of SVM and KNN classifiers. In his analysis Bilal [54] analyzed three classifiers for the Roman-Urdu dataset. He discovered that NB performed better than Decision Tree and K-NN. The evaluation metrics they used are accuracy, recall, and f-measure. The characteristics of classifying news origin in Urdu text are discussed in this research [23]. They performed a comparative study using SVM, K-NN and Decision Tree on a large Urdu dataset containing 16,678 documents, the majority of which were news pieces from the Urdu publication The Daily Roshni. For feature selection TF-IDF weighting scheme is used. Their results proved SVM to be better than other two classifiers in terms of accuracy.

In [45] authors focus on multi-class classification of Pakistani News dataset. They applied various machine learning algorithms i.e Logistic Regression, SVM, NB and Random forest for both single and multi-class. Their comparative analysis shows SVM to be best for binary classification whereas Logistic Regression performs best for multi-class classification.

Nabeel [38] provides a benchmark Urdu dataset and evaluate various ML and DL techniques in their study. They also examined the effects of transfer learning for Urdu language by Bidirectional Encoder Representations via Transformers approach. The findings from their extensive comparative research shows that a technique of feature selection, Normalised Difference Measure combined with ML and DL classifiers outperforms and uplifts the efficiency quite significantly. Three well-known and common DL models are explored in [51]. They compared results with ML models and used various preprocessing methods. They concluded that Convolutional Neural Networks (CNN) outshines other DL and ML models.

2.2 Approaches to Inappropriate Content Detection

Over the past ten years, many NLP researchers have neglected to pay attention to the identification of offensive language. The improvements in NLP methods for everyday tasks were particularly encouraging for overcoming the difficulties of identifying hate speech in social networks. It can be difficult to detect violent language, especially when it's hard to tell it apart from other objectionable content or even harmless stuff where there may be vocabulary overlap. It is common to witness the usage of derogatory or vulgar language in sarcastic or amusing contexts. The NLP community has been concentrating on internet platforms including Twitter, YouTube, Instagram, Facebook, and online blogs to identify harmful language [26], [18], [31], [21].

2.2.1 Machine Learning approaches

Several studies have explored ML techniques to identify inappropriate language in different languages. SVM, Logistic Regression, K-NN are Decision trees are the most commonly employed ML algorithms by many studies for this task. Figure 2.1 shows an example of how decision trees algorithm works. Various feature selection techniques such as lexicon based and chi-square are explored by multiple studies for abusive or offensive language detection. In [27], [30], [42] authors relied on character n-gram feature extraction methods. Some studies [27],[18], [34], [31] also explored word n-gram and its variants for this purpose. In [27] SVM with lin-

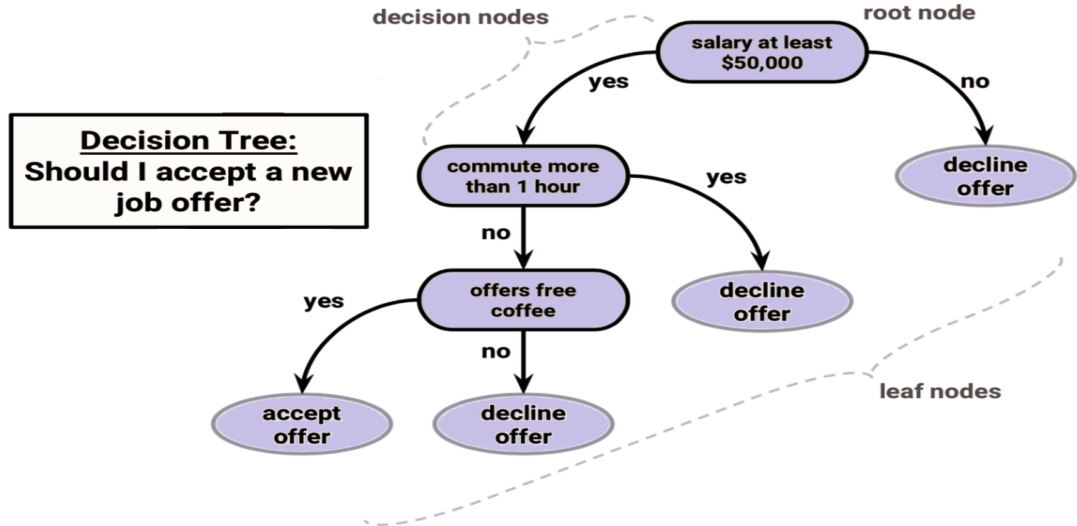


Figure 2.1: Decision Trees Example

ear kernel is proposed to achieve best results in detecting inappropriate language in Bengali dataset. In their suggested approach, both Unicode Bengali characters as well as Unicode emoticons were taken into consideration as acceptable input in our suggested approach. Authors in [53] employed ML classifiers SVM, Logistic Regression, Decision Trees and Random forest etc. along with modular cleaning of twitter dataset and built a tokenizer.

Threatening comments in Twitter tweets were categorised using NB in [22]. Tasks focused on identifying abusive language in social media were presented at the International Workshop in [35]. Identification of objectionable content, automatic classification of offensive categories, and identification of offensive targets comprised the three main subtasks. The messages were categorised as either not offensive or offensive for the first sub-task. If a tweet used abusive language, it was flagged as offensive. The ML classifiers used were NB, Logistic Regression, SVM and Random Forest for this purpose.

Authors in [31] utilized n-gram feature extraction technique up to 8-gram to evaluate its effects with respect to tasks presented at International conference as discussed in previous paper. They trained three automated systems for three subtasks. For first subtask linear SVM along with uni-gram and bi-gram model was introduced. A linear SVM model combined with n-grams upto 4-gram model was proposed for second task. Lastly, they employed Decision Trees model with uni-

gram to 8-gram feature selection method. They achieved great accuracy results and concluded that n-gram implementation takes very less time and is quite simple to employ.

For a poor resource language like Urdu, finding publicly available dataset is quite hard to find. In [36] a offensive language annotated dataset in Urdu script is created and made publicly available. To explore the full potential of ML algorithms they thoroughly experimented multiple ML algorithms on both Urdu and Roman Urdu script dataset and provided a detailed comparative analysis using n-gram feature selection method. They achieved best performance with regression-based ML models. They have extracted characteristics from the dataset using character and word n-grams. This indicates that they are merely attempting to obtain local context data, which is insufficient for obtaining the whole context necessary for hate speech identification. Their analysis concluded that although regression ML models gave outclass accuracy when compared with other models, but they require more time to create these models.

Multi-class classification in Roman Urdu content extracted from YouTube comments is presented by [41]. For feature selection n-gram and TF-IDF techniques are employed first. For normalization, L1 and L2 approaches are applied. They also used SMOTE to balance classes since some labels had comparatively less instances. For comparative analysis , Logistic Regression, NB, SVM and SGD classifier is utilized. Their findings demonstrated that SVM combined with n-gram feature selection, L2 normalization and TD-IDF feature values outperforms all other models on their dataset. The hyper parameters of ML models were also tuned using 10-fold cross-validation. Additionally, they created a web interface for YT Monitor. Its purpose is to first scrape user comments from a keyword or given URL link and then classify it to respective hate content categories.

2.2.2 Deep Learning approaches

NLP characteristics such as spelling and grammar errors, contextual ambiguity, polysemy, and semantic variations makes it challenging to identify inappropriate language. A hybrid deep learning model by combining CNN and Bidirectional Long short term memory (Bi-LSTM) is proposed in this study [24] for automat-

ically identifying such inappropriate language as shown in figure 2.2. They were particularly interested in finding a solution for the following two application scenarios:(a) search engine query completion recommendations,and (b) chats of user in messenger.

In [33] authors employed DL models to automatically recognise Facebook post-

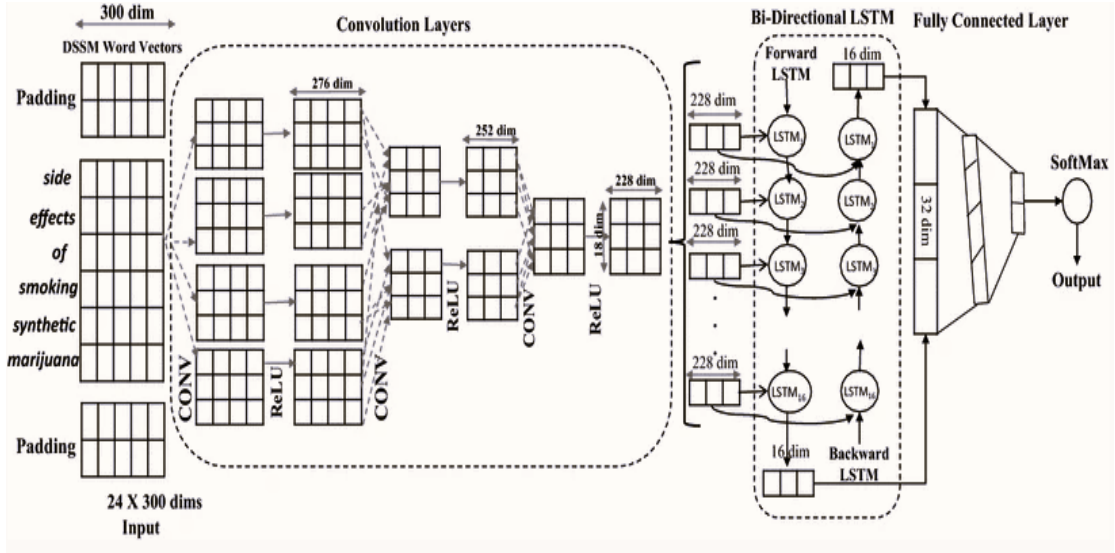


Figure 2.2: A CNN and Bi-LSTM model approach.[24]

ings from users who require emergency help as a result of domestic violence. In order to develop a multi-class identification model that recognises individuals who are critically in need of assistance as a result of domestic violence situations, the authors gathered data from Facebook posts related to domestic violence. They experimented with multiple DL classifiers as well as word embeddings. A gated recurrent unit (GRU) classifier along with word embeddings had the best performance in accuracy.

In [25] authors used an ensemble method to advance the application of DL models for hate speech content identification. The majority of the posts in the datasets used for hate content identification are taken from social media sites like Twitter. When compared to other categories, the proportion of hate speech incidents in the real world is quite low. This distribution between hate speech and other categories can be concluded from the majority of the statistics gathered from social media. The lack of hate speech instances in datasets is recognised to be a difficulty for its detection tasks. Various NLP tasks can benefit from transfer learning approaches

including universal language model fine-tuning (ULMFiT), generative pre-trained transformer (GPT), embedding from language models (ELMO) and Bidirectional Encoder Representations from Transformers (BERT) [49]. The issue of identifying inappropriate language in Dravidian languages i.e Malayalam, Tamil and Kannada taken from Youtube comments is viewed in this study [44] as a multi-class classification problem. The study then presents the accuracy estimates for ML models on the training data. They have establish an inflection point by significant advancements, particularly in activities where data is scarce. To increase the effectiveness of hate speech identification, the authors tested a variety of fine-tuning techniques. The authors have conclude that the CNN-based model and BERT transfer learning models can perform better than other approaches that have been looked at.

The recently released study by [40] is possibly the most important work in detecting offensive language on Roman Urdu dataset. The researchers have presented their findings for both coarse grained and fine grained classification tasks using a variety of widely used baseline ML and DL models. They demonstrated how their unique BERT and CNN-ngram hybrid model may be used for transfer learning and attain an outclass F1-score on a coarse-grained classification problem. [43] provides details on how to recognize threatening language and identify targets in Twitter posts written in Urdu. The authors of this research offered a dataset that consists of 3,564 Twitter messages that have been manually classified as either harmful or non-harmful by human specialists. The target further categorises the threatening tweets into one of two categories: threats against an individual or threats against a group. Numerous experiments using a variety of ML and DL techniques revealed that the best threatening content detection accuracy was achieved by MLP classifier combined with word n-gram model. Whereas, SVM along with fastText word embedding outperforms for target identification task.

In this study [52], they formulated an annotated hate speech lexicon for the Urdu language of 10,526 tweets. They also employed a variety of ML methods for the detection of hate content as baseline experiments. Additionally, they applied transfer learning approaches to take use of multilingual BERT and FastText Urdu word embeddings for their assignment. They tested four alternative BERT versions and

achieved encouraging results for multi class classification problem.

2.3 Comparative Analysis

As already established, when it comes to the identification of hate speech content, a lot of the study is concentrated on the English language. All feature extraction methods, pattern discovery procedures, and models are consequently built for English. In addition, there is a dearth of structured data that may be used for research in poor resource languages. For English, there are some quite large annotated publicly available datasets, but the same cannot be stated for resource scarce languages like Urdu for this problem space. Additionally, to the best of our knowledge, there isn't any extensive published research on the identification of inappropriate content in the Urdu language at this time. Due to this, it is currently difficult to determine which method would work best for a given data set or how the training data should be modified to derive the best predictions.

We summarize the existing literature related to inappropriate/offensive language detection in Table 2.2. The first column lists all the papers from which the literature review is conducted. The next column describes the platform from dataset is collected. From the literature it was observed that most of the hate speech content data source is social media sites. As large community of people having backgrounds from all around the world has easy access to social media sites. Hence, these sites are the best source for collecting data related to inappropriate content detection domain.

The "Features" column lists all the feature extraction techniques that are most commonly used. TF-IDF is a fairly straightforward but intuitive method of weighing words, making it a perfect starting point for a number of tasks. For Deep learning model use of word embeddings is recommended. Word2vec is the most commonly used word embedding. It is also available in many languages including Urdu. The next two columns describe that most of the studies are carried out in English and very few are conducted in Urdu specifically Urdu Nastaliq script. And those research that analyse the Urdu script either have extremely tiny dataset sizes to investigate deep learning methods or have a very low number of words in

each label to effectively extract the true features. Lastly, The different ML and DL models are listed in last column.

We highlight the following research gaps that we address in our work as a result of our study of the literature in the Urdu language:

- Lack of inappropriate content datasets in the Urdu language.
- Exploring advanced deep learning model with attention layer.
- Comparison with baseline DL models
- Impact of word embeddings on Inappropriate content dataset

Our research seeks to contribute in these fields, by developing and making available a larger dataset in Urdu, comparing the effectiveness of embedding features, and evaluating DL models to assess their effectiveness.

The baseline DL models used in this work will be detailed with respect to their framework, evolution and applications in the next chapter. We will discuss their methodology briefly. The last section will lay out the use of word embedding layer in a DL model and the type of embedding used in this work.

Paper	Platform	Features	Language	Data size	Technique
[27]	Facebook	TF-IDF, n-gram	Bengali	Small	MNB,SVM,CNN-LSTM
[53]	Twitter	TF-IDF, W2V, FastText	English	Large	MLP, SVM, RF,LR, GB, DT, AdaBoost, NB
[22]	Twitter	TF-IDF, n-grams	English	Large	NB, LR, SVM, RF, GBT, CNN, RNN
[35]	Twitter	TF-IDF, CV, Glove	English	Moderate	NB, LR, SVM, RF, LSTM
[31]	Twitter	n-gram	English	Moderate	linear SVM, DT
[36]	Twitter	n-gram	Urdu, Roman Urdu	Moderate	17 ML algos
[41]	YouTube comments	TF-IDF, n-gram, L1, L2	Roman Urdu	Moderate	LR,SVM,SGD,NB
[24]	search engine, messenger	-	English	Large	LSTM,C-BiLSTM, BLSTM
[33]	Facebook	W2V, GloVe	English	Large	CNN,RNN,GRU, LSTM,BLSTM
[25]	Twitter	Keras EL	English	Moderate	Ensemble CNN
[44]	YouTube comments	TF-IDF	Dravidian	Large	NB, SVM, KNN, DT, LR,RF
[40]	Twitter	FastText, BERT	Roman Urdu	Moderate	hybrid DL models, CNN-gram
[43]	Twitter	n-gram, FastText	Urdu	Small	LR,RF,AdaBoost, MLP,SVM,1D-CNN, LSTM
[52]	Twitter	TF-IDF,CV, w2v,FastText, BERT	Urdu	Moderate	ML, CNN, Bi-GRU

Table 2.2: Summary of research on Inappropriate language detection

Deep Neural Networks for Inappropriate Language Detection

As a low-resource language, Urdu has restricted access to data sets due to which it lacks fundamental NLP-based research. Additionally, there is no open source resources for such a morphologically faithful language. In addition to these facts, Urdu is gradually maturing for NLP-based applications[39] as they are being developed with time. Artificial Neural Networks (ANN), which are used in the advanced domain of machine learning, or DL, are based on the human neural system and are employed to learn the features of huge experimental data using statistical techniques to forecast unseen test data. Deep learning models are frequently used by researchers for Sentiment Analysis and Opinion Extraction on resource-extensive languages. These models outcomes have encouraged the scholars' confidence in artificial intelligence (AI).

This chapter delves deeper into the detailed explanation of Deep Learning models, particularly RNN, LSTM, BiLSTM, TCN (Temporal Convolutional Networks) and GRU. An outline of the Urdu word embeddings will be discussed later in the chapter.

3.1 Deep Learning Models

Deep learning is an emerging machine learning methodology in recent years. In this approach, numerous characteristics of data are learnt, and then, using those features, state-of-the-art results are generated. Deep learning has had tremendous success in recent years in a variety of practical fields, such as computer vision and speech recognition. Artificial neural networks are used in deep learning to learn tasks using a multi-layered network. Artificial neural networks (ANNs), which are modelled after the intricate structure of the biological brain, have several neurons stacked at various levels and that cooperate with one another. It mimics the learning process of a biological brain by adjusting the weights between neurons as it learns to accomplish various tasks. To extract features and transform them, deep learning employs numerous layers of nonlinear information processing units. Higher layers of the neural network often learn complicated features, whereas the lower layers, which are closer to the input data, learn basic features.

Deep learning approaches differ from conventional machine learning models in that, prior to applying a predictive model, they transform the early representation of predicting factors into a highly abstract set of data characteristics. When classifying text using deep learning, the model is fed both the outcome variable and a meaningful vector representation of the predictive variables. Prior to delivering the input data to the model layer, the deep learning model first learns the usable collection of features from the input data. This capability of deep learning models helps to build the most effective predictive characteristics from the initial text representation. This is helpful because it is a tough process for humans to exactly specify predictive textual features before hand.

When deep learning is applied to text classification, it is anticipated that the algorithm will learn and benefit from complex data like word interactions and word patterns, in contrast to typical ML techniques that consider word score. In order to maximise model performance, these models are subjected to a great deal of parameter tuning, optimization, and constant architecture adjustment.

The primary deep learning algorithms and associated methods, which are mostly used for NLP tasks, are explained in the following number of subsections.

3.1.1 Recurrent Neural Network (RNN):

Traditional Neural Networks (NN) are ineffectual when used with sequence learning, according to the AI paradigm. This is because it is more difficult to correlate the front and back textual sequences. Inputs are coupled in recurrent neural networks (RNN), a sequence learning model. It signifies the generalised feed forward NN in which the hidden layers of the model are connected by nodes, and the sequence characteristics are dynamically learned. The RNN model uses the input phrase (تمیز سے بات کریں) i.e. “Talk appropriately” as its starting point. After word segmentation, each word in the input sentence is transformed into a word vector (w_1, w_2, w_3, w_4). These word vectors are used as the RNN layer’s sequential input. This is how the RNN process operates:

- The hidden layer has initial input w_0 and its output is m_0 .
- m_0 and w_1 are the inputs for the subsequent phase.
- Similar to that, m_1 and m_2 are used as input for the following phase.

$$h_t = f(Xw_t + Wm_{t-1}) \quad (3.1.1)$$

$$y_t = softmax(Ym_t) \quad (3.1.2)$$

During training, RNN learns the sentence context. RNN can transfer semantic information across words, however it cannot capture long-distance semantic links between different words. The gradient continuously decreases while the model is being trained until it disappears entirely. As a result, the length of the sequential data is constrained. This is called vanishing gradient problem in RNNs. RNN architecture is given in Fig. 3.1.

3.1.2 Gated Recurrent Unit (GRU):

GRUs are an enhanced form of the traditional recurrent neural network. The ability of GRUs to provide a gated hidden state is the primary differentiation that can be made between GRUs and vanilla RNNs. GRU makes use of the so-called

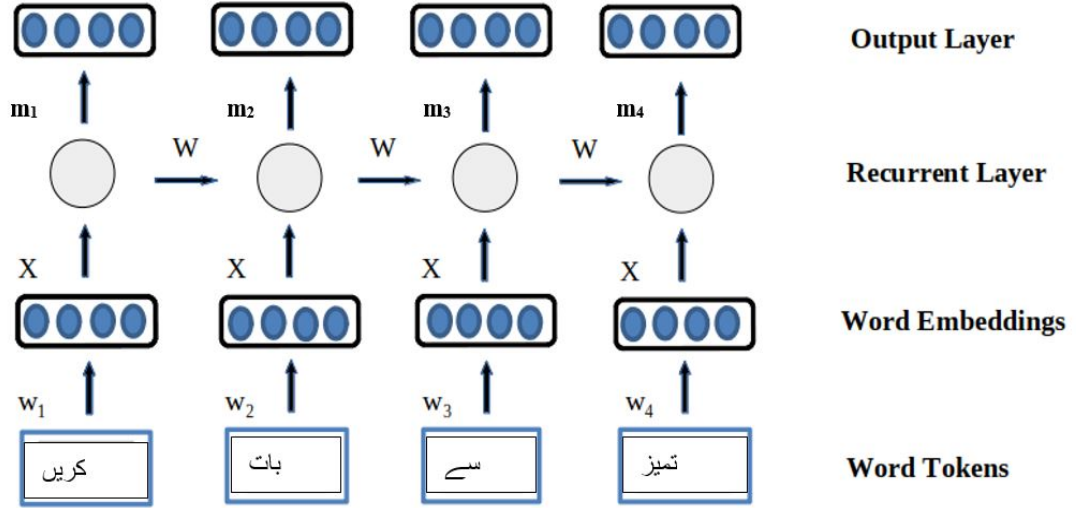


Figure 3.1: RNN Model

update gate and reset gate to address the vanishing gradient issue that affects a normal RNN. This indicates that we have specialised processes for determining when a hidden state ought to be updated as well as when it ought to be reset. For example, if the first token is extremely significant, it will eventually figure out that after the first observation, the token should refrain from updating the hidden state. In a similar vein, we will acquire the ability to skip over temporary observations that are irrelevant. In the end, it will become proficient in resetting the hidden state whenever it is necessary. In essence, these two vectors determine what data should be sent to the output. They have the unique ability to be trained to retain information from the past without having it fade away over time or to discard information that is unrelated to the forecast.

To begin, we will use the following formula to determine the value of the update gate y_t for time step t :

$$y_t = \sigma(W^{(y)}x_t + U^{(y)}h_{t-1}) \quad (3.1.3)$$

When x_t is connected to the network unit, the value of that variable is multiplied by its own weight, W . The same is true for h_{t-1} , which stores data for the units that came before it and is multiplied by its own weight, U . After adding the two sets of findings together, a sigmoid activation function is used to bring the total

value down to a range between 0 and 1, inclusive. The update gate provides the model with the assistance it needs to assess how much of the knowledge obtained from earlier time steps is necessary to be carried forward into the next one. This is a particularly significant feature since it allows the model to choose to copy all of the information from the past, removing the possibility of an issue with vanishing gradients.

The purpose of reset gate in the model is, essentially, to determine how much of the information from the past should be forgotten. The reset gate functions in a manner that is broadly analogous to that of the LSTM's Forget gate in that it categorises data that is unrelated and instructs the model to forget about this data and proceed without it.

3.1.3 Long Short Term Memory (LSTM):

(AI) [1] introduced the Long Short-Term Memory (LSTM) model to address the RNN problem. This model uses three gates—the input gate, forget gate, and the output gate—to learn long-term dependencies between various words.

A deep neural network variant known as LSTM introduces special hidden units known as memory blocks. The temporal state of the neural network is preserved in memory cells within the memory block that have recurrent connections. The special multiplicative units in this block, known as gates, govern the information flow via each unit. According on the weights the model is learning, the input gate accepts or rejects sequential data, and the forget gate activates or deactivates a neuron. The output gate chooses the units' output value for the LSTM.

Each memory block has an input and an output gate in the basic LSTM architecture (AI)[9]. The output gate regulates the flow of activations from the present memory cell to the remainder of the neural network, while the input gate regulates the flow of activations into the memory cell. The original LSTM architecture has the issue of making it difficult for LSTMs to process continuous input streams if the input stream is not sub-sequenced. Later, a forget gate was developed to enable adoptive forgetfulness as a solution to this problem. The LSTM cell's memory can be reset or forgotten with the help of the forget gate. The LSTM's basic

architecture is shown in Fig. 3.2.

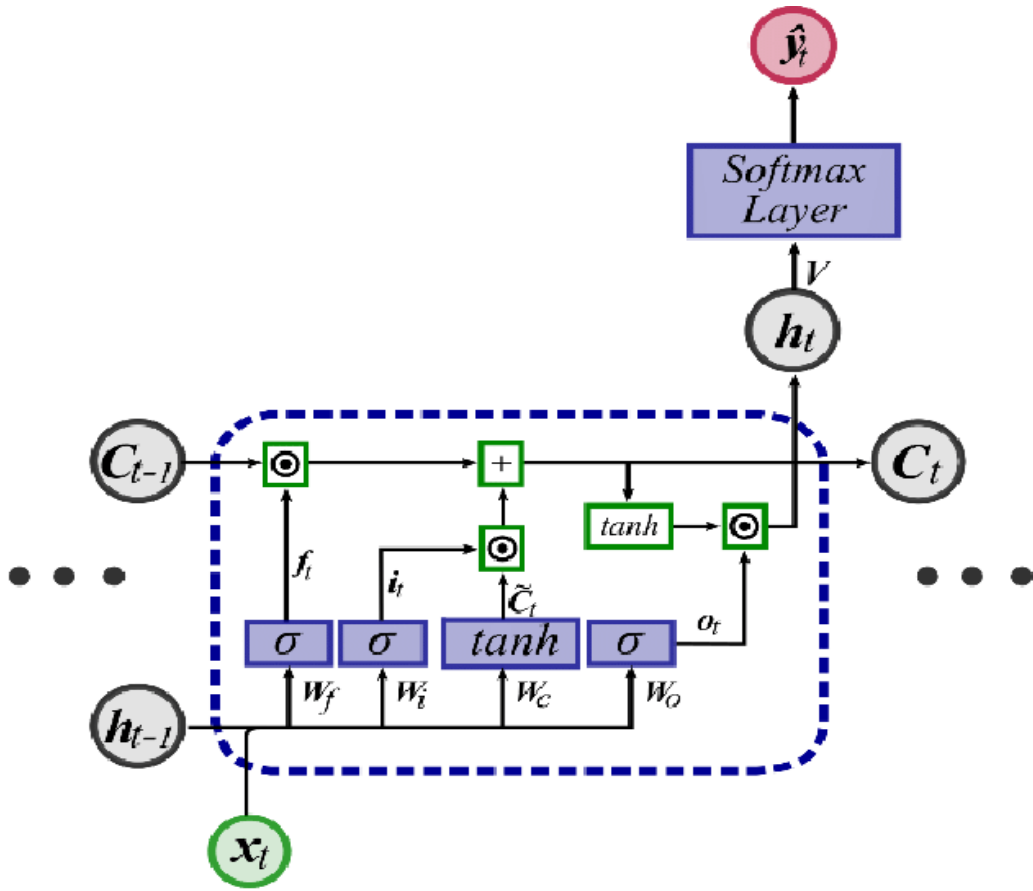


Figure 3.2: LSTM Model[17]

3.1.4 BiDirectional LSTM (BiLSTM):

The conventional RNN model and the LSTM are only able to convey information in the forward direction. Because of this characteristic, these models are able to depend on information that was processed before a specific time. Bidirectional LSTM is utilized [2] as a solution to respond to this difficulty. It has been shown to be very helpful when the context of the input is important. Architecture of BiLSTM is given in Fig. 3.3.

The downstream output of the perceptron layers of RNNs is fed back upstream as an input to the following layers. They are thought to be suitable for temporal data, such as text, because they can detect sequential correlations in the incoming data. Word order in text plays a crucial role in the meaningful understanding of the text,

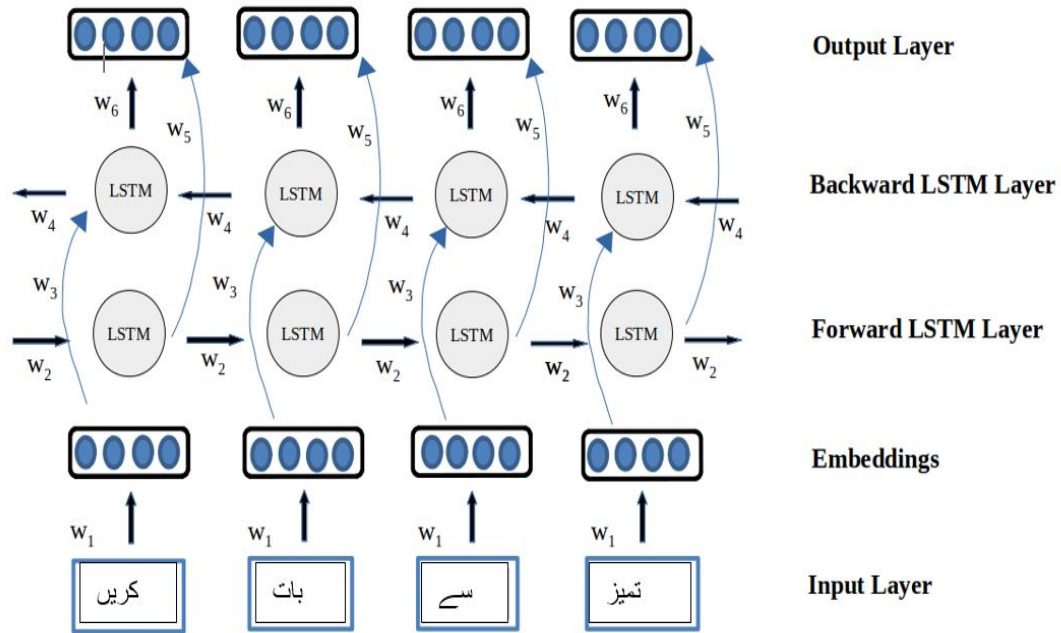


Figure 3.3: BiLSTM Model

making such networks appropriate for text classification problems. When words in a phrase are spaced widely apart in time, standard RNNs cannot understand the correlations between those inputs. Perceptrons, which have been designed to use gates to intentionally forget prior inputs, are less complex brain units than neurons. The LSTM cell is a key illustration of a gated neural network, which is widely used in sequential data or text classification categorization. In contrast to perceptrons, which produce a single hidden layer value, LSTMs output a memory state. In the LSTM architecture, gates are used to specify which portions of the memory state are to be updated with current inputs, which portions of the memory state are to be forgotten, and which combinations of current inputs and memory state are to be output. With this method, the memory units can keep track of long-range correlations between words that are spread out across the text.

For the purpose of taking into account all contextual information, BiLSTM combines both LSTM and bidirectional RNN. Due to the employment of two hidden layers, it has a tendency to process the data in both forward and backward directions. Input data from the past and future can be used to forecast better outcomes by using two directions of time. A lexical item in a text may have a link to both preceding and following words, which is why BiLSTMs can be helpful

in capturing bi-directional correlations in the text. Each bi-LSTM cell creates two different text representations in categorical text classification. The output of many BiLSTM cells, each of which tends to concentrate on a distinct textual feature, is then supplied into an output layer, which uses an activation function to obtain the results.

3.1.5 Temporal Convolutional Network (TCN):

Sequence modelling has previously been mostly related to recurrent neural network architectures like LSTM and GRU in the context of deep learning. The capacity of recurrent neural networks (RNN) to store historical data as time series makes them a popular choice for sequential tasks. The vanishing gradient issue in RNN is resolved by Long Short Term Memory (LSTM) Neural Networks [4] and Gated Recurrent Unit (GRU) [8], which are successfully used in speech and natural language processing. This way of thinking is outdated, according to S. Bai et al. [19], who believes that convolutional networks should be one of the top options when representing sequential data. They were able to demonstrate that convolutional networks are capable of outperforming RNNs in a variety of tasks while avoiding recurrent models' typical flaws, such as the exploding/vanishing gradient problem and poor memory retention. Additionally, because convolutional networks allow for parallel calculation of outputs, employing them in place of recurrent ones can enhance performance. They term this suggested architecture, a Temporal Convolutional Network (TCN).

To forecast the following l values of input time series, a temporal convolutional network is trained. Assume that a set of inputs $x_0, x_1, x_2, \dots, x_L$ provided, and that it is desired to forecast some corresponding output $y_0, y_1, y_2, \dots, y_L$ each time step, whose values are the same as the inputs shifted ahead l time steps. The key restriction is that it can only use the inputs $x_0, x_1, x_2, \dots, x_L$ when forecasting the output y_t that has been previously observed at some time step t . TCN architecture is shown in Figure 3.4.

Two major restrictions on the TCN are that it can only use data from previous time steps and that its output should be the same length as its input [28]. A 1-

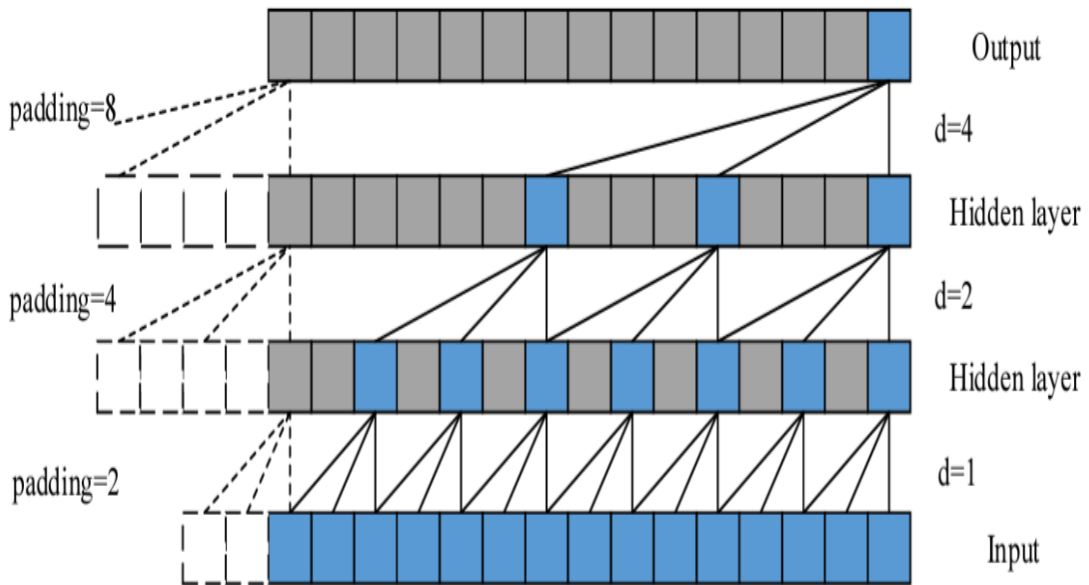


Figure 3.4: Temporal Convolutional Network Architecture[19]

D fully CNN architecture[11] is employed in TCN to satisfy these temporal rules because all of its convolution layers have the same length and zero padding ensures that higher levels are the same length as lower layers. Additionally, TCN employs causal convolutions, wherein each layer's output is computed at time step t using only the region computed at time step t or earlier in the layer before it. It is simple to perform the causal convolution for 1-D data by changing the output of a regular convolution by several time steps.

3.2 Word Embedding:

The adoption of neural network-based models known as "word embeddings," which combine the information as dense and dispersed vectors, has expanded as a result of recent advancements in NLP [7]. Multiple NLP applications have shown performance improvements as a result of these embeddings. Additionally, by assisting DL algorithms in more easily learning textual patterns from the better representations of words and obtaining better generalised output from less input, they hold the key to improving NLP results for resource scarce languages. In addition to enhancing the performance of deep learning algorithms in numerous NLP ap-

plications, the use of word embedding has inspired critically needed research on resource scarce languages.

The outcomes of word embedding are frequently used as input characteristics in deep learning models for NLP [3]. Words are represented in word embedding as vectors that encode their semantic characteristics. The process of language modelling and feature learning known as word embedding involves the transformation of tokens into vectors of progressive real values. This method typically entails embeddings—where each word is treated as a dimension—from a sparse vector of high-dimension, such as one-hot encoding, to a low-dimensional dense vector space. These embedding vectors each have a dimension that represents a hidden feature of a word. These word vectors have syntactic regularities and patterns encoded within them.

Matrix factorization and the usage of neural networks are two popular techniques for learning word embeddings [6]. Word2Vec is a widely used word embedding system that efficiently trains word embeddings from text using a neural network prediction model. There are two models in it: the Skip-gram (SG) model and the Continuous Bag of Words (CBoW) model. In contrast to the SG model, which uses the target words to predict the context words, the CBoW model uses context words to predict the target word. The CBoW approach, which works well with tiny datasets, treats the full text context as a single observation. The SG model, however, treats each context and the target word pair as a separate observation and performs best with big datasets. For the Urdu language, pre-trained word embeddings can be created using the Word2Vec model [48].

The next chapter will cover the details of dataset collection process, its annotation and statistics. We will discuss the pre processing techniques employed briefly illustrate the proposed architecture. It will be concluded with the explanation of setup developed to carry out model experiments and layering of deep neural networks.

Design and Methodology

In this chapter we briefly illustrated the methodology of our proposed model for identification of inappropriate content detection. DL has yet to be fully investigated for detection of inappropriate content in Urdu unicode script. By using a hybrid DL strategy, the use of our suggested architecture and basic DL models tends to close the gap that has been identified in the studied literature. This chapter illustrates the process of data collection, pre-processing of data and in the last a thorough explanation of proposed architecture based on hybrid DL algorithms for Inappropriate content detection.

4.1 Dataset

As its already established, Urdu is a resource scarce language. To collect domain specific data for such languages is the first most important and difficult task of any NLP task. It must have specific abusive/harmful words for proper identification of text. We were successful in finding the appropriate data sources to fulfill our requirements. For this research problem, data is collected from different sources which is explained in detail in the coming section.

4.1.1 Dataset collection

Three publicly available datasets in Urdu native from the online Internet source are obtained for this problem space.

- The first dataset¹ consists of twitter tweets and is obtained using twitter Application Programming Interface (API) with tweets containing violent or abusive content labeled ‘1’ and neutral content labelled as ‘0’.
- Second dataset² is obtained from YouTube videos Urdu comments and is annotated manually by native speakers.
- Third dataset³ obtained was in Roman Urdu language and obtained from twitter just like the other two datasets. It has same labels identifying inappropriate content as ‘1’ and neutral as ‘0’. This dataset is first converted from Roman Urdu to Urdu script using online website called ijunoon⁴.

Finally our dataset is formed with the combination of these three datasets. To study the impact of size of dataset, the dataset is partitioned in variable sized groups. It is combined to form two groups i.e One is UrduInAsmall and other is UrduInAlarge. UrduInAsmall is formed by combining first two datasets and UrduInAlarge is the combination of all three datasets.

Both data sets are have two categories i-e Inappropriate they are labeled as ‘1’and Appropriate that are labeled as ‘0’. Table 4.1 shows the two categories and its labels. A sample of dataset presented in figure 4.1 where the types of text in two classes can be understood easily.

Class	Label	Label Name
Class A	1	Inappropriate
Class B	0	Appropriate

Table 4.1: Dataset categories.

4.1.2 Dataset annotation and Statistics

Gaining ground truth, or arriving at a situation in which the annotated data perfectly fits the requirements, is the fundamental goal of data annotation. Mainly,

¹https://github.com/MaazAmjad/Threatening_Dataset

²<https://github.com/pervezbcs/Urdu-Abusive-Dataset>

³https://github.com/haroonshakeel/roman_urdu_hate_speech

⁴<http://https://www.ijunoon.com/>

Text	Label
یہ خبر افسوس ناک ہے	0
جناب محترم سلمان وفادارصاحب کیا پارلیمنٹ آئینی ادارہ نہیں جس کی اینٹ سے اینٹ بجائے والا ایک شخص	0
چھوٹے قد کی لڑکی اور شہد کی مکھی دونوں ہی بہت خطرناک ہوتی ہیں	0
بکواس مت کرو	1
تمہاری ہیپجڑا فورس ایک نہتے کے سامنے بکری بنی ہے ڈوب مرو	1
آفیسر سمیتبھارتی فوجی جہنم واصل، ہنکرز تباہ بھارت کو پتہ لگ جائے...	1

Figure 4.1: Sample of Dataset.

data is annotated in two ways either automatically or manually. Automatic annotation is thought to be less precise than manual annotation, although it can label several more datasets in a shorter amount of time than a single person can. However, dedicated annotators must pay close attention and be exact while using manual annotation, which is more accurate overall.

For a text to identify as inappropriate content, there are set of words in every language that constitutes as abusive or violent word set. These tweets are manually annotated by the researchers of dataset providers by keeping in mind this list of words to label them accordingly. They hired native Pakistani annotators to achieve maximum efficiency. They were well educated and were advised to stay on neutral grounds in case of addressing text political conflicts. This is important to mention that abusive or violent words can either be only one or multiple words in a single tweet.

In UrduInAsmall, there are total 5734 entries of tweets from which 2890 instances

Characteristics	UrduInAsmall	UrduInAlarge
Total lines	5734	14946
Inappropriate	2890	7181
Appropriate	2844	7765
Maximum words in sentence	198	240

Table 4.2: Statistics of dataset.

are categorized as Inappropriate class and other 2844 are labelled as Appropriate class. In UrduInAlarge, there are 14946 text instances in our dataset that are

divided into two classes i-e Inappropriate and Appropriate. From 14946 instances of total classes there are 7181 tweets in Inappropriate class, they are labeled as ‘1’ and 7765 items in Appropriate class that are labeled as ‘0’. Table 4.2 shows the statistics of two data sets which can also be visualized in figure 4.2 and figure 4.3. The comparison of two different-sized data sets aims to investigate how

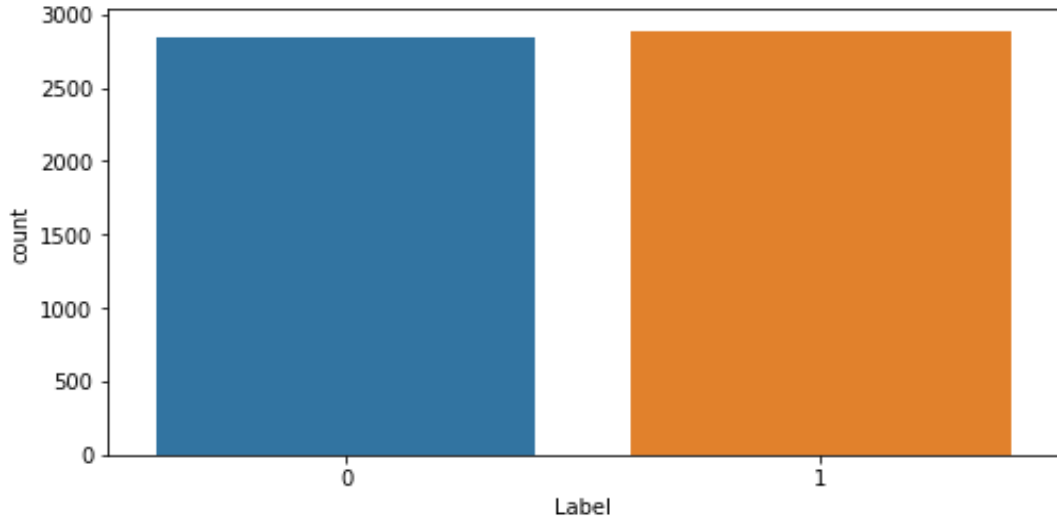


Figure 4.2: UrduInASmall Dataset Statistics Graph.

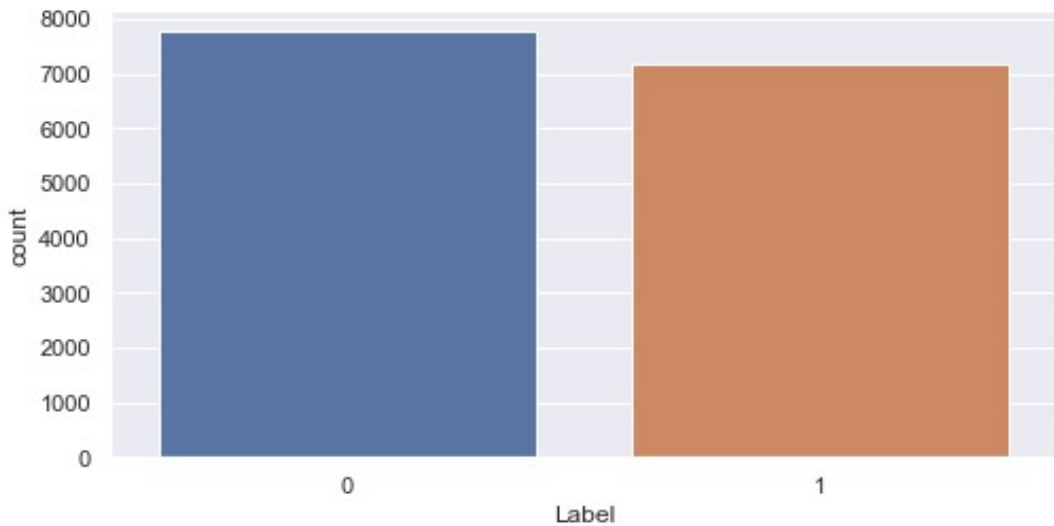


Figure 4.3: UrduInALarge Dataset Statistics Graph.

these data sets affect the performance of DL models. This data set is also checked via Inter-Annotator Agreement (IAA) using Cohen’s Kappa coefficient which is a statistic used to determine whether two annotators can be relied upon. A 90

percent Kappa coefficient was obtained as a result of the measurement by the researchers.

4.2 Dataset Pre-Processing

The next core task in every NLP problem is the preprocessing of dataset. It helps in organizing the information by applying basic operations on it before it is ready to feed into a neural network. Other operations of the process include removal of white spaces and unnecessary words, converting words into their root forms, elimination of redundant words, and tokenizing of the translated sentences and developing a lexicon for source languages. It transforms the raw dataset into a organized and meaningful dataset for further processing.

To get accurate and best results, an well organized, thoroughly cleaned form garbage data, and normalized data is preferred. Pre-processing keeps data clean and free of noise and redundant information. Researchers use this method frequently to obtain cleaned data for improved model interpretation. We used text pre-processing for standardisation on our data sets to implement our suggested model.

For the first step the text is normalized to correct the issue with proper Urdu character encoding and swaps out incorrect Arabic characters with proper Urdu ones. This also brings all the Urdu characters inside the designated unicode range i.e (0600-06FF).It also removes or places blank spaces in such a way that no extra or error words are added in the dictionary.

Next step include removal of punctuation and diatrics marks from the text. So any instances like ‘,;’ and zer, zabar pesh as discussed in chapter 2 section 2.1.1 are removed.

The other symbols like currency, URLs, numerical digits, emails and English alphabets are removed in the third step. Any extra spaces or line breaks were also removed in this step. Another important step of pre-processing is removal of stop words from the text. Stop words in any language are words that have no meaning in itself but are used to give meaning to a sentences or words. For building a dictionary they are not important to consider. For this purpose a list of stop word

in Urdu language is prepared and then used to remove from our dataset text. Finally, This text is then passed through a tokenizer to build a word dictionary. Each word in a text line is considered as a single token (word) after the tokenization process. The preprocessed data is then fed into the neural network model in order to evaluate the model's performance.

4.3 Proposed Methodology

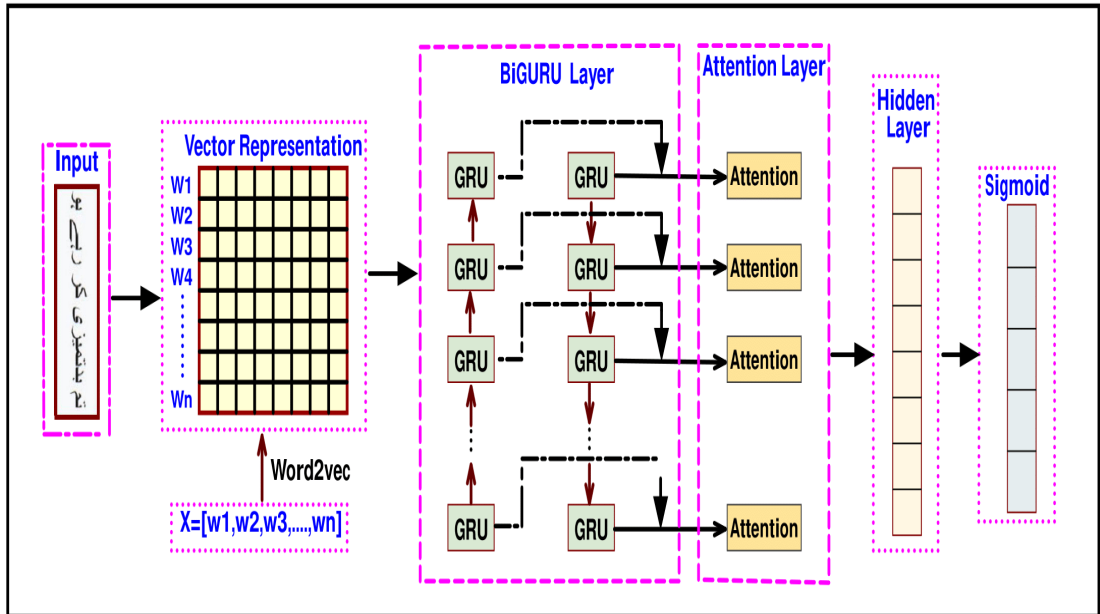


Figure 4.4: Attention based Bidirectional Gated Recurrent Unit BiGRU-A - Proposed model

The proposed model integrates elements of several well-known NN models, specifically the Attention-based Bi-directional GRU. Figure 4.4 is an illustration of the suggested Bi-GRU model with the attention mechanism. Both the Bidirectional GRU and the attention layer are well-known for their applications in text classification, which is why this hybrid model was merged in order to evaluate the adaptability of the former with the latter. Both the Bi-GRU and the attention layer serve distinct functions during the text classification process. The bidirectional gated recurrent unit was implemented so that we could manage the unique aspect of polarity in text classification data and obtain independent context se-

semantic information in the forward and backward passes. The attention mechanism is used to give weights to the features based on how much they contribute to classification. In addition to our suggested model, GRU, LSTM, BiLSTM, and TCN have all been used on a curated dataset. The suggested method provides better classification accuracy for differently sized data sets.

4.3.1 Bidirectional GRU

The suggested bidirectional RNN is utilised to handle the problem where the prior output is not only associated to the prior state, but also to the subsequent state. A Bi-RNN may learn both the forward and the backward properties of the data. A forwarding and backward network combination like this will suit data better than a unidirectional RNN. Text classification frequently makes use of RNN. When dealing with lengthy sequences, the standard RNN is exposed to the issues of vanishing gradient and explosion. The bidirectional GRU is a unique variation of the bidirectional RNN that divides the regular GRU into two directions: a forward direction associated with historical data and a reverse direction associated with future data. This allows for simultaneous use of the input observational data and future data. The unidirectional GRU's classification performance can be significantly enhanced by this configuration. In comparison to RNN, it has greater advantages in handling long sequence texts and solves the gradient disappearance and explosion problems by adding update gate and reset gate to neurons. It also extracts text context information more successfully. This study represents a deep learning text classification technique based on a hybrid BiGRU-Attention model. The following list represents the main contents:

1. Word embedding methodology is used to train the word vector, and the text data is encoded as a low space dense matrix.
2. To extract text context characteristics, Bi-GRU is implemented.
3. The Attention layer receives the output of BiGRU as an input to compute the attention score.

4.3.2 Attention

An attention mechanism is a part of a neural network. The essence of the attention process is the weight distribution of token. The more significant the words with higher weights are in the entire text and the more significant their role in the entire classification task. At each decoder stage, it judges what source elements are more important. The encoder in this arrangement does not have to vectorize the entire sentence; instead, it gives representations for each source token, such as the entire set of RNN states rather than just the most recent one. The basic idea is that a network can determine which input elements are more important at each level. Everything in this scenario is differentiable, allowing for a model based on attention to be trained from start to finish. You don't need to explicitly train the model to select the terms you want; it will figure out how to choose crucial information on its own and It is added individually to the attention layer. At each decoder step h_t , attention receives input from all encoder states ($s_1, s_2, s_3, \dots, s_t$), among others, and computes the attention scores. For the decoder state, attention determines the importance of each encoder state. In essence, it executes an attention function that receives input from a single encoder state and a single decoder state and produces a scalar value. The most often used techniques for determining attention scores are:

1. The simplest way is dot-product.
2. To more efficiently encode each source word, the encoder makes use of two RNNs that read input in the opposite directions forward and backward.
3. The attention score is calculated using a bi-linear function and a unidirectional encoder.

Additionally, deep bidirectional GRU with attention layer offers stronger expressiveness and learning capabilities. Attention layer is a suggested technique for simplifying the modelling of long-term dependency. A more direct relationship between the model state at various periods in time is made possible by adding this layer[12].

4.4 Experimental Setup

After collection of data the next step i.e data pre-processing is completed. Urdu pre-processing is a laborious process by itself, however an Urdu pre-processing library named Urduhack⁵ has made it quite simple. For setting hyperparameters of deep learning models, in depth study of literature is conducted and by implementing grid search a set of parameters were carefully selected. To obtain optimized results data set is divided into train and test sets using sklearn library of ML. To avoid overfitting of data the training set is further divided into validation set using DL library keras. Along with our proposed model Bi-GRU with attention layer for this task, we utilized DL baseline models namely LSTM, Bi-LSTM, TCN, and GRU to verify the performance of our model.

4.4.1 Sequence normalization

Since, all the text instances does not have same number of words we first normalize the sequences. For this task, the maximum number of words in one sequence from the whole dataset is calculated and then zeros are added in other sequences using zero padding technique.

4.4.2 Layering of Proposed model

The basic hierarchy of proposed model is discussed as follows:

1. Input Layer

The first layer of every DL network is the Input layer. After converting the pre-processed data into vectors and splitting them into training, test and validation sets, the training vectors then become the input of any neural network. They are identified as the input layer of a DL network.

2. Word Embedding Layer

In Input vector set each word is given a distinct ID and a meaningful sequence of words. The word embedding layer allows similar meaning words to have

⁵<https://pypi.org/project/urduhack/>

a resembling representation. This layer assign different weights to words randomly and it learns to embed words that are included the training data set eventually. The main purpose of this layer is to learn word embeddings to be used in other models in the future. The best part is that the word embeddings learned by this layer can be utilized in any other model studies. The pre-trained word embeddings used in this study are publicly available word2vec Urdu word embedding. The performance of model is analyzed with or without using word embedding.

3. **Bi-GRU Layer**

Different Neural network layers are added here depending the model under implementation. Bi-GRU layer is the next layer of our model its input is the embedding vectors and the number of neurons are selected by tuning the parameters.

4. **Attention Layer**

The states obtained from GRU are then fed into the Attention layer. This layer helps in focusing the best and most meaningful words.

5. **Dense Layer**

Dense layer is the most basic and simple layer added in a neural network. The parameters of dense layer that are used in our model are dropout for regularization that helps in dealing with over fitting issues and activation function, and learning rate to optimize the performance of model.

The details of optimized parameter settings used in our proposed architecture are given in Table 4.3.

In the following chapter we will briefly discuss the experiments carried out using variable data set size, different models applied and will evaluate their outcomes. We will provide a comparative analysis of impact of word embeddings on DL models. A detailed comparison of DL models will also be conducted.

Parameters	UrduInASmall	UrduInALarge
Activation Function	sigmoid	sigmoid
Dropout	0.5	0.5
Loss Function	binary crossentropy	binary crossentropy
Learning Rate	0.001	0.001
Optimizer	adam	adam

Table 4.3: Optimizing parameters used in both dataset.

Results and Discussion

There has been a rise of interest in DL models application for all type of multi-lingual NLP tasks. But still the existing literature lacks in detailed experimental exploration of DL models on Urdu language Inappropriate content detection.

This chapter briefly explore the methods used while implementing the suggested model. The results from proposed model and other baseline models on two datasets will be discussed in details. We will address the identified research gaps and analyze the results with the evaluation metrics used. The performance will be observed in both datasets and impact of word2vec word embedding will be studied.

5.1 Evaluation Metrics

The effectiveness of categorization models is frequently assessed by researchers using a variety of evaluation metrics. The Evaluation metrics used in our research are Precision, Recall, F1-score and accuracy.

A Precision metric counts how many correctly positive predictions were made. So the accuracy of class with minority can be measured using Precision. It is the ratio of :

$$Precision = \frac{TP}{TP + FP} \quad (5.1.1)$$

where TP stands for True Positives i.e correct positive predictions and FP stands for false positives i-e incorrectly predicted positive. And (TP + FP) indicates total positive predictions.

Recall measures the proportion of positives that are correctly predicted among all

possible positive predictions. Its ratio is:

$$Recall = \frac{TP}{TP + FN} \quad (5.1.2)$$

where FN stands for false negatives it occurs when the model incorrectly predicts the negative class. Precision and recall can be combined into one metric using F-Measure or F1-score, which covers both characteristics. Its can be measured as:

$$F1 - Score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (5.1.3)$$

Accuracy in classification problems is used widely, due to the fact that it is a single metric that summarises the performance of model. It is the ratio of predictions that are predicted correctly by the model. It is as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (5.1.4)$$

where TN are those predictions that are correctly predicted negatives by the model.

5.2 Experiments

Our proposed model BiGRU with attention layer was first trained on both datasets. It outperformed all other baseline DL models used in this study. The comparison of results is carried out by keeping in mind the size of dataset, evaluation measures and use of embedding layer. We run the experiments multiple times by altering the optimization parameters in order to achieve best results from each model for perfect comparison. For our proposed architecture, After repeated experiments we concluded to use the parameters presented in Table 4.3. Activation function ‘sigmoid’ is used for for binary classification as the output of this function always either ‘0’ or ‘1’. Similarly, loss function used was ‘Binary cross entropy’ is preferred for binary classification. Adam optimizer achieved best result as it handles noisy or sparse gradient problems much better than others. We found dropout rate ‘0.5’ ideal for our problem case as increasing or decreasing it does not improve our results. Using learning rate, helps the model to adpat to the problem quickly. In our case we used ‘0.001’ learning rate.

Model	Test Accuracy	Test Loss	F1-score	Precision	Recall
LSTM	0.779	0.432	0.783	0.758	0.810
Bi-LSTM	0.773	0.545	0.763	0.787	0.740
GRU	0.770	0.493	0.768	0.763	0.773
TCN	0.770	0.442	0.743	0.825	0.676
BiGRU-A	0.789	0.496	0.781	0.797	0.766

Table 5.1: Results Comparison of baseline Model with our proposed model on UrduInASmall dataset without using Word2Vec layer.

Model	Test Accuracy	Test Loss	F1-score	Precision	Recall
LSTM	0.726	0.483	0.726	0.715	0.783
Bi-LSTM	0.712	0.500	0.741	0.665	0.837
GRU	0.690	0.533	0.643	0.744	0.567
TCN	0.663	0.586	0.614	0.707	0.542
BiGRU-A	0.730	0.476	0.710	0.756	0.669

Table 5.2: Results Comparison of baseline Model with our proposed model on UrduInASmall dataset with using Word2Vec layer.

Model	Test Accuracy	Test Loss	F1-score	Precision	Recall
LSTM	0.827	0.385	0.828	0.833	0.823
Bi-LSTM	0.825	0.431	0.808	0.847	0.773
GRU	0.810	0.493	0.799	0.807	0.791
TCN	0.807	0.459	0.806	0.772	0.844
BiGRU-A	0.842	0.367	0.827	0.866	0.792

Table 5.3: Results Comparison of baseline Model with our proposed model on UrduInALarge dataset without using Word2Vec layer.

Model	Test Accuracy	Test Loss	F1-score	Precision	Recall
LSTM	0.748	0.511	0.752	0.708	0.803
Bi-LSTM	0.747	0.670	0.726	0.749	0.704
GRU	0.523	0.692	0.500	0.620	0.634
TCN	0.682	0.558	0.720	0.635	0.832
BiGRU-A	0.760	0.489	0.745	0.756	0.735

Table 5.4: Results Comparison of baseline Model with our proposed model on UrduInAlarge dataset with using Word2Vec layer.

5.3 Results comparison

Results obtained through our experiments are presented Table 5.1 - 5.4. The ‘Model’ column represents our baseline models and proposed model. Evaluation metrics are then presented in the next columns for respective DL models. Our suggested model BiGRU-A yeilds best performance i-e 84% accuracy as compared with other models. We will compare these outcomes on the basis of following three points:

- **Dataset size**

Table 5.1 and Table 5.2 represents the results obtained for small sized Urdu data set UrduInAsmall. Whereas, Table 5.3 and Table 5.4 represents the results obtained for large sized Urdu data set UrduInAlarge. If we compare the accuracy achieved by our suggested model for both dataset we can observe a visible increase in the value as the size of dataset increases. This shows that as we increase the training data, the DL model has more training examples to train. Hence, it can learn a lot better than the small data set training examples. Therefore, all the models yielded best accuracy performances for UrduInAlarge dataset when compared with UrduInAsmall dataset.

- **Effect of word embedding**

Since word embedding can more clearly show the relationship and information between words, its application has been researched to enhance model performance[46]. Table 5.1 - 5.4 shows the results from two datasets with or

without using word embedding word2vec layer. Our results shows that using word embeddings has yielded poor performances whereas it achieves best results without using word embedding. This is because the Inappropriate class in our dataset contains a lot of swear words that are not included in pre-trained Word2Vec word embedding [47].

- **Model Comparison**

On the basis of evaluation metrics shown in Table 5.1 - 5.4 visible difference in performance of our model can be observed for two datasets. Precision, Recall and F1 measure of our suggested model has out performed all other models overall. Also the loss of test data is calculated for all models and our model has performed well in this metric too.

Figure 5.1 compares and visualize the evaluation metrics of baseline model and suggested architecture.

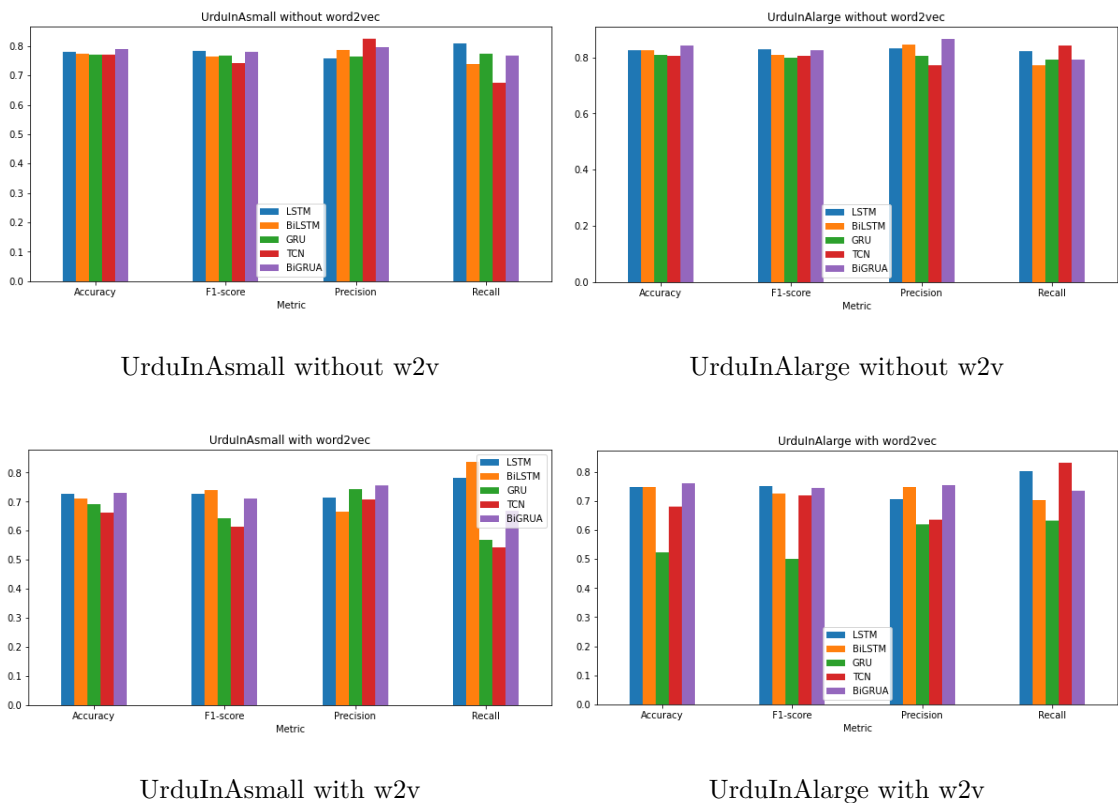


Figure 5.1: Evaluation Metrics Comparison of both data sets with or without Word2vec

5.4 Discussion on Proposed Model

The outcomes from all of the aggregations shown here demonstrate how the method described in this research offers a higher level of accuracy than the baseline models. Our hybrid Attention based Bidirectional GRU (BiGRU-A) presents best performance on both datasets with or without using word2vec layer. Figure 5.2 shows the accuracy comparison graphs of our proposed architecture. The first two graphs are the accuracy comparison of two datasets without using word embedding layer over 5 epochs. The last two graphs are the accuracy comparison of two datasets with using word embedding layer over 50 epochs. From figure 5.2 and Tables 5.1 - 5.4 it can be observed that the accuracy obtained in UrduInAlarge dataset has increased quite efficiently from the UrduInAsmall dataset.

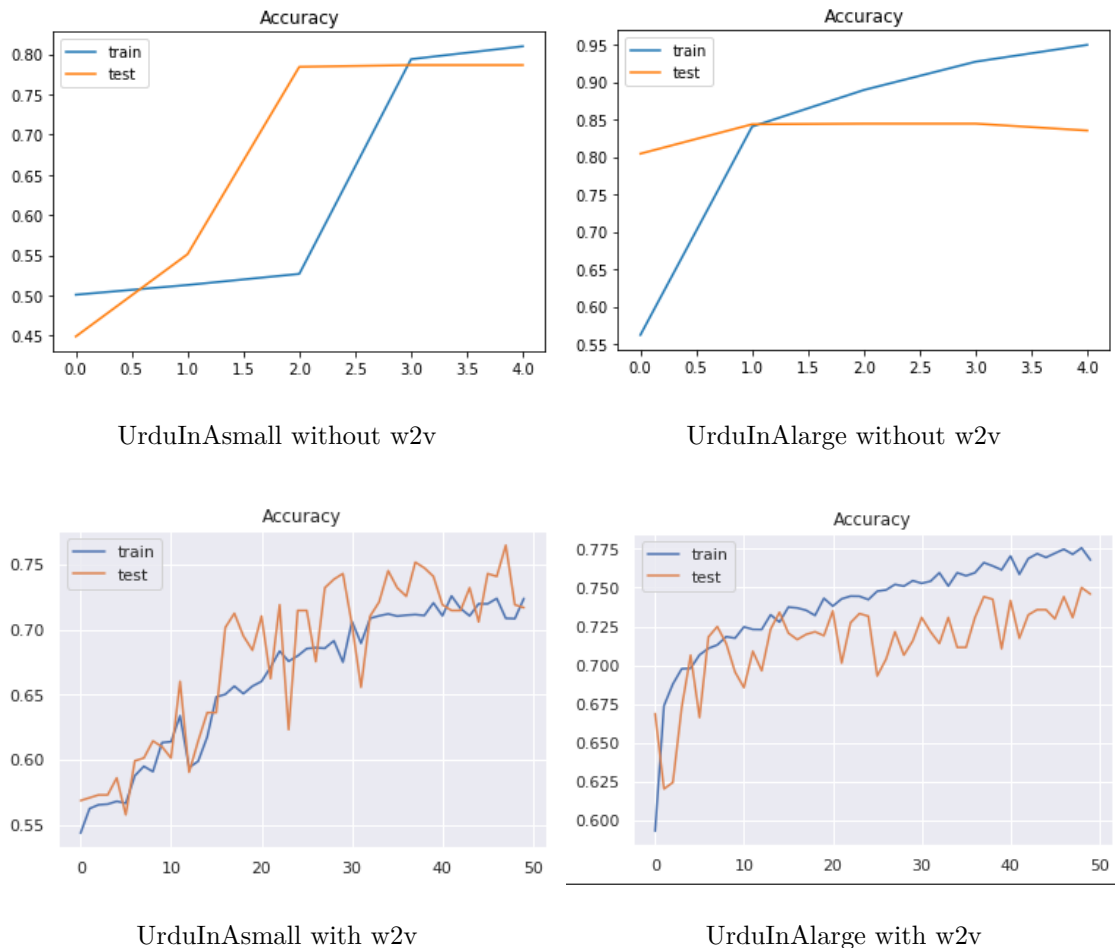


Figure 5.2: Accuracy Comparison of both data sets with or without Word2vec

To sum up, the technique utilised in the suggested hybrid model, BiGRU-A, is

relatively simple. No feature selection technique is engaged throughout the experimentation process, giving it an edge over other DL models proposed in many studies. Our tests clearly show that an attention layer can enable a model to grab specific important points in a sequence while computing its output. The attention layer also helps handling long and variable length sentences.

Hence our hybrid model not only reduces the cost and time of implementation but also enhance the performance of DL model. The completed research work makes a contribution in the domain of Inappropriate content identification in Urdu language using DL techniques.

In the last chapter of this work we will summarize what we have achieved so far. We will also point out the limitations of the conducted research and issues faced during this course of time. We will end our thesis with future work suggestions deduced with this work.

Conclusion and Future Work

The chapter summarizes the work presented through this research. It also briefly explains the issues and challenges unique to the Urdu language. The upcoming sections will describe the conclusion of this research.

6.1 Synopsis

Many studies have explored the field of automatic inappropriate content detection in European or English language. But very few studies have considered investigating inappropriate language detection in Urdu unicode text. With the studied literature we observed the major work is either done using Roman Urdu dataset or the dataset used has very small size. Also, in these studies mostly ML algorithms have been explored. Another identified gap is that due to the insufficient inappropriate content resources the datasets used have imbalanced class distribution which can create problems in identification of minority class.

This goal to conduct this study is to develop a larger sized balanced dataset that can be used for future studies. And to implement bidirectional Gated Recurrent Unit along with attention layer on variable sized datasets to detect Inappropriate content in Urdu unicode text language. The main purpose of this study was to not only draw attention of academics researchers on problem solving tasks in Urdu unicode text language but also to make a contribution in the literature of poor resource language Urdu. For this purpose, a thorough comparative analysis has been conducted with baseline deep learning models and the impact of word em-

bedding layer is also studied deeply. Furthermore, they are evaluated extensively using multiple evaluation metrics to make it a standard study in this field. The experimental results of our model revealed that it achieved overall high scores in every aspect no matter the dataset size variations and the use of word embedding layer. It yielded 84% accuracy without using word embedding layer. Through our research we have established that use of word embedding layer for inappropriate content detection decreases the efficiency of model as this dataset contains a lot of swear words that are not included in pre-trained word2vec embedding. Moreover, larger the size of dataset greater will be the performance of DL models.

6.2 Challenges of the Research

The collection and annotation of the Inappropriate content Urdu unicode text dataset was the major challenge faced in this research. When the Roman Urdu dataset was converted to Urdu script using online tool the issue of incorrect translation of misspelled words was faced. Because there are multiple ways of writing an Urdu word in Roman Urdu. People on social media not only have their own perspective of spelling but also used short forms as they wish. Human annotators can identify the real meaning behind short forms but there is a limit to what can be detected by automatic systems. We tried to correct the misspelled Urdu words translated from Roman Urdu to the best of our knowledge to improve the accuracy of the dataset.

While training the DL models choosing the best number of neurons in the dense layer, number of dense layer and optimization of parameters is tedious task for implementing a numbers of models to solve multiple tasks. The primary drawback is the potential for each model's training to proceed slowly. Each model must be trained once for each potential set of parameters in order to determine the best parameters. Since it is difficult to test every potential combination, we move on to a few sets that ought to work well. To achieve best accuracy performance of all models to perfectly compare with the proposed models different combination of parameters, layers and neurons are utilized to finally conclude this study.

6.3 Limitations

Through our experiments we can highlight the following limitations of this study:

- The word2vec word embeddings can be fine-tuned further to our dataset to improve the accuracy of DL model.
- The size of dataset can be increased further.

6.4 Future Work

From the experienced limitation we plan to improve our work by fine-tuning the word2vec word embedding to our dataset. We will also try different available word embeddings to further study the impact of embedding layer. The dataset size can be increased further for analysis in this domain. Many subtasks can be added to enhance the work with this dataset for example Inappropriate content can be further divided into sub classes to identify what level of harm is intended in the text. It can be used to list out the content that should be blocked or censored in Urdu language.

Moreover, this study can be advanced further by implementing transfer learning approaches like BERT, Transformer and additional analysis can be conducted in the future.

6.5 Applications

Automatic Inappropriate Content detection can be applied to solve real life problems such as:

- By media regulatory authorities and social networking sites to monitor the type content broadcasts through social media
- Prevention of cyber bullying content.
- Control of spread of violent and derogatory content on time before its too late to stop.

Bibliography

- [1] Sepp Hochreiter and Jürgen Schmidhuber. “Long short-term memory.” In: *Neural computation* 9.8 (1997), pp. 1735–1780.
- [2] Mike Schuster and Kuldip K Paliwal. “Bidirectional recurrent neural networks.” In: *IEEE transactions on Signal Processing* 45.11 (1997), pp. 2673–2681.
- [3] Ronan Collobert et al. “Natural language processing (almost) from scratch.” In: *Journal of machine learning research* 12.ARTICLE (2011), pp. 2493–2537.
- [4] Alex Graves. “Supervised sequence labelling.” In: *Supervised sequence labelling with recurrent neural networks*. Springer, 2012, pp. 5–13.
- [5] Kashif Riaz. “Comparison of Hindi and Urdu in computational context.” In: *Int J Comput Linguist Nat Lang Process* 1.3 (2012), pp. 92–97.
- [6] Tomas Mikolov et al. “Efficient estimation of word representations in vector space.” In: *arXiv preprint arXiv:1301.3781* (2013).
- [7] Marco Baroni, Georgiana Dinu, and Germán Kruszewski. “Don’t count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors.” In: *ACL*. 2014.
- [8] Junyoung Chung et al. “Empirical evaluation of gated recurrent neural networks on sequence modeling.” In: *arXiv preprint arXiv:1412.3555* (2014).
- [9] Hasim Sak, Andrew W Senior, and Françoise Beaufays. “Long short-term memory recurrent neural network architectures for large scale acoustic modeling.” In: (2014).
- [10] Njagi Dennis Gitari et al. “A lexicon-based approach for hate speech detection.” In: *International Journal of Multimedia and Ubiquitous Engineering* 10.4 (2015), pp. 215–230.

- [11] Jonathan Long, Evan Shelhamer, and Trevor Darrell. “Fully convolutional networks for semantic segmentation.” In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 3431–3440.
- [12] Colin Raffel and Daniel PW Ellis. “Feed-forward networks with attention can solve some long-term memory problems.” In: *arXiv preprint arXiv:1512.08756* (2015).
- [13] Tehseen Zia, Muhammad Pervez Akhter, and Qaiser Abbas. “Comparative study of feature selection approaches for Urdu text categorization.” In: *Malaysian Journal of Computer Science* 28.2 (2015), pp. 93–109.
- [14] Kashif AHMED et al. “Framework for Urdu News Headlines Classification.” In: *Journal of Applied Computer Science Mathematics* 10 (Apr. 2016), pp. 17–21.
- [15] Mondher Bouazizi and Tomoaki Otsuki Ohtsuki. “A pattern-based approach for sarcasm detection on twitter.” In: *IEEE Access* 4 (2016), pp. 5477–5488.
- [16] Yashar Mehdad and Joel Tetreault. “Do characters abuse more than words?” In: *Proceedings of the 17th annual meeting of the special interest group on discourse and dialogue*. 2016, pp. 299–303.
- [17] JD Bermúdez et al. “Evaluation of recurrent neural networks for crop recognition from multitemporal remote sensing images.” In: *Anais do XXVII Congresso Brasileiro de Cartografia*. 2017, pp. 800–804.
- [18] Azalden Alakrot, Liam Murray, and Nikola S. Nikolov. “Towards Accurate Detection of Offensive Language in Online Communication in Arabic.” In: *ACLING*. 2018.
- [19] Shaojie Bai, J Zico Kolter, and Vladlen Koltun. “An empirical evaluation of generic convolutional and recurrent networks for sequence modeling.” In: *arXiv preprint arXiv:1803.01271* (2018).
- [20] Muhammad Okky Ibrohim and Indra Budi. “A dataset and preliminaries study for abusive language detection in Indonesian social media.” In: *Procedia Computer Science* 135 (2018), pp. 222–229.
- [21] Ho-Suk Lee et al. “An abusive text detection system based on enhanced abusive and non-abusive word lists.” In: *Decis. Support Syst.* 113 (2018), pp. 22–31.
- [22] Younghun Lee, Seunghyun Yoon, and Kyomin Jung. “Comparative studies of detecting abusive language on twitter.” In: *arXiv preprint arXiv:1808.10245* (2018).

- [23] Imran Rasheed et al. “Urdu text classification: a comparative study using machine learning techniques.” In: *2018 Thirteenth International Conference on Digital Information Management (ICDIM)*. IEEE. 2018, pp. 274–278.
- [24] Harish Yenala et al. “Deep learning for detecting inappropriate content in text.” In: *International Journal of Data Science and Analytics* 6.4 (2018), pp. 273–286.
- [25] Steven Zimmerman, Udo Kruschwitz, and Chris Fox. “Improving hate speech detection with deep learning ensembles.” In: *Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018)*. 2018.
- [26] Vimala Balakrishnan et al. “Cyberbullying detection on twitter using Big Five and Dark Triad features.” In: *Personality and Individual Differences* (2019).
- [27] Puja Chakraborty and Md Hanif Seddiqui. “Threat and abusive language detection on social media in bengali language.” In: *2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)*. IEEE. 2019, pp. 1–6.
- [28] Yangdong He and Jiabao Zhao. “Temporal convolutional networks for anomaly detection in time series.” In: *Journal of Physics: Conference Series*. Vol. 1213. 4. IOP Publishing. 2019, p. 042050.
- [29] Thomas Mandl et al. “Overview of the hasoc track at fire 2019: Hate speech and offensive content identification in indo-european languages.” In: *Proceedings of the 11th forum for information retrieval evaluation*. 2019, pp. 14–17.
- [30] Pushkar Mishra et al. “Abusive language detection with graph convolutional networks.” In: *arXiv preprint arXiv:1904.04073* (2019).
- [31] Priya Rani and Atul Kr Ojha. “KMI-coling at SemEval-2019 task 6: exploring N-grams for offensive language detection.” In: *Proceedings of the 13th International Workshop on Semantic Evaluation*. 2019, pp. 668–671.
- [32] Anna Schmidt and Michael Wiegand. “A survey on hate speech detection using natural language processing.” In: *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media, April 3, 2017, Valencia, Spain*. Association for Computational Linguistics. 2019, pp. 1–10.
- [33] Sudha Subramani et al. “Deep learning for multi-class identification from domestic violence online posts.” In: *IEEE Access* 7 (2019), pp. 46210–46224.

BIBLIOGRAPHY

- [34] Marcos Zampieri et al. “Predicting the Type and Target of Offensive Posts in Social Media.” In: *NAACL*. 2019.
- [35] Marcos Zampieri et al. “SemEval-2019 Task 6: Identifying and Categorizing Offensive Language in Social Media (OffensEval).” In: **SEMEVAL*. 2019.
- [36] Muhammad Pervez Akhter et al. “Automatic detection of offensive language for urdu and roman urdu.” In: *IEEE Access* 8 (2020), pp. 91213–91226.
- [37] Amitha Mathew, P Amudha, and S Sivakumari. “Deep learning techniques: an overview.” In: *International conference on advanced machine learning technologies and applications*. Springer. 2020, pp. 599–608.
- [38] Muhammad Nabeel Asim et al. “Benchmark Performance of Machine And Deep Learning Based Methodologies for Urdu Text Document Classification.” In: *arXiv e-prints* (2020), arXiv–2003.
- [39] Ali Nawaz et al. “Extractive text summarization models for Urdu language.” In: *Information Processing & Management* 57.6 (2020), p. 102383.
- [40] Hammad Rizwan, Muhammad Haroon Shakeel, and Asim Karim. “Hate-speech and offensive language detection in roman Urdu.” In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2020, pp. 2512–2522.
- [41] Tauqeer Sajid et al. “Roman Urdu Multi-Class Offensive Text Detection using Hybrid Features and SVM.” In: *2020 IEEE 23rd International Multitopic Conference (INMIC)*. 2020, pp. 1–5.
- [42] Gudbjartur Ingi Sigurbergsson and Leon Derczynski. “Offensive Language and Hate Speech Detection for Danish.” In: *LREC*. 2020.
- [43] Maaz Amjad et al. “Threatening language detection and target identification in Urdu tweets.” In: *IEEE Access* 9 (2021), pp. 128302–128313.
- [44] Judith Jeyafreeda Andrew. “JudithJeyafreedaAndrew@DravidianLangTech-EACL2021: offensive language detection for Dravidian code-mixed YouTube comments.” In: *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*. 2021, pp. 169–174.

BIBLIOGRAPHY

- [45] Anum Ilyas, Surayya Obaid, and Narmeen Zakaria Bawany. “Multilevel Classification of Pakistani News using Machine Learning.” In: *2021 22nd International Arab Conference on Information Technology (ACIT)*. IEEE. 2021, pp. 1–5.
- [46] Teng Jinbao et al. “Text classification method based on BiGRU-attention and CNN hybrid model.” In: *2021 4th International Conference on Artificial Intelligence and Pattern Recognition*. 2021, pp. 614–622.
- [47] Lal Khan et al. “Urdu sentiment analysis with deep learning methods.” In: *IEEE Access* 9 (2021), pp. 97803–97812.
- [48] Sajadul Hassan Kumhar et al. “Word embedding generation for Urdu language using Word2vec model.” In: *Materials Today: Proceedings* (2021).
- [49] Yogesh Yadav et al. “A Comparative Study of Deep Learning Methods for Hate Speech and Offensive Language Detection in Textual Data.” In: *2021 IEEE 18th India Council International Conference (INDICON)*. IEEE. 2021, pp. 1–6.
- [50] Qing Yu, Ziyin Wang, and Kaiwen Jiang. “Research on text classification based on bert-bigru model.” In: *Journal of Physics: Conference Series*. Vol. 1746. 1. IOP Publishing. 2021, p. 012019.
- [51] Muhammad Pervez Akhter et al. “Exploring deep learning approaches for Urdu text classification in product manufacturing.” In: *Enterprise Information Systems* 16.2 (2022), pp. 223–248.
- [52] Raza Ali et al. “Hate speech detection on Twitter using transfer learning.” In: *Computer Speech Language* 74 (2022), p. 101365.
- [53] Parisa Hajibabae et al. “Offensive language detection on social media based on text classification.” In: *2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC)*. IEEE. 2022, pp. 0092–0098.
- [54] Muhammad Aasim Qureshi et al. “Sentiment analysis of reviews in natural language: Roman Urdu as a case study.” In: *IEEE Access* 10 (2022), pp. 24945–24954.