A Dual Attention-Based Representation for the Detection of Abusive Language in Texts and Memes.

Technical Report No. CCC-22-005

by

Horacio Jesús Jarquín Vásquez

Doctoral Advisors:

Dr. Hugo Jair Escalante Balderas, INAOE
Dr. Manuel Montes y Gómez, INAOE

Instituto Nacional de Astrofísica, Óptica y Electrónica
©Coordinación de Ciencias Computacionales

June, 2022
Santa María de Tonantzintla, Puebla, CP 72840

Abstract

The use of attention mechanisms in deep learning approaches has become popular within the computer vision and natural language processing communities, due to its outstanding performance. The use of these mechanisms allows managing the importance of the elements of a sequence in accordance to their context, however, this importance has been observed independently between the pairs of elements of a sequence (self-attention) and between the application domain of a sequence (contextual attention), leading to the loss of relevant information and limiting the representation of the sequences. To tackle these particular issues, we propose a dual attention mechanism, which trades off the previous limitations, by considering the internal and contextual relationships between the elements of a sequence. By combining these two approaches, we will explore the performance of the proposed mechanism in the abusive language detection task in text and memes, where the management of internal and contextual relationships between pairs of words and image regions are essential for the detection of abusive language. As preliminary results, we proposed a new dual attention mechanism called: Self-Contextualized Attention (SCA); as a first step, the proposed mechanism was evaluated in the detection of abusive language in text, through the integration of the SCA mechanism into two well-performing deep neural network architectures. The preliminary results show a better performance with the use of the SCA mechanism, compared to the independent use of the current attention mechanisms, in addition to this, competitive results were obtained with respect to state-of-the-art approaches.

Keywords: Deep Learning, Attention Mechanisms, Abusive Language Detection

Contents

1 Introduction

This technical report No. CCC-22-005 presents my doctoral dissertation proposal.

The integration of social media platforms into the everyday lives of billions of users has increased the number of online social interactions, promoting the creation of a wide variety of content, as well as the exchange of different opinions and points of view that would otherwise be ignored by traditional media. The use of these social media platforms has revolutionized the way people communicate and share information. Unfortunately, not all of these interactions are constructive for all users, as the presence of Abusive Language (AL) has spread to these media. The rise of this phenomenon also seems to be influenced by anonymity given to users and the lack of effective regulation provided by social media platforms [1].

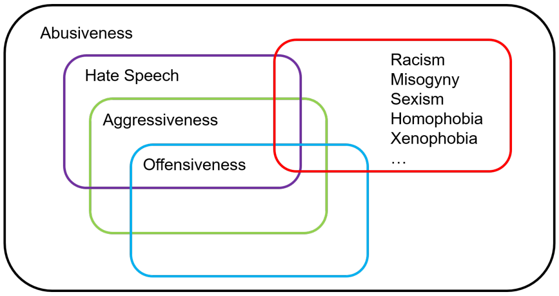


Figure 1: General taxonomy in abusive language phenomena. Figure inspired by [2].

Although there is no global consensus on the definition of terms such as: Hate Speech (HS) and AL, in this work we use the definition of AL as: *verbal messages which use harsh, rude, offensive, and/or insulting words in an inappropriate way and which may also include profanity and slurs to demean the dignity of an individual or group of people* [3]. The AL term is usually used as an umbrella expression, which covers several related phenomena from the use of simple obscene and profanities to the use of threats and severe insults [4]. In recent studies [2], the relationship between different phenomena such as: HS, offensive language, aggressiveness, abusiveness/toxicity, and the other manifestations of hatred towards certain targets such as misogyny, racism, and homophobia, has been mapped. Figure 1 shows a taxonomy of these relationships, as it can be seen, the AL contains the aforementioned phenomena. Specifically, in this work we will focus on the detection of AL in social media.

Due to its negative social impact [5], the automatic detection of AL has stimulated the interest of supplier companies and governments. On the one hand, there is the direct damage to users who are target of AL, where they can present some psychological damage, and even in extreme cases commit suicide. On the other hand, there is the indirect damage to society, which causes the deterioration of the public discourse, leading to the creation of a more polarized society [6, 7]. The detection of AL in social media is not an easy task; the use of content filters and moderators is far from being a good and sustainable solution to this problem, due to the large amount of information that is currently generated on these platforms. Derived from this, multiple efforts have been made to combat the proliferation of AL, ranging from the establishment of norms and regulations of the content posted on social media [8], to the use of Machine Learning (ML) for the computational analysis of information from social media [9].

Regarding the norms and regulations, different countries have restricted the publication of potentially offensive content. For example, the European Union in conjunction with different social media platforms such as: Facebook[1], Twitter[2], Microsoft and YouTube have recently signed a code of conduct[3], pledging to review the majority of valid notifications for removal of illegal AL content in less than 24 hours. However, these efforts are not a scalable and long-term solution to this problem. On the other hand, the ML approaches for the detection of AL have been addressed from a supervised perspective, where most of these approaches have focused on the detec-

[1] https://time.com/5739688/facebook-hate-speech-languages/

[2] https://help.twitter.com/en/rules-and-policies/hateful-conduct-policy

[3] http://ec.europa.eu/justice/fundamental-rights/files/hate_speech_code_of_conduct_en.pdf

tion of AL in text [2]. Predominantly, the Natural Language Processing (NLP) has approached this task from the perspective of the computational language analysis. A great variety of methods have been proposed, ranging from the use of bags of words (BoW) and traditional classification approaches [10, 6], to the use of Deep Learning (DL), Attention Mechanisms (AM), and Transformer-based neural language models, which constitute the state-of-the-art [11, 12]. Despite the encouraging results in the AL detection task, most of the current approaches has been mainly focused on the analysis of textual information. This presents a limitation, since the social media content has a multimodal nature, including the use of: audio, text, image, and video.

One of the most common examples of multimodal information found in social media are the memes [13], which are defined as: *the conjunction of a text and an image, which provides a meaning (mostly humorous or ironic) and that the absence of one of its elements (text or image) will not result in the same interpretation* [14]. Despite its daily use in humorous and ironic publications, the use of memes to transmit AL transcends the social media platforms [15]. The detection of AL in memes presents a challenging task, since the interpretation of a meme is highly dependent on both, the text and image. With the intention of taking a step further, in this work we will focus on the detection of AL in text and memes, from the perspective of a deep representation of language, and a deep joint representation of vision and language, respectively.

In this research proposal we aim to make the following contributions: 1) Novel attention-based mechanisms, which model the alignment of features based on the contextual relationships between the pairs of elements of a sequence. Considering pairs of words (text), as well as pairs of image regions, and words (memes). 2) The creation of a deep representation of language, and a deep joint representation of vision and language based on the previously proposed attention mechanisms; in order to automatically align the contextual relationships between the elements of a sequence. 3) The integration of deep feature fusion approaches, in order to combine and unify different modality features with the previously proposed representations. The proposed contributions will be evaluated in the detection of AL in text and memes, using different publicly available datasets.

The remainder of this document is organized as follows: Section 2 introduces a brief discussion about the related work to the detection of AL in text and memes. Section 3 presents the research proposal, which covers the problem statement, hypothesis, research questions, objectives, expected contributions, and scope and limitations. Section 4 contains the Methodology to accomplish the expected contributions and objectives proposed in the previous section. Section 5 presents the work schedule, which illustrates the different steps to finish the dissertation. Section 6 includes the preliminary work to support this dissertation proposal, and present the preliminary conclusions and published papers derived from this work. Finally, Section 7 presents and describes in detail the background concepts and techniques needed for this dissertation proposal.

2 Related Work

This section presents an analysis of the previous related work, as well as different techniques and approaches, for the detection of AL in social media. This section is divided three-folded, the first subsection covers the unimodal and multimodal approaches for the detection of AL in social media. The second subsection presents some of the most relevant evaluation campaigns related to the AL detection in social media. Finally, the third subsection discusses the current limitations of the main approaches in the detection of AL in unimodal and multimodal scenarios.

2.1 Abusive language detection in social media

A wide variety of works related to the detection of AL focus on the detection of sexist, racist, hateful, aggressive, and offensive social media content [6, 16]. Most of these works are approached from a supervised perspective; with the use of different data preprocessing techniques, the use of different representations and a wide variety of traditional machine learning, as well as, deep learning approaches [9]. The following subsections, describe the most relevant approaches related to the detection of AL from a unimodal (text) and multimodal (memes) perspective.

2.1.1 Detection of abusive language in text

Different methods have been proposed to detect AL by using textual information. The proposed approaches range from traditionally NLP-based ones, to the use of deep learning based models; which constitute the state of the art in this particular task [9, 7]. A great variety of features have been used to tackle this problem; initial approaches used bag-of-words representations, considering word and character n-grams as input features [17, 18, 19, 20]. Aiming to improve the generalization of the classifiers, other approaches have also considered word embeddings as features

[18, 21, 11]. Recently, some approaches have used sophisticated text representations by applying pre-trained Transformer-based neural language models, and deep contextualized word representations models, such as: ELMO [22], GPT-2 [23], and BERT [24], and fine-tune their parameters to the AL detection task [25, 26, 27, 12, 28].

Regarding the classification stage, different approaches and techniques have also been proposed. These approaches could be divided in two categories; the first category relies on traditional classification algorithms such as Support Vector Machines (SVM), Naive Bayes, Logistic Regression and Random Forest [19, 17, 29, 9, 20, 6, 30]; on the other hand, the second category includes deep learning based approaches, which employ Convolutional Neural Networks (CNN) for word and character based feature extraction [31, 32, 33, 34], Recurrent Neural Networks (RNN) for word and character dependency learning [31, 35, 36, 11, 33], and the combination of both for creating powerful structures that capture order information between the extracted features [21, 37, 38].

Recent works in the AL detection task have considered deep learning approaches with the use of attention mechanisms, providing models with the ability to automatically weight features [39]. One of the first works introducing attention into the task employed the self-attention mechanism to detect abuse in portal news and Wikipedia [40]. Subsequently, the use of the contextual attention mechanism, introduced by [41] has shown encouraging results in this task [11, 42, 43], by allowing the improvement of the sentence representation. Finally, the use of the aforementioned Transformers has become popular in recent years, and it constitutes the current state of the art in this task [44, 45].

2.1.2 Detection of abusive language in memes

The detection of AL has been approached mainly with the use of textual resources [9, 7]. In recent years there has been an interest in extending the detection of

AL to a multimodal perspective; one of the phenomena that best describes these perspectives is the detection of abusive memes in social media, due to the integration of textual and visual information that together generate a context [46]. The memes classification task can be seen as a combined Vision & Language (VL) multimodal problem, where a wide variety of approaches have been proposed based on: 1) the fusion of multimodal features, and 2) the use of pre-trained multimodal models [47, 13]. Figure 2 presents a general scheme for the detection of AL in memes based on the aforementioned approaches.

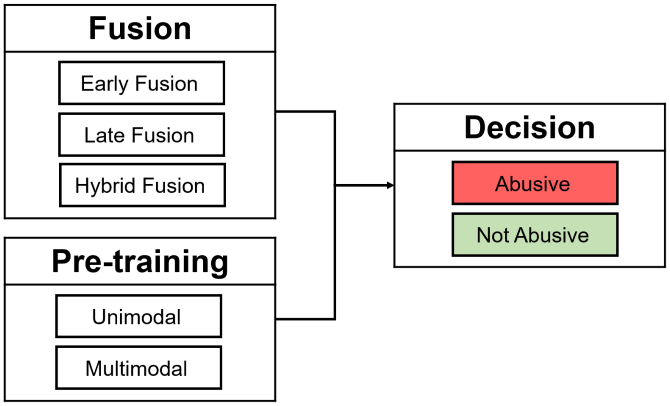


Figure 2: General scheme for the detection of AL in memes.

Regarding the proposed approaches for the detection of AL in memes, a great number of them follow the classification scheme shown in Figure 2. Concerning the fusion-based approaches, a wide variety of them have used different techniques based on early, late, and hybrid fusion of multimodal features from image and text [48, 49, 50, 51, 52]. Most of these VL features are extracted from unimodal pre-trained models such as: BERT [24], ELMO [22], GPT-2[23] for the linguistic features, and AlexNet [53], VGG [54], GoogLeNet [55] for the visual features. On the other hand, other approaches have opted for the use of pre-trained multimodal models, which use joint multimodal information (text and image) [15, 56, 52, 57]; the latter have shown to have a better performance in the detection of AL, since their training considers both modalities (text and image) simultaneously, which allows a better alignment between them [14, 58]. Within the approaches based on the use of pre-trained multimodal models, the most used representations have been those based on transformers, such as: ViLBERT [59], VisualBERT [60] and VL-BERT [61].

2.2 Evaluation campaigns for the abusive language detection in social media

Considering the well-acknowledged increase of AL on social media platforms, a wide number of workshops and evaluation campaigns have been proposed in order to mitigate the impact of such kind of content [2]. Most of this content is based on the detection of abusive messages. In 2018 the first workshop on *Trolling, Aggression and Cyberbullying (TRAC-1)*[4] was held at the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), which aims to classify Overtly Aggressive, Covertly Aggressive and Non-aggressive text data [5]. Likewise, the *Automatic Misogyny Identification (AMI)*[5] task was proposed as part of the Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2018), and focuses in the automatic identification of misogynous content in English and in Italian tweets [62]. Finally, in 2019 and 2020 the OffensEval shared-tasks on *Identifying and Categorizing Offensive Language in Social Media*[6][7] were presented at the International Workshop on Semantic Evaluation (SemEval), which focuses in the offensive language identification, the automatic categorization of offense types, and the offense target identification [63, 64].

Regarding the detection of AL in memes, in recent years there has been an increasing interest in the community, and supplier companies to mitigate this problem.

[4] https://sites.google.com/view/trac1/shared-task

[5] https://amievalita2018.wordpress.com/

[6] https://competitions.codalab.org/competitions/20011

[7] https://competitions.codalab.org/competitions/22917

For example, in 2020 Facebook held the contest: *Hateful Memes Challenge*[8], which focuses on the detection of hate speech in memes, with the use of a 10k manually annotated dataset [13]. In that same year, in the 14th International Workshop on Semantic Evaluation (SemEval-2020), it was proposed the *Memotion Analysis*[9] shared-task; which aims to identify sarcastic, humorous, and offensive memes [14]. Finally, as part of the 16th International Workshop on Semantic Evaluation (SemEval-2022), the *Multimedia Automatic Misogyny Identification (MAMI)*[10] shared-task was proposed. This particular task has two main sub-tasks, a basic task (Sub-task A) about misogynous meme identification, and an advanced task (Sub-task B), where the type of misogyny should be recognized among potential overlapping categories such as stereotype, shaming, objectification and violence.

2.3 Discussion

A wide variety of approaches have been proposed for the detection of AL in text and memes, ranging from the use of traditional fusion and classification techniques [9, 6, 2, 16], to the use of Deep Learning (DL) based ones [13, 65, 44, 45, 16]. Due to the powerful representation ability with multiple levels of abstraction, DL-based representations has excelled in recent years [66, 60]. Within the DL multimodal representations, the vision & language representations have highlighted in recent years. The use of the Transformer Neural network (TNN) architecture [67] has been applied in this context to create pre-trained vision & language representations, aiming to capture the relationships between the pairs of image regions and words, through the self-attention mechanism, thus obtaining cutting-edge results for a wide variety of vision & language tasks, including the detection of AL in memes [61, 59, 60]. Regarding the detection of AL in text, DL-based approaches constitute the state

[8] https://ai.facebook.com/blog/hateful-memes-challenge-and-data-set/

[9] https://competitions.codalab.org/competitions/20629

[10] https://competitions.codalab.org/competitions/34175

of the art [65, 16]. Specifically, the use of pre-trained language models (e.g. BERT, and GPT-2) based on the TNN architecture, has become popular over the last few years, due to: 1) the fine tuning strategy which facilitates the training process, and 2) the use of the self-attention mechanism, which allows modeling the relationships between pairs of words [27, 12].

Despite the encouraging results obtained with the use of Transformer-based pre-trained representations in unimodal and multimodal scenarios, different areas of improvement have been reported regarding these pre-trained models [68, 69, 70]. Among these, the most prominent ones involves: 1) the handling of missing data (e.g. out vocabulary words), and 2) the lack of contextual information in domain-specific tasks, such is the case of the detection of AL. These particular issues are generated since these pre-trained models are trained with general-purpose datasets and pre-trained tasks [24, 60, 61], which limits the self-attention mechanisms in the extraction of relationships between different features. Some approaches have addressed these issues by training models with domain-specific data [71, 68]; for example, in regards to the detection of AL in text, in [69] the HateBERT pre-trained model was proposed, which was re-trained with potentially offensive social media data. However, the creation of these pre-trained models requires a large amount of data and a high computational cost [23, 59, 68].

3 Research Proposal

This section presents in detail the research proposal. In the first part we introduce the motivation and justification, as well as the problem statement; the next part presents the hypothesis, the research questions, the objectives, and the expected contributions. In the final part we present the scope and limitations of this dissertation proposal.

3.1 Motivation and Justification

The use of TNN-based pre-trained models constitutes the current state of the art in the detection of AL in unimodal (text) and multimodal (memes) scenarios [72, 6, 46, 70, 16]. The outstanding performance of TNN-based architectures is mainly due to the use of the Self Attention (SA) mechanism [39], which is used to capture the internal relationships between the elements of a sequence. Specifically, the use of SA in the deep representation of language captures the internal relationships between each pair of words in a given sequence, on the other hand, the use of SA in the deep representation of vision & language captures the internal relationships between each word and the image regions of a given image and text pair. This is very useful in the application domain of this doctoral research proposal, since the use of SA allows modeling the internal relations between the elements of the texts and memes. In this work we will focus on the extension of the SA mechanism, applied to pre-trained language models, as well as, vision & language models.

3.2 Problem Statement

The use of AMs has gained relevance within the DL approaches, due to the ability they provide to the classification models to focus on a subset of inputs (or features), as well as their capability to model long-term dependencies between the elements of a sequence [73]. The use of AMs has been applied to a large number of DL architectures, obtaining cutting-edge results in a wide variety of domains. According to [39] AMs can be classified according to the input domain into two main categories: 1) the SA mechanisms, and 2) the Contextual Attention (CA) mechanisms. Despite their outstanding results, two main limitations have been identified in the use of both AMs. On the one hand, the CA mechanisms ignores the internal relationships between the elements of a sequence, on the other hand, the SA mechanisms does not consider the global relationships within the elements of different sequences, which causes the loss of relevant information in the application domain (training task). Clearly, these limitations are complementary.

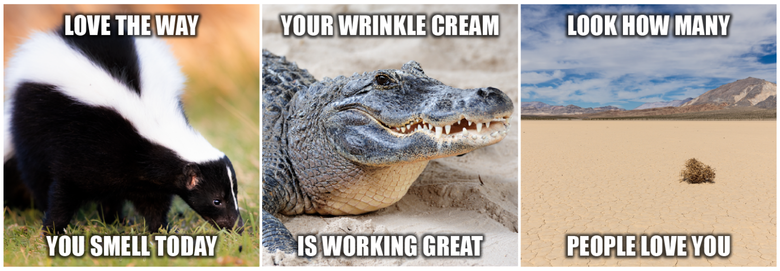


Figure 3: Examples of offensive memes. Figure taken from: [13]

The integration of the SA mechanisms in DL architectures (e.g. TNN), has become popular in domains where the correct interpretation of an instance strongly depends on the internal context of the elements of a sequence (local dependencies); which is not always the case in the detection of AL in text and memes. This unique integration present a limitation when incorporating relevant information related to the application domain (global dependencies), which generates the loss of relevant information to improve the image and text representations. An example of this can be seen in the memes shown in Figure 3, where the integration of a global context is necessary to improve the interpretation of the instances. Considering the meme on the left-hand side, it is necessary to know that the smell of a skunk is strong, to correctly interpret the irony of the meme. As a strategy to overcome this issue, in this work we focus on the extension of the SA mechanisms by proposing novel approaches that integrate the CA mechanisms. Since these approaches combine both, the SA and CA mechanisms, we will refer to them as: dual attention mechanisms. The proposed mechanisms will be integrated into a variety of DL architectures, and evaluated in the AL detection task in text and memes.

3.3 Hypothesis

The integration of the SA and CA mechanisms in pre-trained deep representations of vision & language, could improve the performance of the AL detection task in text and memes, generating a model that is both modular and scalable, as well as easy to train in terms of the number of parameters to optimize.

3.4 Research Questions

This doctoral research proposal raises the following questions:

1. How to integrate the SA and CA mechanisms for the representation of local and global dependencies?
2. Which similarity measures and loss functions obtain the best performance results in the integration of the SA and CA mechanisms?
3. Which DL architectures best fit with the proposed AMs, in terms of the best performance results in the detection of AL in text and memes?
4. What are the most relevant textual and visual features for the deep representation of text and images?

3.5 Objectives

The aim of this doctoral research work is:

Develop a novel dual attention mechanism based on the integration of the CA mechanism into the SA mechanism, in order to extract the internal and contextual relationships between the elements of a sequence. Subsequently, evaluate the proposed mechanism in the detection of AL in text and memes, with the use of a variety of DL architectures and improve the results obtained by traditional and state of the art approaches.

The specific objectives are listed below:

1. Propose a novel dual attention mechanism based on the integration of the SA and CA mechanisms, which integrates the extraction of the internal and contextual relationships within the elements of a sequence.
2. Integrate the proposed attention mechanism in a variety of standard and well-known DL architectures, including pre-trained models based on the TNN architecture, in unimodal (text) and multimodal (text and image) scenarios for the detection of AL in text and memes.
3. Develop a method for the detection of AL in memes that allows the dynamic incorporation of the relevance of each modality with the proposed dual attention mechanism.
4. Evaluate qualitatively and quantitatively the effectiveness of the proposed dual attention mechanism in a wide variety of AL datasets for text and memes.

3.6 Expected Contributions

Through this doctoral research are expected to obtain the following contributions, where the first two contributions are the most important:

1. New approaches based on the integration of the SA and CA mechanisms (dual attention mechanisms), which model the contextual relationships between the pairs of elements of a sequence. Considering all the pairs of words in a sequence (text), as well as the different image regions and the pairs of words (memes).
2. The creation of a language and vision & language representation based on the proposed dual attention mechanisms, with the aim of automatically modeling the contextual relationships between the elements of a sequence.
3. The integration of different feature fusion approaches, in order to unify in a weighted approach the different modalities in the previously proposed representations (vision & language).
4. have a better understanding of the benefits that unimodal and multimodal AMs can have in the detection of AL in text and memes.

3.7 Scope and Limitations

This research work covers the design, implementation and evaluation of the proposed dual attention mechanisms, as well as the deep-based language and vision & language representations. The proposed contributions will be evaluated on the detection of AL in text and memes, using different publicly available English datasets, as well as a variety of collections presented in different evaluation forums.

4 Methodology

This section describes the proposed methodology to achieve the objectives proposed in this doctoral research proposal. Figure 4 presents the outline of the proposed methodology, which consists of 4 major stages. The proposed methodology is designed under an iterative approach, between each of the different phases.

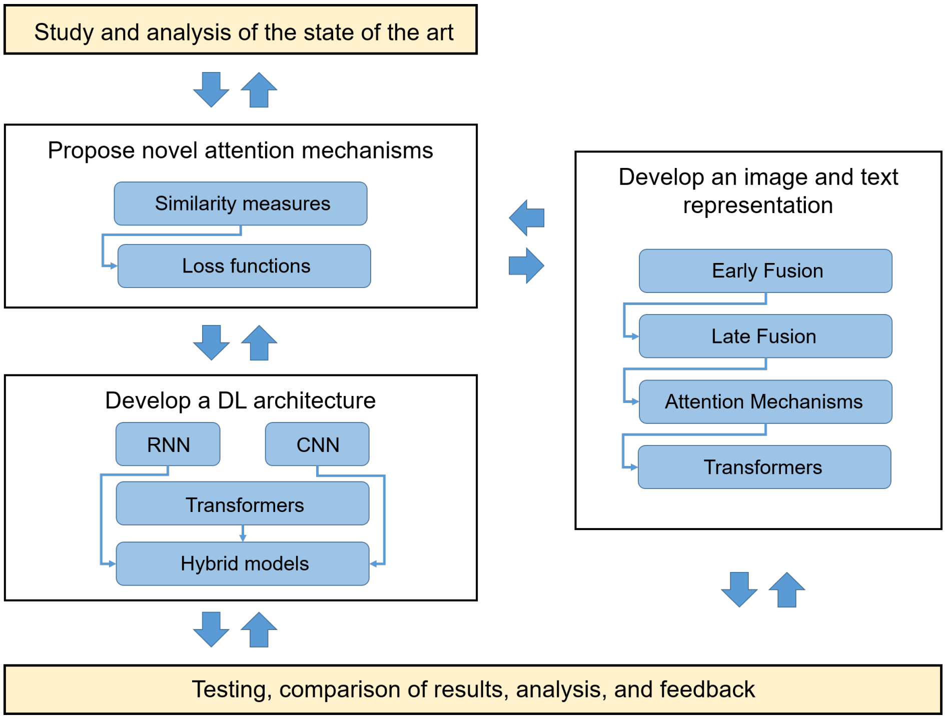

Study and analysis of the state of the art
Propose novel attention mechanisms
Similarity measures
Loss functions
Develop an image and text representation
Early Fusion
Late Fusion
Attention Mechanisms
Transformers
Develop a DL architecture
RNN
CNN
Transformers
Hybrid models
Testing, comparison of results, analysis, and feedback

Figure 4: Outline of the proposed methodology.

Below, each of the phases of the methodology is detailed:

1. **Study and analysis of the state of the art.**
   (a) Identify and obtain labeled datasets related to the AL detection task in text and memes.
   (b) Analyze the features of the obtained datasets.
   (c) Implement the approaches of the state of the art related to the detection of AL in text and memes, using the previously obtained datasets.

2. **Propose dual attention mechanisms to extract and incorporate local and global dependencies in the elements of a sequence.** The attention mechanisms have been incorporated into DL architectures, due to their ability to find local (use of SA) [40] and global (use of CA) [41] relationships in the elements of a sequence [39]. The development of a novel attention mechanism is proposed in order to create a mechanism that considers both relationships, the following strategies are proposed:

    (a) Propose similarity Measures: A variety of similarity and normalization measures have been proposed in attention mechanisms [39]. It is proposed to develop novel similarity measures which incorporates the SA and CA mechanisms.

    (b) Propose loss functions: These functions quantify the predicted output (or label) against the actual output. The loss functions are used to determine the penalty for an incorrect classification of an input data [74, 75]. The development of a novel loss function based on the query similarity of the contextual attention vectors is proposed.

3. **Develop a DL architecture for the detection of AL in text, which integrates the previously proposed dual attention mechanisms.** This stage includes the development of a DL architecture that integrates the previously proposed attention mechanisms, in order to measure the efficiency of this mechanisms in the detection of AL in text, with the use of different architectures. Some examples of DNN architectures to consider are:

    (a) Recurrent Neural Networks (RNNs): Design a DL architecture based on RNNs. The RNN take as their input, not just the current input example, but also what they have received previously. With this process, the network creates a memory of what they previously learn and it finds correlations between the elements of a sequence [76].

(b) Convolutional Neural Networks (CNNs): The use of this architecture has been used within NLP for the extraction of important features at a word and character n-gram level [32, 34, 38]. We proposed the integration of the proposed mechanisms in a CNN in order to find relationships between the most relevant features extracted from a CNN.

(c) Transformers: The use of this architecture has become popular in recent years, due to their outstanding performance in a wide variety of tasks [24, 77]. Its outstanding results are based on the use of SA, which allows capturing long-range relationships between words [67]. It is proposed to develop an architecture based on the use of Transformers with the integration of the proposed attention mechanisms.

(d) Hybrid models: In addition to integrating the proposed mechanisms, the development of hybrid models based on two or more DL architectures is proposed, in order to integrate the strengths of the DL architectures. For example, the integration of the CNN and RNN networks has excelled in the detection of AL in text, due to their ability to extract relationships between the most relevant features [21, 38, 37].

4. **Develop a model to create a representation that combines the different modalities (image and text), for the detection of AL in memes.** This phase involves the development of a model that automatically combines the different modalities (image and text features), and creates a new vision & language representation. Traditional approaches based on ensemble of classifiers or the concatenation of features, tend to learn some hierarchical structure of the information, ignoring the relationship between the modalities. To overcome this problem, the use of DL approaches has become popular since these approaches learn to combine and or give importance to a wide variety of information. The following approaches will be used in this work:

(a) Early Fusion: Develop an approach to early fusion the feature vectors of different modalities. The information from the different modalities is taken as one vector, and then a classifier (DNN) is used to learn this representation [78].

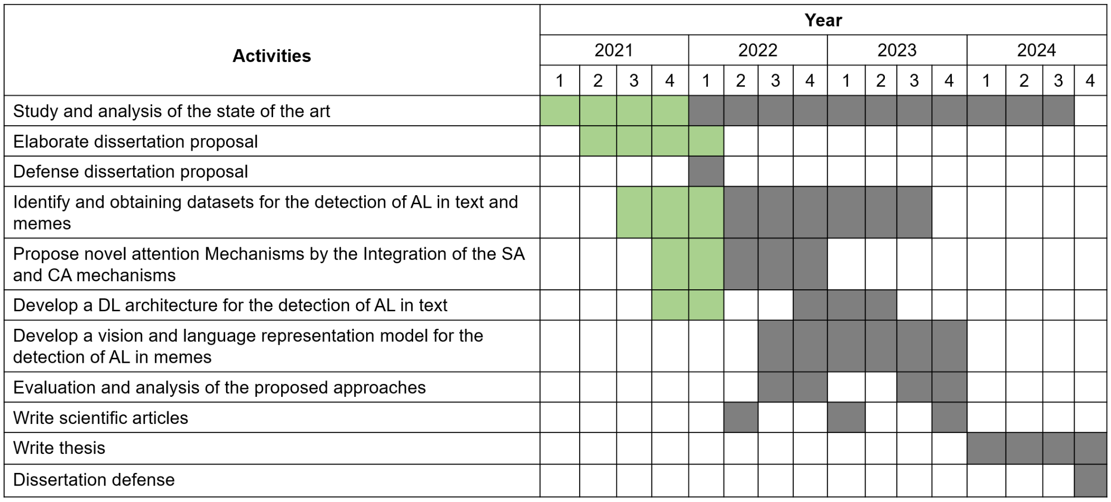
(b) Late Fusion: In this phase each modality is used independently in the classifier, followed by a fusion vector at a decision-making stage [78].

(c) Attention mechanisms: In order to weigh the relevance of each modality, it is proposed to develop a model with the attention mechanisms that could learn the importance of the features in the different modalities, extracting the most relevant parts from each modality [39].

(d) Transformers: Similar to the previous stage, the development of a Transformer-based model for the representation of vision & language is proposed, which considers the relationship between words and image regions [59, 61]. This is essential for the correct interpretation of memes [13].

5 Work Plan

This section presents the work plan for this dissertation proposal; Figure 5 presents a general working schedule, and it includes the most relevant activities that are planned. The schedule is divided into four periods per year, each of them has an extension of three months.



Activities	Year															
	2021				2022				2023				2024			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Study and analysis of the state of the art	🟩	🟩	🟩	🟩	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	
Elaborate dissertation proposal		🟩	🟩	🟩	🟩											
Defense dissertation proposal					▨											
Identify and obtaining datasets for the detection of AL in text and memes			🟩	🟩	🟩	▨	▨	▨	▨	▨	▨					
Propose novel attention Mechanisms by the Integration of the SA and CA mechanisms				🟩	🟩	▨	▨	▨								
Develop a DL architecture for the detection of AL in text				🟩	🟩			▨	▨	▨						
Develop a vision and language representation model for the detection of AL in memes							▨	▨	▨	▨	▨	▨				
Evaluation and analysis of the proposed approaches							▨	▨			▨	▨				
Write scientific articles						▨			▨			▨				
Write thesis													▨	▨	▨	▨
Dissertation defense																▨

Figure 5: Work Schedule for the completed and pending activities for this doctoral research proposal.

6 Preliminary Results

This section presents the preliminary results that support this research proposal. In order to address the first two stages of our methodology, a first approach to the dual attention mechanism was proposed to combine the SA and CA mechanisms. For the sake of clarity, we will refer to this dual attention mechanism as: *Self-Contextualized Attention (SCA) mechanism*. The proposed approach was addressed from a unimodal perspective (only text), and evaluated with 6 different collections related to the AL detection task. This section is divided as follows: Section 6.1 introduce the evaluation datasets, lately in Section 6.2 we present in detail the proposed SCA mechanism. Section 6.3 presents the integration of the proposed mechanism into a Bi-GRU, and a TNN architecture. Section 6.4, 6.5 and 6.6 comprises the comparison baselines and implementation details, and the quantitative and qualitative results, respectively. Finally, in section 6.7, 6.8 and 6.9 we present our preliminary conclusions, the work in progress, as well as the published papers.

6.1 Datasets for Abusive Language identification

AL can be of different types, its main divisions are distinguished by the target and severity of the insults. Accordingly, different collections and evaluation campaigns have considered different kinds of AL for its study [9]. Below we present a brief description of the six English datasets we used in our experiments. From now on we will refer to them as DS1, DS2, ..., DS6. Figure 6 shows the classes distribution.

DS1 [29], DS2 [19] and DS3 [79] were some of the first large-scale datasets for abusive tweet detection; DS1 focuses on the identification of racist and sexist tweets, DS2 focuses on identifying tweets with abusive language and hate speech, whereas DS3 focuses on the detection of harassment in tweets. On the other hand, DS4 [63] and DS5 [62] were used in the *SemEval-2019 Task 6*, and in the *Evalita 2018* Task on

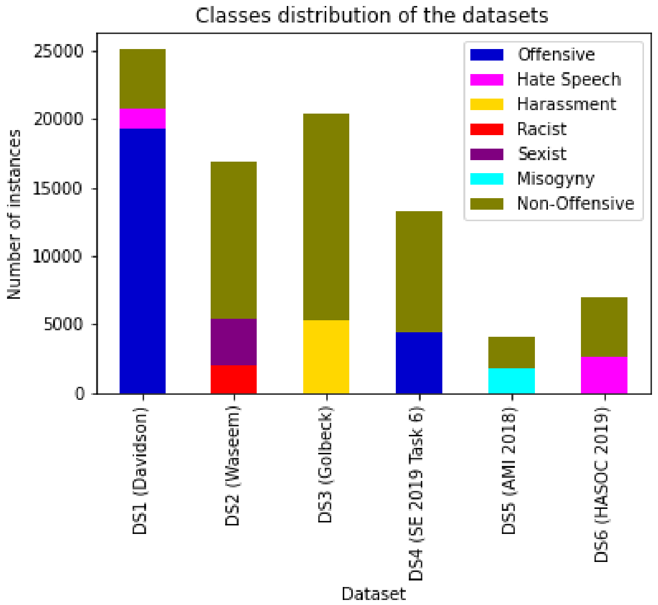


Figure 6: Classes distribution of the 6 used datasets.

Automatic Misogyny Identification (AMI) respectively. DS4 focuses on identifying offensive tweets, whereas DS5 focuses on identifying misogyny in tweets. Finally, DS6 [80] was presented at the *11th Forum for Information Retrieval Evaluation (FIRE)*, in the Hate Speech and Offensive Content Identification (HASOC) shared-task, where the main goal is the classification of Hate Speech and non-offensive online content in Indo-European Languages; in this work we will only focus on the English language. The last three shared-tasks provide a fine-grained evaluation through different sub-tasks; in this work, we focus on the sub-task A (binary classification of offenses, misogyny, and hate speech, respectively).

6.2 Self-contextualized attention mechanism

This subsection introduces the proposed SCA mechanism. For a better understanding of the attention mechanisms, we refer the reader to subsection 7.3.1. The proposed mechanism can be applied to any sequence of encoding features H. For the purposes of this work, each element of the sequence is represented by the word encoding features h_i, which are extracted from a deep neural network, either an RNN or TNN.

Given a sequence of encoding features $H = \{h_1, h_2, ..., h_n\}$, where $H \in \mathbb{R}^{k \times n}$, k is the number of the encoding features and h_i refers to the *i-th* element of H, the purpose of our proposed SCA mechanism is to generate a global context-aware representation G, that considers both the internal and external relationships between the encoding features of H. Figure 7 shows the general architecture of our proposed SCA mechanism. This architecture is divided into three major stages, each of them is illustrated by the 3 rectangles, corresponding to the SA, CA and SCA stages. Below, we present in detail the aforementioned stages.

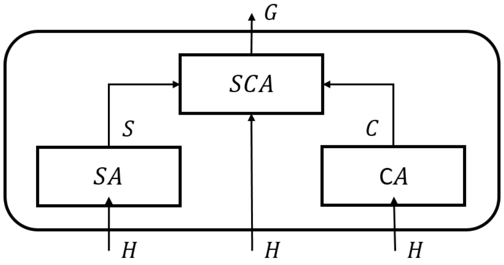


Figure 7: Proposed self-contextualized attention mechanism.

SA stage: the main purpose of SA is the building of connections within the elements of the same sequence, but at different positions. The use of SA allows the modeling of both long-range and local dependencies, this is captured by the attention filter $\alpha_s \in \mathbb{R}^{n \times n}$ defined in the Equation 1. This attention filter is calculated with the dot product similarity between all the pairs of elements of H, later these values are smoothed with the use of a softmax function. Finally, the context-aware representation $S \in \mathbb{R}^{k \times n}$ shown in the Equation 2, is calculated with the matrix multiplication of H and α_s^T, where α_s is used to highlight and filter out the most and less relevant encoding features, respectively.

$$\alpha_s = softmax(H^T \cdot H) \tag{1}$$

$$S = H\alpha_s^T \tag{2}$$

CA stage: unlike the previous stage, the CA mechanism uses a context vector $u_h \in \mathbb{R}^k$, which is randomly initialized and jointly learned during the training process, this vector is used as a query vector in order to obtain the attention values $\alpha_c \in \mathbb{R}^n$ by measuring the similarity between the elements of the sequence H and the application domain represented by u_h. This similarity is calculated in the Equation 3 by calculating the scalar dot product of u_h^T and H; the resulting values are smoothed with the use of a softmax function. Contrasting the CA mechanism proposed by [41], instead of using a weighted sum between each attention value and its corresponding encoding features for the final sequence representation, our context-aware representation $C \in \mathbb{R}^{k \times n}$ shown in Equation 4, takes all the information of the attention values, by doing an element-wise multiplication $\odot$, within each scalar of α_c and its corresponding encoding features h_i.

$$\alpha_c = softmax(u_h^T \cdot H) \tag{3}$$

$$C = \alpha_c \odot H \tag{4}$$

SCA stage: since the previous stages generate two different context-aware representations S and C, respectively. The purpose of this stage is to merge these representations in order to create a global context-aware representation $G \in \mathbb{R}^{k \times n}$ that integrates both, the internal and external relationships. These relationships are captured with the global attention filter $\alpha_g \in \mathbb{R}^{n \times n}$, which is calculated by the smoothed dot product similarity between S and C, as shown in Equation 5. This attention filter can be seen as a high level attention representation, since it is calculated based on the local dependencies and the application domain. Finally, the global context-aware representation G is calculated in Equation 6 with the matrix multiplication of H and α_g^T.

$$\alpha_g = softmax(S^T \cdot C) \tag{5}$$

$$G = H\alpha_g^T \tag{6}$$

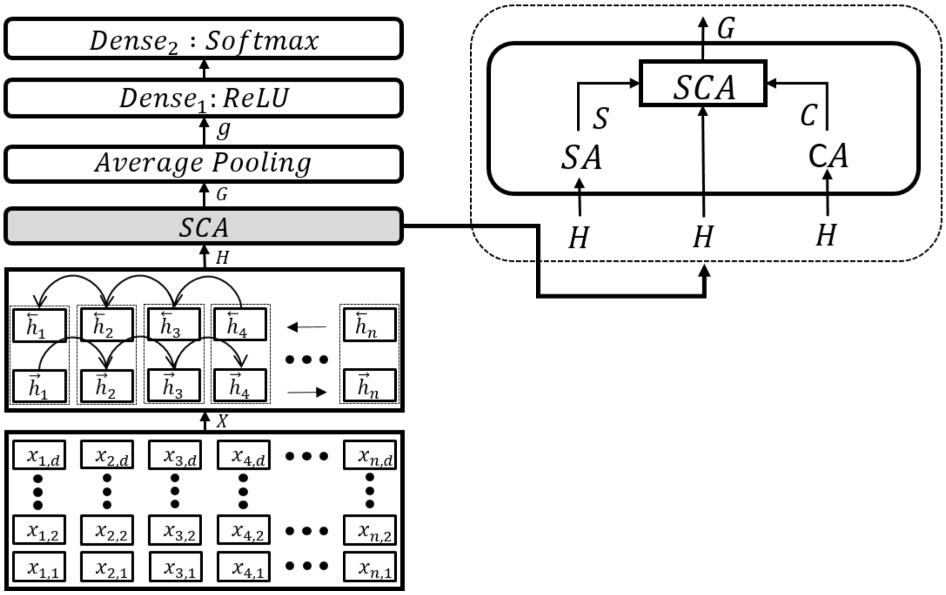
6.3 Integration of the SCA mechanism in deep neural networks

This subsection presents the integration of the proposed SCA mechanism in deep neural network architectures. Two different architectures were selected in accordance to the previously mentioned state-of-the-art approaches, presented in section 2. Subsequently, this subsection is divided two-folded: First, the integration of the SCA mechanism in RNNs is presented. Next, the integration of the SCA mechanism in the TNN architecture is presented.

6.3.1 Integrating the SCA mechanism into RNNs

In order to integrate the proposed mechanism, we adapt a modular and well-performing DNN architecture. This architecture was presented in [41] and its designed to modularly manage different AM. The adapted architecture is shown in Figure 8; it consists of four main stages, which are described below.

The *first* and *second stages* correspond to the input and encoding stages, respectively. The *input stage* is integrated by the embedding matrix $X \in \mathbb{R}^{d \times n}$, which is represented by a sequence of n d-dimensional word vectors x_i. Subsequently, the



$Dense_2 : Softmax$
$Dense_1: ReLU$
g
$Average\ Pooling$
G
SCA
H
$\overleftarrow{h}_1$ $\overleftarrow{h}_2$ $\overleftarrow{h}_3$ $\overleftarrow{h}_4$ $\overleftarrow{h}_n$
$\overrightarrow{h}_1$ $\overrightarrow{h}_2$ $\overrightarrow{h}_3$ $\overrightarrow{h}_4$ $\overrightarrow{h}_n$
X
$x_{1,d}$ $x_{2,d}$ $x_{3,d}$ $x_{4,d}$ $x_{n,d}$
$x_{1,2}$ $x_{2,2}$ $x_{3,2}$ $x_{4,2}$ $x_{n,2}$
$x_{1,1}$ $x_{2,1}$ $x_{3,1}$ $x_{4,1}$ $x_{n,1}$
G
SCA
S
C
SA
CA
H
H
H

Figure 8: Adapted RNN architecture.

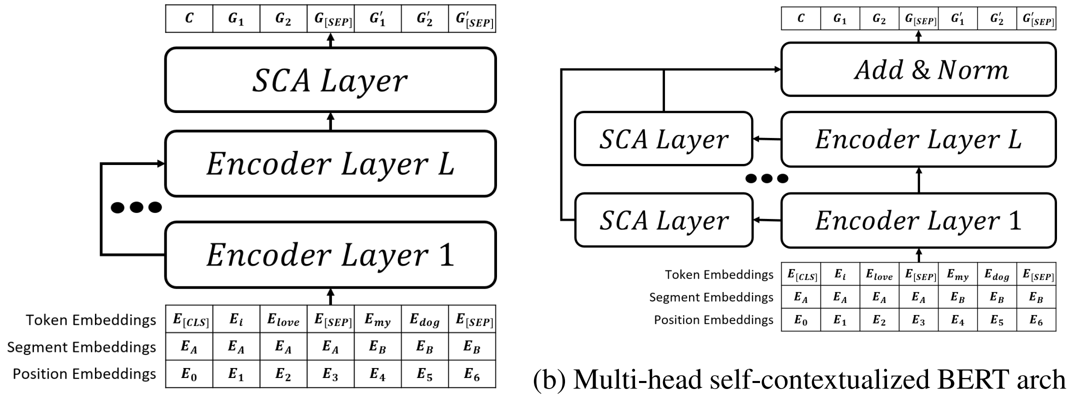
embedding matrix X passes as input to the *encoding stage*, which is conformed by a Bidirectional Gated Recurrent Unit (Bi-GRU) layer. The Bi-GRU layer accomplish the sequence encoding task by summarizing the information of the whole sequence X centered around each word annotation; the producing encoding stage generates a sequence of encoding features $H \in \mathbb{R}^{k \times n}$.

Since not all words contribute equally for the meaning and representation of a sequence, the *third stage* corresponds to the attention stage, including the SCA mechanism and the average pooling layer. Specifically, the sequence encoded features H are passed as input to the SCA mechanism, which generates a global context-aware representation G (as described in Section 6.2); since the next stage uses a vector for the classification layers, the matrix G is reduced with the average pooling layer, generating a high level representation vector $g \in \mathbb{R}^k$, which summarizes the most relevant information from G. Finally, the *Fourth stage* uses the representation vector g as input for the classification layers; two layers handle the final classification, a dense layer with a Rectified Linear Unit (ReLU) activation function, and a fully-connected softmax layer to obtain the class probabilities and get the final classification.

6.3.2 Integrating the SCA mechanism into TNNs

The integration of the proposed SCA mechanism into the TNN architecture is presented as a second step in the integration of the SCA mechanism in deep neural network architectures. Since there are currently many pre-trained TNN-based models, we selected the BERT (Bidirectional Encoder Representations from Transformers) model [24], which is currently the most widely used pre-trained model, obtaining cutting-edge results in one- and two-sentence classification tasks [68].

Unlike the original TNN architecture, BERT only uses the encoding layer (for more information, we refer the reader to section 7.3.3), which allows the model the representation of text sequences [24]. The model has two main pre-trained models: $BERT_{large}$ and $BERT_{base}$, which differentiate according to the following parameters: $BERT_{base}$ $(L = 12, H = 768, A = 12)$ and $BERT_{large}$ $(L = 24, H = 1024, A = 16)$, where L represents the number of encoding layers (Transformer blocks), H the hidden size of each encoding layer, and A the number of self-attention heads.


C G_1 G_2 $G_{[SEP]}$ G'_1 G'_2 $G'_{[SEP]}$
SCA Layer
Encoder Layer L
Encoder Layer 1
Token Embeddings $E_{[CLS]}$ E_i E_{love} $E_{[SEP]}$ E_{my} E_{dog} $E_{[SEP]}$
Segment Embeddings E_A E_A E_A E_A E_B E_B E_B
Position Embeddings E_0 E_1 E_2 E_3 E_4 E_5 E_6
C G_1 G_2 $G_{[SEP]}$ G'_1 G'_2 $G'_{[SEP]}$
Add & Norm
SCA Layer
Encoder Layer L
SCA Layer
Encoder Layer 1
Token Embeddings $E_{[CLS]}$ E_i E_{love} $E_{[SEP]}$ E_{my} E_{dog} $E_{[SEP]}$
Segment Embeddings E_A E_A E_A E_A E_B E_B E_B
Position Embeddings E_0 E_1 E_2 E_3 E_4 E_5 E_6

(a) Self-contextualized BERT architecture

(b) Multi-head self-contextualized BERT architecture

Figure 9: Proposed architectures for the integration of the SCA mechanism in TNN-based models.

For the purpose of one-sentence classification, the last encoding layer is commonly used, since the information from the previous layers is contained in it (the model has residual connections between the different encoding layers). Specifically, the <CLS> token of the last encoding layer is used for classification tasks [24]. Recent studies suggest that the use of different levels of encoding layers may have a better performance impact in different tasks [81]. In order to compare the use of all the encoding layers in contrast to the use of the last one, two different architectures are proposed: the *Self-contextualized BERT* architecture and the *Multi-head self-contextualized BERT* architecture. Figure 9 illustrates the proposed architectures.

Regarding the *Self-contextualized BERT* architecture, the last T_L encoding layer is used as input to the proposed SCA mechanism (as illustrated in the left-hand side of Figure 9), which generates a global context-aware representation G, maintaining the same dimensions $M \times H$ (as the previous layer), where M represents the number of tokens in the sentence. Subsequently the classification token G_1 is then passed to a fully-connected softmax layer to obtain the class probabilities and get the final classification. On the other hand, the *Multi-head self-contextualized BERT* architecture uses each of the encoding layers T_i as input to an independent layer of the SCA mechanism (each layer of the SCA mechanism corresponds to a specific encoding layer, as illustrated in the right-hand side of Figure 9); thus generating L different global context-aware representations G_i. Subsequently, these representations pass through the concatenation and normalization layer, in order to maintain the same dimensions $M \times H$. Finally, as in the previous architecture, the classification token passes to the classification layer.

6.4 Comparison baselines and implementation details

the first architecture is based on a simple Bi-GRU network, which receives word embeddings as input but does not use any attention layers; the second and third architectures employ the same Bi-GRU network with the addition of a SA and CA layer, respectively. Finally, in order to compare the integration of our proposed

SCA mechanism into the BERT model, the fourth baseline is based on a fine-tuned $BERT_{BASE}$[11] model, built with the addition of the task-specific inputs and an end-to-end fine-tuning strategy of all parameters. As described in [24], we take the last layer encoding of the classification token <CLS> and use it as input for the softmax classification layer. These four baselines architectures are referred in the experiments as: $Bi\text{–}GRU$, $Bi\text{–}GRU_{SA}$, $Bi\text{–}GRU_{CA}$, and $BERT_{BASE}$. The proposed integrations of our proposed mechanism into the $Bi-GRU$ DNN and the two proposed BERT-based architectures, are referred in the experiments as: $Bi-GRU_{SCA}$, $SC-BERT$, and $MHSC-BERT$, respectively.

Regarding the implementation details, different text preprocessing operations were applied: user mentions and links were replaced by the default tokens: <*user*> and <*url*>; in order to enrich the vocabulary, all hashtags were segmented by words (*e.g.* #BuildTheWall - build the wall) with the use of the ekphrasis library, proposed in [82]; in addition to this, all emojis were converted into words (*e.g.* ☺ - smiley face) using the demoji[12] library; stop words were removed, with the exception of personal pronouns; all text was lowercased and non-alphabetical characters as well as consecutive repeated words were removed. For word representation we used pre-trained fastText embeddings [83], trained with subword information on Common Crawl, which have been recognized as useful for this task according to the study presented in [8]. All the non-BERT based DNNs were trained for a total of 12 epochs, with a learning rate of 1e–4, using the Adam optimizer [84] and a Dropout rate of 10%. The BERT-based ones were trained for a total of 3 epochs, following the same hyperparameter settings.

[11] `https://tfhub.dev/tensorflow/small_bert/bert_en_uncased_L-12_H-768_A-12/1`

[12] `https://pypi.org/project/demoji/`

6.5 Quantitative Effectiveness of the SCA mechanism

Table 1 shows the results of the evaluation applied to the proposed architectures integrated with the SCA mechanism ($Bi{-}GRU_{SCA}$, $SC{-}BERT$, and $MHSC{-}BERT$), as well as the four baselines architectures $Bi-GRU$, $Bi-GRU_{SA}$, $Bi-GRU_{CA}$ and $BERT_{BASE}$. In order to compare the results against the SOTA approaches, DS1, DS2 and DS3 were evaluated with the weighted-average F_1 score, DS4 and DS6 were evaluated using the macro-average F_1 score, and DS5 was evaluated using the accuracy.

Model	**DS1**	**DS2**	**DS3**	**DS4**	**DS5**	**DS6**
$Bi-GRU$	0.842	0.873	0.691	0.795	0.691	0.727
$Bi-GRU_{SA}$	0.879	0.896	0.708	0.813	0.713	0.742
$Bi-GRU_{CA}$	0.881	0.907	0.716	0.819	0.725	0.745
$Bi-GRU_{SCA}$	0.891	0.915	0.731	0.826	0.738	0.753
$BERT_{BASE}$	0.874	0.921	0.718	0.814	0.714	0.768
$SC-BERT$	0.887	0.927	0.724	0.821	0.725	0.772
$MHSC-BERT$	**0.903**	0.929	**0.732**	**0.835**	**0.742**	0.781
$SOTA$	0.88	**0.93**	0.727	0.829	0.704	**0.788**
Reference	[85]	[85]	[11]	[25]	[86]	[87]

Table 1: Comparison results from our four baselines, our proposed architectures (integrating the SCA mechanism) and state-of-the-art approaches in six datasets for the AL identification.

Centering the analysis of results on the first three baselines and in our $Bi-GRU_{SCA}$ architecture (rows 2 - 5), the results indicate that the use of AM outperformed the base Bi-GRU network (row 2 vs rows 3 - 5) by at least a margin of 2.2%. In addition, the use of the CA outperformed the use of SA (row 4 vs row 3), which is consistent according to the results obtained in [11]. Finally, comparing the use of our proposed SCA mechanism against the use of SA and CA (row 5 vs rows 3 and 4), better results are obtained in the six evaluation datasets.

Table 1 also compares the results from the integration of our proposed SCA mechanism into the TNNs. Regarding the $BERT_{BASE}$ baseline (row 6 vs. rows 7 - 8), it is shown that the integration of our proposed mechanism into the BERT model, outperforms the $BERT_{BASE}$ model. The integration of the SCA mechanism at the different levels of encoding in the BERT model ($MHSC-BERT$) shows a consistent improvement, with respect to all the proposed approaches (row 7 vs rows 2 - 6). In order to compare the results of our proposed SCA-based architectures, against the four baselines, the Friedman statistical test [88] was performed, obtaining statistically significant results at $p \leq 0.005$. In addition to this, Figure 10 presents the Nemenyi test [89], applied to the proposed SCA-based approaches, and the $Bi-GRU$ and $BERT_{BASE}$ baselines. As shown in the figure, the results are statistically significant, when comparing the TNN-based approaches against the RNN-based ones.

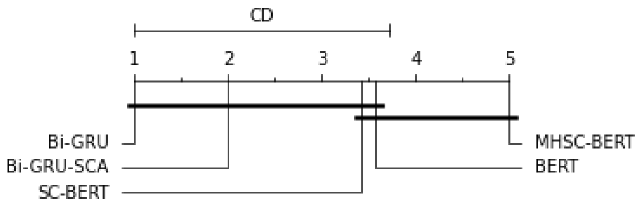


Figure 10: Comparison of all the proposed SCA-based approaches, and the $Bi-GRU$ and $BERT_{BASE}$ baselines against each other with the Nemenyi test.

Finally, Table 1 presents a comparison between state-of-the-art approaches and the proposed ones (row 9 vs rows 2 - 8). It is shown that our $MHSC-BERT$ architecture obtains better results in 4 of the 6 datasets.

6.6 Qualitative effectiveness of the SCA mechanism

NOTE: This subsection contains examples of language that may be offensive to some readers, these do not represent the perspectives of the authors.

In order to understand the effectiveness of our proposed SCA mechanism in the improvement of the sequences representation, this subsection presents the qualitative results of the analysis and visualization of the attention values. Since the SCA mechanism integrates both, the SA and CA mechanisms, the attention values were considered at these three different levels, with the analysis of the α_s, α_c and α_g attention filters, which correspond to the SA, CA and SCA mechanisms.

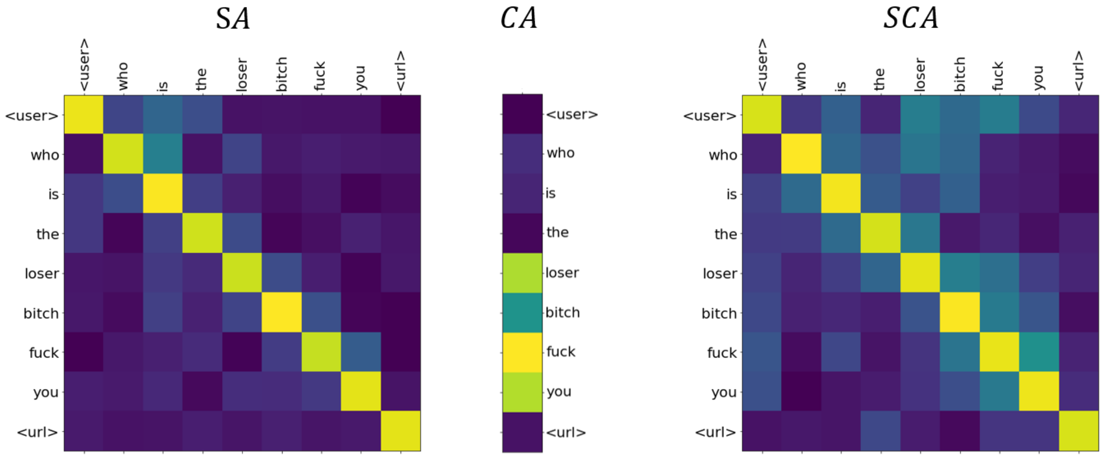


Figure 11: Attention heatmaps visualization, corresponding to the α_s, α_c, and α_g attention filter values. The Example shown in the attention heatmaps was taken from the DS3 dataset.

Figure 11 shows the visualization of the attention heatmaps corresponding to the three attention filters values integrated by the SCA mechanism. The example shown in the figure *"user who is the loser bitch fuck you url"* corresponds to an offensive instance taken from the DS3 dataset. As shown in the figure, the values of the attention filter α_s, corresponding to the SA, tend to be more relevant with respect to their own elements and their closest neighbors, for example, in the case of the most relevant words to *"who"*, the same word *"who"* is found, followed by the word *"is"*, likewise, in the case of the most relevant words to *"fuck"*, the words *"fuck"*, *"you"* and *"bitch"* are found. On the other hand, the values of the attention filter α_c, corresponding to the CA, indicate the most relevant words for the AL identification; as can be seen in the central heatmap from the Figure 11, the most relevant words are: *"loser"*, *"bitch"* and *"fuck"*, which indeed correspond to words potentially used in offensive contexts.

Finally, the values of the attention filter α_g, corresponding to the SCA, are shown in the right heatmap from Figure 11. The attention filter α_g shows the combination of both AM, which improves the representation of an instance. For example, in the produced visualization from the most relevant words to *"user"*, a closer relationship to offensive words is now presented, highlighting the words: *"loser"*, *"bitch"* and *"fuck"*, which are often used to offend, something similar is presented with the words *"who"* and *"is"*. On the other hand, the words *"fuck"*, *"you"* and *"bitch"*, in addition to having a better relationship with other offensive words as *"loser"*, are also related to the target of the offense: *"user"*.

6.7 Conclusions

One of the main problems in the use of current AMs is the loss of contextual or internal information between the elements of a sequence. To tackle this particular issue we proposed the SCA mechanism, which integrates the SA and CA mechanisms for the construction of a novel representation that considers both, the internal and contextual relationships between the elements of a sequence. Due to the highly context-dependent interpretation of words in the AL detection task, in this work we explore the use of the proposed SCA mechanism in the AL detection task. The results obtained in six collections, considering different kinds of AL, were encouraging. In addition to this, the SA and CA mechanisms were evaluated against the SCA mechanism, the results show a quantitative improvement in the use of our proposed SCA mechanism, which allowed preliminary concluding that the use of the SCA mechanism is useful for discriminating between offensive and non-offensive contexts.

In addition to this, the integration of the proposed mechanism in the BERT model shows a considerable improvement with respect to the $BERT_{BASE}$ model. The $MHSC-BERT$ model was the one that obtained the best performance results of all the proposed approaches, which allows us to preliminary conclude that the use of the SCA mechanism at different encoding levels is useful for the detection of AL in text. Likewise, it opens the prospect for its implementation in the detection of AL in memes, with the use of TNN-based vision & language models.

6.8 Work in progress

Motivated by the obtained results in the integration of the SCA mechanism into the different encoding layers of the BERT model,it is proposed to extend the combination of the context-aware global vectors G_i with the use of a Gated Multimodal Unit (GMU) [90], this with the purpose of combining the information in a weighted manner, according to the different encoding layers. It is planned to publish our approaches and results in the journal: Information Processing & Management, which has an impact factor of: 6,222. As a future task we plan to integrate the SCA mechanism into TNN-based vision & language models, for the detection of AL in memes.

6.9 Published Papers

The proposed SCA mechanism and its integration into a Bi-GRU network was presented at the *Ninth International Workshop on Natural Language Processing for Social Media*.

- H. Jarquín-Vásquez, H. J. Escalante, and M. Montes, “Self-contextualized attention for abusive language identification,” in Proceedings of the Ninth International Workshop on Natural Language Processing for Social Media, (Online), pp. 103–112, Association for Computational Linguistics, June 2021.

7 Background Concepts

This section describes an overview of the different techniques and core concepts needed for this dissertation proposal. Since the scope of this work covers the unimodal (text) and bimodal (text and image) representations; this section is divided as follows: first, a description of text classification and Multimodal Machine Learning (MML) is presented. Then, the concept of DL is introduced, as well as, some of the main ideas that are behind it, with some relevant DNN architectures useful for the representation of the data.

7.1 Text Classification

Text Classification (TC) is the process of assigning categories or tags to a text or a document according to its content; TC can be used to categorize and structure a set of documents, for example, topics, languages, and conversations. TC has a wide range applications such as: sentiment analysis, intent detection, and information filtering [91].

TC can work in two different ways: i) automatic, that applies machine learning to classify text faster and with less cost, and ii) manual, where a human annotator review the text and categorize it accordingly to how interprets the content [65]. TC has become an important part of business as it allows to get insights from the data and automate analysis for different processes. Figure 12 describes the general process for TC; the model receive an input text and return a label as an output.

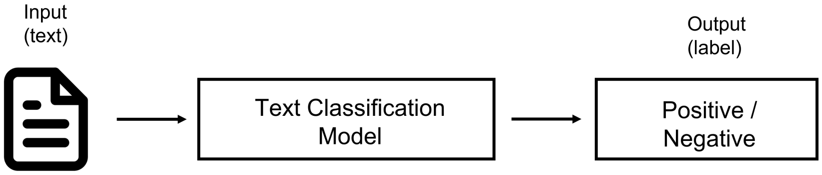


Figure 12: General process for text classification.

7.2 Multimodal Machine Learning

MML aims to build models that can process and relate information from multiple modalities, according to [47] there are five core technical challenges surrounding the MML, Figure 13 presents this taxonomy.

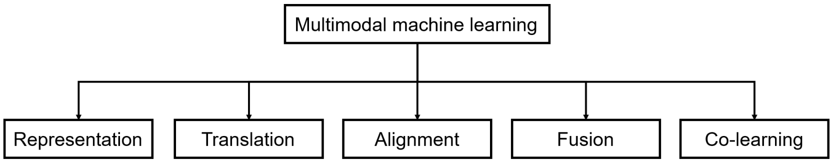


Figure 13: Taxonomy of the Multimodal machine learning.

The following describes the five core technical challenges in the MML:

- **Representation:** A first fundamental challenge is learning how to represent and summarize multimodal data in a way that exploits the complementarity and redundancy of multiple modalities. The heterogeneity of multimodal data makes it challenging to construct such representations.
- **Translation:** A second challenge addresses how to translate (map) data from one modality to another.
- **Alignment:** A third challenge is to identify the direct relations between sub-elements from two or more different modalities.
- **Fusion:** A fourth challenge is to join information from two or more modalities to perform a prediction.
- **Co-learning:** A fifth challenge is to transfer knowledge between modalities, their representation, and their predictive models.

In this work we focus on the representation and alignment of vision and language; based on the identification of direct relations between the sub-elements of both features modalities through an attention-based DL approach. In the following subsections DL is described in detail, with its main components, including the AMs; as well as, some relevant DNN architectures such as: RNNs, and the transformer neural network.

7.3 Deep Learning

DL is a sub-field within ML, this sub-field focuses on learning models at multiple levels of representation and abstraction from input data such as images, sound and text. Historically the concept of DL was originated within the context of artificial neural networks [92]. The family of DL methods has become increasingly extensive, also encompassing a variety of supervised and unsupervised learning algorithms [74].

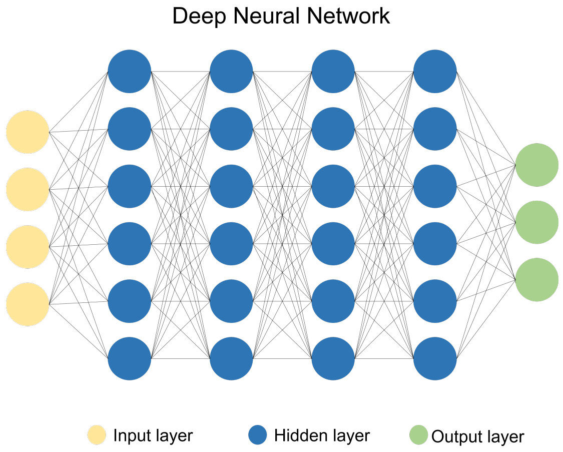


Figure 14: Representation of a deep neural network.

The Deep Neural Networks (DNN) are an example of a DL model. They can be defined as a multi-layer perceptron that consists of an Artificial Neural Network (ANN) formed by multiple layers, in such a way that it has the ability to solve problems that are not linearly separable. Generally, this type of network consists of an input layer, a hidden layer and an output layer (following a simple configuration, this is not considered part of a DNN). Each layer is composed of neurons that are connected between the layers and are responsible for transferring the weights through the network. Figure 14 shows an example of a DNN architecture.

A DL architecture consists of a multi-layer representation that applies activation functions to perform non-linear transformations of the inputs which can be described as follows:

$$f_l^{W,b} = f_l(\sum_{j=1}^{N_l} W_{lj}X_j + b_l), 1 \geq l \leq L \tag{7}$$

Where the number of hidden units is given by N_l. The predictor is in charge of modeling a high-dimensional mapping F through the composition of functions, as can be defined in equation 8.

$$Y(X) = F(X) = (f_1^{W_1,b_1} \circ \ldots \circ f_L^{W_L,b_L}) \tag{8}$$

The final output is the answer of Y, this can be categorical or numerical. The explicit structure of the deep prediction rule is:

$$\begin{aligned} Z^{(1)} &= f^{(1)}(W^{(0)}X + b^{(0)}), \\ Z^{(2)} &= f^{(2)}(W^{(1)}Z^{(1)} + b^{(1)}), \\ &\ldots \\ Z^{(L)} &= f^{(L)}(W^{(L-1)}Z^{(L-1)} + b^{(L-1)}), \\ Y^{(X)} &= W^{(L)}Z^{(L)} + b^{(L)} \end{aligned} \tag{9}$$

Where $Z^{(L)}$ is defined as the L-th layer, $W^{(L)}$ is the weight matrix and $b^{(L)}$ is the bias. $Z^{(L)}$ contains the extracted hidden features, in other words, the deep approach uses hierarchical predictors that comprise a series of non-linear transformations in L applied to X. Each of the L transformations refers to a layer where the original input is X, the output of the first transformation is the input of the second layer and so on until the output Y as the layer $(L+1)$. $l \in \{1, ..., L\}$ is used to index the layers, which are called hidden layers. The number of layers L represents the depth of the deep architecture.

Notable approaches have emerged in DL such as the use of attention mechanisms, CNNs, RNNs, Auto-Encoder (AE), Deep Belief Network (DBN), Generative Adversarial Network (GAN), Deep Reinforcement Learning (DRL) and the Transformer Neural Network (TNN) [75]. Specifically, in this work we focus on the extension of the AMs, through the use of the TNN and RNNs; this applied to unimodal and bimodal scenarios.

7.3.1 Attention Mechanisms

One of the most used approaches in DL is the use of AMs, the idea behind the use of the AM is to provide the classification model with the ability to focus on a subset of inputs (or features), handling in this way the importance of features in accordance to their context. Due to their outstanding performance in many NLP and Computer Vision (CV) tasks, several AM have been proposed in recent years [39], which can be divided into two main approaches: Self-Attention (SA) [67] and Contextual Attention (CA) [41] mechanisms. Specifically, SA takes the relationships among features within the same sequence, whereas, CA selectively focuses on features with respect to some external query vector, which adjusts according to the training task. The more important the feature is in determining the answer to that query, the more focus it is given. Figure 15 illustrate these approaches.

The main purpose of the SA mechanism is the building of connections within the elements of the same sequence, but at different positions. The use of SA allows the modeling of both long-range and local dependencies, this is captured by the attention filter $\alpha_s \in \mathbb{R}^{n \times n}$ defined in the Equation 10. This attention filter is calculated

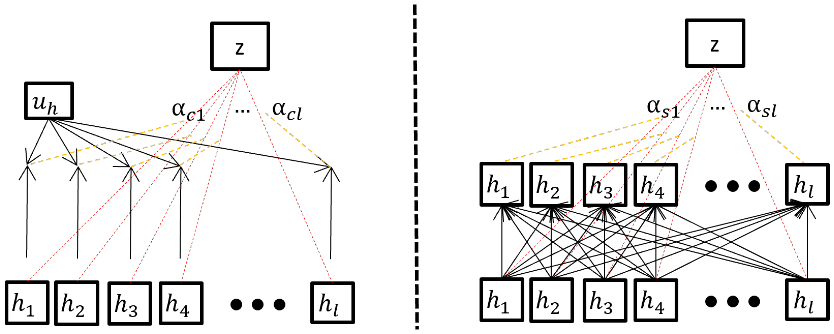


Figure 15: Contextual attention vs. self-attention representation.

with the dot product similarity between all the pairs of elements of $H \in \mathbb{R}^{k \times n}$, where H is sequence of encoding features (usually extracted with an RNN); later these values are smoothed with the use of a softmax function.

$$\alpha_s = softmax(H^T \cdot H) \tag{10}$$

Unlike the SA mechanism, the CA mechanism uses a context vector $u_h \in \mathbb{R}^k$, which is randomly initialized and jointly learned during the training process, this vector is used as a query vector in order to obtain the attention values $\alpha_c \in \mathbb{R}^n$ by measuring the similarity between the elements of the sequence H and the application domain represented by u_h. This similarity is calculated in the Equation 11 by calculating the scalar dot product of u_h^T and H; the resulting values are smoothed with the use of a softmax function.

$$\alpha_c = softmax(u_h^T \cdot H) \tag{11}$$

7.3.2 Recurrent Neural Network

A recurrent neural network (RNN) is a class of ANN with connections between nodes form a directed or undirected graph along a temporal sequence. This allows it to exhibit temporal dynamic behavior. Derived from feed-forward neural networks, RNNs can use their internal state (memory) to process variable length sequences of inputs [76]. RNNs are widely used to perform the sequence analysis process as they are designed for extracting the contextual information by defining the dependencies between various time stamps.

RNNs preserve the sequential information in the hidden states of the network, and affect the processing of each new example to find correlations between events that are separated for different moments. Just as human memory travels in a sequence way through our brain, affecting the behavior without using the full information; the information that travels in the hidden states of the RNNs affect the decisions without revealing all learned. The process of preserving memory in these networks are represented by $h_t = \phi(Wx_t + Uh_{t-1})$ where the hidden state at time step t is h_t. In this function, the input at the same step x_t is modified by a weight matrix W and is added to a hidden state of the previous time step that is represented by h_{t-1} multiplied by the hidden state in the previous time in matrix U. Figure 16 presents a simple example of a RNN unit.

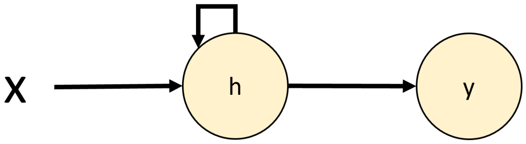


Figure 16: Example of a RNN unit.

RNNs are very good dynamic systems, but they present a problem maintaining the relation of long sequences because the back-propagated gradient shrink at each time step and after many steps vanish [93]. To solve this problem, the use of explicit memory is proposed. Two main neural architectures stand out: the Long Short-Term Memory (LSTM) and the Gated Recurrent Unit (GRU) networks. Both NN use special hidden units and learn to remember inputs of the sequence for a long time. These hidden units are called memory cells, a gated neuron that leaks information through the time. Each memory cell has a connection to itself at the next time step, where it copies the value of the actual state and accumulates the new values, and have a multiplicative gate by another memory cell that learns to decide to clear or keep the content of the memory [76]. For more details related to the LSTM and GRU neural networks, we refer the reader to the following works: [94, 95].

7.3.3 Transformer Deep Neural Network

TNN is a neural network architecture based on the self-attention mechanism, it dispenses the usage of recurrence and convolutions. The TNN was proposed by [67] and arose in the context of sequence-sequence learning; specifically in the task of NMT, where the RNNs used to be the most popular and powerful architecture for the Encoder-Decoder structure to solve NMT problems. However, in recent works [24, 96, 77], the usage of this architectures improves the results in sequence related tasks, such as: one and two sentence classification tasks. The Transformer model with its encoder and decoder components is illustrated in Figure 17, both the encoder and decoder are composed of multiple identical encoders and decoders that can be stacked on top of each other Nx times. The encoder stack and the decoder stack share the same number of Nx.

Encoder: the encoder block is a stack of Nx identical layers. Each layer has a multi-head self-attention mechanism sub-layer followed by a position-wise fully connected feed-forward network sub-layer, as seen in the left part of Figure 17.

Decoder: the decoder block is also a stack of Nx identical layers. In addition to the two sub-layers in each encoder layer, the decoder has an extra Masked Multi-Head Attention sub-layer to avoid this attention sub-layer looking into later stages.

For more details related to the TNN, we refer the reader to the following work: [67].

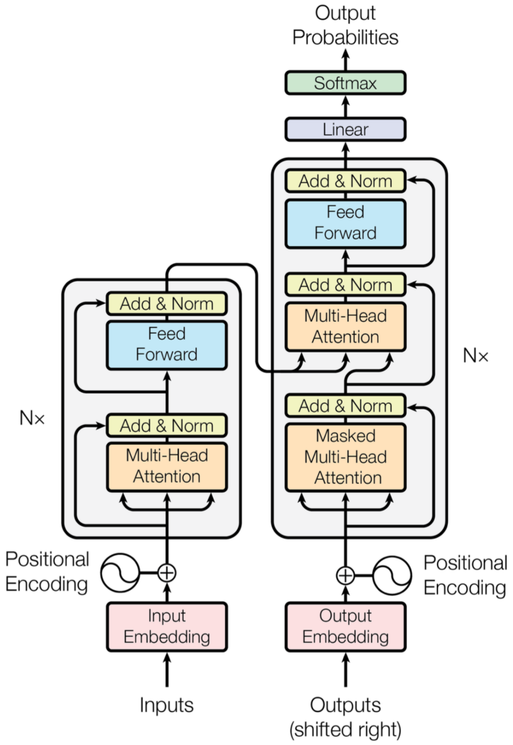


Figure 17: Transformer DNN architecture. Figure taken from [67].

References

[1] J. Guberman and L. Hemphill, “Challenges in modifying existing scales for detecting harassment in individual tweets,” *Proceedings of 50th Annual Hawaii International Conference on System Sciences (HICSS)*, 2017.

[2] F. Poletto, V. Basile, M. Sanguinetti, C. Bosco, and V. Patti, “Resources and benchmark corpora for hate speech detection: a systematic review,” *Language Resources and Evaluation*, 09 2020.

[3] N. Cecillon, V. Labatut, R. Dufour, and G. Linarès, “Abusive language detection in online conversations by combining content- and graph-based features,” *Frontiers in Big Data*, vol. 2, 2019.

[4] S. Kiritchenko and I. Nejadgholi, “Towards ethics by design in online abusive content detection,” *ArXiv*, vol. abs/2010.14952, 2020.

[5] R. Kumar, A. N. Reganti, A. Bhatia, and T. Maheshwari, “Aggression-annotated corpus of Hindi-English code-mixed data,” in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, (Miyazaki, Japan), European Language Resources Association (ELRA), 2018.

[6] S. MacAvaney, H.-R. Yao, E. Yang, K. Russell, N. Goharian, and O. Frieder, “Hate speech detection: Challenges and solutions,” *PLOS ONE*, vol. 14, pp. 1–16, 08 2019.

[7] U. Naseem, S. K. Khan, M. Farasat, and f. ali, “Abusive language detection: A comprehensive review,” *Indian Journal of Science and Technology*, vol. 12, pp. 1–13, 12 2019.

[8] M. Corazza, S. Menini, E. Cabrio, S. Tonelli, and S. Villata, “A multilingual evaluation for online hate speech detection,” *ACM Transactions on Internet Technology*, vol. 20, pp. 1–22, 03 2020.

[9] A. Schmidt and M. 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*, (Valencia, Spain), pp. 1–10, Association for Computational Linguistics, 2017.

[10] P. Fortuna and S. Nunes, “A survey on automatic detection of hate speech in text,” *ACM Computing Surveys*, vol. 51, no. 4, 2018.

[11] T. Chakrabarty, K. Gupta, and S. Muresan, “Pay “attention” to your context when classifying abusive language,” in *Proceedings of the Third Workshop on Abusive Language Online*, pp. 70–79, Association for Computational Linguistics, 2019.

[12] R. T. Mutanga, N. Naicker, and O. O. Olugbara, “Hate speech detection in twitter using transformer methods,” *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 9, 2020.

[13] D. Kiela, H. Firooz, A. Mohan, V. Goswami, A. Singh, P. Ringshia, and D. Testuggine, “The hateful memes challenge: Detecting hate speech in multimodal memes,” in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[14] C. Sharma, D. Bhageria, W. Scott, S. PYKL, A. Das, T. Chakraborty, V. Pulabaigari, and B. Gambäck, “SemEval-2020 task 8: Memotion analysis- the visuo-lingual metaphor!,” in *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, (Barcelona (online)), pp. 759–773, International Committee for Computational Linguistics, 2020.

[15] S. Suryawanshi, B. R. Chakravarthi, M. Arcan, and P. Buitelaar, “Multimodal meme dataset (MultiOFF) for identifying offensive content in image and text,” in *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, pp. 32–41, European Language Resources Association (ELRA), 2020.

[16] Y. Wenjie and Z. Arkaitz, “Towards generalisable hate speech detection: a review on obstacles and solutions,” *PeerJ Computer Science*, 2021.

[17] P. Burnap and M. Williams, “Us and them: identifying cyber hate on twitter across multiple protected characteristics,” *EPJ Data Science*, vol. 5, p. 11, 12 2016. https://doi.org/10.1140/epjds/s13688-016-0072-6.

[18] C. Nobata, J. Tetreault, A. Thomas, Y. Mehdad, and Y. Chang, “Abusive language detection in online user content,” in *Proceedings of the 25th International Conference on World Wide Web*, WWW ’16, (Republic and Canton of Geneva,

CHE), p. 145–153, International World Wide Web Conferences Steering Committee, 2016.

[19] W. Zeerak and H. Dirk, “Hateful symbols or hateful people? predictive features for hate speech detection on twitter,” in *Proceedings of the NAACL Student Research Workshop*, pp. 88–93, Association for Computational Linguistics, 2016.

[20] A. Gaydhani, V. Doma, S. Kendre, and L. B B, “Detecting hate speech and offensive language on twitter using machine learning: An n-gram and tfidf based approach,” in *IEEE International Advance Computing Conference 2018*, 09 2018.

[21] Z. Zhang, D. Robinson, and J. A. Tepper, “Detecting hate speech on twitter using a convolution-gru based deep neural network,” in *ESWC*, 2018.

[22] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, “Deep contextualized word representations,” in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, (New Orleans, Louisiana), pp. 2227–2237, Association for Computational Linguistics, June 2018.

[23] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, “Language models are unsupervised multitask learners,” ., 2019.

[24] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Association for Computational Linguistics, 2019.

[25] P. Liu, W. Li, and L. Zou, "NULI at SemEval-2019 task 6: Transfer learning for offensive language detection using bidirectional transformers," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, pp. 87–91, Association for Computational Linguistics, 2019.

[26] A. Nikolov and V. Radivchev, "Nikolov-radivchev at SemEval-2019 task 6: Offensive tweet classification with BERT and ensembles," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, (Minneapolis, Minnesota, USA), pp. 691–695, Association for Computational Linguistics, 2019.

[27] M. Mozafari, R. Farahbakhsh, and N. Crespi, "A BERT-based transfer learning approach for hate speech detection in online social media," in *Complex Networks 2019: 8th International Conference on Complex Networks and their Applications*, vol. Studies in Computational Intelligence book series (SCI, volume 881) of *Complex Networks and Their Applications VIII : Volume 1, Proceedings of the Eighth International Conference on Complex Networks and Their Applications*, (Lisbonne, Portugal), pp. 928–940, Springer, 2019.

[28] A. Shrivastava, R. Pupale, and P. Singh, "Enhancing aggression detection using gpt-2 based data balancing technique," in *2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS)*, pp. 1345–1350, 2021.

[29] T. Davidson, D. Warmsley, M. W. Macy, and I. Weber, "Automated hate speech detection and the problem of offensive language," in *Proceedings of the Eleventh International Conference on Web and Social Media, ICWSM 2017, Montréal, Québec, Canada, May 15-18, 2017*, pp. 512–515, AAAI Press, 2017.

[30] S. Abro, S. Shaikh, Z. H. Khand, Z. Ali, S. Khan, and G. Mujtaba, "Automatic hate speech detection using machine learning: A comparative study," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 8, 2020.

[31] P. Badjatiya, S. Gupta, M. Gupta, and V. Varma, "Deep learning for hate speech detection in tweets," in *Proceedings of the 26th International Conference on World Wide Web Companion*, (Republic and Canton of Geneva, CHE), International World Wide Web Conferences Steering Committee, 2017.

[32] B. Gambäck and U. K. Sikdar, "Using convolutional neural networks to classify hate-speech," in *Proceedings of the First Workshop on Abusive Language Online*, pp. 85–90, Association for Computational Linguistics, 2017.

[33] G. D. Ashwin, I. Irina, and F. Dominique, "Classification of hate speech using deep neural networks.," *Revue de l'Information Scientifique et Technique*, vol. 25, no. 1, pp. 1–12, 2020.

[34] P. K. Roy, A. K. Tripathy, T. K. Das, and X.-Z. Gao, "A framework for hate speech detection using deep convolutional neural network," *IEEE Access*, vol. 8, pp. 204951–204962, 2020.

[35] A. S. Saksesi, M. Nasrun, and C. Setianingsih, "Analysis text of hate speech detection using recurrent neural network," in *2018 International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC)*, pp. 242–248, 2018.

[36] G. K. Pitsilis, H. Ramampiaro, and H. Das, Langseth, "Effective hate-speech detection in twitter data using recurrent neural networks," *Applied Intelligence*, vol. 48, pp. 4730–4742, 2018.

[37] T. Huynh, D.-V. Nguyen, K. Nguyen, N. Nguyen, and A. Nguyen, "Hate speech detection on vietnamese social media text using the bi-gru-lstm-cnn model," in *Proceedings of the Sixth International Workshop on Vietnamese Language and Speech Processing (VLSP 2019)*, 10 2019.

[38] R. Duwairi, A. Hayajneh, and M. Quwaider, "A deep learning framework for automatic detection of hate speech embedded in arabic tweets," *Arabian Journal for Science and Engineering*, vol. 46, pp. 1–14, 02 2021.

[39] S. Chaudhari, G. Polatkan, R. Ramanath, and V. Mithal, "An attentive survey of attention models," *Association for Computing Machinery*, vol. 37, 2020.

[40] J. Pavlopoulos, P. Malakasiotis, and I. Androutsopoulos, "Deeper attention to abusive user content moderation," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, (Copenhagen, Denmark), pp. 1125–1135, Association for Computational Linguistics, 2017.

[41] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1480–1489, Association for Computational Linguistics, 2016.

[42] G. L. D. la Peña Sarracén, R. G. Pons, C. E. Muñiz-Cuza, and P. Rosso, "Hate speech detection using attention-based lstm," in *EVALITA@CLiC-it*, 2018.

[43] H. J. Jarquín-Vásquez, M. Montes-y Gómez, and L. Villaseñor-Pineda, "Not all swear words are used equal: Attention over word n-grams for abusive language identification," in *Pattern Recognition* (K. M. Figueroa Mora, J. Anzurez Marín, J. Cerda, J. A. Carrasco-Ochoa, J. F. Martínez-Trinidad, and J. A. Olvera-López, eds.), (Cham), pp. 282–292, Springer International Publishing, 2020.

[44] P. Rani, S. Suryawanshi, K. Goswami, B. R. Chakravarthi, T. Fransen, and J. P. McCrae, "A comparative study of different state-of-the-art hate speech detection methods in Hindi-English code-mixed data," in *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, (Marseille, France), pp. 42–48, European Language Resources Association (ELRA), May 2020.

[45] G. Kovács and R. Alonso, Pedro andSaini, "Challenges of hate speech detection in social media," *SN Computer Science*, vol. 2, 2021.

[46] T. H. Afridi, A. Alam, M. N. Khan, J. Khan, and Y. Lee, "A multimodal memes classification: A survey and open research issues," *CoRR*, vol. abs/2009.08395, 2020.

[47] T. Baltrusaitis, C. Ahuja, and L.-P. Morency, "Multimodal machine learning: A survey and taxonomy," *IEEE Transactions on Pattern Analysis and Machine Intelligence.*, vol. 41, no. 2, p. 423–443, 2019.

[48] B. Oriol, C. Canton-Ferrer, and X. G. i Nieto, "Hate speech in pixels: Detection of offensive memes towards automatic moderation," in *NeurIPS 2019 Workshop on AI for Social Good*, (Vancouver, Canada), 09/2019 2019.

[49] V. Keswani, S. Singh, S. Agarwal, and A. Modi, "Iitk at semeval-2020 task 8: Unimodal and bimodal sentiment analysis of internet memes," in *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, 07 2020.

[50] R. Gomez, J. Gibert, L. Gómez, and D. Karatzas, "Exploring hate speech detection in multimodal publications," in *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1459–1467, 2020.

[51] M. G. Constantin, D.-S. Pârvu, C. Stanciu, D. Ionascu, and B. Ionescu, "Hateful meme detection with multimodal deep neural networks," in *2021 International Symposium on Signals, Circuits and Systems (ISSCS)*, pp. 1–4, 2021.

[52] H. Kirk, Y. Jun, P. Rauba, G. Wachtel, R. Li, X. Bai, N. Broestl, M. Doff-Sotta, A. Shtedritski, and Y. M. Asano, "Memes in the wild: Assessing the generalizability of the hateful memes challenge dataset," in *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, (Online), pp. 26–35, Association for Computational Linguistics, Aug. 2021.

[53] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems* (F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds.), vol. 25, Curran Associates, Inc., 2012.

[54] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings* (Y. Bengio and Y. LeCun, eds.), 2015.

[55] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, 2015.

[56] R. K.-W. Lee, R. Cao, Z. Fan, J. Jiang, and W.-H. Chong, "Disentangling hate in online memes," in *Proceedings of the 29th ACM International Conference on Multimedia*, MM '21, (New York, NY, USA), p. 5138–5147, Association for Computing Machinery, 2021.

[57] Y. Zhou, Z. Chen, and H. Yang, "Multimodal learning for hateful memes detection," in *2021 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, (Los Alamitos, CA, USA), pp. 1–6, IEEE Computer Society, jul 2021.

[58] Z. Yu, J. Yu, Y. Cui, D. Tao, and Q. Tian, "Deep modular co-attention networks for visual question answering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[59] J. Lu, D. Batra, D. Parikh, and S. Lee, "Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks," in *Advances in Neural Information Processing Systems*, vol. 32, Curran Associates, Inc., 2019.

[60] L. H. Li, M. Yatskar, D. Yin, C. Hsieh, and K. Chang, "Visualbert: A simple and performant baseline for vision and language," *CoRR*, vol. abs/1908.03557, 2019.

[61] W. Su, X. Zhu, Y. Cao, B. Li, L. Lu, F. Wei, and J. Dai, "Vl-bert: Pre-training of generic visual-linguistic representations," in *International Conference on Learning Representations*, 2020.

[62] E. Fersini, D. Nozza, and P. Rosso, "Overview of the evalita 2018 task on automatic misogyny identification (AMI)," in *Proceedings of the Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2018) co-located with the Fifth Italian Conference on Computational Linguistics (CLiC-it)* (T. Caselli, N. Novielli, V. Patti, and P. Rosso, eds.), vol. 2263 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2018.

[63] Z. Marcos, M. Shervin, N. Preslav, R. Sara, N. Farra, and R. Kumar, "Semeval-2019 task 6: Identifying and categorizing offensive language in social media (offenseval)," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, pp. 75–86, Association for Computational Linguistics, 2019.

[64] M. Zampieri, P. Nakov, S. Rosenthal, P. Atanasova, G. Karadzhov, H. Mubarak, L. Derczynski, Z. Pitenis, and c. Çöltekin, "SemEval-2020 Task 12: Multilingual Offensive Language Identification in Social Media (OffensEval 2020)," in *Proceedings of SemEval*, 2020.

[65] Y. Zhou, "A review of text classification based on deep learning," in *Proceedings of the 2020 3rd International Conference on Geoinformatics and Data Analysis*, ICGDA 2020, (New York, NY, USA), p. 132–136, Association for Computing Machinery, 2020.

[66] W. Guo, J. Wang, and S. Wang, "Deep multimodal representation learning: A survey," *IEEE Access*, vol. 7, pp. 63373–63394, 2019.

[67] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems* (I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds.), vol. 30, Curran Associates, Inc., 2017.

[68] A. H. Mohammed and A. H. Ali, "Survey of BERT (bidirectional encoder representation transformer) types," in *2nd International Conference on Physics and Applied Sciences (ICPAS 2021)*, vol. 1963, p. 012173, IOP Publishing, jul 2021.

[69] T. Caselli, V. Basile, J. Mitrović, and M. Granitzer, "HateBERT: Retraining BERT for abusive language detection in English," in *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, (Online), pp. 17–25, Association for Computational Linguistics, Aug. 2021.

[70] S. Khan, M. Naseer, M. Hayat, S. W. Zamir, F. S. Khan, and M. Shah, "Transformers in vision: A survey," *ACM Comput. Surv.*, dec 2021. Just Accepted.

[71] I. Beltagy, K. Lo, and A. Cohan, "SciBERT: A pretrained language model for scientific text," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, (Hong Kong, China), pp. 3615–3620, Association for Computational Linguistics, Nov. 2019.

[72] A. Mogadala, M. Kalimuthu, and D. Klakow, "Trends in integration of vision and language research: A survey of tasks, datasets, and methods," *ArXiv*, vol. abs/1907.09358, 2019.

[73] Z. Niu, G. Zhong, and H. Yu, "A review on the attention mechanism of deep learning," *Neurocomputing*, vol. 452, pp. 48–62, 2021.

[74] D. Li and Y. Dong, "Deep learning: Methods and applications," *Foundations and Trends® in Signal Processing*, vol. 7, no. 3–4, pp. 197–387, 2014.

[75] M. Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M. S. Nasrin, M. Hasan, B. C. Van Essen, A. A. S. Awwal, and V. K. Asari, "A state-of-the-art survey on deep learning theory and architectures," *Electronics*, vol. 8, no. 3, 2019.

[76] Y. Yu, X. Si, C. Hu, and J. Zhang, "A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures," *Neural Computation*, vol. 31, pp. 1235–1270, 07 2019.

[77] V. Cohen and A. Gokaslan, "Opengpt-2: Open language models and implications of generated text," *XRDS*, vol. 27, p. 26–30, sep 2020.

[78] N. Gaw, S. Yousefi, and M. R. Gahrooei, "Multimodal data fusion for systems improvement: A review," *IISE Transactions*, vol. 0, no. 0, pp. 1–19, 2021.

[79] J. Golbeck, Z. Ashktorab, R. O. Banjo, A. Berlinger, S. Bhagwan, C. Buntain, P. Cheakalos, A. A. Geller, Q. Gergory, R. K. Gnanasekaran, R. R. Gunasekaran, K. M. Hoffman, J. Hottle, V. Jienjitlert, S. Khare, R. Lau, M. J. Martindale, S. Naik, H. L. Nixon, P. Ramachandran, K. M. Rogers, L. Rogers, M. S. Sarin, G. Shahane, J. Thanki, P. Vengataraman, Z. Wan, and D. M. Wu, "A large labeled corpus for online harassment research," in *Proceedings of the 2017 ACM on Web Science Conference*, WebSci '17, (New York, NY, USA), p. 229–233, Association for Computing Machinery, 2017.

[80] T. Mandl, S. Modha, P. Majumder, D. Patel, M. Dave, C. Mandlia, and A. Patel, "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*, FIRE '19, (New York, NY, USA), p. 14–17, Association for Computing Machinery, 2019.

[81] K. Clark, U. Khandelwal, O. Levy, and C. D. Manning, "What does BERT look at? an analysis of bert's attention," *CoRR*, vol. abs/1906.04341, 2019.

[82] C. Baziotis, N. Pelekis, and C. Doulkeridis, "Datastories at semeval-2017 task 4: Deep lstm with attention for message-level and topic-based sentiment analysis," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, (Vancouver, Canada), pp. 747–754, Association for Computational Linguistics, 2017.

[83] T. Mikolov, E. Grave, P. Bojanowski, C. Puhrsch, and A. Joulin, "Advances in pre-training distributed word representations," in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, European Language Resources Association (ELRA), 2018.

[84] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings* (Y. Bengio and Y. LeCun, eds.), 2015.

[85] M. Mozafari, R. Farahbakhsh, and N. Crespi, "A bert-based transfer learning approach for hate speech detection in online social media," in *Complex Networks and Their Applications VIII - Volume 1 Proceedings of the Eighth International Conference on Complex Networks and Their Applications COMPLEX NETWORKS 2019, Lisbon, Portugal, December 10-12, 2019*, vol. 881 of *Studies in Computational Intelligence*, pp. 928–940, Springer, 2019.

[86] P. Saha, B. Mathew, P. Goyal, and A. Mukherjee, "Hateminers : Detecting hate speech against women," *CoRR*, vol. abs/1812.06700, 2018.

[87] B. Wang, Y. Ding, S. Liu, and X. Zhou, "Ynu_wb at HASOC 2019: Ordered neurons LSTM with attention for identifying hate speech and offensive language," in *Working Notes of FIRE 2019 - Forum for Information Retrieval Evaluation, Kolkata, India, December 12-15, 2019* (P. Mehta, P. Rosso, P. Majumder, and M. Mitra, eds.), vol. 2517 of *CEUR Workshop Proceedings*, pp. 191–198, CEUR-WS.org, 2019.

[88] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine Learning Research*, vol. 7, no. 1, pp. 1–30, 2006.

[89] J. Gardner and C. Brooks, "A statistical framework for predictive model evaluation in moocs," in *Proceedings of the Fourth (2017) ACM Conference on Learning @ Scale*, L@S '17, (New York, NY, USA), p. 269–272, Association for Computing Machinery, 2017.

[90] J. Arevalo, T. Solorio, M. Montes, and F. González, "Gated multimodal networks," *Neural Computing and Applications*, pp. 1433–3058, 02 2020.

[91] C. Aggarwal and C. Zhai, "A survey of text classification algorithms," *Mining Text Data*, vol. 9781461432234, pp. 163–222, Aug. 2012.

[92] Y. Bengio, "Learning deep architectures for ai," *Foundations and Trends® in Machine Learning*, vol. 2, no. 1, p. 1–127, 2009.

[93] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[94] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[95] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *3rd International Conference on Learning Representations, ICLR 2015*, Jan. 2015.

[96] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems*

(H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, eds.), vol. 33, pp. 1877–1901, Curran Associates, Inc., 2020.