Affective behavior modeling on social networks

Waleed Ragheb

THESIS SUBMITTED FOR THE DEGREE OF DOCTOR OF PHILOSOPHY (PH. D.) FROM THE UNIVERSITY OF MONTPELLIER

In Computer Science

Information, Structure, Systems (I2S) graduate school

Laboratory of Informatics, Robotics and Microelectronics of Montpellier (LIRMM), France

Affective Behavior Modeling on Social Networks

Presented by Waleed RAGHEB
On November 6th, 2020

Under the supervision of Jérôme AZE, Sandra BRINGAY and Maximilien SERVAJEAN

In front of a jury composed of

Alexis JOLY, DR, INRIA Sophia-Antipolis, ZENITH team – LIRMM - France	President
Osmar ZAÏANE, PR, Department of Computing Science - University of Alberta - Canada	Reviewer
Patrice BELLOT, PR, Aix-Marseille University – LSIS - France	Reviewer
Aurélie NEVEOL, MCF HDR, Paris Saclay University, CNRS, LIMSI- France	Examiner
David LOSADA, MCF, University of Santiago de Compostela - Spain	Examiner
Jérôme AZE, PR, University of Montpellier - France	Director
Sandra BRINGAY, PR, Paul Valéry University - Montpellier 3 - France	Director
Maximilien SERVAJEAN, MCF, Paul Valéry University - Montpellier 3 - France	Co-Director

UNIVERSITÉ DE MONTPELLIER

IUT Béziers

THÈSE POUR OBTENIR LE GRADE DE DOCTEUR DE L'UNIVERSITÉ DE MONTPELLIER

En Informatique

École doctorale Information Structures Systèmes (I2S)

Le Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier (LIRMM), France

Modélisation des sentiments sur les réseaux sociaux

Présentée par Waleed RAGHEB

Le Novembre 6ème, 2020

Sous la direction de Jérôme AZE, Sandra BRINGAY et Maximilien SERVAJEAN

Devant le jury composé de

Alexis JOLY, DR, INRIA Sophia-Antipolis, ZENITH team – LIRMM - France	Président
Osmar ZAÏANE, PR., Department of Computing Science - University of Alberta - Canada	Rapporteur
Patrice BELLOT, PR, Aix-Marseille Université – LSIS - France	Rapporteur
Aurélie NEVEOL, MCF HDR, Université Paris Saclay, CNRS, LIMSI- France	Examinateur
David LOSADA, MCF, University of Santiago de Compostela - Spain	Examinateur
Jérôme AZE, PR, Université de Montpellier - France	Directeur
Sandra BRINGAY, PR, Paul Valéry University - Montpellier 3 - France	Directeur
Maximilien SERVAJEAN, MCF, Paul Valéry University - Montpellier 3 - France	Co- Directeur

UNIVERSITÉ DE MONTPELLIER

IUT Béziers

University of Montpellier

Doctoral Thesis

Affective Behavior Modeling on Social Networks

Author:
Waleed Ragheb

Supervisor:
Prof. Jérôme Azé
Prof. Sandra Bringay
Dr. Maximilien Servajean

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

in the

ADVANSE: ADVanced Analytics for data SciencE
Laboratoire d'Informatique, Robotique et Microélectornique de Montpellier (LIRMM)

Declaration of Authorship

I, Waleed Ragheb, declare that this thesis titled, "Affective Behavior Modeling on Social Networks" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: _____

Date: _____

"*True wisdom is less presuming than folly. The wise man doubteth often, and changeth his mind; the fool is obstinate, and doubteth not; he knoweth all things but his own ignorance.*"

Pharaoh Akhenaton

UNIVERSITY OF MONTPELLIER

Abstract

Computer Science
Laboratoire d'Informatique, Robotique et Microélectornique de Montpellier
(LIRMM)

Doctor of Philosophy

Affective Behavior Modeling on Social Networks

by Waleed RAGHEB

Affective Computing (AC) is an emerging area of research that aims to develop intelligent computer systems that can recognize, synthesize, and respond to the various concepts of human affect. It is one of the most important tracks in Artificial Intelligence (AI) research. AC is deemed to be the tipping point to move from the narrow cognitive definition of AI to more general, sentient , and emotional AI. As emotions plays an essential role in human-to-human communications, machines should also have the fluidity or flexibility to have emotionally-driven responses to situations. Humans use multiple pathways to communicate their affect including facial expressions, gesture, body language, tone of voice, language and verbal cues. With the vast increase of textual user-generated content on social media networks, the detection of human affect from text became an imperative need. Many tasks in Natural Language Processing (NLP) are directly related to affect recognition such as sentiment analysis, opinion mining, abusive language, at-risk user detection, and also those concerning human-computer interactions such as conversational frameworks and chatbots. Subjective and affective concepts in NLP research including feelings, intentions, emotions, moods, and sentiments are used interchangeably. However, bearing in mind the differences of these affect-related terms helps for more reliable and efficient detection systems. Many traditional systems and their modern extensions employ extensive feature engineering steps for text representation including hand-crafted, lexical features, or classical static word embedding. However these models may focus on the important parts of the input text, they disregard other parts and aspects which may harm model generalization for different affective states of the different affective concepts.

In order to mitigate these limitations, we introduce different models that use/extend advanced NLP deep learning models for more reliable text representation. These models use the transfer learning capabilities and are empowered with the attention mechanisms to consider all the contextual information with varied emphasis on different parts with a higher influence on the decisions. Moreover, the proposed models accord special attention to the characteristic differences of the different affective concepts. We consider the affective characteristics of the most important conscious affect-driven subjectivity concepts, precisely, the sentiment, emotion, and mood:

- **Sentiment:** We addressed the problem of sentiment analysis and proposed a deep learning model that applies transfer learning and multi-levels of self-attention layers to focus on the most important parts of the text that have a high influence on sentiments. The model is evaluated on several datasets and shows very competitive results. Furthermore, we evaluate the impact of attention mechanisms on the model's interpretability and user perceptions.

- **Emotion:** We tackle the problem of detection and classification of basic emotions in textual dialogues. We extend the basic model used for sentiment classification to model textual conversations and track the emotion over turns. We participate in the SemEval-2019 shared task on contextual emotion detection in text. The model shows very competitive results and ranked 9^{th} out of more than 150 participants.

- **Mood-I:** However user mood can be classified into two main types - positive and negative mood, mood disturbances inflict various mental illnesses/disorders. We consider the problem of early detection of depression, anorexia, and self-harm using users' writings on Reddit. We proposed a new multi-stage architecture that models users' temporal mood variations. We participated in eRisk-2018 and eRisk-2019 tasks. The proposed models perform comparably to other contributions and ranked the 2^{nd} out of 13 teams in eRisk-2019.

- **Mood-II:** We foster the study of the mood consequences to include the problem of suicide thoughts detection. Therefore, we propose a novel backbone-independent model that uses state-of-the-art Transformer-based models through Negative Correlation Learning (NCL) configuration. We evaluate the model on different tasks for at-risk users detection. The models achieve significant improvements over the existing state-of-the-art results reported for five out of six tasks for the different risk sources.

UNIVERSITÉ DE MONTPELLIER

Résumé

Département Informatique
Laboratoire d'Informatique, Robotique et Microélectornique de Montpellier
(LIRMM)

Docteur en Philosophie

Modélisation des sentiments sur les réseaux sociaux

par Waleed Ragheb

L'informatique affective (AC) est un domaine de recherche émergent qui vise à développer des systèmes informatiques intelligents capables de reconnaître, de synthétiser et de répondre aux différents concepts de l'affect humain. C'est l'une des pistes les plus importantes de la recherche sur l'intelligence artificielle (IA). L'AC est considérée comme le point de basculement permettant de passer de la définition cognitive étroite de l'IA à une IA plus générale, sentimentale et émotionnelle. Comme les émotions jouent un rôle essentiel dans les communications inter-humains, les machines doivent également avoir la fluidité ou la souplesse nécessaires pour réagir aux situations en fonction des émotions. Les humains utilisent de multiples moyens pour communiquer leurs affects, notamment les expressions faciales, les gestes, le langage corporel, le ton de la voix, le langage et les indices verbaux. Avec l'augmentation considérable du contenu textuel généré par les utilisateurs sur les réseaux de médias sociaux, la détection de l'affect humain à partir du texte est devenue un besoin impératif. De nombreuses tâches du traitement du langage naturel (TALN) sont directement liées à la reconnaissance de l'affect, comme l'analyse des sentiments, l'exploration des opinions, la détection du langage abusif, la détection des utilisateurs à risque, et aussi celles concernant les interactions homme-machine, comme les cadres de conversation et les chatbots. Les concepts subjectifs et affectifs dans la recherche en TALN, y compris les sentiments, les intentions, les émotions, les humeurs et les émotions sont utilisés de manière interchangeable. Cependant, garder à l'esprit les différences de ces termes liés à l'affectivité permet d'obtenir des systèmes de détection plus fiables et plus efficaces. De nombreux systèmes traditionnels et leurs extensions modernes utilisent des étapes d'ingénierie de caractéristiques étendues pour la représentation de textes, y compris des caractéristiques lexicales artisanales ou l'intégration de mots statiques classiques. Cependant, ces modèles peuvent se concentrer sur les parties importantes du texte d'entrée, ils ignorent d'autres parties et aspects qui peuvent nuire à la généralisation du modèle pour différents états affectifs des différents concepts affectifs.

Afin d'atténuer ces limitations, nous introduisons différents modèles qui utilisent ou étendent des modèles avancés d'apprentissage profonds utilisés en TALN pour une représentation plus fiable du texte. Ces modèles utilisent les capacités d'apprentissage par transfert et sont dotés de mécanismes d'attention permettant de prendre en compte toutes les informations contextuelles en mettant l'accent sur différentes parties ayant une plus grande influence sur les décisions. En outre, les modèles proposés accordent une attention particulière aux différences caractéristiques des différents concepts affectifs. Nous considérons les caractéristiques affectives des concepts les plus importants de la subjectivité affective consciente, précisément, le sentiment, l'émotion et l'humeur :

- **Sentiment :** Nous avons abordé le problème de l'analyse des sentiments et proposé un modèle d'apprentissage approfondi qui applique l'apprentissage par transfert et des couches d'auto-attention à plusieurs niveaux pour se concentrer sur les parties les plus importantes du texte qui ont une grande influence sur les sentiments. Le modèle est évalué sur plusieurs jeux de données et présente des résultats très compétitifs. En outre, nous évaluons l'impact des mécanismes d'attention sur l'interprétabilité du modèle et les perceptions des utilisateurs.

- **Émotion :** Nous abordons le problème de la détection et de la classification des émotions de base dans les dialogues textuels. Nous étendons le modèle de base utilisé pour la classification des sentiments pour modéliser les conversations textuelles et suivre l'émotion au fil du temps. Nous avons participé à la tâche partagée SemEval-2019 sur la détection des émotions contextuelles dans les textes. Le modèle a obtenu des résultats très compétitifs et s'est classé à 9^{ieme} sur plus de 150 participants.

- **Mood-I :** Cependant, l'humeur des utilisateurs peut être classée en deux types principaux : l'humeur positive et l'humeur négative, les troubles de l'humeur infligeant diverses maladies/désordres mentaux. Nous examinons le problème de la détection précoce de la dépression, de l'anorexie et de l'automutilation en utilisant les écrits des utilisateurs sur Reddit. Nous avons proposé une nouvelle architecture à plusieurs niveaux qui modélise les variations temporelles de l'humeur des utilisateurs. Nous avons participé aux tâches eRisk-2018 et eRisk-2019. Les modèles proposés ont des performances comparables aux autres contributions et ont permis d'atteindre la 2^{eme} place sur 13 équipes dans eRisk-2019.

- **Mood-II :** Nous encourageons l'étude des conséquences sur l'humeur afin d'inclure le problème de la détection des idéations suicidaires. C'est pourquoi nous proposons un nouveau modèle indépendant de l'ossature qui utilise des modèles de pointe basés sur les transformateurs grâce à la configuration de l'apprentissage par corrélation négative (NCL). Nous avons évalués le modèle sur différentes tâches pour la détection des utilisateurs à risque. Les modèles ont permis d'apporter des améliorations significatives par rapport aux meilleurs résultats existants rapportés pour cinq des six tâches pour les différentes sources de risque.

Acknowledgements

A Ph.D. is a long ride in roads riddled with opportunities, difficulties, ups, and downs. I am eternally grateful to many people I met and who escorted me across all ways. Firstly, I would like to thank my Ph.D. supervisors Sandra Bringay, Jérôme Azé, and Maximilien Servajean. Thank you all for the freedom you accorded to me to follow my interests and curiosity. Sandra, thank you for your advice and motivational inflammatory words that always came on time. Jérôme, thank you for your professionalism, substantial support, and providing an encouraging research environment at LIRMM and IUT of Béziers. Maximilien, thank you for the fruitful discussions and the inspiring view along the route.

I'm eternally grateful for all my family and single out my Mother, wife, the two lovely daughters -Yasmine and Hana-,and my sister. They endured a lot along the way with love, support, encouragement, and trust. I'll always owe them a lot. Thank you all; You all accepted and dealt with my mental and/or physical absence from our life in the last three years without imposing any pressure and with all satisfaction

I would like to thank my professors, colleagues, and friends at LIRMM especially the wonderful ADVANSE team. I will be always surprised by their skills, talents, attitudes, and the hard-working. I learned and got a bunch of different experiences. I would like to make a special mention here to our team leader Prof. Pascal Poncelet. Thank you so much; it was a real honor to be part of such a team and looking for more possible cooperation in the future.

I wish to thank all the other jury members of my dissertation committee - Alexis Joly, Osmar Zaïane, Patrice Bellot, Aurélie Neveol, and David Losada - for their time, support, comments, and reviews of this document.

I am thankful for all the wonderful people I had the opportunity to know at various conferences, summer, and spring schools. I am looking forward to seeing you at future events.

Finally, I would like to acknowledge La Région Occitanie and l'Agglomération Béziers Méditerranée which finance the thesis as well as INSERM and CNRS for their financial support of the CONTROV project.

Contents

List of Figures

List of Tables

To my dear father
- to whom I promised to dedicate this dissertation
before he left our world
. . .

Chapter 1

Introduction

Since time immemorial, people believed that human emotion is the mysterious part of the soul controlled by divine thoughts [1]. They have sought to understand emotions' characteristics and how to identify them. This continued to be the case for a long time, even after Darwin's theory of natural selection [49] who replaced the mystery with the evolution. Shortly thereafter, it is marked the birth of psychology. Wilhelm Wundt [118] – the father of experimental psychology – distinguished psychology as a science from philosophy and biology. The emotion research has progressed and developed towards the physical basis of emotions. This was a golden age of emotion research when the mythical inner feeling of emotion became real. Researches have continued to refine and reformulate human intangibles like thoughts, emotions, feelings, personality traits, moods, sentiments, and temperament that have been deemed detectable and measurable. Most of these subjective terms commonly used interchangeably and are used to be covered below the generic terminology of *affect* [237]. The study of human affect crossed disciplinary boundaries between psychology, neuroscience, philosophy, cognitive science, and get into computer science [29].

With the information technology revolution, it became mandatory – not just an option – for many computer systems to express and recognize affects to attain creative and intelligent behavior. The fundamental concepts were first presented by R.W. Picard [187] who introduced the Affective Computing (AC) in 1995 as a new field of study and systems development that can recognize, understand, reproduce, and express human affects. AC has become an emerging and important branch of Artificial Intelligence (AI). The overarching goal is to create systems that can interpret the emotional state of humans and adapt its behavior in order to provide intuitive and appropriate emotionally informed responses [235].

1.1 Affect Detection in Texts

Human affects are not linguistic constructs, language provides convenient access to them. Numerous researches in social psychology studied language as a way of expressing emotions [113, 175, 184]. With the large quantity of texts (particularly affectively oriented e.g., social media), the detection of affective states and concepts from text is becoming increasingly important for more flexible, intelligent, and creative systems. In many Natural Language Processing (NLP) researches, subjective terms of affective concepts like intentions, beliefs, feelings, emotions, mood, and sentiment are used in the same sense. However, considering the differences between them plays an important role in increasing the effectiveness of detection methods.

[1] The ancient Egyptian Weighing of the heart (the seat of emotion) ceremony by Maat. https://en.wikipedia.org/wiki/Ancient_Egyptian_conception_of_the_soul

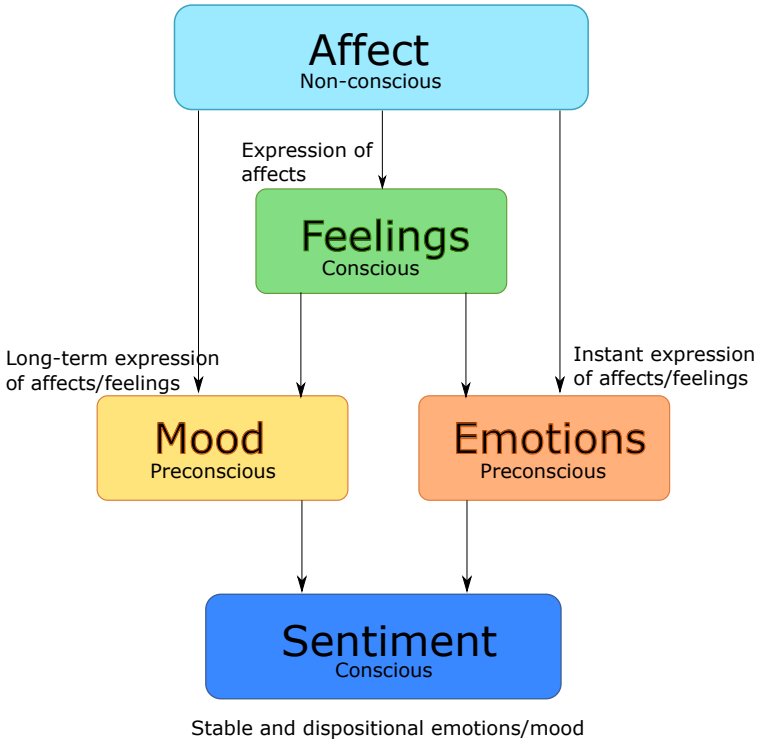


FIGURE 1.1: A visualization summarizes the affect-related terms considered in the thesis – adapted[2]from [46]

A recent study by Munezero et al. [46] considered the differences between these subjective terms in the literature to understand how they relate to each other. The study has led to consider the term *affect* as the most abstract and difficult to realize. It has a non-conscious behavior and is the predecessor to feelings and emotions. In contrast, feelings are personal phenomena that have direct conscious access. Moreover, emotions are preconscious social expressions of feelings and affect unleashed by an external or internal motive. However all emotions are necessarily the projection/display of feelings, not all feelings are considered emotions. Feelings with physical sensations such as hunger or pain are not considered as emotions. Regarding sentiments, it is considered partly social constructs of emotions that developed and continue for a time. The study does not consider, for practical reasons, one of the most crucial affect-related terms - mood. Similarly to human emotions, the mood is directly linked to the feelings, but emotions have high-intensity nature and are very brief and last for an instant. On the other hand, the mood tends to be less intense than an emotion, but it lasts longer in time [203]. Another important perspective differentiating between mood and emotions is the cause. Emotion is caused by a specific event or incident but the mood does not necessarily need a contextual stimulus [69]. The mood is heavily influenced by several factors like the environment, physiology, and mental state. Additionally, the mood impacts the impression of information and person memory [76]. It may have influences on everyday person-perception judgments. Therefore, Emotions and moods could both prompt sentiments.

In the course of this thesis, we considered the detection of sentiment, emotion, and mood from texts. Figure 1.1 summarizes the differences between those terms. The models proposed in this thesis utilize these definitions and differences to tackle the detection problem for the corresponding affective terms and concepts.

[2]The original study does not consider the mood and go further to include the personal interpretations of information for opinions

1.1.1 Motivation

The amount of data and information in general and more particularly those generated by normal users continue to increase rapidly. For example, Figure 1.2 [3] shows the number of active users in online social networks over the last 10 years. The number is tripled to be around 3 billion users in 2019 and constantly growing. About 50% of the users mainly prefer to use social media platforms to express themselves [4]. This is considered a double-edged weapon. On one hand, all this stream of data could have many potentials and provides several opportunities for many sectors such as business, marketing, medical care, education, entertainment, and many more. But, on the other hand, it puts responsibilities on the social media providers in managing, moderating, monitoring, and optimum utilization of this vast amount of data. This increases the demand for more intelligent systems that can detect user affect and intents and differentiate between content types for more appropriate decisions.

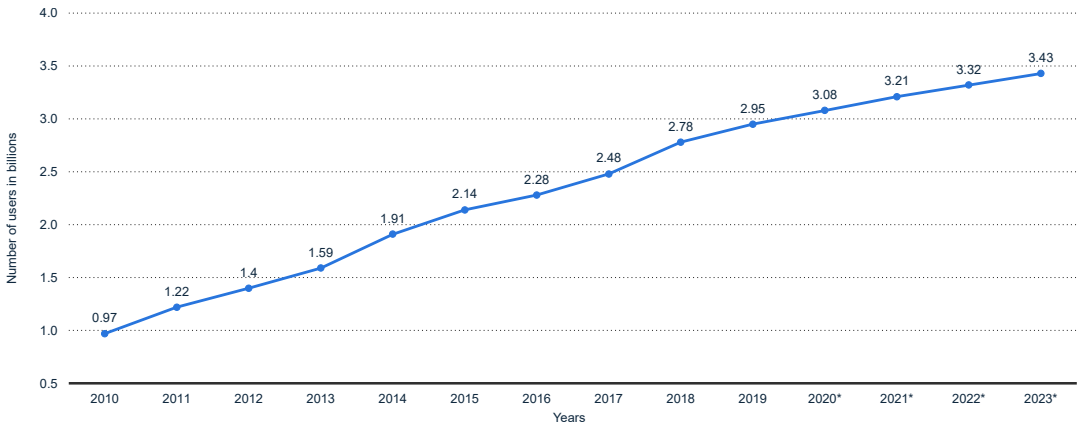


FIGURE 1.2: Number of active social network users worldwide (in billions)

From a related perspective, AI can be classified from business and market point of view into three different types [114]:

- *Analytical AI* has only characteristics consistent with cognitive intelligence generating cognitive representation of the world;
- *Human-inspired AI* has elements from cognitive and emotional intelligence; understanding human emotions and affects;
- *Humanized AI* shows characteristics of all types of competencies (i.e., cognitive, emotional, and social intelligence), is able to be self-conscious and is self-aware in interactions with others.

The classification puts emotional intelligence as a cut-off point to move from analytical AI to human-inspired AI. Therefore, emotional intelligence or, more generally, AC is the major factor towards a strong, broad, and super AI. Many researchers attempt to measure different aspects of emotions and affect covering broad areas, noticeably in Human-Computer Interaction (HCI). Early researches and applications of AC were centering around audio and image modalities. For example, recognition and

[3] Source: https://www.statista.com/statistics/278414/number-of-worldwide-social-network-users/

[4] Source: https://www.statista.com/statistics/1015131/impact-of-social-media-on-daily-life-worldwide/

synthesis of facial expression, and the synthesis of voice intonation and inflection. Furthermore, more than one modality can be combined and fused. Other supplementary modalities also considered for HCI such as brain signals, body, and hand gestures. Due to the tremendous increase of the textual data available, the text is considered a valuable source for detecting human affects. Compared to other modalities and other NLP research points (see Figures 1.3 and 1.4), NLP-based AC is evolving and considered as a hot area of research. More specifically, detection and classification of subjectivity and affective status are still a difficult NLP task and attract the attention of many researchers in AI.

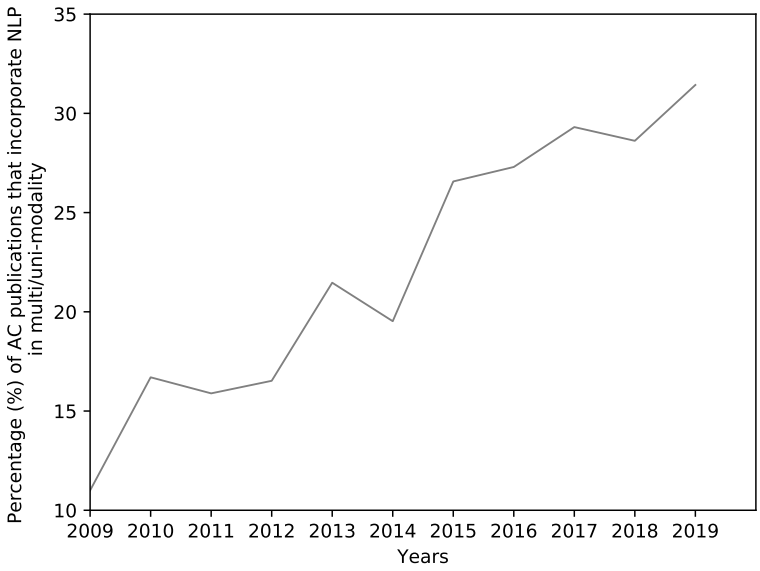


FIGURE 1.3: NLP-based Affective Computing (AC) research over the last 10 years[5]

Additionally, the market for AC is emerging and growing fast, empowered by advancements in technology and motivated by use cases in industries like automotive, healthcare, and customer service. AC uses hardware and software to identify human feelings, behaviors, opinions, moods, tone, and cognitive states through the facial, body (language), and voice detection and recognition technologies. Major technology companies are investing in the AC space such as Amazon (Rekognition capability on AWS), Microsoft (Emotion API), and IBM (Tone Analyzer). Also, intelligent assistants like Alexa (Amazon), Google Assistant or Siri (Apple) is changing the way we collect, qualify, and process everyday data to become more emotional[7] [8].

In addition, enterprises across various verticals work towards enriching and facilitating the interactivity between human and machine through emotional intelligence

[5]Scopus Search API https://dev.elsevier.com/documentation/ScopusSearchAPI.wadl and Google scholar Advanced search https://scholar.google.com/

[6]Statistics on the top 10 NLP venues in the list https://scholar.google.com/citations?view_op=top_venues&hl=en&vq=eng_computationallinguistics

[7]Source: https://developer.amazon.com/en-US/blogs/alexa/alexa-skills-kit/2019/11/new-alexa-emotions-and-speaking-styles

[8]Source: https://www.applegazette.com/siri-2/apples-siri-might-understand-and-interpret-human-emotions-in-the-near-future/

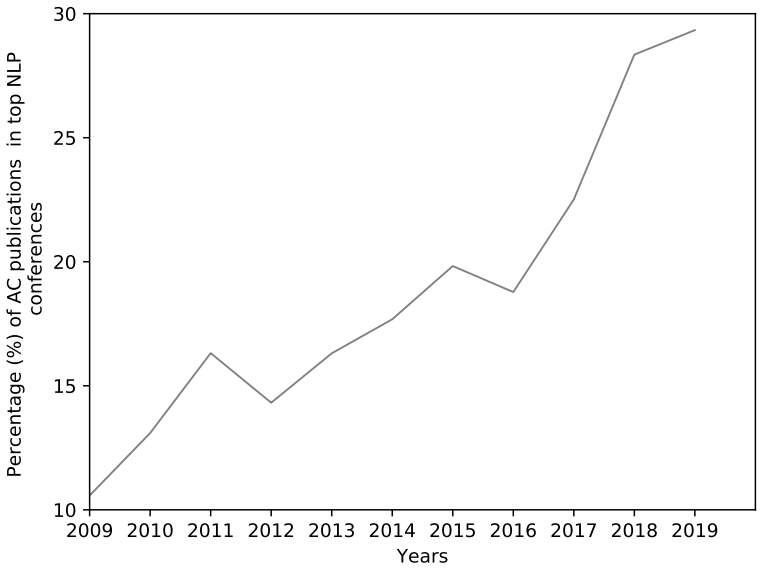


FIGURE 1.4: Affective Computing (AC) publications in top NLP conferences over the last 10 years [6]

and AI. Companies such as Affectiva [9], Kairos AR[10], Nemesysco[11], Neurodata [12], and Sensumco[13] started providing services to the market that empower applications and digital platforms with an affective and emotional capabilities in multimodal manner including NLP. Other enterprises such as Receptiviti[14], Cognitum[15], and DAVI[16] focus mainly on providing AC services through language processing.

1.2 Research Questions and Challenges

The main broad research question posed in this context is *How can we detect and classify the affective status for different affective concepts from the text?*

We considered here all affective concepts that have conscious or preconscious affective behavior (see Figure 1.1). This includes different types of emotions, categories of sentiments, and mood consequences. Detection of affective states from text is well defined as a supervised classification problem in the machine learning context. Accordingly, there are more particular questions which are clustered around the main general question.

Finding good representations of inputs is a very crucial and challenging step to train any machine learning model. Classical NLP models incorporate many handcrafted [27, 30] and lexicon-based [202, 162] features for text representations. The process seemed to be time-consuming, expensive, and fail to adapt to cross-domain

[9] https://www.affectiva.com
[10] https://www.kairos.com
[11] https://www.nemesysco.com/
[12] https://neurodatalab.com
[13] https://sensum.co
[14] https://www.receptiviti.com
[15] https://www.cognitum.eu
[16] https://davi.ai/en/home

tasks. Many works in the literature have investigated finding the mapping of words to vectors into continuous spaces. The process is called text vectorization and we introduce more literature review in Chapter 2. These bring some questions such as can we reply on only text-based features to do the task without using any lexical, behavioral, or demographic features? How can we employ transfer learning to work properly across different affective concepts? Would it be beneficial for these models to be fine-tuned for specific tasks? Would it be helpful while processing the overall text to pay attention to some important segments that have an impact on the decision? We can generalize all the input representation related question into:

Q1: How can we well represent the input without external or domain-specific feature engineering steps while giving varying emphasis to different segments?

Nevertheless, only considering the syntax and semantics in text for affect-related tasks is not enough. Different affective concepts have different characteristics. As discussed in Section 1.1, the sentiment is often exploited for detecting polarity. It reflects the deeper and stable psychological state of the holder, enabling people to reason why they like, dislike something, or to what extent. On the other hand, the temporal behavior in emotions and moods plays an important role in differentiating between them. An emotion follows its eliciting stimuli closely or even instantaneously. While the Mood typically unstable and lasts for longer. Besides, emotions could be classified into a predefined set of categories. A common example is Plutchik's wheel of emotions [189] which classify emotions into 8 basic emotions: joy, trust, fear, surprise, sadness, anticipation, anger, and disgust. Regarding the mood, the most widely accepted states are either positive or negative moods. Unlike emotions, it is difficult to define states that collectively capture the entire content domain of mood [69], but psychologically, it is possible to observe some of its consequences. Taking into account all these differences, the following question arises:

Q2: How can we develop predictive models that are adapted to model the affective characteristics of different affective concepts?

1.3 Thesis Contributions

The thesis addresses the above two questions and introduces pragmatic solutions by proposing four new models for the detection and classification of sentiments, emotions, and moods. We do not rely on any hand-crafted features or affect-related lexicons in the text representation step for all the proposed models. Moreover, we have accorded a special concern to the basic characteristics and feature for each affective concept.

1.3.1 Attentive-based Sentiment Classification Model

We address the problem of sentiment analysis through user review classification in different domains. We propose a new deep learning model that makes use of 1) transfer learning, rather than the classical shallow methods of static word embedding and 2) multi-levels self-attention mechanisms to focus on the most important parts of the text that highly influence the sentiments for different abstraction representation of

the input. Our model was evaluated on several datasets and shows very competitive results. Moreover, we evaluate the impact of attention mechanisms that enable users to have an efficient and meaningful interpretation. We show that such visualization brings them deeper knowledge about how the system proposes sentiment and help to produce explainable systems.

1.3.2 Emotional Classification in Textual Conversations

We consider the problem of classifying emotions that varies through textual conversations. Textual conversations could have many forms such as those existing in the hierarchy of comments on social media posts, chatbots, or text private messages. We proposed a new model for the detection and classification of the basic emotions in users' turns of the conversations. Our proposed model makes use of attentive-based deep transfer learning models and turn-based conversational modeling for classifying the emotions. The model can track users' emotions in each turn in multiple party conversations. Our model was evaluated on the data provided by the SemEval-2019 shared task on contextual emotion detection in text [32]. The model shows very competitive results and ranked 9^{th} out of more than 150 participants.

1.3.3 Temporal Mood Variation Models

We consider the detection of the mood from user writings in social media. We have taken a narrower, and standard definition of mental illness/disorders that are affected by user mood and behaviors, including depression, anorexia, and self-harm. We tackle the problem of early detection of these mental disorders using users' posts on Reddit. We proposed a new architecture that models the temporal mood variation detected from user writings. Our proposed architectures use only textual information and two learning phases through the exploration of state-of-the-art text vectorizations and deep language models. We participated in eRisk-2018 [142] and eRisk-2019 [143] competitions. The proposed models perform comparably to other contributions and ranked the 2^{nd} out of 13 teams in eRisk-2019.

1.3.4 Negatively Correlated Noisy Learners Ensembles

We extend the study of mood detection to include the problem of detecting suicidal ideation and thoughts. We propose the Negatively Correlated Noisy Learners (NCNL) model. The novel deep learning ensemble architecture makes use of multiple noisy base learners in a Negative Correlation Learning (NCL) configuration for text classification. NCNL is designed to be, backbone-independent, and we examine it with modern Transformer-based architectures. We evaluate our models on six different tasks for at-risk user detection and classification. Our models achieve significant improvements over existing state-of-the-art results reported for five out of six tasks for the different risk sources.

1.4 Thesis Outlines

Figure 1.5 shows the organization of the remaining chapters of this thesis. Each chapter references its related papers that have been published within the thesis. The list of these papers is presented in the next section 1.5. This work is organized as follows:

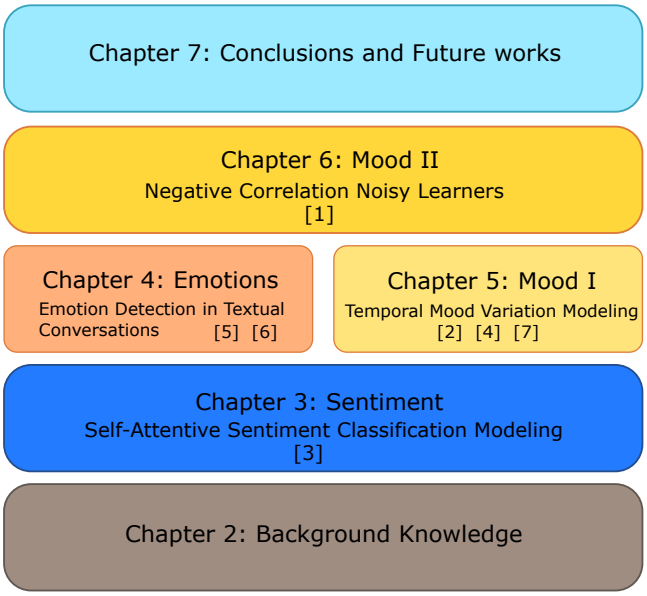


FIGURE 1.5: Organizations of the remaining chapters in the thesis

- Chapter 2 provides an overview of background information and the related work to the contents of this thesis. We review some important definitions of social networks and briefly go through the affect control theory. We furthermore discuss the text vectorization methods in NLP with more emphasis on modern deep learning models. We additionally review the attention mechanism used in these models. We finish this chapter with the state-of-the-art models for affect detection in texts corresponding to each affective term subject to this thesis.

- Chapter 3 tackles the problem of document-level sentiment classification of user reviews. We start with the problem formulation and the related work. We propose a self-attention based deep learning model exploring its transfer learning capabilities through different domains' (movies, product, and restaurant) reviews. We asses through an empirical study how the attention mechanism proposed in the model provides some sort of decision interpretability.

- Chapter 4 addresses the problem of emotion detection and classification in textual conversation. We begin by identifying the problem. We proposed a new model that models the conversational turn-based behavior. In addition, we proposed a set of model variants to analyze the effects of different model elements.

- Chapter 5 presents a new architecture for early detection of some mood and mental disorder. We consider the detection of depression, anorexia, and self-harm through processing a sequence of user writings in social medial. We start by problem definition and task description. We split the chapter into two parts according to the procedure used in processing users' writings -either by chunks or by item. In each procedure, we present possible model variants.

- Chapter 6 tackles the detection and classification of mood and mental disorder. We address the problems mentioned in the previous chapter (depression, anorexia, and self-harm) adding the detection of suicidal ideation and thoughts. We start with the problem definition and related work. We propose a novel framework that uses a set of negatively correlated noisy learner ensembles. The proposed framework is independent of the used models. We examine model possible variants. We finish the chapter with an exhaustive study on the diversity of the proposed ensemble.

- Chapter 7 finally draws conclusions and discussions and provides an outlook into the future.

1.5 Publications

The work on this thesis have led to the following publications:

International Journals

[1] Waleed Ragheb, Jérôme Azé, Sandra Bringay, Maximilien Servajean: Negatively Correlated Noisy Learners for At-risk User Detection on Social Networks: A Study on Depression, Anorexia, Self-harm and Suicide. IEEE Transactions on Knowledge and Date Engineering. (Under Review)

International Conferences

[2] Waleed Ragheb, Jérôme Azé, Sandra Bringay, Maximilien Servajean: Language Modeling in Temporal Mood Variation Models for Early Risk Detection on The Internet. Proceedings of the 10th International Conference of the CLEF Association (CLEF 2019). Lugano, Switzerland (2019)

National Conferences

[3] Waleed Ragheb, Jérôme Azé, Sandra Bringay, Maximilien Servajean: Pourquoi dois-je croire ta prédiction ? Comment expliquer les résultats d'une classification automatique de sentiments à partir de textes. Journées francophones d'Ingénierie des Connaissances (IC-2019). Toulouse, France (2019)

Workshops

[4] Waleed Ragheb, Jérôme Azé, Sandra Bringay, Maximilien Servajean: Attentive Multi-stage Learning for Early Risk Detection of Signs of Anorexia and Self-harm on Social Media. at the CLEF eRisk-2019 Tasks for Early Risk Detection on The Internet (working notes). Lugano, Switzerland (2019)

[5] Waleed Ragheb, Jérôme Azé, Sandra Bringay, Maximilien Servajean: LIRMM-Advanse at SemEval-2019 Task 3: Attentive Conversation Modeling for Emotion Detection and Classification. SemEval@NAACL-HLT 2019: 251-255. Minnesota, USA (2019)

[6] Waleed Ragheb, Jérôme Azé, Sandra Bringay, Maximilien Servajean: Attention-based Modeling for Emotion Detection and Classification in Textual Conversations. 2ndWorkshop on Humanizing AI (HAI)@IJCAI'19. Macao, China (2019)

[7] Waleed Ragheb, Bilel Moulahi, Sandra Bringay, Jérôme Azé, Maximilien Servajean: Temporal Mood Variation: at the CLEF eRisk-2018 Tasks for Early Risk Detection on The Internet (working notes). Avignon, France (2018)

[8] Waleed Ragheb, Bilel Moulahi, Sandra Bringay, Jérôme Azé, Maximilien Servajean: LIRMM@DEFT-2018 – Modèle de classification de la vectorisation des documents. DEFT-2018. Rennes, France (2018)

Chapter 2

Background Knowledge

There is no method but to be very intelligent.

Thomas Eliot

Contents

2.1 Introduction

The chapter contains the related works and background relevant to the contributions presented in this thesis. The chapter provides a good reference concerning many technical and architectural concepts pushing modern NLP systems in general and affective computing in particular. More in-depth related works are included in each chapter.

The chapter starts with an introduction to social media with important definitions, types, and characteristics. Section 2.3 summarizes the Affect Control Theory (ACT). A comprehensive overview of the text vectorization approaches is presented in section 2.4. Section 2.5 gives a brief introduction to attention mechanisms. A brief state-of-the-art on affect detection from textual data is discussed in section 2.6 for the three considered affective terms (emotion, sentiment, and mood).

2.2 Social Media Networks

Social media is usually referred to those types of media that enable interaction and participation among individuals. In the broadcast age, media were centralized. Radio or television station, newspapers, or a movie production studio transmitted their content and messages to the public. The interaction, including the feedback, was indirect, very limited, and delayed. Digital and mobile technologies open the doors for a new media age that enables more interaction and sharing ideas between individuals. Social media users could share their opinions with many and get possible immediate feedback. Almost all media sources tend to involve social media into their regular work for wider and broader reach, more user engagement, and to increase community interactions. However all social media are directly linked to the digital platform, not everything that is digital is considered as social media. In this section, we explore the definitions, types, and characteristics of social media. Besides, we highlight the important role of moderation in managing social media.

2.2.1 Definitions

Kaplan and Haenlein in [115] defined social media as digital platforms, or more specifically, Internet-based applications that enable and allow users to generate, share their contents and react to each other. The term User-Generated Content (UGC) describes the data generated by the end-user that can have many forms and types. According to the Organisation for Economic Co-operation a Development (OECD) [248], UGCs could be images, videos, texts, and audios that should satisfy three conditions and requirements:

1. UGC should be published to accessible and public websites or on social media platforms. This excludes contents like e-mails, private or group instant messages.
2. It should contain user input not only a repetition of other contents without modification or comments. This excludes the content like those generated by just propagating the exact news articles as it is without mention any other information or opinion.
3. It should be created without professional purpose or in a professional routine. This excludes the data created in the commercial market context with the expectation of profit or remuneration.

With the recent rise of information technologies, the growth of the internet, and after the development of Web 2.0 framework, the definition of online UGC is generalized to any content generated by ordinary internet users not crafted content created

by content providers. Users can generate content for many purposes, for example, but not limited to, criticism or review of services or products, connecting with peers, reporting and sharing news, achieving a certain level of fame, the desire to express oneself, advertising, vigilantism, or entertainment. These make just copy/paste or hyperlinking, even without modifications or comments, are considered to be UGC [199]. Text is the most common type of content on various social media platforms. Besides, texts exist in the description of other types of media like images and videos as metadata. Processing textual social media content is considered a challenging task [74]. These are mainly due to the complex characteristics of this content, such as the high length variation, noisy content, explicit and implicit information, multilingualism, and relationships.

2.2.2 Types and Characteristics

According to the previously mentioned definitions, there are many types of social media applications and platforms. Although there is no systematic way of categorizing social media networks, a lot of existing trials exist in the literature. Social networks could be categorized according to the UGC into profile-based, microblogging, and content-based social networks [178]. In profile-based social networks, like Facebook, MySpace, and Google+, users used to express themselves and share content mostly related to professional or personal interests and activities. For microblogging networks, like Twitter, Tumblr, and Plurk, people share content related to a specific and current incident or news. In content-based social networks, such as YouTube, Flickr, and Spotify, they are revolved around the contents shared by their users.

From another perspective and according to the social presence theory [45] and the concept of self-presentation [81], social media networks could be classified into six different types shown in Table 2.1. With modern social media applications, features, and add-ons, one social media platform can be classified to be in more than one category.

Type	Self-presentation *	Social presence **	Example [1]
Blogs	✓	↓	Reddit
Social networking	✓	~	Facebook
Virtual worlds	✓	↑	Second Life
Collaborative Projects	✗	↓	Wikipedia
Content-based Networking	✗	~	YouTube
Virtual game worlds	✗	↑	CrowdStar

* (✓) for high and (✗) for limited self-presentation or self-disclosure.
** high (↑), medium (~) and low (↓) social presence or media richness

TABLE 2.1: Classification of social media networks based on self-presentation and social presence. Adapted from: [115]

Through the thesis, we use different types of UGC including microblogging and social networking writings. In chapter 3, we consider the microblogging user reviews for sentiment classification. In chapter 4, we uses textual conversational contents in social networking. In Chapters 5 and 6, the used datasets contains user writings to some mental health related sub-blogs in Reddit.

[1] Frequently updated list of social networking sites and application in: https://en.wikipedia.org/wiki/List_of_social_networking_websites: last access on February 5, 2021

2.2.3 Content Moderation

The massive use of social networks creates a large volume of content with very high and increasing speed. However, the phenomenon of UGC has numerous sets of benefits to the society, it needs to be monitored and moderated [247]. In the market and commerce point of view, monitoring the UGC has a lot of benefits in modern marketing strategies. As a common example, processing social media users' reviews helps the organizations to point at the weak and strong aspects of their products or services. Analysis of these results help increasing user engagement to the brand.

Content moderation could be mandatory to monitor submissions and applying a set of rules which define what is acceptable and what is not. In almost social media communities, moderators are assigned to filter undesirable content, e.g. hate speech, defamation, cyberbullying, and pornography, to maintain some sense of order and to avoid legal issues [204]. The moderation requires essential human resources. The vast increasing streams of UGC leads to make the moderation process tends to be very complex and costly. Usually, moderation models involve different moderation stages to guarantee the quality of the input. Many moderation types compromise between the high control of the content and fast - real-time - submissions. They are also different in terms of the workload of the human moderators. Some of the most common moderation types are [247]:

- Pre-moderation: Users' submissions are placed in queue waiting for moderation action to be published.
- Post-moderation: submissions are published in real-time. Moderators should review published items in a short time frame.
- Reactive moderation: Moderators rely on the community members to point to any undesirable content. It may look risky, but it allows the community to scale-up.
- Distributed moderation: It involves guidance from different senior moderators. It is usually rare and used for legal and branding reasons.
- Automated moderation: It uses technical tools and solutions to process UGC. These tools apply a set of rules to help the moderators to reject or approve submissions. Automated moderation contribute to saving human intervention and scaling up the community while preserving the rules.

The proposed models discussed in this thesis could be used for automated moderation. The thesis presents various models for the detection of sentiment, emotion, and user mood in social media users' submissions. Some of these models - discussed in chapter 5 and chapter 6 - are designed for early risk detection of some mental and mood disorders, like, depression, anorexia, and suicidal ideation. Early detection of at-risk individuals helps community moderators, especially, in active and dynamic communities.

2.3 Affect Control Theory

Affect Control Theory (ACT), introduced by Heise [96, 94], propose a socio-mathematical model of affect. ACT proposes that individuals maintain affective meanings through their actions and interpretations of events. The theory considers many affective concepts, such as identities, behaviors, settings, personal attributes,

emotions, feelings, and mood. It assumes that these concepts vary in three different dimensions in latent space. Firstly, the evaluation, this describes the goodness or badness of a given concept. The second one is the potency, which scores the powerfulness or powerlessness of such concepts. the third dimension is the activity, which measures the energy or excitedness levels. Measuring the affective concepts using these dimensions yield to positioning them is EPA (Evaluation, Potency, and Activity) space.

Within the EPA space, each concept is defined by it's EPA profile. This profile is decomposed of the three different values in the EPA space, in addition to, the transient affective meaning and the associated sentiment [83]. This transient is complying with the impact and effect created by external recent events. These events modify the three values of the EPA dimensions in a complex manner. The mapping between the transient and original affective concept issued by a set of events could be modeled by a non-linear function obtained through empirical studies and surveys [96]. This function is controlled by the *deflection*, a key element in the theory which is defined as the distance between the *pre-event* transient and *post-event* transient [110]. Events are the smallest elements of a situation in which an actor performs some behavior upon an object person. This means that the perception of the affective situations (events) is defined by three concepts - actor, behavior, and object - each is defined by three different values in the EPA space.

In the general mathematical model given by ACT, the pre-event transient vector (f) is given by:

$$f = [a_e \quad a_p \quad a_a \quad b_e \quad b_p \quad b_a \quad o_e \quad o_p \quad o_a] \tag{2.1}$$

The vector contains the EPA profiles corresponding to the three concepts - actor(a), behavior (b), and object (b). The change equation $G(f)$ defines a polynomial, multiplicative combination of the pre-event transients such that:

$$\begin{aligned} G(f) = [1 \quad & a_e \quad a_p \quad a_a \quad b_e \quad b_p \quad b_a \quad o_e \quad o_p \quad o_a \quad a_e b_e \quad a_e o_p \quad a_p b_p \quad a_a b_a \\ & b_e o_e \quad b_e o_p \quad b_p o_e \quad b_p o_p \quad a_e b_e o_e \quad a_e b_e o_p] \end{aligned} \tag{2.2}$$

Let M defines the level of impact of each element in $G(f)$ to modify the corresponding element in f. M could be defined as a matrix of $|f|$ rows and $|G(f)|$ columns, where describes the impact of the j^{th} coefficient of $G(f)$ to the i^{th} element of the pre-event transient. The post-event transient $\widehat{f}$ is defined by:

$$\widehat{f_i} = M^T_{i*}.G(f) \tag{2.3}$$

Where M_{i*} is the i^{th} row of the M matrix. The deflection is measured by the square Euclidean distance between the pre-event and post-event transient:

$$deflection = \sum_i (f_i - M^T_{i*}.G(f))^2 \tag{2.4}$$

Theoretically, Heise [95] stated that high deflections maintained over time generate psychological stress. He also observed that humans, in general, prefer the behaviors that minimize the deflection for the affective concepts related to the set of actions. Many ACT-based models used for affects detection and generation employ the deflection phenomena in statistical or machine learning models [98, 99, 5, 169, 10].

In our models, we utilize modern Natural Language Processing (NLP) and text vectorization techniques to model the deflection for emotion and mood detection in social media writings. In chapter 4, we propose a model that pays close attention to the deflection changes through textual conversations. Besides, in chapter 5, we propose a framework architecture that models the deflection by the temporal changes in the mood of at-risk individuals to detect early signs of mental disorders.

2.4 Text Vectorization

2.4.1 An Overview

In NLP, models are formed by a set of different modules stacked in a sequential pipeline. These steps, usually, include data preprocessing, feature engineering, and model training. The model is then used for predicting outputs for novel unseen data. Features in machine learning are numerical attributes. In the feature engineering step, the textual data is transformed to feature vectors. This step is also known as text vectorization. It is one of the most crucial steps for obtaining good performance on a given NLP task. It should provide a good representation of the text to the model to make it easier to be trained.

In traditional NLP, feature engineering may require to have the good domain knowledge to determine the best combination of these features. However several tasks use similar features, they are so varied and task-specific. It needs access to many external resources. Some of the common features are, Part-of-Speech tags, word counts, word shape, n-grams (bi-gram and tri-gram), and lexicon-based features [208] . The most common disadvantages in classical feature engineering are:

- The process is time-consuming and may need an expert point of view.
- Features are incomplete. They are extracted from only parts of the text and lose the contextual information.
- It is very sensitive to the quality of the input text and highly affected by the preprocessing steps.
- It loses the sense of generalization and usability. It should be defined/updated for each different task.

Some other alternatives for general text vectorization that does not require specific domain knowledge were Bag-Of-Words (BOW) and Term Frequency–Inverse Document Frequency (TF-IDF). BOW is a very simple representation of text that describes the occurrence of words in the vocabulary within a document [82]. The problem with scoring word frequency is that highly frequent words start to dominate in the document (e.g. larger score), but may not contain as much informational content to the model as rarer. TF-IDF is the product of two statistics, term frequency, and inverse document frequency [209]. Term frequency is a scoring of the frequency of the word in the current document. Inverse document frequency: is a scoring of how rare the word is across documents. The terms with the highest TF-IDF score are often distinct terms (contain useful information) in a given document.

On the other hand, unsupervised pretraining models [188] is considered a scalable approach for feature extractors. Theoretically, unsupervised learning is more general and mimics human learning capabilities with the absence of supervising. Pretrained

models utilize unlabeled data to learn the structure and meaning of language. As shown in Figure 2.1, two main types for unsupervised pretraining. The first one is to use matrix factorization techniques to factorize a word-to-word co-occurrences. The most common two examples are, Latent Semantic Analysis (LSA) [59] and Latent Dirichlet Allocation (LDA) [16].

Latent Semantic Analysis (LSA) is one of the foundational techniques for learning dense representations of words. Given M documents and N words in the vocabulary, the model can construct an $|M| \times |N|$ matrix (A) in which each row represents a document and each column represents a word. The matrix A is computed by singular value decomposition (SVD), which decomposes it into the product of three matrices:

$$A = U_t S_t V_t^T \tag{2.5}$$

The variable (t) is a hyperparameter that reflects the dimensionality of the representation of a word and S is a diagonal matrix of singular values.

Latent Dirichlet Allocation (LDA) is considered as a probabilistic LSA. It uses a probabilistic method instead of SVD to tackle the problem. In particular, it uses Dirichlet priors for the document-topic and word-topic distributions, lending to better generalization. The unlabeled data serves as the training data for the Dirichlet distribution of document-topic relationship. The most common use case for LDA is topic detection and text classification. Word representation in LDA is determined by the word distribution across all topics for the corresponding word.

The second approach for unsupervised pretraining is language modeling. These kinds of models are trained using auxiliary tasks for language modeling to learn language features and capture semantic information in the text. In the rest of this section, we will discuss the language modeling pretraining techniques, more specifically, the neural-based approach. Finally, we look at a new age of word embedding using deep pretrained models.

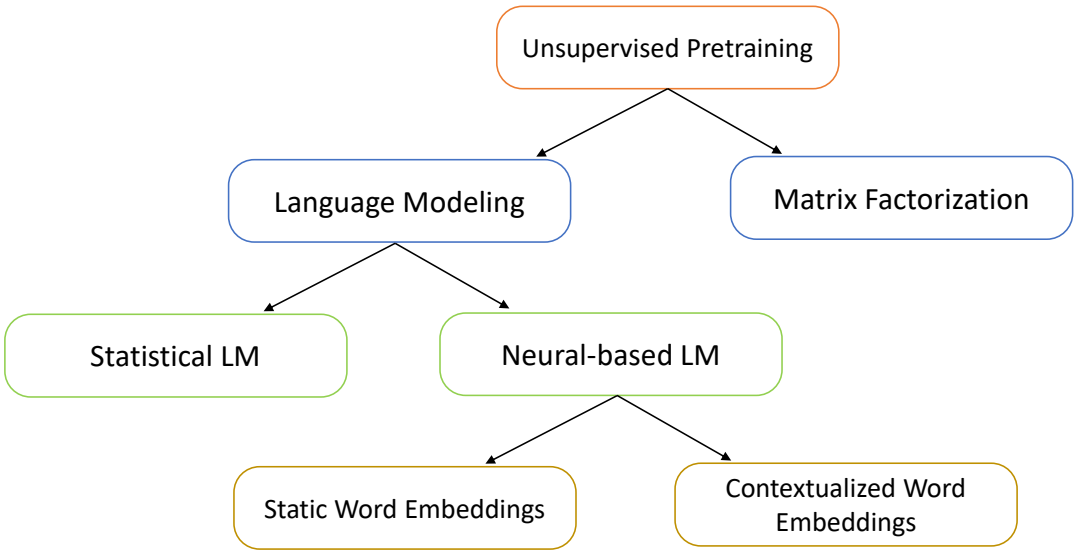


FIGURE 2.1: Types of unsupervised pretraining approaches

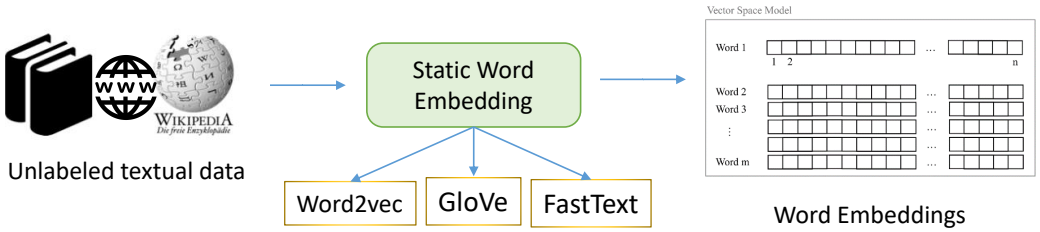


FIGURE 2.2: Static word embedding: inputs, outputs, and examples

2.4.2 Language Modeling

Language modeling is the task of predicting a word or a set of words given its contextual information. It is considered one of the most important key elements in recent NLP applications. The concept of language modeling has a long history and can be traced back to 1948 when Claude Shannon propose his theory of communication [213]. We consider two main approaches for building language models; statistical and neural-based models.

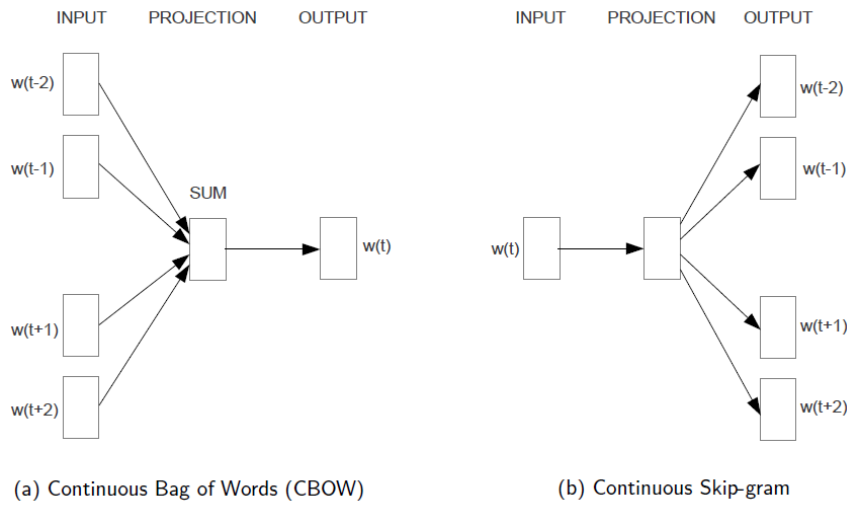
The very simplest kind of statistical language models are *N-gram* models. The basic idea is to consider the structure of a text, corpus, or language as the probability of different words occurring alone or in sequence. The simplest model, the uni-gram model, treats the words in isolation [88]. The basic idea behind higher-order N-gram models is to consider the probability of a word occurring as a function of its immediate context. For example, in a bi-gram model, this context is the immediately preceding word ($p(w_i|w_{i-1})$). The problem with N-gram language models is that it only considers the immediate context, not the whole context. Some probability models are, specifically, designed for sequence modeling which is similar to textual data. These models could be used to directly encode probability values in linguistic formalism. The most common examples are the Hidden Markov Model (HMM) language models [104] and Probabilistic Context-free Grammars (PCTG) [221]. The problems of these approaches are the limited ability to model long, structural, and lexical dependencies. These models are the closest non-neural-based models to the currently dominant approach of pretraining language models [105].

Text vectorization (embeddings) has been revolutionized recently by the development of Neural Network Language Models (NNLM), also known as neural-based LM. The idea is to train a neural network in an unsupervised manner to learn text embeddings. Neural-based LM proves to be low-dimensional, dense, and more expressive than traditional approaches [14]. Embeddings are pretrained on vast amounts of textual data instead of training them on a target (frequently a small) datasets. This allows the knowledge to be transferred across models, domains, and even tasks. Two types of word embedding exist in the literature; static and contextualized embeddings.

2.4.3 Static Word Embeddings

Neural-based static word embeddings is a neural network mapping function that maps each word to a single vector through a language modeling task. As shown in figure 2.2, static word embeddings leverage off the word vectors as the output of the pretrained models. This output can be considered as a dictionary of words-embeddings pairs. The word embeddings are then used for the downstream (target) tasks.

Word2vec: The avalanche of neural-based embeddings started in 2013 by a group of NLP researchers in Google when Miklove et. al. [158] proposed the Word2vec



(a) Continuous Bag of Words (CBOW) (b) Continuous Skip-gram

FIGURE 2.3: Word2vec models' architectures form [158]

model. The model trains a small neural network to calculate words' embeddings based on the words' context. Two models are proposed by Word2vec, as shown in figure 2.3. Both models use shallow networks with only one hidden-layer but trained on different objective functions for language modeling. In the continuous bag of words (CBOW) model, the network tries to predict which word is most likely given its context (C). The model receives as an input the window of context C and predicts a given word (w_t) by minimizing the negative log-likelihood of the following loss function:

$$\ell_{CBOW} = -\frac{1}{|C|}\sum_{t=1}^{|c|} log\, P(w_t|w_{t-C},...,w_{t-1},w_{t+1},...,w_{t+C}) \tag{2.6}$$

Where the probability $P(w_t|w_{t-C},...,w_{t-1},w_{t+1},...,w_{t+C})$ is computed by softmax; such that:

$$\begin{aligned} P(w_t|w_{t-C},...,w_{t-1},w_{t+1},...,w_{t+C}) &= \frac{exp(\tilde{X_t}^{\top} X_s)}{\sum_{i=1}^{|V|} exp(\tilde{X_i}^{\top} X_s)} \\ X_s &= \sum_{j=-C}^{C} X_{t+j} \qquad j \neq 0 \end{aligned} \tag{2.7}$$

In the vocabulary of size $|V|$, the embeddings of the word (w_i) is given by X_i, it's context embedding is denoted by $\tilde{X}_i$, and X_s is the sum of the context embeddings.

In Skip-gram models, the idea is very similar, but the network works the other way around. This time, the network uses the target word to predict its context. The model is considered an approximation of the language model that focuses on learning efficient word representations rather than accurately modeling word probabilities [131]. The objective function to be minimized during training is:

$$\begin{aligned} \ell_{SG} &= -\frac{1}{|C|}\sum_{t=1}^{|C|}\sum_{j=-C}^{C} log\, P(w_{t+j}|w_t) \quad j \neq 0 \\ P(w_{t+j}|w_t) &= \frac{exp(\tilde{X}_{t+j}^{\top} X_t)}{\sum_{i=1}^{|V|} exp(\tilde{X_i}^{\top} X_t)} \end{aligned} \tag{2.8}$$

Both CBOW and skip-gram models are trained using negative sampling [86] that trains the models to distinguish a target word w_t from negative samples drawn from a noisy distribution. Word2vec showed that it is possible to use vectors to properly represent words in a way that captures semantic or meaning-related relationships. Interesting observations are seen after training on a large corpus[157]. The similarity between words like "Paris" and "France" is closely the same as that is between "Cairo" and "Egypt". In addition, mathematical operations, like addition, give interesting results. For example, adding the embedding of the word "King" to the word "woman" gives the embedding closed to "Queen".

Many extensions to CBOW and skip-gram models have been proposed. The extensions use modifications of the same architectures in different ways, for example, incorporate n-gram features [177], jointly leaning LDA and skip-gram models (lda2vec) [166], and adding a vector that represents the paragraph or document into the learning process to get document embedding (Doc2vec) [128].

GloVe: Pennington et. al., a group of researchers at Stanford, proposed the Global Vectors for Word Representation (GloVe) model [185]. It aims to combine the count-based matrix factorization and the context-based skip-gram model together. However Word2vec learns the relation between the target words and their context, it ignores the co-occurrence of words. GloVe models link word vectors directly to the probability of these words' co-occurrence in the corpus. The intuition is that the word meanings are captured by the ratios of co-occurrence probabilities rather than the probabilities themselves [185]. GloVe minimizes the difference between the embeddings of a word w_i with its context word c_t and the logarithm of their number of co-occurrences by the following loss function:

$$\ell_{GloVe} = \sum_{i,j=0}^{|V|} f(C_{ij})(X_i^\top \tilde{X}_j + b_i + \tilde{b_j} - log(C_{ij}))^2 \tag{2.9}$$

Where C_{ij} represents the frequency of the co-occurrence of words w_i and w_j with their biases b_i and b_j respectively. $f(.)$ is a weighting function that assigns relatively lower weight to rare and frequent co-occurrences.

FastText: It is considered as an extension to the skip-gram Word2vec model. FastText embeddings are proposed by a group of researchers on Facebook that is based on the skip-gram model [18]. The main problems of almost previously mentioned word embedding are the unknown (out-of-vocabulary) words. Besides, these models ignore the morphological structure of words by assigning different embeddings to each word. These limitations are magnified for large-vocabulary and morphologically rich languages, like Arabic and Hebrew. FastText model is proposed for better generalization. The word embeddings outputted by FastText look very similar to the ones provided by Word2Vec. However, they are not calculated directly. Instead, they are a combination of lower-level embeddings of word parts. It uses the negative sampling skip-gram model with the same objective function and proposed to apply it to the subword model. Each word w_t is represented as a bag of characters n-grams $\mathbb{G}_{w_t} \subset \{g_1, ..., g_M\}$, where M is the number of n-grams appearing in w_t. The embedding of the word w_t (X_t) is the sum of the vector representations of its n-grams, such that, $X_t = \sum_{j=1}^{M} Z_j$, where Z_j denote the embeddings of the subword n-gram g_j.

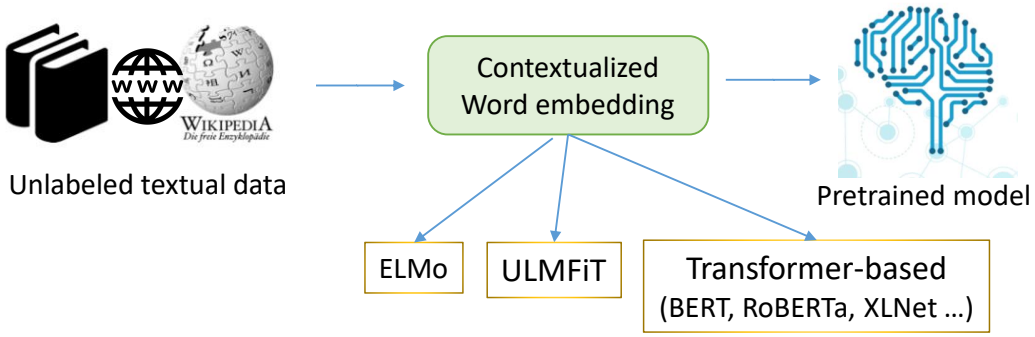


FIGURE 2.4: Contextualized word embedding: inputs, outputs and examples

2.4.4 Contextualized Word Embeddings

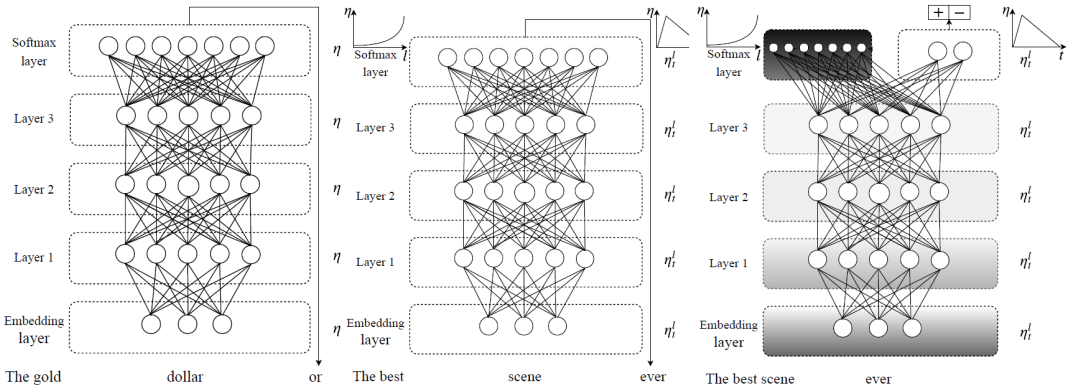
Static word embeddings models generate the same embedding for the same word in different contexts. Linguistically, static embeddings assume the same meaning for all polysemous words. Contextualized (Dynamic) words embeddings capture word semantics in different contexts to address the issue of Polysemy and the context-dependent nature of words. As shown in Figure 2.4, the output after training the model is a pretrained model, not just vectors. In the literature, there are a lot of trials to move from static to contextualized embeddings. We will focus on the most important milestones and state-of-the-art models that are closely relayed to the models proposed in the thesis.

ELMo: It stands for the Embeddings from Language Models [186]. It uses two stacked bi-directional LSTM (Bi-LSTM) trained on language modeling tasks. In the bi-directional language model (biLM), a target word (w_t) in a sequence ($w_1, ..., w_t, ..., w_M$) is predicted by two independent forward and backward language Model. The biLM jointly maximizes the log-likelihood of the following loss function:

$$\ell_{ELMo} = \sum_{t=1}^{N} [log\ P(w_t|w_1, w_2, ..., w_{t-1}) + log\ P(w_t|w_{t+1}, w_{t+2}, ..., w_N)] \tag{2.10}$$

ELMo uses character-based embedding to construct word representations. It employs a character Convolutional Neural Network (CNN) as a first layer to process the input word sequences. As a result, the biLM provides three layers of representations for each input token. The embedding for each word in a given context is computed by concatenation followed by weighted summation of all layers. The model is pretrained on the general 1B Word Benchmark [34] and could be fined tuned on domain-specific data. Regarding the downstream tasks, ELMo has significantly improved the state of the art in six NLP tasks including question answering, textual entailment, semantic role labeling, named entity extraction, co-reference resolution, and sentiment analysis [186].

ULMFiT: The great success of ELMo caught the attention of the NLP community concerning the contextualized representation of words and transferring linguistic information captured from large unlabeled text to downstream tasks. However, fine-tuning of these models leads to significant drops in the language modeling capabilities. Unlike the behavior of modern transfer learning Computer Vision (CV) models, NLP language models suffered catastrophic forgetting in downstream tasks fine-tuning. Universal Language Model Fine-tuning (ULMFiT) has been proposed



(A) LM pre-training (B) LM fine-tuning (C) Classifier fine-tuning

FIGURE 2.5: ULMFiT pretraining and fine-tuning steps from [100]

to address these issues and enable robust inductive CV-like transfer learning for any NLP task [100]. ULMFiT not only introduced a language model but also a process to effectively fine-tune this model for various tasks.

ULMFiT makes use of the Average stochastic gradient descent - Weighted Dropout LSTM (AWD-LSTM) for language modeling [154]. The AWD-LSTM has been dominating the state-of-the-art for word-level language modeling but has not been tested for downstream task transfer learning. ULMFiT use the three layers of AWD-LSTM model with the same hyperparameters proposed in the original model [154] and proposed the three following main steps, as shown in Figure 2.5:

1. A language model is trained on a large general-domain corpus to learn general linguistic features. The model used the language model on Wikitext-103 [155], consisting of more than 28K Wikipedia articles and 103 million words. (LM pre-training)

2. The language model is then fine-tuned on the target task corpus to capture task-specific information. (LM fine-tuning)

3. Fine-tuning the overall model with target classification task. (Classifier fine-tuning)

ULMFiT proposed several methods and tricks for adaptation that facilitate the transfer of information to the target task:

- *Slanted Triangular Learning Rates (STLR)*: It uses a learning rate scheduler that increases and decreases linearly but with short increase and a long decay.

- *Discriminative fine-tuning*: The model applies different learning rates for different layers groups. The intuition is that each layer captures different types of information.

- *Gradual unfreezing*: In the classification fine-tuning step, the model's layers are gradually unfreezed starting from the last layer. The main reason is that parts of the model are pretrained while the other ones are trained from scratch.

- *Bi-directional ensemble*: The AWD-LSTM is a unidirectional LSTM. ULMFiT proposed the ensemble of the forward and backward versions of the same models.

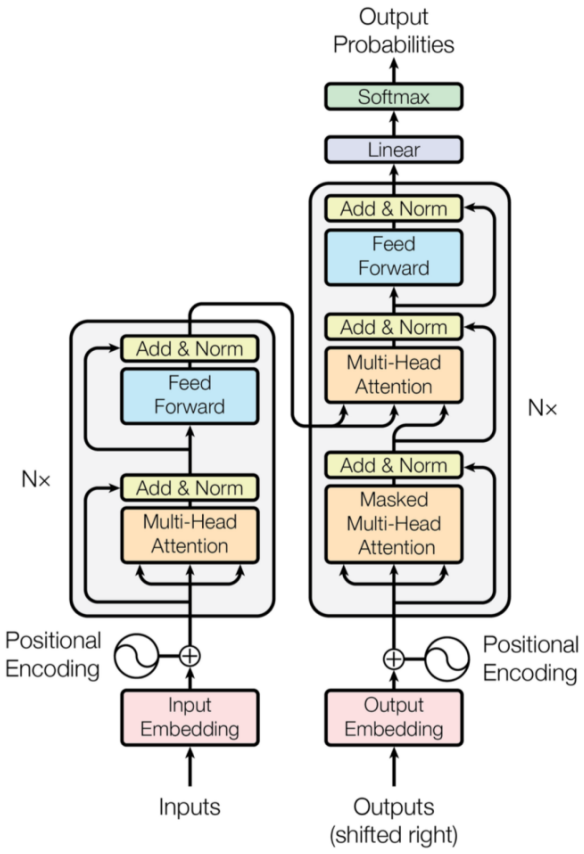


FIGURE 2.6: The Transformer architecture from [246]

Eventually, thanks to ULMFiT, the NLP community finds out a new way to do transfer learning similar to CV models. ULMFiT achieves significant improvements over the state-of-the-art on six representative text classification tasks [100].

Transformer-based Models: Sequential nature of recurrent models, e.g. ELMo and ULMFiT, do not allow the parallelization during training. These limits the ability of recurrent models to capture long-range dependencies. Vaswani et. al. proposed the Transformer model to replace the recurrent connections with multi-head self-attentions for sequence-to-sequence (seq2seq) modeling [246]. As shown in Figure 2.6, Transformer model consists of (N) bi-directional encoders and uni-directional decoder blocks. The attention mechanism - discussed in section 2.5 - allows connections between the hidden states of the output vectors for each encoder and decoder blocks. In constructing the target sequence, each target word is predicted based on a combination of vectors, rather than just the last hidden state of the decoder, this mechanism gives the decoder access to all the hidden states of the encoder. The Transformer outperforms all sate-of-the-art models in machine translation tasks.

BERT: Generative Pre-Training (GPT) model propose using Transformer decoders blocks (N=12 block) for language modeling task [196]. The problem of the GPT model is the uni-directional behavior of the Transformer decoders which limits the view of the model to the left or right context only during pretraining. Devlin et. al. proposed the Bi-directional Encoder Representations from Transformers (BERT) model [60]. The model uses the Transformer encoder and proposed two new auxiliary tasks for pretraining. The first one is the Masked Language Modeling (MLM). The objective of MLM is to predict a percentage of randomly masked tokens given

its context. The idea of using MLM in pretraining is that the bi-directional nature of Transformer decoders allows each word to indirectly see itself in a multi-layered context. The second proposed task is Next Sentence Prediction (NSP). It is proposed to capture the relationship between sentences that is not directly modeled by normal language modeling. The NSP task is modeled as a binary classification problem to predict whether a sentence A actually follows sentence B. This task helps the model handling the relationships between multiple sentences which is needed for many downstream tasks, like, question answering, natural language inference, and semantic similarity [60].

BERT represents the input word sequence using Word-Piece [256], positional, and segment embeddings. Two models are proposed -$BERT_{BASE}$ and $BERT_{LARGE}$. The models are different in the number of Transformer encoder blocks and the size of the hidden representation of the sequence items. The model is pretrained using 16 GB of textual data (3.3 billion words) from Google books corpus and Wikipedia. BERT was considered a breakthrough in the NLP community. It outperforms all the state-of-the-art models on 11 NLP tasks.

XLNet: Except for BERT, all previously discussed pretrained language models are *autoregressive* (AR) models. These models estimate the probability distributions in text corpus by language modeling tasks. BERT is considered an *autoencoder* (AE) model. Instead of explicit density estimation if AR models, BERT, and all AE models, reconstruct the original data from corrupted input. BERT uses masked input text and bidirectional contexts for reconstruction. BERT predicts only the masked words and assumes that these tokens are independent. These may harm the ability to model high-order, long-range dependencies.

Yang et. al. considered all these limitations and proposed a generalized AR pretraining model, XLNet [260]. The model introduces permutation language modeling which brings the advantages of both AR and AE methods. XLNet maximizes the expected likelihood of all permutations in random order. The model uses Transformer-XL [47] blocks instead of the Transformer encoders used in BERT. The model is pre-trained on 113 GB of text (33 billion words). It uses SentencePiece [124] tokenization to represent input sequence. XLNet has outperformed BERT on 20 tasks, and improve state-of-the-art results by a large margin in different tasks including question answering, natural language inference, sentiment analysis, and document ranking.

RoBERTa: It is a Robustly optimized BERT approach (RoBERTa) proposed by Liu et. al. [139]. The model improved BERT training methodology and use more pretraining data. RoBERTa removes the Next Sentence Prediction (NSP) task and introduces dynamic masking so that the masked token changes during the training epochs. The model uses longer sequences, larger batches, and more training steps than used in BERT for pretraining. The model uses 160 GB of text for pre-training, including the 16GB used in BERT. RoBERTa uses Byte-Pair Encoding (BPE) [212] for input sequence representation and embedding. RoBERTa outperforms BERT in all the downstream tasks. The new model matches XLNet performance in some tasks and achieves new state-of-the-art results in four out of nine tasks.

In all the models proposed in the thesis, we did not use any hand-crafted or lexicon-based features to represent input text. Instead, we apply the modern text vectorization techniques based on unsupervised pretraining. We tested various static and contextualized language modeling based word embeddings.

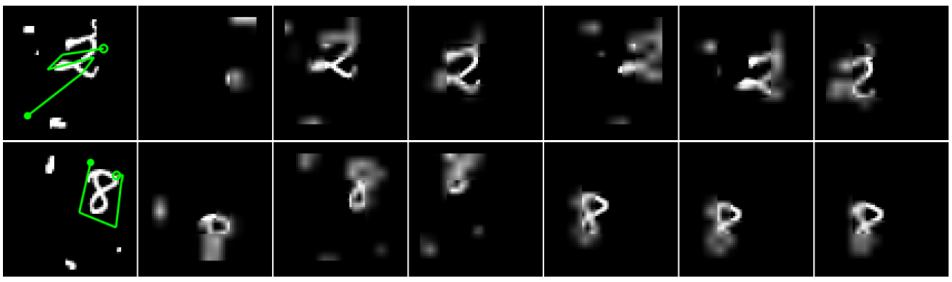


FIGURE 2.7: An example of glimpse-based model from [160]. The first column shows the input image and glimpse path in green. The other columns show the glimpses the network chooses. The center of each image shows the full resolution glimpse

2.5 Attention Mechanism

Attention mechanism is one of the recent trends in NLP. Attention Mechanisms in neural networks are inspired by the visual attention mechanism found in humans. Human is being able to focus on a certain region of an image with a high resolution while perceiving the surrounding image in a low resolution, and then adjusting the focal point over time. This is why the early applications for attention were in the field of image recognition and computer vision [238, 72, 127].

In the computer vision literature, especially in deep learning, models suffer from the computational limitations of working with large images and the cost of the widespread sliding window paradigm [160]. As a solution, a series of glimpses could be taken from a large image to formulate an approximation impression of the image before making a prediction. These glimpse-based modifications are considered as attentional guidance. As shown in Figure 2.7, visual attention models integrate information over time for each step. These models apply essential iterative steps including, read operator - to read the input image -, glimpse sensor - to extract features -, and a locator - to predict the next location of the next read operator [50].

In the NLP context, as discussed in section 2.4.4, one of the most popular architectures are the Encoder-Decoder or sequence to sequence models. One limitation of these models is that they encode the input sequence to a fixed-length internal representation. These models experienced performance degradation for long input sequences. Attention mechanism overcomes this limitation and allows the network to learn where to pay attention in the input sequence for each item in the output sequence. Neural Machine Translation (NMT) is one of the early birds that make use of attention mechanism in a typical NLP application [13].

As shown in Figure 2.8, the decoder was allowed to attend to different parts of the source sentence at each step of the output generation. The model learns where to attend to in the input sentence given what it has been produced so far. Each output word depends on a weighted combination of all the input states not only the last hidden state. Attention scores are the weights that define how much of each input state should be considered for each output. A big advantage of attention is its ability to interpret and visualize what the model is doing. For example, by visualizing the attention weight matrix when a sentence is translated, we can understand how the model is translating.

Unlike human attention which is something that's supposed to save computational resources, attention comes with a cost. Attention is looking at everything in detail

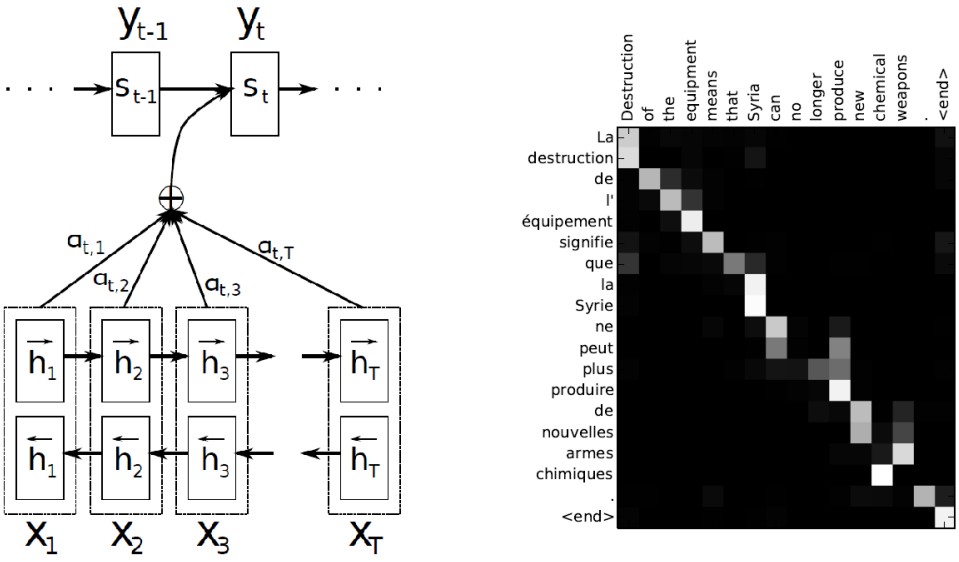


FIGURE 2.8: A graphical illustration of the attention mechanism for Seq2Seq models, next to it an example of attention visualization for NMT model proposed in [13]

before deciding what to focus on. It is needed to calculate an attention value for each combination of input.

Besides the multi-head attention used in the Transformer-based contextualized embeddings, we proposed different types of attention integration to the models presented in the thesis. In chapter 3 we proposed multi-level self-attention integration with the ULMFiT model discussed in section 2.4.4. In chapter 4 and 5 we proposed utilizing self-attention to enable the model to focus on the most relevant parts of the text that influence the model decisions.

2.6 Detection of Affects from Text

In this section, we present the research challenges, datasets, and recent advances for the three considered affective concepts in the thesis - emotion, sentiment, and individuals' mood.

2.6.1 Emotions

We consider the problem of emotion detection from textual conversations. Unlike classical emotion recognition of sentences/utterances, emotion recognition in textual conversation ideally requires context modeling of the individual utterances. The context can completely change the emotion of the same utterance in ongoing dialogues. Emotion detection in conversation draws the attention of the NLP community.

Textual conversations could have many forms including users' comments on social media, digital assistants, and conversational agents. However, few publicly available datasets exist. Table 2.2 briefly describe six datasets for emotion recognition in conversational dialogues. The datasets IEMOCAP, SEMAINE, and MELD are multimodal. In addition to the textual information, these datasets contain acoustic and visual information. Except for SEMAINE, all datasets are categorical. SEMAINE is annotated by four real-valued affective attributes (valence, arousal, expectancy, and

power).

Dataset	Multimodal	Training size *	Testing size *	No. of emotions
SEMAINE [153]	✓	63	32	-
IEMOCAP [25]	✓	120	31	6
EmotionLines [37]	✗	800	200	7
MELD [194]	✓	1153	280	7
DailyDialog [134]	✗	12118	100	7
EmoContext [32]	✗	32913	5508	4

* Training and testing size are given by number of conversations.

TABLE 2.2: Conversational datasets for emotion detection and classification

Features engineering approaches have been explored for emotion detection in text. Emotional lexicons, e.g. SentiWordNet [71], WordNet-Affect [228] and EMOLex [162], have been created to help in emotion-specific feature extraction. Recent deep learning models show a significant improvement for emotional classification and hence, which exist in textual dialogues. Poria et. al. [192] considered the multimodality emotion detection in conversations. They combined feature engineering approaches with CNN-based for textual features. The model ignores the relationships and dependencies among the utterances. The sequential nature of the conversational patterns motivates the research towards using Recurrent Neural Network (RNN). The model proposed in [193] uses LSTM-based architecture to enables utterances to capture contextual information. The textual Features are extracted using Word2vec followed by CNN model. Hazarika et. al. proposed the Conversational Memory Network (CMN) [91]. The model uses Gated Recurrent Units (GRU) and input/output memory units for modeling emotion transitions. The model uses attention mechanism to filter relevant memories over multiple input/output hops. This work has been extended in [92] which proposed the Interactive Conversational Memory Network (ICON). The model interconnects these memories hierarchically to model self and inter-speaker emotional influence. Majumder et. al. proposed the DialogueRNN model [149] that uses hierarchical multi-stage GRU units with attention mechanism. On the multimodal setup, DialogueRNN showed the ability to model multiparty conversations and outperformed all previous models by a good margin.

On the unimodal formulation of the problem, the task becomes more challenging in the absence of facial expressions and voice modulations [32]. Agrawal et. al [2] proposed the Neural and Lexical Combiner (NELEC) model that combines hand-crafted with neural-based features. Besides, the model jointly train LSTM and GRU units. Huang et. al. [103] proposed the Hierarchical LSTMs for Contextual Emotion Detection (HRLCE) model. The model uses different embeddings (GloVE, ELMo and DeepMoji [73]) through hierarchical RNN to model utterance encoding and context information. The model shows a competitive performance by ensemble it with BERT. Along the same line, ensemble methods that combines more than one text vectorization techniques show good performances [257, 135, 254].

2.6.2 Sentiment

In sentiment analysis tasks, three levels of granularity are considered in the literature - document level, sentence level, and aspect level [200]. In document-level sentiment classification, the model assigns an overall sentiment or polarity level to a given review document. The prediction could be for each sentence in the sentence-level sentiment classification. Aspect-level sentiment classification is to predict the sentiment of sentences/documents toward a given aspect. However aspect-level sentiment analysis provides additional information besides the target sentiment, it requires high quality annotated datasets with the set of predefined aspects [234].

Annotating the datasets for target sentiment is less expensive, there is a much greater quantity of automatically labeled data than the manually annotated ones. Therefor, large datasets have been grouped and preprocessed to be publicly available [171]. In the literature, many datasets have been used for document-level sentiment classification of user reviews. The datasets have different sizes, domains, and the number of classes used for labeling. The datasets are split into predefined training and testing sets to enable head-to-head comparisons of different models and approaches. Three major different domains for most of the datasets used; movies reviews [179, 223, 146], product reviews [266, 109, 190, 191], and restaurant reviews [266, 190, 191, 234]. Zhang et. al. [266] used the Stanford Network Analysis Project (SNAP) for data collection and produce four large-scale datasets for products and restaurants reviews (Yelp-bi/Full and Amazon-bi/Full). The datasets are either binary-polarity or full-polarity (5 classes (stars) of sentiment levels). More than 100K reviews for each polarity in the Yelp-bi/Full training sets. Amazon-bi/Full contain more than 3M total reviews in the training sets for different polarities. These datasets have been used to train and test most of the deep learning models in sentiment analysis, so far.

There are different approaches for Sentiment Classification of textual reviews. Lexicon-based approaches were commonly used for sentiment detection using existing sentiment lexicons. Numerous sentiment lexicon varying in format and size exist for general and domain-specific sentiment analysis; e.g. Opinion Lexicon [101], General Inquirer [227], SentiSense [53], Micro-WNOp [31], SO-CAL [232], and Subjectivity Lexicon [202]. Deep learning models outperform all the previous machine learning and lexicon-based approaches in sentiment analysis [148]. CNN and RNN models have been extensively used in the literature for sentiment classification. Kalchbrenner et. al. [112] proposed the Dynamic CNN (DCNN) model that applies the dynamic pooling functions besides the global pooling operation in CNN to handle long sequence. Zhang et. al. [266] conducted a large scale comparative study between traditional models, such as BOW, n-grams, and TF-IDF variants and proposed character-level CNN for sentiment analysis. Yin et. al. proposed the Multichannel Variable-Size CNN (MVCNN) model [262]. The model used different pretrained word embeddings, including Word2vec and GloVe, and phrases features trained using variable-size convolution filters. Wang et. al. [252] used multi-scale feature attentions for input representation and proposed the Densely Connected CNN (DCCNN) for Text Classification.

In addition to CNN, RNN are widely used for sentiment classification. Most of these models use attention mechanisms in different ways. Yang et. al. proposed the Hierarchical Network (HN) and proposed two levels of attention for words and sentences in (HN-ATT) model [261]. A similar attention-based hierarchical neural network has been proposed in the Contextual Sentiment Classification (CSC) model to incorporate user preferences and product characteristics [165]. Letarte et. al.

proposed architecture of Self-Attention Networks(SANet) [130] that models the interactions between all input word pairs. Lin et. al. proposed a self-attention based model for extracting an interpretable sentence embedding in (SA-Embedding) model [136]. Tutek et. al. proposed the iterative recursive attention model (IRAM) [245] which recursively updates the input representation by the result of attention scores previously computed. Some models combine CNN and RNN model [252, 234, 65]. For instance, the Convolutional-based with Recurrent-based Attentions Network (CRAN) score the convolution-based features by recurrent attention scores for input sequence representation.

The classification model comes with Transformer-based models have been tested for all large-scale document sentiment classification datasets [60, 139, 260]. The models showed a good performance and open the door for further improvements. BERT model has been used for Unsupervised Data Augmentation (UDA) [259] model and obtain the state-of-the-art for Yelp-bi/Full and Amazon-bi/Full datasets.

2.6.3 Mood

Detection of mental disorders, mood, and psychological state from textual data has a long history and closely related to the psycho-linguistics proposed by Jacob Kantor in 1936 [113]. In the information age, where increasing textual data streams of UGC on social media create the need for automatic mood detection. In the NLP literature, three levels of granularity are considered for mood detection: text, user, and population levels. We focus on the user-level approaches which incorporate the text-level processing methods as one of its core components.

Datasets have been collected from different sources and platforms. Twitter [2] is one of the most popular sources of mental-health-related datasets [207, 55, 56, 42, 58]. By default, almost all the activities on Twitter are public. Twitter users could simply broadcast their posts to whoever wants to listen. For instance, The first shared tasks in the Workshop on Computational Linguistics and Clinical Psychology (CLPsych-2015) [42] consisted of three user-level binary classification tasks on depression and Post Traumatic Stress Disorder (PTSD) From Twitter. The notable limitation of Twitter datasets is that each tweet is a short message which may provide limited insight, especially for mood detection tasks. Facebook [3] is considered one of the potential sources for textual mental health datasets [44, 122]. However Facebook posts could get pretty lengthy which may be better, its users employ more control to form a closed community of friends and family members. Therefore, this makes it difficult to obtain a sufficient amount of data by researchers outside Facebook. Another important data source for mood and mental disorders detection is Reddit [4] [57, 142, 143, 141, 269]. For instance, the first pilot task in eRisk-2017 workshop [141] - held as a part of the Conference and Labs of the Evaluation Forum (CLEF) - was an exploratory task on early risk detection of depression using user posts on Reddit. The following version of the workshop (eRisk-2018 and eRisk2019) used Reddit for the detection of other mental disorders, e.g. anorexia and self-harm). In addition to eRisk, the three shared for suicidal ideation detection proposed in (CLPsych-2019) workshop [269] used the University of Maryland Suicidality Dataset (UMSD) [217] which is constructed using data from Reddit. Most UGC data in Reddit is available for open access besides, no

[2] https://www.twitter.com
[3] https://www.facebook.com
[4] https://www.reddit.com

limitation for writings size exists. All previously mentioned datasets are annotated either by direct self-reporting e.g. through users' survey, indirect self-reporting (self-labeling), experts, or crowdsourcing.

Feature engineering approaches have been applied for the detection of signs of mental disorders in text. Machine learning and statistical models use combinations of various types of these hand-crafted features. The most important types of features are lexical/textual-based, behavioral, and demographic features [30, 58, 244, 67]. For instance, Cacheda et. al. [27] combines behavioral and text-based features to build dependent classifiers for positive and negative users' groups. Besides, Funez et. al. in [77] proposed a flexible temporal variation of terms (FTVT) and sequential incremental classification (SIC) models to detect signs of depression and anorexia in users writings on Reddit. Burdisso et. al. [24] generalized the previous model and proposed the SS3 classifier designed for early classification and explainability of the results using the temporal variation of language terms.

Various studies proposed different deep learning models to tackle the problem. Trotzek et. al. [242] proposed a CNN-based model with plenty of hand-crafted features represent the user-level linguistic metadata. Paul et. al. [183] proposed an RNN-based with pretrained GloVe and FastText embeddings. Mohammadi et. al. [164] combined CNN and RNN based feature encoding in a complex pipeline. The model employed a Support Vector Machine (SVM) classifier to predict on user-level. Matero et. al. [151] used attention mechanism through a dual contextual process for modeling positive and negative content separately. Different embedding methods have been tested with that model and the BERT-based model (DualContextBert) outperforms all other embeddings for suicide risk assessment.

In this thesis, we propose different models for emotion detection in textual conversations, sentiment classification and analysis, and detection of at-risk individuals diagnosed by bad mood and mental disorders. We compared the performance of these models with the state-of-the-art results reported in the literature. Our models perform competitively and obtained a new state-of-the-art for detection of depression, anorexia, self-harm, and suicide thought indicators in users' social media writings.

Chapter 3

Sentiment: Self-Attentive Sentiment Classification Modeling

"It's a most distressing affliction to have a sentimental heart and a skeptical mind."

Naguib Mahfouz

Contents

3.1 Introduction

Sentiment analysis and opinion mining are one of the fastest growing research areas in the machine learning community [150]. These areas have many applications ranging from e-commerce, marketing, reputation management, customer support, to politics and more. Currently, many tools enable us to detect subjective information, such as sentiments (positive or negative) and emotions (fear, happiness, etc.) from texts. Models for detecting feelings become more and more effective. However, they still suffer from the defects in the following areas.

Firstly, most of the existing approaches only provide sentiments and polarity identifications for specific domain or type of texts, such as [144] for financial sentiment analysis of mainstream financial websites or for sentiment analysis of customer reviews about laptops or restaurants [133]. In aspect level sentiment classification, the model should identify opinions from text about specific entities and their aspects [102]. For example, given a sentence "great food but the service was dreadful", the sentiment polarity about aspect *food* is positive while the sentiment polarity about *service* is negative. For aspect level sentiment the model must be trained using aspect based classified datasets which could be costly similarly to argumentation mining problems [64]. In this context, transfer learning or domain adaptation that has been widely used in machine learning, especially in the era of deep neural networks, could help to reuse models developed and trained in a source task to another target task. The power of transfer learning is very clear when the features learned from the source or base task are general and can be repurposed to target tasks. Computer Vision (CV) models are the most common and widely used models that make use of domain adaptation. Today, most CV models base extracting the feature to a pretrained models like AlexNet, ResNet, MS-COCO, etc. [250]. In Natural Language Processing (NLP), transfer learning and domain adaptation have been proposed before and published under different names. Since the formulation of the problem in [52] and with the emerging field of deep learning, there are many other trials for deeper transfer learning and domain adaptation models [80, 158, 128, 186]. The idea did not obtain that success until Universal Language Model Fine-tuning (ULMFiT) proposed by Jeremy et al [100] was released [1].

Secondly, to obtain a comprehensive sentiment analysis, explainable justifications are necessary not only the predicted labels. One recent trend in deep learning models is the attention mechanism [13, 263]. Attention in neural networks are inspired from the visual attention mechanism found in humans. The main principle is being able to focus on a certain region of an image with “high resolution” while perceiving the surrounding image in “low resolution”, and then adjusting the focal point over time. This is why early applications for attention were also in the field of computer vision. In NLP, most competitive neural sequence translation models have an encoder-decoder structure [246]. A limitation of these architectures is that it encodes the input sequence to a fixed length internal representation. This causes the results to get worse as the length of the sequence increases. Attention tries to overcome this limitation by guiding the network to learn where to pay close attention in the input sequence. Neural Machine Translation (NMT) is one of the early birds that make use of attention mechanism [13]. This has recently been applied to the problem of sentiment analysis

[1] The work reported in this chapter was carried out at the beginning of the thesis and at that time ULMFiT was very competitive and the state-of-the-art for many sentiment classification datasets. Later, the Transformer-based models -used in chapter 6- outperformed ULMFiT.

[145]. One of the most interesting side-effects of applying the attention is visualization. For each analyzed text, the relative relevance of each sentiment can be displayed to the user. This allows them to understand which parts of the text are more relevant to particular sentiment and interpret the score attributed to a classification. This can be considered an interpretation to which parts and segments in the input sequence that highly influence the decision of the network. To our knowledge, there has been no study showing the contribution of attention mechanisms on user interpretation.

The objective of this research is twofold: first, to show how beneficial the attention mechanism is to sentiment analysis when added to the ULMFiT architecture; Second, to evaluate how a visualization based on the attention mechanism can serve as an explanatory tool to estimate sentiment attributes since the user has a better idea of which part of the text contributes the most to the estimation of the sentiment.

The remainder of this chapter is organized as follows. In section 3.2, we give an overview of the related work and background of our proposed model which is presented in section 3.3. In section 3.4, the experimental setup, datasets used and basic results are explained. We show some discussions in section 3.5. Conclusions are presented in section 3.6.

3.2 Related Work

To address the previously mentioned limitations of the current state-of-the-art models in terms of efficiency and explainability, we propose a method based on Language Modeling and Attention learning.

Language Modeling (LM) which aims to predict the next word given a list of previous words or context is a vital and important basics in most NLP applications. Not only because it tries to understand the long-term dependencies and hierarchical structure of the text, but for its open and free resources. LM is considered as an unsupervised learning step as it needs only the corpus of an unlabeled text. The problem is that LMs get overfitted to small datasets and suffer from catastrophic forgetting when fine-tuned with a classifier. Compared to Computer Vision (CV), neural network LM models were typically more shallow and therefore required different fine-tuning methods. One of the Major steps towards deeper language model is the development of ELMo (Embeddings from Language Models) [186]. The ELMo model uses a concatenation of vectors that are generated by bidirectional LSTMs. These vectors are independently pre-trained on a large text corpus. The model proves that lower level LSTM architectures can compute syntax-based aspects of a word, while high level LSTM can capture context-dependent sentence-level features of word meaning. This way, ELMo provides a deep contextualized word and sentence representations embedding.

The development of ULMFiT is considered like ELMo as moving from shallow to deep pre-training word representation [100]. But ULMFiT introduced methods to effectively utilize a lot of what the model learns during pre-training. It introduced a language model and a process to effectively fine-tune that language model for various tasks. This idea has been proved to achieve CV-like transfer learning for many NLP tasks. ULMFiT makes use of state-of-the-art AWD-LSTM (Average stochastic gradient descent - Weighted Dropout LSTM) LM proposed by Merity et. al. in [154]. The same 3-layer LSTM architecture with the same hyperparameters and no additions other than tuned dropout hyperparameters are used. The classifier layers above the base LM encoder is simply a pooling layer (maximum and average pool) followed by fully-connected linear layer block. The overall model significantly outperformed

existing state-of-the-art on six text classification tasks including three tasks for sentiment analysis. Google AI lab, introduced a new word embedding named Bidirectional Encoder Representations from Transformers (BERT) [60]. Unlike ELMo, this architecture trains the vectors on the left and right contexts in all layers using bidirectional Transformer architecture [246]. One additional output layer can then tune the output. This architecture can be used for pre-training tasks as well as fine-tuning procedures. BERT is a very huge model. It has a large number of encoder layers (Transformer Blocks) – twelve for the *Base* version, and twenty four for the *Large* version. The large model achieved advanced state-of-the-art results for some NLP tasks, namely: question answering, named-entity recognition, and next sentence prediction. In our proposed model, we decided to use ULMFiT for its powerful fine-tuning capabilities and reasonable model size. In addition, ULMFiT was reported to be the state-of-the-art for most large-scale sentiment classification datasets.

One recent trends in NLP models is an attention function. This can be described as mapping a query and a set of key-value pairs to an output, where the query, keys, values, and output are all vectors. The output is computed as a weighted sum of values, where the weight assigned to each value is computed by a compatibility function of the query with the corresponding key. Self-attention, also known as intra-attention, is an attention mechanism relating different positions of a single sequence in order to compute a representation of the same sequence [136]. This can be seen as taking a collection of vectors—whether it is a sequence of vectors representing a sequence of words, or an unordered collections of vectors representing a collection of attributes and summarizing them into a single vector. This summarization is done by scoring each input sequence with a probability-like scores which can be outputed from the attention. Self-attention has been successfully applied to many tasks, including reading comprehension, abstractive summarization, textual entailment, learning task-independent sentence representations, machine translation and language understanding [229]. The result of the attention layer in text classification applications has already been used for visualization [252, 136] but to our knowledge, there has been no qualitative study of the impact of this visualization on the interpretation that users can make of this supplementary information.

3.3 Proposed Architecture

Our architecture is composed of 4 components described in this section.

3.3.1 Attention-based AWD-LSTM Encoder

Traditional LSTM has three gates: an input gate i_t, a forget gate f_t, an output gate o_t and a memory cell c_t. They are all vectors in $\mathbb{R}^d$ which correspond to the d dimensional vector representation. The LSTM transition equations are:

$$\begin{aligned} f_t &= \sigma(W_f x_t + U_f h_{t-1}) \\ i_t &= \sigma(W_i x_t + U_i h_{t-1}) \\ o_t &= \sigma(W_o x_t + U_o h_{t-1}) \\ \tilde{c}_t &= tanh(W_c x_t + U_c h_{t-1}) \\ c_t &= f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \\ h_t &= o_t \odot tanh(c_t) \end{aligned} \tag{3.1}$$

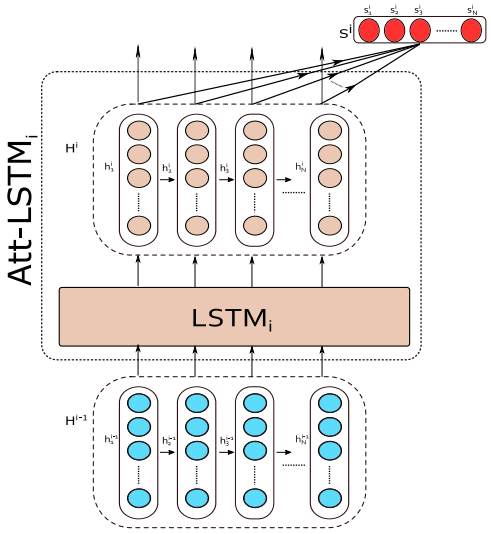


FIGURE 3.1: Attention based LSTM

where x_t is the input at the current time step, σ is the sigmoid function and $\odot$ is the element-wise multiplication operation, $W_{\{i,f,o,c\}}$, $U_{\{i,f,o,c\}}$ are all sets of learned weight parameters.

In our model, we use the hidden state vector of each time step as the representation of the corresponding word in a sentence. As a way to prevent overfitting in LSTM training, AWD-LSTM proposed [154] a solution by applying a dropout on the hidden-to-hidden connections. It applies DropConnect [251] to weight matrices $U_{\{i,f,o,c\}}$. We used the same three tied layers of LSTM in addition to applying self-attention to the hidden state vectors of each time step. Figure 3.1 shows the way we apply Att-LSTM on previous input states H^{i-1} to get new hidden states H^i and a corresponding attention scores S^i. The input hidden state sequence $H^{i-1} = \{h_1^{i-1}, h_2^{i-1}, \ldots, h_N^{i-1}\}$, where N is the input sequence length, is passed to the LSTM layer states. The output states has the form of $H_i = \{h_1^i, h_2^i, \ldots, h_N^i\}$. The attention layer takes the encoded input sequence and computes the attention scores $S^i = \{s_1^i, s_2^i \ldots s_N^i\}$. The attention layer can be viewed as a linear layer without bias.

$$\begin{aligned} \alpha^i &= \{V^i.H^i\} \\ S^i &= exp(\alpha^i) / \sum_{j=1}^{N} exp(\alpha_j^i) \end{aligned} \tag{3.2}$$

Where V^i and α_j^i are the weights and logits of the self-attention layer of the i^{th} Att-LSTM respectively.

3.3.2 Multi-level Self-Attention Aggregation

The proposed architecture uses a stacking of three Att-LSTM on top of each other exactly the same way of regular AWD-LSTM model. Figure 3.2 shows an overview of the model. At each layer, attention scores are obtained according to a specific level of sequence encoding and then aggregated to obtain global attention scores $\overline{S}$. The aggregation function is the log average of the three levels of self-attention scores.

$$\overline{S} = log \sum_{i=1}^{3} S^i / 3 \tag{3.3}$$

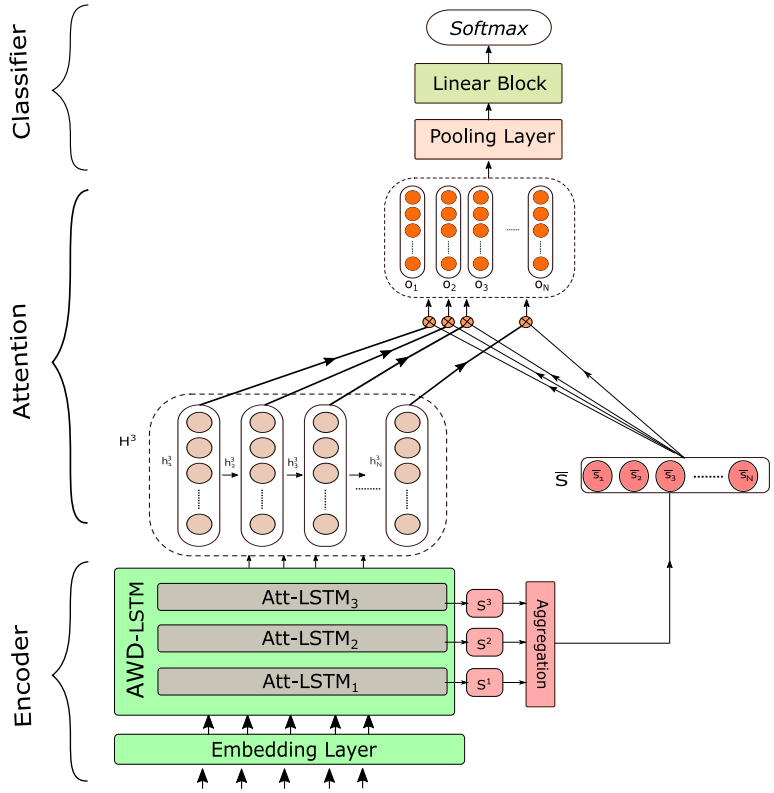


FIGURE 3.2: Proposed Model Architecture

Experiments shows that log operator behaves better than normal average especially for long sequences. The global attention scores $\overline{S}$ are used to compute the scored sequence $O = \{o_1, o_2, \ldots, o_N\}$ where o_i is the inner product of the corresponding attention score and the output of the last Att-LSTM layer such that:

$$o_i = \overline{s}_i \otimes h_i^3 \tag{3.4}$$

3.3.3 Classification Layers

After aggregating the information from multi-level attention and scoring with the encoder output, we convert the resulting representations of all positions in O to a fixed-length vectors with pooling. We used three pooling functions. We apply an attention pooling X_{att} such that:

$$X_{att} = \sum_{i=1}^{N} exp(\overline{s}_i) \otimes h_i^3 \tag{3.5}$$

Also, we apply maximum X_{max} and average pooling X_{avg} to O in order to get the final representations of the input text after encoding and attention. which is given by.

$$X_{in} = [X_{att} \oplus X_{max} \oplus X_{avg}] \tag{3.6}$$

then feed it into a classifier linear block. This block is consisted of two different sizes fully connected dense layers followed by a Softmax to determine the output sentiment class.

3.3.4 Model Training

Training the overall model comes into three main steps:

1. The LM is randomly initialized and then trained by stacking a linear decoder on top of the encoder. The LM is trained on a general-domain corpus. This helps the model to understand the general features of the language.

2. After training, the same full LM is used as an initialization to be fine-tuned using the data of the target task (sentiment datasets). In this step, we limit the vocabulary of the LM to the frequently used words (repeated more than twice) from the target task.

3. We keep the encoder and replace the decoder with the classifier and both are fine-tuned on the target task.

In the first step and for training the language model, we used the Wikitext-103 dataset [155]. With more than 28K of Wikipedia articles and 103 million words, the model determines the main structure and hierarchy of the language by sequence-to-sequence modeling. We train the language model encoder only once and it is fine-tuned for each target model. For training the overall classification model, the model is trained in an end-to-end way in a supervised learning framework, the aim of this training is to optimize all the parameters so as to minimize the objective function (loss function) as much as possible. In our work, let y_i be the correct sentiment polarity, which is represented by one-hot vector, and $\hat{y}_i$ denotes the predicted sentiment polarity for the given sentence. We regard the cross-entropy as the loss function, and the formula is as follows:

$$loss = -\sum_{<T>} y_i \log(\hat{y}_i) + \lambda \|\theta\|^2 \tag{3.7}$$

Where λ is the regularization factor, θ contains all model parameters and T is all the training examples.

The training of the architecture is done using slanted triangular learning rates (STLR) which change the learning rate for each iteration in triangular fashion. We used only once cycle, as recommended in [100]. The model was trained by discriminative fine-tuning which uses different learning rates for each layer group. The model is trained gradually by freezing and unfreezing layers for different groups.

We trained the model on the forward and backward LMs for both the general-domain and task specific datasets. Both LMs -backward and forward- are used to build two versions of the same proposed architecture. The final decision is the ensemble of both.

We used Pytorch[2] to build the whole model and make use of Fastai[3] libraries for applying the training strategies and fine-tuning the language models. For text preprocessing, the text is first normalized and tokenized. Special tokens were added for capitalized and repeated words. We kept the punctuation and the sentiments symbols in text. We used Spacy[4] and the wrapper of FastText[5]. The models are trained and tested on four Nvidia GEFORCE GTX 1080 GPUs. We released the source code of the model and all the experiments on github[6].

[2]https://pytorch.org/
[3]http://www.fast.ai/
[4]https://spacy.io/
[5]https://fasttext.cc/
[6]https://github.com/WaleedRagheb/MLSA

3.4 Experimental Setup

3.4.1 Datasets

We applied the model on different sentiment classification datasets. Table 3.1 shows a brief statistics of these datasets. The IMDB is a dataset for binary sentiment classification containing highly polar movie reviews for training and testing [146]. There is additional unlabeled data that we used for language model fine-tuning as well. We also used the binary and full classes versions of Yelp and Amazon user reviews datasets [266]. All of these datasets are balanced in terms of the number of training and test examples for each class.

Dataset	# Training Examples	# Testing Examples	#classes
IMDB	25K	25K	2
Yelp-bi	560K	38K	2
Yelp-Full	650K	50K	5
Amazon-bi	3.6M	400K	2
Amazon-Full	3M	650K	5

TABLE 3.1: Used sentiment datasets and number of training and testing examples

3.4.2 Baselines and Results

We compared our model with several existing state-of-the-art competitive baselines that make use of the attention mechanism for document and sentiment classifications:

- **HN-ATT** [261] the model mirrors the hierarchical structure of documents through two levels of attentions in words and sentences.
- **DCCNN-ATT** [252] this model is a Convolutional neural network with dense connections and multi-scale feature attentions.
- **SANet** [130] this model uses self-attention to model the interactions between all input word pairs.
- **SA-Embedding** [136] this model is based on extracting an interpretable sentence embedding by self-attention.
- **CSC** [165] this model uses attention-based hierarchical neural networks that incorporate user preferences and product characteristics into sentiment classification tasks.
- **CRAN** [65] the model combines both Convolutional-based with Recurrent-based attentions.
- **IRAM** [245] it is an attention model, which incrementally constructs representations of input data through reusing results that is previously computed in recursive fashion.

Also, we compared the proposed model with the default ULMFiT model [100] and the base model of BERT trained for sentiment classification task [36]. Table 3.2 shows the testing error of the proposed model and all the baselines on the testsets. For all the baselines we used the results reported in the original paper. The proposed model outperforms almost the attention based models with a significant margin. Compared

Models	IMDB	Yelp-bi	Yelp-Full	Amazon-bi	Amazon-Full
HN-ATT	-	-	-	-	36.40
DCCNN-ATT	-	2.64	30.58	**3.32**	34.81
SANet	-	4.77	36.03	4.52	38.67
SA-Embedding	-	5.10	36.60	-	40.20
CSC	-	6.90	35.97	4.90	39.89
CRAN	7.90	-	-	-	-
IRAM	8.80	-	-	-	-
BERT	7.00	3.00	-	-	-
ULMFiT	4.60	**2.16**	29.98	-	-
Ours	**4.51**	2.25	**29.76**	3.43	**34.78**

TABLE 3.2: Test error rates (%) of our proposed model and all the baselines

to the default ULMFiT, the proposed model is competitive and has improved state-of-the-art results for the IMDB, full version of Yelp and amazon datasets.

3.5 Discussions

In this section we will conduct ablation study of the model and discuss in more details the impact of applying attention to increase user interpretability.

3.5.1 Model Ablation Analysis

To further investigate the efficacy of the key components of our proposed model, we perform ablation study as shown in Table 3.3. We test the performance of six different variants of the model on all test datasets. To assist the impact of attention in the model, we tried one variant without any attention and another with Uni-Level (UL) attention rather than the Multi-Level (ML). In UL attention, we skip the attention aggregation step and use the attention scores of the last Att-LSTM. To test the overall model in terms of its transfer learning cabapilities, we proposed to train the model from scratch without pre-training the language model encoder. We also propose pre-training the same encoder and make it shared for all models of different datasets. In shared encoder model, only attention and classification layers are learned from scratch. In addition, we tried also to train two classifiers and make it shared for all models. The two shared classifiers are one for binary and other for five classes datasets.

The results show that adding attention to the model is helpful and using multi-level outperforms the uni-level attention architecture. Pre-training the encoder has a significant impact on the results. It is very obvious for smaller size datasets. Using either a pre-trained shared encoder or shared classifier for all the models gives good comparable results compared to other baselines. This proves the language modelling and generalization capabilities of the proposed model's building blocks to model long-term and short term dependencies in different domains.

3.5.2 Attention Visualizations

One of the most attractive outcomes of applying the attention mechanism is its ability to process all the input sequences with different weights of attentions. It usually pays closer attention to the most important parts that influence the network decision.

Model Variants	IMDB	Yelp-bi	Yelp-Full	Amazon-bi	Amazon-Full
-No Attention	4.60	2.16	29.98	3.84	35.00
- UL Attention	4.58	2.21	29.81	3.54	34.80
-No pre-training + No Attention	9.98	4.01	33.97	4.40	36.49
-No pre-training + ML Attention	8.87	3.85	31.77	4.25	36.43
-Shared Encoder + ML Attention	4.76	**2.11**	29.84	3.49	34.84
-Shared Classifier + ML Attention	5.21	2.66	31.18	3.53	34.86
-Pre-training + ML Attention	**4.51**	2.25	**29.76**	**3.43**	**34.78**

TABLE 3.3: Test error rates (%) of proposed variants of the model

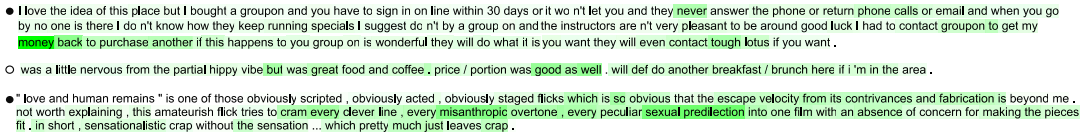


FIGURE 3.3: Attention visualization examples of positive (white bullet) and negative (black bullet) restaurants and films reviews

Figure 3.3 shows some examples of correctly classified positive and negative reviews for a restaurant and a movies review. The attention scores are processed (scaled and normalized) before reflecting it on the text for a clearer visualization.

From a first glance we could easily recognize many sentiments and emotional phrases. The model focuses on words for which it is easier to deduce the sentiment. To validate these findings, we compared the most important tokens in terms of attention scores with sentiment and emotional lexicons. We found EmoLex proposed by Mohammad et. al. in [162] a good example. EmoLex is created with a high-quality, moderate-sized, emotion and polarity lexicon. Using EmoLex gives a good results in large variety of task related to emotion, sentiment and stance classification [161]. It has entries for more than 10,000 word-sense pairs. Table 3.4 shows the results of matching the top attention scored tokens with EmoLex in both testsets. For easy interpretation of the table and for example, we can say that 88.11% from the total IMDB testset examples contain emotions and polarity words in the top 5% from highly attention tokens in the text. This reflects the precision of attention mechanism in focusing on these words and phrases.

Dataset	Top 5%	Top 10%	Top 20%
IMDB	88.11%	97.30%	99.65%
Yelp-bi	51.86%	78.81%	94.37%
Yelp-Full	64.11%	84.65%	95.47%
Amazon-bi	55.71%	80.18%	94.65%
Amazon-Full	57.67%	81.02%	94.84%

TABLE 3.4: Sentiments and emotions in the top attention scored parts of the text in testsets

3.5.3 Empirical Study

However attention models may focus on some emotional words, only paying close attention to these words may be misleading. The used Self-attention mechanism looks to some parts of the text which have a higher impact on the final decision of

FIGURE 3.4: Example of an attention-based Word cloud image

the network. Based on some examples, as in figure 3.3, we could easily find parts of the highlighted text that are not emotionally related. In order to determine to what extent only paying attention to important words could induce a polarity decision, we propose to run a word cloud survey. This allows us to know which set of words could be more helpful. In this section, we present the word cloud survey task that makes a comparison between lexicon based and attention based ways of highlighting.

Word Cloud survey

This is a closed-ended question survey that asks participants to guess the polarity of a review given a word cloud image representing some of its words. Although this does not take into account the word order but it will prove if the proposed model focuses on words for which it is easier to deduce the sentiment. The word cloud images are generated in three different ways, either lexicon based, attention based or mixing them together. For attention based questions, the words sizes and positions in the images are determined by the attention weights associated to it as shown in the example in figure 3.4. In the case of lexicon based word images, these were determined by lexicon word frequencies and the presence order in the document review. For mixed based images, we combined both weights together.

Each participant answered 30 questions, 10 questions for each type. The question is to guess the polarity of either positive or negative reviews associated to the word cloud. We let the participant know if the image is originating from which type of review: film, restaurant or an other product. The question selections are randomized for each question type, they are also shuffled for each participant.

Results

The total number of participant is 85. For each question types, we compute the average facility index F_i as:

$$F_i = 100.\frac{\bar{x_i} - x^i_{min}}{x^i_{max} - x^i_{min}} \tag{3.8}$$

where $\bar{x_i}$ is the average score for question type i, x^i_{min} and x^i_{max} is the minimum and maximum scores for type i respectively. This gives the average scores of a question type as a percentage that indicates how easy the question is. We compute the standard deviation of all the scores for each question type. In addition, we compute the discrimination index D_i as:

$$D_i = 100.\frac{\mathbb{E}[(X_i - \mu_{X_i})(T - \mu_T)]}{\sigma_{X_i}\sigma_T} \tag{3.9}$$

Which is the correlation percentage between the scores of each question type and the total score for each participant in the overall survey T. This gives more information about the participant agreement. Table 3.5 shows the word cloud survey results for the three question types.

Question Types	Facility Index	Standard Deviation	Discrimination Index
Attention	**72.12%**	**43.90%**	**7.70%**
Lexicon	58.90%	49.86%	-16.10%
Mixed	68.27%	49.19%	6.70%

Table 3.5: Word Cloud Survey Results

The results show that the task is some how difficult and guessing the sentiment without the complete text is not an easy task. However, the attention word cloud questions are more easy than others with average facility index of 72.12% and standard deviation of 43.90%. Also the positive correlation and agreement of the discrimination index. The task becomes more difficult when only using lexicon, using the same measures.

3.6 Conclusions

In this chapter, we consider the problem of document-level sentiment classification problems and proposed a new deep learning model for classifications of user reviews on the internet. The proposed model uses ULMFiT and introduces multi-levels of self-attention layers to enhance the transfer learning capability of the original model. The proposed attention mechanism provides interpretations of network decisions. Form experiments, We can conclude that the idea of transfer learning is very effective in NLP applications in particular for sentiment analysis. It performs better than the classical shallow learning models of word embedding. Also, adding a self-attention mechanism directly impacts the performance of these models and increases user interpretability of the results. The proposed model was evaluated for sentiment classification problems on five common datasets. Our experiments show competitive results for our model compared to current state-of-the-art attentive based models and the default ULMFiT.

The visualization of the attention scores is useful with a higher impact on user perception compared to lexicon-based methods. However, we encourage further experiments to assess the behavior of modern attention-based models on users' perceptions on similar tasks. Moreover, the proposed model is general and could be applied to different classification tasks such as topic labeling and spam detection. Besides, the model could be modified to work for other natural language understanding tasks. For example, modifications could be needed for other forms of multi-field textual datasets such as those used in textual conversations and sequential datasets.

Chapter 4

Emotions: Emotions Detection in Textual Conversations

> "I don't want to be at the mercy of my emotions. I want to use them, to enjoy them, and to dominate them."
>
> Oscar Wilde

Contents

4.1 Introduction

Emotions play a crucial role in human life, as they are important in interpersonal relationships and contribute to decision making and reasoning. It is a short-lived feeling in response to our interpretation of an immediate trigger [203]. Despite the culture and language differences, emotions could be classified into basic predefined emotional states. Various researches proposed many theories of emotions with different views [168]. Most notably, Ekman in [70] argued for the existence of six basic emotions; anger, disgust, fear, joy, sadness, and surprise. Additionally, Plutchik studied the problem and added two more basic emotions -trust and anticipation. He also introduced the opposing pairs of emotions (e.g. joy versus sadness, anger versus fear, trust versus disgust, and surprise versus anticipation) and proposed the wheel of emotions shown in Figure 4.1. As humans can naturally capture and express different emotions, machines should be able to infer them as well. The process is widely known as emotional intelligence.

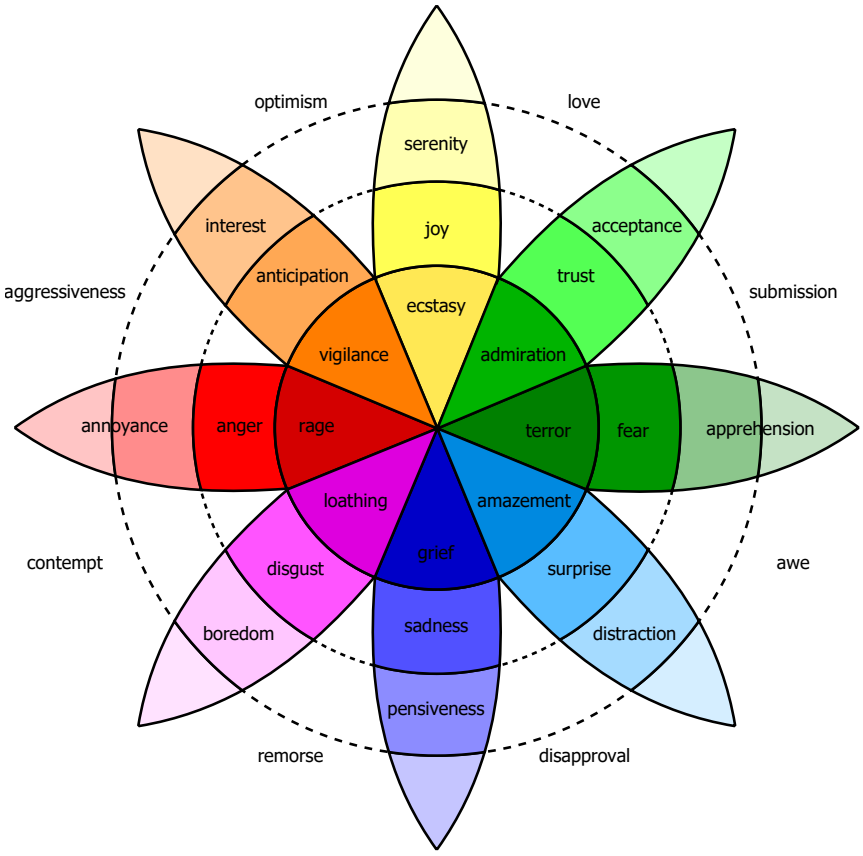


FIGURE 4.1: Plutchick's Wheel of emotion from [189]

Emotional intelligence has played a significant role in many application in recent years [121]. It is one of the essential abilities to move from narrow to general human-like intelligence. Being able to recognize expressions of human emotion such as interest, distress, and pleasure in communication is vital for helping machines choose more helpful and less aggravating behavior. Human emotions are a mental state that can be sensed and hence recognized in many sources such as visual features in images or videos [20], as textual semantics and sentiments in texts [28] or even patterns in EEG brain signals [108]. With the increasing number of messaging platforms and with the growing demand of customer chat bot applications, detecting the emotional

state in conversations becomes highly important for more personalized and human-like conversations [267].

This chapter addresses the problem of modeling a conversation that comes with multiple turns for detecting and classifying emotions. We refer to our participation in the Semeval-2019 Task-3 - EmoContext: Contextual Emotion Detection in Text [32]. The proposed model makes use of transfer learning through the universal language modeling that is composed of consecutive layers of Bi-directional Long Term Short Term Memory (Bi-LSTM) units. These layers are learned first in language modeling task on a general text and then fine-tuned to a specific target task. The model also makes use of an attention mechanism in order to focus on the most important parts of each text turn. Finally, the proposed classifier models the changing of the emotional state of a specific user across turns. The proposed model gives competitive performance and ranked 9^{th} out of more than 150 participants.

The chapter is organized as follows. In Section 4.2, the related work is introduced. Then, we present a quick overview of the task and of the datasets in Section 4.3. Section 4.4 describes the proposed model architecture, some variants and hyperparameters settings. The experiments and results are presented in Section 4.5. Section 4.6 concludes the study.

4.2 Related Work

Transfer learning or domain adaptation has been widely used in machine learning especially in the era of deep neural networks [84]. In natural language processing (NLP), this is done through Language Modeling (LM). Through this step, the model aims to predict a word given some context. This is considered as a vital and important basics in most of NLP applications. Not only because it tries to understand the long-term dependencies and hierarchical structure of the text but also for its open and free resources. LM is considered as unsupervised learning process which needs only corpus of unlabeled text. The problem is that LMs get overfitted to small datasets and suffer catastrophic forgetting when fine-tuned with a classifier. Compared to Computer Vision (CV), NLP models are typically more shallow and thus require different fine-tuning methods. The developing of the Universal Language Model Fine-tuning (ULMFiT) [100] is considered like moving from shallow to deep pre-training word representation. This idea has been proved to achieve CV-like transfer learning for many NLP tasks. ULMFiT makes use of the state-of-the-art AWD-LSTM (Average stochastic gradient descent - Weighted Dropout) language model [154]. Weight-dropped LSTM is a strategy that uses a DropConnect [251] mask on the hidden-to-hidden weight matrices, as a means to prevent overfitting across the recurrent connections.

On the other hand, as a recent trend in deep learning [263], attention mechanism is inspired from the visual attention mechanism that exists in humans. The main principle is being able to focus on a certain region of an image with "high resolution" while perceiving the surrounding image in "low resolution", and then adjusting the focal point over time. This is why the early applications for attention were in the field of image recognition and computer vision [127]. In NLP, most competitive neural sequence transduction models have an encoder-decoder structure [246]. A limitation of these architectures is that it encodes the input sequence to a fixed length internal representation. This cause the results going worse performance for very long input sequences. Simply, attention tries to overcome this limitation by guiding the network to

learn where to pay close attention in the input sequence. Neural Machine Translation (NMT) is one of the early birds that make use of attention mechanism [13]. It has recently been applied to other problems like sentiment analysis [145] and emotional classification [149].

4.3 Datasets

In our experiments, we used the datasets provided by the task organizers of Semeval-2019 Task-3 [32]. These datasets contain collections of labeled conversations. Each conversation is a three turn talk between two parties. The conversation labels correspond to the emotional state of the last turn. Conversations are manually classified into three emotional states for *happy*, *sad*, *angry* and one additional class for *others*. The datasets restricts the number of emotions to these classes as they are the most popular emotions in conversational data. In general, released datasets are highly imbalanced and contains about 4% for each emotion in the validation (development) set and final test set. Table 4.1 shows the number of conversations examples and emotions provided in the official released datasets.

Dataset	Data size	Happy	Sad	Angry
Training	30160	5191	6357	6027
Validation (Dev)	2755	180	151	182
Testing	5509	369	308	324

TABLE 4.1: Semeval-2019 Task-3 EmoContext datasets

4.4 Proposed Models

In this section, we present the proposed model architecture for modeling a conversation through language models encoding and classification stages. Also, we explain the training procedures used and the external resources for training the language model. In addition to the basic architecture, We will describe the used variants of the model for evaluation. Finally, we will list the hyperparameters used for building and training these models.

4.4.1 Model Architecture

In figure 4.2, we present our proposed model architecture. The model consists of two main steps: encoder and classifier. We used a linear decoder to learn the language model encoder as we will discuss later. This decoder is replaced by the classifier layers. The input conversations come in turns of three. After tokenization, we concatenate the conversation text but keep track of each turn boundaries. The overall conversation is inputted to the encoder. The encoder is a normal embedding layer followed by AWD-LSTM block. This uses three stacked different size Bi-LSTM units trained by ASGD (Average Stochastic Gradient Descent) and managed dropout between LSTM units to prevent overfitting. The conversation encoded output has the form of $C_{Enc} = [T^1_{Enc} \oplus T^2_{Enc} \oplus T^3_{Enc}]$ where T^i is the i^{th} turn in the conversation and $\oplus$ denotes a concatenation operation and $T^i_{Enc} = \{T^i_1, T^i_2, \ldots, T^i_{N_i}\}$. The sequence length of turn i is denoted by N_i. The size of T^i_j is the final encoding of the j's sequence item of turn i.

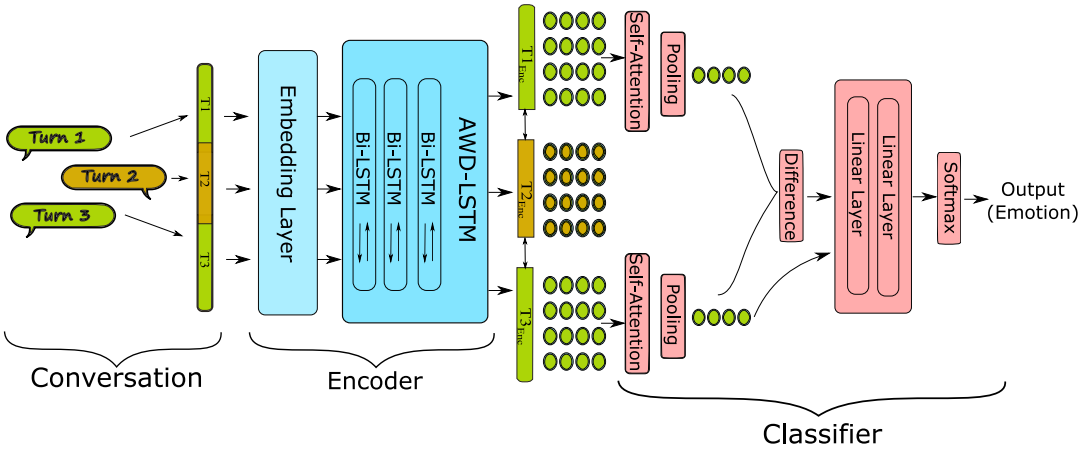


FIGURE 4.2: Proposed model architecture (*Model-A*)

For classification, the proposed model pays close attention to the first and last turns. The reasons behind this are that the problem is to classify the emotion of the first and last turns. Also, the effect of the middle turn appear implicitly on the encoding of the last turn as we used Bi-LSTM encoding on the concatenated conversation. In addition, tracking the difference between the first and the last turn of the same person may be beneficial in modeling the semantic and emotional changes. So, we apply self-attention mechanism followed by an average pooling to get turn-based representation of the conversation. The attention scores for the i^{th} turn S^i is given by:

$$S^i = \text{Softmax}\{W_i.T^i_{Enc}\} \tag{4.1}$$

Where W_i is the weight of the attention layer of the i^{th} turn and S^i has the form of $S^i = \{S^i_1, S^i_2, ..., S^i_{N_i}\}$. The output of the attention layer is the scoring of the encoded turn sequence $O^i = \{o^i_1, o^i_2, \ldots, o^i_{N_i}\}$ which has the same length as the turn sequence and is given by $O^i = S^i \odot T^i_{Enc}$ where $\odot$ is the element-wise multiplication. The difference of the pooled scored output of O^1 and O^3 is computed as O_{diff}. The input of the linear block is X_{in} is formed by:

$$X_{in} = [O_{\text{diff}} \oplus O^3_{\text{pool}}] \tag{4.2}$$

The fully connected linear block consist of two different sized dense layers followed by a Softmax to determine the target emotion of the conversation.

4.4.2 Training Procedures

Training the overall models comes into three main steps:

1. The LM is randomly initialized and then trained by stacking a linear decoder in top of the encoder. The LM is trained on a general-domain corpus. This helps the model to get the general features of the language.

2. The same full LM after training is used as an initialization to be fine-tuned using the data of the target task (conversation text). In this step we limit the vocabulary of the LM to the frequent words (repeated more tan twice) of target task.

3. We keep the encoder and replace the decoder with the classifier and both are fine-tuned on the target task.

For training the language model, we used the Wikitext-103 dataset [155]. We train the model on the forward and backward LMs for both the general-domain and task specific datasets. Both LMs -backward and forward- are used to build two versions of the same proposed architecture. The final decision is the ensemble of both. Our code is released on github[1]. However we tried the uni-directional models, experimental studies shows that the ensemble models give a better performance. Training the self-attention layer uses the same learning rates used in the classification layers group.

We used Pytorch [2] to build the whole model and make use of Fastai [3] libraries for applying the training strategies and fine-tuning the language models. For text preprocessing, the text is first normalized and tokenized. Special tokens were added for capitalized and repeated words. we keep the punctuation and the emotions symbols in text. We used Spacy [4] and the wrapper of FastText [5]. The models are trained and tested on four Nvidia GEFORCE GTX 1080 GPU.

4.4.3 Model Variations

In addition to the model - *(Model-A)* - described by Figure 4.2, we tried five different variants. Each variant modify the classifier layer groups. Studying the effect of these variants will provide a good model ablation analysis.

The first variant -(*Model-B*)- is formed by bypassing the self attention layer. This will pass the output of the encoder directly to the average pooling layer such that $X_{\text{in}}^{B} = [T_{\text{diff}} \oplus T_{\text{pool}}^{3}]$ where T_{diff} is the difference between the first and third pooled encoded turns of the conversations.

-(*Model-C*)- is to input a pooled condensed representation to the whole conversation C_{pool} rather than the last turn to the linear layer block. In this case: $X_{\text{in}}^{C} = [O_{\text{diff}} \oplus C_{\text{pool}}]$. We also studied two versions of the basic model where only one input is used $X_{\text{in}}^{D} = O_{\text{diff}}$ -(*Model-D*)- and $X_{\text{in}}^{E} = O_{\text{pool}}^{3}$ -(*Model-E*). In these two variants, we just change the size of the first linear layer.

Also, we apply the forward direction LM and classifier only without ensemble them with the backward direction and keep the same basic architecture -(*Model-F*).

4.4.4 Hyperparameters

We use the same set of hyperparameters across all model variants. For training and fine-tuning the LM, we use the same set of hyperparameter of AWD-LSTM proposed by [154] replacing the LSTM with Bi-LSTM and keep the same embedding size of 400 and 1150 hidden activations. We used weighted dropout of 0.2 and 0.25 as the input embedding dropout and the learning rate is 0.004. We fine-tuned the LM by all provided datasets in table 4.1. We train the LM for 14 epochs using batch size of 128 and limit the number of vocabulary to all token that appear more than twice. For classifier, we used masked self-attention layers and average pooling. For the linear block, we used hidden linear layer of size 100 and apply dropout of 0.4. We used Adam optimizer [63] with $\beta_1 = 0.8$ and $\beta_2 = 0.99$. The base learning rate is 0.01. We used the same batch size used in training LMs but we create each batch using weight random sampling. We used the same weights provided by the organizers (0.4 for each emotion). We train the classifier on training set for 30 epochs and select the best model on validation set to get the final model.

[1] https://github.com/WaleedRagheb/AttentiveEmocontext
[2] https://pytorch.org/
[3] http://www.fast.ai/
[4] https://spacy.io/
[5] https://fasttext.cc/

4.5 Results & Discussions

Models	Results									
	Happy			Sad			Angry			Micro
	P	R	F1	P	R	F1	P	R	F1	F1
Baseline	0.5123	0.5845	0.5461	0.5163	0.7600	0.6149	0.4777	0.7867	0.5945	0.5861
NELEC [2]	0.7632	0.7148	**0.7382**	0.7938	0.816	0.8047	0.747	0.8322	**0.7873**	**0.7765**
SymantoResearch [15]	0.7380	0.7042	0.7207	0.8193	0.816	0.8176	**0.7807**	0.7886	0.7846	0.7731
ANA [103]	**0.7698**	0.6831	0.7239	**0.8458**	0.812	**0.8286**	0.7198	0.8188	0.7661	0.7709
CAiRE-HKUST [254]	0.7301	**0.743**	0.7365	0.7774	**0.852**	0.813	0.6997	0.8289	0.7588	0.7677
Figure Eight [257]	0.7055	0.7254	0.7153	0.7695	0.828	0.7977	0.6954	**0.8658**	0.7713	0.7608
YUN-HPCC [132]	0.7169	0.6866	0.7014	0.8016	0.824	0.8126	0.7198	0.8188	0.7661	0.7588
Model-A	0.7256	0.7077	**0.7166**	0.8291	0.776	**0.8017**	0.7229	0.8054	**0.7619**	**0.7582**
Model-B	**0.7341**	0.6514	0.6903	0.7401	**0.82**	0.778	0.7049	**0.8255**	0.7604	0.7439
Model-C	0.7279	0.6972	0.7122	0.7765	0.792	0.7842	0.6941	0.8221	0.7527	0.7488
Model-D	0.7214	**0.7113**	0.7163	0.8128	0.764	0.7876	0.6965	0.8087	0.7484	0.749
Model-E	0.7204	0.7077	0.714	0.8205	0.768	0.7934	0.7026	0.8087	0.752	0.7512
Model-F	0.7336	0.669	0.6998	**0.8377**	0.764	0.7992	**0.738**	0.7752	0.7561	0.75

TABLE 4.2: Test set results of the proposed model, its variants, best performing systems, and the baseline

The results of the test set for different variants of the model for each emotion is shown in table 4.2. In addition, we show the results of the baseline system and the best performing teams according to the various classification measures used for each emotion. The table reports the value of precision (P), recall (R) and F1 measure for each emotion and the micro-F1 for all three emotional classes. The micro-F1 score is the official metric used in this task. The baseline model uses GloVe [185] embedding and followed by a simple recurrent architecture. The top performing systems reported used deep learning architectures and ensemble various types of static and contextualized embeddings.

Regarding the proposed model variants, *Model-A* gives the best performance F1 for each emotion and the overall micro-F1 score. However some variants of this model give better recall or precision values for different emotions, *Model-A* compromise between these values to give the best F1 for each emotion. Removing the self-attention layer in the classifier -*Model-B*- degraded the results. Also, inputting a condensed representation of the all conversation rather than the last turn -*Model-C*- did not improve the results. Even modeling the turns difference only -*Model-D*- gives better results over *Model-C*. These proves empirically the importance of the last turn in the classification performance. This is clear for *Model-E* where the classifier is learned only by inputting the last turn of the conversation. Ensemble the forward and backward models was more useful than using the forward model only -*Model-F*.

Comparing the results for different emotions and different models, we notice the low performance in detecting happy emotion. This validate the same conclusion of Chatterjee et.al in [33]. They justify this by the difficulties even for human level annotation to discriminate between happy and many other emotions. The model shows a significant improvement over the EmoContext organizer baseline (F1: 0.5868). Also, comparing to other participants in the same task with the same datasets, the proposed model gives competitive performance and ranked 9^{th} out of more than 150

participants. The proposed model can be used to model multi-turn and multi-parties conversations. It can be used also to track the emotional changes in long conversations.

One of the most attractive outcomes of applying the attention mechanism is its ability to process all the input sequences with different weights of attentions. It usually pays closer attention to the most important parts that influence the network decision. To validate these findings, we compared the most important tokens in terms of attention scores with sentiment and emotional lexicons. We found EmoLex proposed by Mohammad et. al. in [161, 162] as a good example. EmoLex is created with a high-quality, moderate-sized, emotion and polarity lexicon. It has entries for more than 10,000 word-sense pairs. We extracted the words related Emocontext emotions for happy (Joy) and sad (sadness) and angry (anger).

Table 4.3 shows the results of matching the top 20% attention scored tokens with EmoLex in both validation (Dev) and testing sets. The self-attention layers proposed in the first T_1 and last turn T_3 in the conversation seem to pay close attention to the corresponding emotional words. This is clear with the diagonal look in the table. However the mentioned difficulties in Happy emotion detection, the self-attention focuses in parts of text related to joy with a significant difference between the sadness and anger lexicon words. This significance is decreased between the sadness and anger words. However, the attention model is well focused to the correct emotions.

			Lexicon-based		
		Datasets	Joy	Sadness	Anger
Attention-based	Happy	(V)	**42.57%**	4.95%	4.05%
		(T)	**39.27%**	7.97%	7.36%
	Sad	(V)	21.66%	**40.58%**	23.04%
		(T)	20.59%	**32.25%**	26.04%
	Angry	(V)	21.07%	26.05%	**39.73%**
		(T)	22.02%	22.97%	**35.02%**

TABLE 4.3: Matching Percentages of emotion related words in the top 20% attention scored parts of the text in T_1 and T_2 in Validation (V) and Testing (T) datasets

4.6 Conclusions

In this chapter, we consider the problem of emotions detection and classification in textual conversations. We propose a new model which participated to the Semeval-2019 Task-3 [32] for contextual emotion detection in text. The task is to predict the emotional status of the last utterance in two party textual dialogues. The task considered three basic emotion - happy, sad, and angry - that mostly occur in textual conversations. The proposed model makes use of deep transfer learning rather than the shallow models for language modeling. The model assume that the emotional state of a specific utterance is closely connected to the immediate change from previous states. Therefore, the model pays close attention to the first and the last turns written by the same person in 3-turns conversations. The classifier uses self-attention layers and the overall model does not use any special emotional lexicons or feature engineering steps.

The results of the model and its variants show a competitive results compared to the organizers baseline and other participants. Our best model gives micro-F1 score of 0.7582 and ranked the 9^{th} out of more than 150 other participants. The proposed model could be generalized to model multi-parties long conversations. The model can be applied to other emotional and sentiment classification problems and can be modified to accept external attention signals and emotional specific word embeddings.

Emotions tend to be strong and clear feelings, unlike the mood, which is not as intense as emotions. However, mood is long lasting affective status. It lasts longer in time than emotion. Broadly, the mood could be divided into positive and negative categories - a 'bad' or 'good' mood. However it could be interesting in some application to assess and classify an individual's mood, extreme mood changes and swings can have implication for the mental health.

Chapter 5

Mood I: Temporal Mood Variation Modeling

We talk when we cease to be at peace with our thoughts.

Kahlil Gibran

Contents

5.1 Introduction

Mental health is important at every stage of life. It directly impacts the individual's emotional, psychological, and social behavior. Through all of life's stages – from childhood, adolescence, and adulthood – anyone could experience one or more mental health problems. Mental illness is a leading cause of one-third of disability worldwide [249]. The Diagnostic and Statistical Manual of Mental Disorders (DSM) produced by the American Psychiatric Association (APA) categorized mental disorders using a common language and standard criteria [61]. The most common main categories are mood and eating disorders. Mood disorders or sometimes referred to as affective disorders are characterized by a change in mood or affect, usually accompanied by a change in the overall level of activity. Examples of mood disorders are depression, bipolar disorder, self-harm, and suicidal thoughts and ideation. Eating disorders are marked by an obsession with food or body shape caused by several factors like genetics, brain biology, personality traits, and cultural ideals. Common examples of eating disorders are anorexia, bulimia, binge disorder, and pica. However the continuing effort in categorizing mental illness, the DSM states that "there is no assumption that each category of mental disorder is a completely discrete entity with absolute boundaries dividing it from other mental disorders or from no mental disorders" [61, p. 16]. In this chapter, we pay close attention to three common mental health problems - depression, anorexia, and self-harm.

Depression is a common mental disorder. Globally, more than 300 million people of all age stages suffer from depression [129]. It has a direct and indirect effect on economic growth because of its major impact on productivity. Depression also has dramatic consequences not only for those affected but also for their families and their social and work-related environments [239]. It may be the psycho-physiological basis for panic and anxiety symptoms. Panic disorder has been increasingly focused on health services and the media, where it affects young people aged 20-40. The incidence of these disorders affects 22% of the adult world population. At its worst consequences, depression is one of the major causes of suicide [255].

Anorexia is considered one of the most common eating disorder. It is characterized by low weight, the worry of gaining weight, and a powerful need to be skinny, leading to food restriction. Many who suffer from an eating disorder see themselves as overweight although they could be thin [111]. Individuals with eating disorders have also been shown to have lower employment rates, in addition to an overall loss of earnings. Eating disorder sufferers who are experiencing an overall loss in earnings associated with their illness are also magnified by the excess of health-care costs. According to the National Eating Disorder Association (NEDA), up to 70 million people worldwide suffer from eating disorders [236]. Eating disorder symptoms are beginning earlier in both males and females. As estimated, 1.1 to 4.2 percent of women suffer from anorexia at some point in their lifetime [97]. Young people between the ages of 15 and 24 with anorexia have 10 times the risk of dying compared to their peers of the same age.

Self-harm is a very common problem, and many people are struggling to deal with it [119]. Several illnesses are associated with self-harm, including borderline personality disorder, depression, eating disorders, anxiety, or emotional distress [62]. Self-harm occurs most often during the teenage and young adult begin around age 14 and carry on into their 20s, though it can also happen later in life [119]. There is also an increased risk of suicide in individuals who self-harm and it is found in 40% to 60% of

suicides [90].

Individuals living with mental disorders do not always receive adequate treatment. Even in economically-advantaged societies, a large treatment gap exists which refers to the difference in the proportion of at-risk people who have disorders and the proportion of those individuals who receive care [116]. There are always ongoing trials to close this gap by the integration of mental health services into primary care [201] or by task sharing and capacity building [1]. However, mental disorders and at-risk individuals may find other ways to reduce stigma and discrimination. The WHO's mental health action plan for the next two decades [173] calls for supporting "information systems, evidence and research," which requires new development and improvements in global mental health surveillance capabilities. Therefore, research on mental health has turned to use web data sources in particular social media data.

Social media has indeed become increasingly used, not only by adults but also at different age stages with over 3 billion active users worldwide [225]. Mental disorder sufferers turn to online social media and web forums not only to ask for information on specific conditions but also for direct and indirect emotional support, learning coping strategies, sharing experiences, and reducing the feeling of isolation [107]. Even though social media can be used as a very helpful tool in changing a person's life, it may cause conflicts that can have a negative impact. This adds responsibilities for content and community management for monitoring and moderation. With the increasing number of users and their content, these operations turn out to be extremely difficult. Many social media providers try to deal with this problem by reactive moderation. In reactive moderation, users report any inappropriate, negative, or risky user-generated content. However, if it may reduce the workload or the cost of moderating, it is not enough, especially for handling at-risk user threads or posts.

This problem motivates the eRisk-2017 task organizers to initiate the pilot task for detecting depression from user posts on Reddit[1] as a Part of the Conference and Labs of the Evaluation Forum (CLEF) [141]. In eRisk-2018 the extension of the study was planned to include detection of anorexia. The datasets composed of user writings grouped by 10 chunks. Each user data chunk contains 10% of the total writings for the corresponding user. In eRisk-2019, a continuation of anorexia task in addition to another task for early detection of signs of self-harm. In this task, no training dataset is provided. Tasks organizers proposed employing new evaluation measures besides the traditional classification measures. Besides, they change the processing manner and move from chunk-based release the datasets to item-based processing. The main idea in all these tasks is to detect such problems or signs of risks from users posts as early as possible using the minimum amount of user writings.

In this chapter, we present our participation in both versions of the competitions - eRisk-2018 (chunk-based) and eRisk-2019 (item-based). We propose the Temporal Mood Variation (TMV) architecture to model the mood variation detected from user posts through multi-stage learning phases. Besides, we propose the Deep Mood Evaluation Module (DMEM) and integrate it into the TMV architecture. We test multiple model variants and obtain interesting results.

[1]Reddit is an open-source platform where community members (red-ditors) can submit content (posts, comments, or direct links), vote submissions, and the content entries are organized by areas of interests (subreddits).

The organization of the chapter is as follows. In section 5.2, the related work is introduced. Section 5.3 describes the problem definition, proposed models, and results of the chunk-based processing for depression and anorexia datasets of eRisk-2018. Section 5.4 elaborates on the differences in the proposed models, their variants, and results for anorexia and self-harm datasets of eRisk-2019.

5.2 Related Work

5.2.1 Language of At-risk users

Previous researches on social media have established the relationship between an individual's psychological state and his/ her linguistic and conversational patterns [182, 167]. Many language features could characterize an individual's mental disorder and emotional distress. Theoretically, this is based on psycho-linguistics which was first introduced by an American psychologist in 1936 [113]. It is the study of the aspects associating the mental state with the language and speech of individuals. Psycho-linguistics postulates that words and features used in everyday spoken and written language can reveal individuals' thoughts, emotions, and motivations. Empirically, on social media posts, a study on twitter [58] discriminated the users diagnosed with depression by their increased use of first-person pronouns and fewer references to third persons. Another study used differences in word count, references to ingestion, sadness, swear words, article words, and positive emotion. Even for other languages, a similar study in Japanese [244] concluded that at-risk users have significantly higher ratios in using first-person and possessive pronouns and negative emotion words in their posts. This is consistent with a recent meta-analysis study [67] that points out the use of first-person singular pronoun as a linguistic marker for depression. This reflects their loneliness, self-focused attention, and psychological distancing from others. Other studies combined these basic markers with more features like higher character counts [253], fewer references to past and present tense [211], readability scores [242] and perceptual processes for feeling [68]. However, most of the studies in this area have focused on depression, other features have been observed for different mental health problems.

Regarding eating disorders, specifically anorexia, a recent study analyzed the content related to anorexia that is shared on Tumblr[2] [54]. The author captures diagnostic information by observing some affective, social, cognitive, and topic modeling features in social media. Examples of these features are selected variables from psycho-linguistic lexicons, lexical density, and verbal fluency measures. Other features for detecting individuals diagnosed with anorexia are related to author profiling features [198], domain-related vocabulary [174], and temporal variations behavior [197].

However, while all the previous studies could identify features that allow the classification between controlled and at-risk social media users, the language differences in communicating about different mental health problems remains an open question [79]. This makes it difficult to have unified feature sets that can detect different sources of mental health risks on online forums. In addition, having such good feature combinations is more influenced by mental health professionals than linguistics. In this context, and for a better sense of generalization, the proposed models introduced in this chapter use the text sequences of user posts without any handcrafted or lexicon features. This enables the proposed model to be used for detecting other mental

[2] https://www.tumblr.com/

health problems and in general for text classification.

Regarding the definitions of the affective concepts mentioned in chapter 1, the temporal aspect plays an important role in characterizing the mood and its consequences. Recent psychological studies showed the correlation between an individual's mental status and mood variation over time[19, 181]. It is also evident that some mental disordered may have chronic week-to-week mood instability. It is a common presenting symptom for people with a wide variety of mental disorders, with as many as 8 of 10 patients reporting some degree of mood instability during assessment[141]. These studies suggest that clinicians should screen for temporal mood variation across most common mental health disorders.

5.2.2 Text Classification

Affective Computing (AC) is the study and development of systems and devices that can recognize, interpret, process, and simulate human affects. It is an interdisciplinary field spanning computer science, psychology, and cognitive science [29]. AC has become an emerging and important branch of Artificial Intelligence (AI). The overarching goal is to create systems that can interpret the emotional state of humans and adapt its behavior to provide intuitive and appropriate emotionally informed responses [235]. Since subjectivity is a human mind feature, it is not available for objective observation or verification. Subjective experiences are created from the subject perspective, and they reflect the individuals' intentions, beliefs, feelings, emotions, sentiment, or even an individual's mental state. In NLP research, the detection of these kinds of subjectivity can be formulated as text classification and categorization task. It is the process of assigning tags or categories to text according to its content. There are many approaches for automatic detection of affects in text. One of the most well-established approaches in text classification is Machine Learning (ML).

Since the features in ML are basically numerical attributes, traditional NLP models start with extracting some important features from the text. These models are considered to be incomplete as they focus only on these features and lose the contextual information in the text. Another way is to apply text vectorization (embedding). As discussed in Section 2.4, traditional NLP modules start with feature extraction from text such as the count or frequency of specific words, predefined patterns, Part-of-Speech tagging, etc. These hand-crafted features should be selected carefully and sometimes with an expert view. However these features are interesting [240], sometimes they lose the sense of generalization. Another recent trend is the use of word and document vectorization methods. These strategies that convert either word, sentences, or even overall documents into vectors take into account all the text not just parts of it. There are many ways to transform a text to high-dimensional space such as term frequency and inverse document frequency (TF-IDF), Latent Semantic Analysis (LSA), Latent Dirichlet Allocation (LDA), etc [146]. This direction was revolutionized by Mikolov et al. [157, 156] who proposed the Continuous Bag Of Words (CBOW) and skip-gram models known as Word2vec. It is a probabilistic based model that makes use of two-layered neural network architecture to compute the conditional probability of a word given its context. Based on this work Le et al. [128] propose the Paragraph Vector model. The algorithm which is also known as Doc2vec learns fixed-length feature representations from variable-length pieces of texts, such as sentences, paragraphs, and documents. Both word and document vectors are trained using stochastic gradient descent and back-propagation shallow neural network language models. The development of Universal Language Model Fine Tuning (ULMFiT) is considered like

moving from shallow to deep contextual pre-training word representation [100]. This idea has been proved to achieve Computer Vision (CV)-like transfer learning for many NLP tasks. ULMFiT makes use of the AWD-LSTM (Average stochastic gradient descent - Weighted Dropout LSTM) language model proposed by Merity et al. in 2017 [154]. The same 3-layer LSTM recurrent architecture with the same hyperparameters and no additions other than tuned dropout hyperparameters are used. The classifier layers above the base LM encoder is simply a pooling layer (maximum and average pool) followed by three fully-connected linear layers. The overall models significantly outperform the state-of-the-art on six text classification tasks including three tasks for sentiment analysis.

Other interesting work on text distributed representation is the bayesian inversion proposed by Taddy in [233] which uses Bayes formula to compute the probabilities of a document belonging to a topic. Given a document d and label y, Bayes formula is:

$$p(y|d) = \frac{p(d|y)p(y)}{p(d)}$$

For classification problems, $p(d)$ can be ignored since d is fixed. $p(d|y)$ is estimated by first training the text vectorization model on a subset of the corpus with label y, then using the skip-gram objective composite likelihood as an approximation. As discussed in [233], Bayesian inversion will not always outperform other classification methods. It rather provides a simple, scalable, interpretable, and effective option for classification whenever distributed representations are used.

The attention mechanism is considered as one of the recent trends in NLP models [13]. It can be described as mapping a query and a set of key-value pairs to an output, where the query, keys, values, and output are all vectors. The output is computed as a weighted sum of the values, where the weight assigned to each value is computed by a compatibility function of the query with the corresponding key. This can be seen as taking a collection of vectors, whether it could be a sequence of vectors representing a sequence of words, or unordered collections of vectors representing a collection of attributes and summarize them into a single vector. This summarization is done by scoring each input sequence with probability-like scores obtained from the attention. This helps the model to pay close attention to the sequence items with higher attention scores. Attention-based models have been successfully applied to many tasks, including reading comprehension, abstractive summarization, textual entailment, learning task-independent sentence representations, machine translation and language understanding [229].

In the proposed models discussed in this chapter, we apply different text vectorization methods and propose a new modification to the default ULMFiT model by proposing self-attention layers and a bi-directional version of the AWD-LSTM. We apply the attention layer to get more accurate representation on the writing-level.

5.3 Chunk-based Processing

In this section, we discuss the problem of early risk detection where the collected user writings are released in chunks. We consider the detection of signs of depression and anorexia in eRisk-2018 [142]. We proposed a new architecture that considers the temporal aspects of users' mood variations. The originality of our approach is to perform the detection through two main learning phases using text vectorizations

and state-of-the-art language modeling. The first phase is to construct a time series representing temporal mood variation through users' posts. The second phase is to build variable-length time series classification models to obtain the proper decision. The main idea is to give that decision once the time series prove clear signs of mental disorder from current and previous mood extracted from the content.

This section is organized as follows. Section 5.3.1 introduces the problem definition of early risk detection and used datasets. Section 5.3.2 presents the proposed model architecture. In Section 5.3.3, we discuss the model variants and hyperparameter settings. The evaluation results and discussions are presented in section 5.3.4. We conclude the study and experiments in section 5.3.5.

5.3.1 Datasets

In eRisk 2018, two tasks are presented [142]. Both tasks are considered as a binary classification problem. The first task is to discriminate between depressed and non-depressed users while the second one is between users diagnosed with anorexia and non-anorexia. The datasets are a dated textual data of user posts and comments -posts without titles- on Reddit. The training and testing datasets are divided into 10 chunks in chronological order. Each chunk contains 10% of the user's posts. A brief summary and statistics for these datasets are provided in Tables 5.1 and 5.2. The goal is not only to perform classification but also to do it as early as possible using the minimum amount of data or chunks for each user. The decision corresponding to each user data chunk could be one of the classes or could be postponed for future chunks. At the end of the 10^{th} chunk, all classification propositions must have been submitted.

	Training Dataset	Testing Dataset
No. of Users (Depressed/Non-Depressed)	886 (135/752)	820 (79/741)
No. of Submissions	531,394	544,447
Avg. No. of Submissions/User	608.04	663.95
No. of Sentences	1,157,230	1,336,379
Avg. No. of Sentences per Submission	2.29	2.45
Avg. Sentence Size (words)	14.31	14.26
Vocabulary Size	234,181	222,201

Table 5.1: Summary on eRisk-2018 Task.1 - Depression Datasets

	Training Dataset	Testing Dataset
No. of Users (Anorexia/Non-Anorexia)	152 (20/132)	320 (41/279)
No. of Submissions	84,834	168,507
Avg. No. of Submissions/User	558.12	526.58
No. of Sentences	193,026	370,281
Avg. No. of Sentences per Submission	2.28	2.12
Avg. Sentence Size (words)	14.74	14.30
Vocabulary Size	81,497	103,380

Table 5.2: Summary on eRisk-2018 Task.2 - Anorexia Datasets

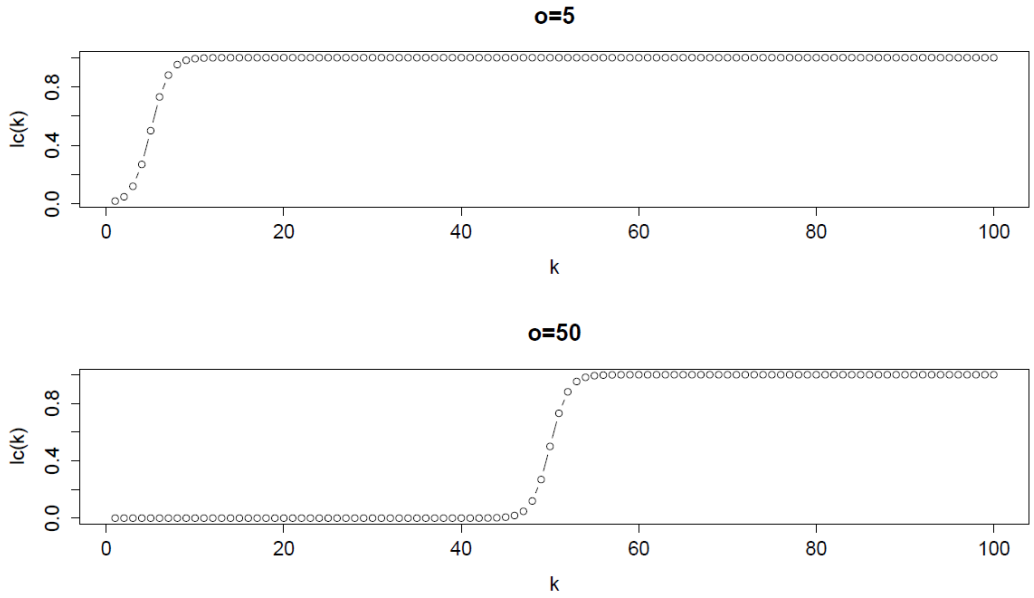


FIGURE 5.1: Figure from [142] showing the latency cost function: $lc_5(k)$ and $lc_{50}(k)$

For evaluation, the classical classification performance measures (Precision, Recall, and F1) are computed for each run. In addition, eRisk organizers incorporate an error measure called Early Risk Detection Error (ERDE) [26] into the evaluation process. ERDE considers both the correctness of the decision and the delay taken by the model to make the decision. It introduces a cost function $lc_o(k)$ for true positive decisions, where (k) denotes the delay in terms of the number of processed user writing before the model decision (d). Suppose that gt denotes the golden truth, The ERDE is given by:

$$ERDE_o(d,k) = \begin{cases} c_{fp} & d = positive \quad and \quad gt = negative \\ c_{fn} & d = negative \quad and \quad gt = positive \\ lc_o(k) & d = gt = positive \\ 0 & d = gt = negative \end{cases} \tag{5.1}$$

The values of (c_{fp}) and (c_{fn}) depends on the application domain and the interpretation of false positives and false negatives. In eRisk taks, the organizers fixed the value of $c_{fn} = 1$ and $c_{fp} = 0.1296$. The value of (c_{fp}) is computed according to the proportion of positive cases in eRisk-2017 test sets and fixed to this value for eRisk-2018 tasks. The cost function for true positives $lc_o(k) \in [0,1]$ is defined by:

$$lc_o(k) = 1 - \frac{1}{1 + e^{(k-o)}} \tag{5.2}$$

Figure 5.1 shows the cost function for $lc_5(k)$ and $lc_{50}(k)$. The function is a monotonically increasing function of k. The cost grows quickly, exactly after the value of (o). Two version of ERDE is used by setting $o = 5$ and $o = 50$ ($\text{ERDE}_{5,50}$). It takes into account the correctness of the (binary) decision and the delay taken by the system to make the decision [141].

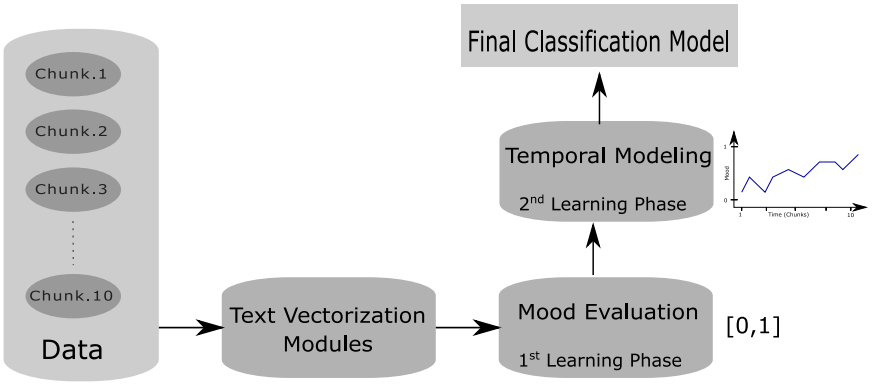


FIGURE 5.2: Block diagram of the main architecture of Temporal Mood Variation (TMV) model

5.3.2 Proposed Models

Temporal Mood Variation Model

The temporal aspects of the eRisk-2018 tasks inspired us to model the temporal mood variation trough user's text content. The average number of days ranging from the first submission to the last submission is approximately 600 days [140]. So, determining how user's posts and comments vary from positive to negative and vise versa through time is worth inspecting. In the proposed models, time aspects are given as chunks. The main idea is to process user submissions for each chunk and determine the probability of how positive or negative the chunk is. The proposed architecture of our models is shown in Figure 5.2.

Step 1 - Text Vectorization Module: The input of this module is the list of textual information divided into ten chunks. The chunks are chronologically ordered as discussed in Section 5.3.1. The first step is to build a text vectorization model using all the text chunks. Two static text vectorization models are used. These models are the Word2vec and its evolution, the Doc2vec [158, 128]. We have tested the two alternatives to Doc2vec specifications - distributed memory (DM) and distributed bag-of-words (DBOW) [128]. We also keep track of the text for each user in every chunk and its label embedded in the model. Also, we built a vectorization model for positive and for negative and did not use any external resources. This module can be considered as an unsupervised learning phase.

Step 2 - Mood Evaluation Module: Our models are based on the work of Matt Taddy in [233] about Bayesian inversion. One of the interesting conclusions from this work is that any distributed representation can be turned into a classifier through inversion via Bayes rule. In our proposed model, we segmented the text of each chunk into sentences and scored each sentence through each vectorization model. The mood of the overall chunk is evaluated simply by normalizing the count of positive and negative sentences using the inversion technique. Each chunk will have a number between [0,1]. This can be considered as the probability of how positive (risky) the chunk is. Processing all chunks leads to a ten-points time series for the ten chunks for each user in the training datasets. Mood evaluation using the inversion technique is considered as the first learning phase in our proposed architecture.

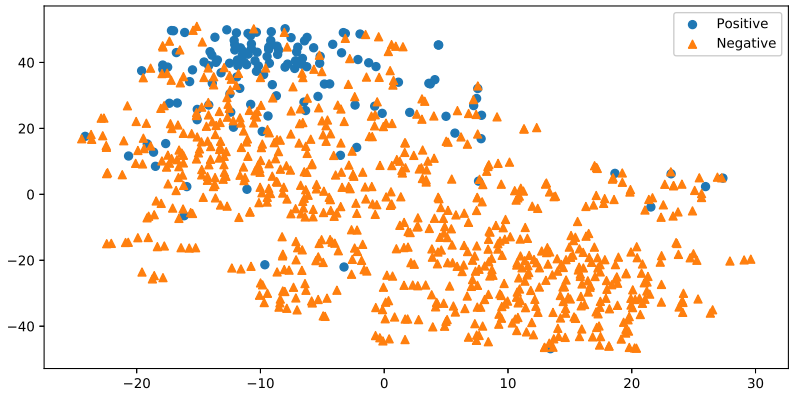


FIGURE 5.3: t-SNE reduced time series information for ten chunks per user in eRisk-2018 Task-1 - Depression training dataset

Step 3 - Temporal Modeling Module: Another learning phase is to build machine learning models to learn some patterns from the resulted time series to come up with the final classification model. In the ideal case and for the complete time series, we would have only one model. But since we should not wait for the complete time series we built multiple models for different sizes of time series to be able to give a decision without having to wait for the ten chunks. Figure 5.3 shows an example of a two-dimensional representation of the complete time series for the depression task using t-SNE [147]. These time series will be the training set of the second learning phase. The separation between positive and negative users is obvious. It is expected that this separation would not be as ideal as this in testing but it will exist.

We tried also to encapsulate text vectorization and mood evaluation modules and proposed Deep Mood Evaluation Module (DMEM). This module is based on ULMFiT architecture [100] and the idea of transfer learning for language modeling in addition to using attention layers for classifications.

Deep Mood Evaluation Module (DMEM)

We propose a modification of the basic architecture of the ULMFiT mainly by adding attention to the model. The proposed architecture will help the model to focus on the important parts of the text that influence the network decision. Figure 5.4 shows the proposed model and the separation between encoder layers (text vectorization module) and classifier layers (mood evaluation module).

The input sequence is passed to the embedding layer then the three Bi-LSTM layers to form the output of the encoder. The encoder output has the form of $X_i = \{x_1^i, x_2^i, x_3^i, \ldots, x_N^i\}$ where N is the sequence length. The attention layer takes the encoded input sequence and computes the attention scores S^i. The attention layer can be viewed as a linear layer without bias.

$$\begin{aligned} \alpha^i &= \{W^i.X^i\} \\ S^i &= log[\frac{exp(\alpha^i)}{\sum_{j=1}^{N} exp(\alpha_j^i)}] \end{aligned} \tag{5.3}$$

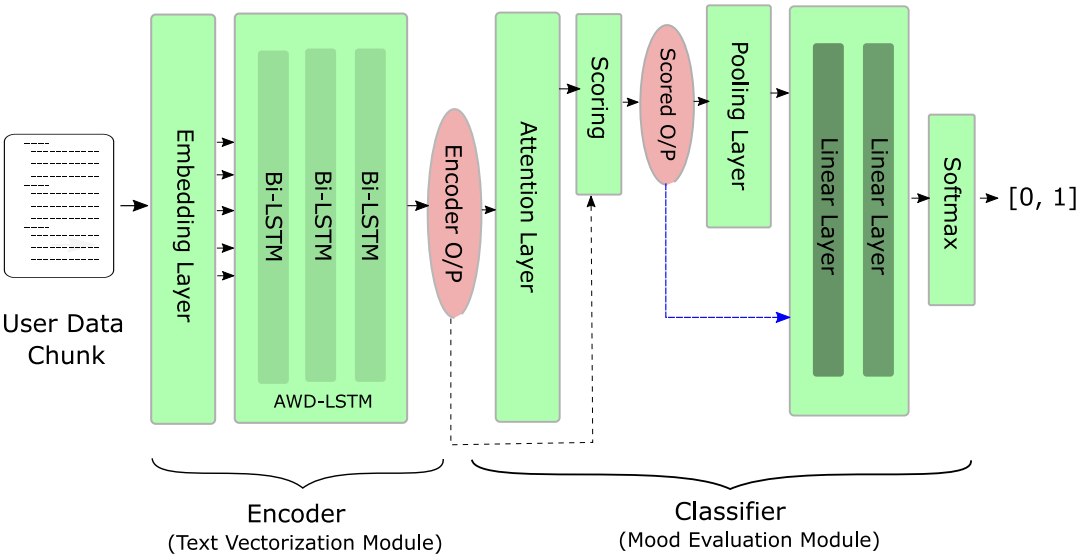


FIGURE 5.4: Deep Mood Evaluation Module (DMEM)

Where W^i is the weight of the attention layer of the i^{th} sequence. The attention scores S^i is used to compute the scored sequence $O^i = \{o_1^i, o_2^i, o_3^i, \ldots, o_N^i\}$ which has the same length as the input sequence.

$$O^i = S^i \odot X^i \tag{5.4}$$

Where $\odot$ is the element-wise multiplication. Since the input sequence to the attention layer (encoder output) resulted from Bi-LSTM layers, the last element in the scored output S_N^i can be used for representing the whole sequence. But as we used attention scores, the whole sequence is represented by the weighted sum of all output sequences $\bar{O}^i$. This is done by:

$$\bar{O}^i = \sum_{<N>} S^i \odot X^i \tag{5.5}$$

We tried this scoring strategy in addition to the base model which skip the attention layer and move the output of the encoder directly to the classifier layers. For classification layers, a simple concatenation between the maximum and average pooling in addition to the scored output is inputted to a group of two different sizes fully connected linear layers. The output of the last linear layer is passed to the Softmax to form the network decision.

Training the over whole models comes into three main steps proposed in [100].

1. The LM is initialized by training the encoder on a general-domain corpus (Wikitext-103 dataset [155]). This helps to capture the general features of the language, preserve low-level representations, and adapt high-level ones.

2. The pre-trained LM is fine-tuned using the training datasets for both tasks.

3. The classifier and the encoder are fine-tuned on the target task using different strategies for each layer group.

The training of the architecture is done using slanted triangular learning rates (STLR), discriminative fine-tuning (Discr) and layers gradual unfreezing proposed for ULMFiT with the same hyperparameters settings [100]. We train the model on the forward language models for both the general-domain and task-specific datasets. Training the attention layer uses the same learning rates and cycles used in the classification layers group.

5.3.3 Experimental Setup

Table 5.3 summarizes the main steps of our proposed system variants for both tasks and the starting chunk number for each run to make the first positive decisions.

	Step 1	Step 2	Step 3	Starting Chunk
LIRMMA	Doc2vec	Bayesian Inversion	MLP	8
LIRMMB	Word2vec*	Bayesian Inversion	MLP	5*
LIRMMC	Word2vec	Bayesian Inversion	RF	3
LIRMMD	Word2vec	Bayesian Inversion + Moving Average	———	1
LIRMME	Word2vec	Bayesian Inversion + Moving Average	———	1
$DMEM_A$	AWD-LSTM (pre-trained)	Attention + Pooling Classifier	MLP	3
$DMEM_B$	AWD-LSTM	Attention + Pooling Classifier	MLP	3
$DMEM_C$	AWD-LSTM (pre-trained)	Pooling Classifier (No Attention)	MLP	3

TABLE 5.3: Summary of the proposed architecture variants for eRisk-2018 tasks. Cells with (*) stand for different selection for Anorexia Task-2

For document vectorization (Doc2vec), the resultant vectors had 200 dimensions. The model used a context window of 10 words and a minimum of two for word counts. It used a negative sampling loss with DBOW version and trained for 20 training epochs. In the word level vectorization, the vector size of a word had a dimension of 200 with a context window size of five words. Hierarchical softmax was used and a minimum count of two words was considered. In the second learning phase and for temporal modeling, the used architecture of the Multi-layered perceptron (MLP) had two hidden layers with ten neurons each. Concerning the Random Forest (RF) classifier, ten estimators were used.

For LIRMMB in anorexia task, we used Doc2vec rather than Word2vec and it starts to detect positive users in the eighth chunk. We expected that Doc2vec could give better results, especially for small size datasets. Hence we proposed LIRMMD and LIRMME to give a decision from the first chunk, we substitute the second learning phase with a window moving average from the output of the Bayesian inversion technique. For LIRMMD, we assumed the positive users will have a risky mood in the first chunks than the lasts. Two varying thresholds were used; one for the number of sentences and the other for the positive probability threshold. The size of the averaging window is three and the probability changing from 0.6 with the number of sentences higher than 100 to 0.8 and zero for sentence count threshold. For LIRMME, the difference comes from the assumption that a higher probability threshold was given to last chunks than the first chunks with the same sentence thresholds. The risk probability starts with 0.8 in the first chunk to 0.6 in the last chunk.

For $DMEM_A$ variant, we use the same set of hyperparameter of AWD-LSTM proposed by [154] replacing the LSTM with Bi-LSTM and keep the same embedding size of 400 and 1150 hidden activations. We used a weighted dropout of 0.2 and 0.25 as the input embedding dropout and the learning rate is 0.004. We fine-tuned the LM by training datasets provided in Tables 5.1 and 5.2. We train the LM for 14 epochs using a batch size of 128 and limit the number of vocabulary to all token that appears more than twice. For classifiers, we used masked self-attention layers and concatenation of maximum and average pooling. For the linear block, we used a hidden linear layer of size 100 and apply dropout of 0.4. We used Adam optimizer [63] with β_1 = 0.8 and β_2 = 0.99. The base learning rate is 0.01. We used the same

batch size used in training LMs. For training the classifier, we create each batch using weighted random sampling to handle the problem of imbalance in the datasets. We train the classifier on the training set for 30 epochs and select the best model on the validation set to get the final model. We tried two other variants of the DMEM. The first one ($DMEM_B$) use the AWD-LSTM encoders without pre-training step while the other one ($DMEM_C$) skip the attention layer for the classification to use only concatenation pooling layer. The three variants of DMEM will give a good ablation analysis of the model.

5.3.4 Results & Discussions

Evaluation Results

Upon the submission of the last chunk, the evaluation process started for all runs results. As mentioned in Section 5.3.1, the two versions of ERDE, in addition to the classical classification measures: Precision(P), Recall (R), and F1-Measure (F1) are used. Tables 5.4 and 5.5 show the evaluation results of all proposed variants for both tasks. Also, we show the results of the best performing models for each evaluation metric in both tasks.

	$ERDE_5$	$ERDE_{50}$	F1	P	R
FHDO-BCSG [243]	9.50%	**6.44%**	**0.64**	0.64	0.65
UNSL [77]	**8.78%**	7.39%	0.38	0.48	0.32
RKMVERI [183]	9.81%	9.08%	0.48	**0.67**	0.38
UDC [27]	15.79%	11.95%	0.18	0.10	**0.95**
LIRMMA	**10.66%**	9.16%	**0.49**	**0.38**	0.68
LIRMMB	11.81%	9.20%	0.36	0.24	**0.73**
LIRMMC	11.78%	9.02%	0.35	0.23	0.71
LIRMMD	11.32%	**8.08%**	0.32	0.22	0.57
LIRMME	10.71%	8.38%	0.37	0.29	0.52
$DMEM_A$	**9.50%**	**6.60%**	**0.61**	**0.52**	0.72
$DMEM_B$	10.12%	7.79%	0.54	0.47	0.63
$DMEM_C$	9.88%	6.82%	0.56	0.62	0.51

TABLE 5.4: Results of the proposed runs for eRisk-2018 Task.1 - Depression

Discussions

From the first look of the results, It is clear that the DMEM models outperform all other variants of the temporal mood variation models. The use of deep AWD-LSTM language modeling rather than the shallow Word2vec and Doc2vec is very useful. The effect of transfer learning is obvious for $DMEM_A$ and $DMEM_C$. The main reason is that the language model encoder is pre-trained by general-purpose text data before being used in the model. This improvement is more remarkable for the anorexia task with much less training data. The attention layer in the classification stage ($DMEM_A$) of the model helps to focus on the most important parts in long text chunk for active users. The use of MLP in the temporal modeling -second learning phase of the model- from the third chunk helps in the early detection of risky users. In contrast with all

	$ERDE_5$	$ERDE_{50}$	F1	P	R
FHDO-BCSG [243]	**11.98%**	**6.61%**	**0.85**	0.87	0.83
UNSL [77]	12.93%	9.85%	0.79	**0.91**	0.71
RKMVERI [183]	12.17%	8.63%	0.67	0.82	0.56
LIRMMA	**13.65%**	13.04%	**0.54**	**0.52**	0.56
LIRMMB	14.45%	12.62%	0.52	0.41	0.71
LIRMMC	16.06%	15.02%	0.42	0.28	**0.78**
LIRMMD	17.14%	14.31%	0.34	0.22	0.76
LIRMME	14.89%	**12.69%**	0.41	0.32	0.59
$DMEM_A$	**12.90%**	**8.16%**	**0.8**	**0.73**	**0.88**
$DMEM_B$	14.66%	10.33%	0.75	0.79	0.73
$DMEM_C$	13.46%	9.02%	0.78	0.72	0.84

TABLE 5.5: Results of the proposed runs for eRisk-2018 Task.2 - Anorexia

runs using Doc2vec (LIRMMA for task-1 and LIRMMA & LIRMMB for task-2) that started giving decisions later (eighth chunk).

Comparing word and document level vectorization, it is clear that Doc2vec behaves better than Word2vec in terms of classical classification measures. The runs with higher recall use word-level vectorization with either MLP or RF as the second learning phase. In the mood evaluation step, fake stories were misleading and made a lot of false-positive predictions. In addition, our models do not discriminate between user posts and comments (posts without titles) which could be beneficial for evaluating user mood.

For some at-risk users, first chunks posts don't have any proof of depression or anorexia and suddenly users started to express their status late. For the second learning phase, the model classifies the overall mood time series and late signs of disorders could not be predicted earlier by our models. So, in some runs (for both tasks) some moderation on the proposed assumptions (classification probability thresholds) are needed. Tables 5.6 and 5.7 show some statistics of all submitted runs compared to the proposed models. The ranking of our official participation best run and proposed DMEM runs for each evaluation metric is also included. The statistics of the depression task are for 45 runs of 11 teams. The anorexia task statistics on results are for 34 runs of 9 teams. Most of the teams have participated in both tasks with at least one run for each. All the variants of our models behave comparably with all other participants' runs. The improvement of the results of using DMEM especially for anorexia task is clear on the ranking for each evaluation measure.

The ERDE-score has been discussed critically as for the 2017 and 2018 chunk based settings [241]. The study and experiments show that it is not a meaningful metric for the described shared tasks. Only the correct prediction of a few positive samples has an effect on this score and the best results can therefore often be obtained by only minimizing false positives.

	$ERDE_5$	$ERDE_{50}$	F1	P	R
Average	10.33%	8.23%	0.42	0.37	0.55
Standard Deviation	1.13%	1.09%	0.12	0.15	0.16
Max	15.79%	11.95%	0.64	0.67	0.95
Min	8.78%	6.44%	0.18	0.1	0.15
Official Runs Rank	31	22	13	15	3
DMEM Runs Rank	8	3	2	10	4

Table 5.6: Statistics on 45 participating runs results and our ranks for eRisk-2018 Task.1 - Depression

	$ERDE_5$	$ERDE_{50}$	F1	P	R
Average	13.31%	10.89%	0.56	0.63	0.58
Standard Deviation	1.62%	2.69%	0.19	0.22	0.2
Max	19.90%	19.27%	0.85	0.91	0.88
Min	11.40%	5.96%	0.17	0.15	0.1
Official Runs RanK	28	27	20	24	4
DMEM Runs Rank	15	8	4	19	1

Table 5.7: Statistics on 34 participating runs results and our ranks for eRisk-2018 Task.2 - Anorexia

5.3.5 Conclusion

In this section, we present the revised participation of LIRMM in the two eRisk-2018 tasks. Both tasks are for early detection of signs of depression and anorexia from users' posts on Reddit. We proposed the Temporal Mood Variation (TMV) model architecture that performs the classification through two phases of supervised learning. The first learning phase builds a time series representing the mood variation while the second learning phase is a classification model that learns patterns from the time series to detect early signs of such mental disorders. The proposed architecture used the text without any handcrafted features or lexicons. We tested multiple model variants that utilize modern text vectorizations and machine learning models combinations. Besides, we proposed a new modification to the TMV architecture which combines the text vectorization and mood evaluation modules into Deep Mood Evaluation Module (DMEM). DMEM uses modern state-of-the-art, attention-based, deep language modeling architecture rather than the shallow text vectorization models like Word2vec and Doc2vec. We proposed three variants of the model. The results show a significant improvement and outperform the official five submitted models in eRisk-2018 for both tasks. Comparing to the other contributions and baselines, the results are competitive for all used evaluation metrics.

Critiques have been raised in both tasks regarding the way of releasing the data. Owing to the fact that some users are more active than others, a high variance in chunks sizes for different users exists. A chunk could contain tens of writings while some other chunks have only a few ones. As a consequence, there are two main comments. The first is processing data by chunks is not realistic since the speed of

the data stream is not similar. The second one regarding the ERDE metric. It changes quickly from very low to very high penalty with the number of writings which could be a lot for active users. Task organizers take that into account in proposing the new tasks in eRisk-2019.

5.4 Item-based Processing

In eRisk-2019 [143], a continuation of the anorexia task with a new test set and a new task on early detection for signs of self-harm was proposed. To make the tasks more realistic, the datasets are released in an item-based manner rather than the chunk-based used in eRisk-2018. Each item is a user's writings (either a post or a comment). Besides, the test set also includes a negative user group who often post about anorexia (e.g. individuals who actively participate in the anorexia threads because they have close relatives suffering from this eating disorder). Developing predictive systems that process the data item-by-item is a good simulation for the actual problems that face social media providers for better moderation. Added to this, in chunk-based processing, the contribution of each user to the performance evaluation has a large variance (different for users with few writings per chunk vs users with many writings per chunk).

In this section, we present our participation in both tasks for early detection of anorexia and self-harm in eRisk-2019. We modify the TMV and DMEM models discussed in section 5.3.2 to work for user data items rather than chunks. In addition, we propose using different models' configurations in the second learning phase of TMV architecture.

This section is organized as follows. In Section 5.4.1, a description of the datasets of both early risk detection tasks and the used evaluation methods are presented. Section 5.4.2 presents the proposed models. The experimental setup and all model variants used are introduced in Section 5.4.3. In Section 5.4.4, the evaluation results and discussions are presented. We conclude the study and experiments in section 5.4.5.

5.4.1 Datasets

In eRisk-2019, three tasks are presented [143]. The first task (T1) is for early detection of signs of anorexia. It is a continuation of the same task in eRisk-2018. The second one (T2) is a new task in 2019 for early detection of signs of self-harm. No training data is provided for this task. Another task was proposed (T3) for measuring the severity of the signs of depression. In this section, we will describe the first two tasks (T1 and T2) that we have participated in.

Both tasks are considered as a binary classification problem. The datasets are a dated textual data of user posts and comments -posts without titles- on Reddit. The training and testing datasets are provided in streams of user writings (posts and comments). The data is ordered chronologically. A brief statistics and summary of these datasets are provided in Table 5.8. Task organizers set up a server that iteratively gives user writings to the participating teams. The goal is not only to perform classification but also to do it as early as possible using the minimum amount of writings for each user. A decision must be sent after processing each user's writing to continue receiving more. This decision could be positive or negative risk cases or postponed for future writings. A detailed description of the tasks and used evaluation metrics can be found in the corresponding task description paper [143].

	T1		T2
	Training	Testing	Testing
No. of Users (At-risk/Controlled)	472 (61/411)	815 (73/742)	340(41/299)
No. of writings	253,341	570,510	170,698
Avg. No. of writings/User	536.74	700.01	502.05
Avg. writings Size (words)	35.38	34.83	33.15
Vocabulary Size	117,090	210,763	105,448

TABLE 5.8: Summary of eRisk-2019 datasets for the two tasks (T1 - Anorexia and T2 - Self-harm)

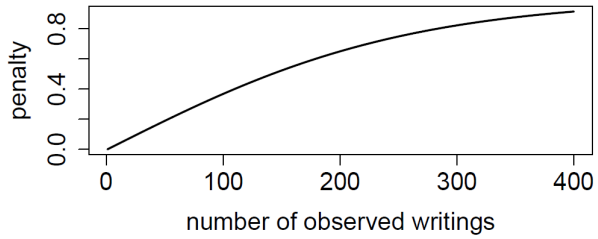


FIGURE 5.5: Figure from [143] showing Latency penalty with the number of processed writings

For evaluation, the classical classification performance measures (Precision, Recall, and F1) are used. Besides, the task organizers replaced the ERDE with a new measure to evaluate the model latency (latency-weighted F1). This new metric tries to overcome the limitations of ERDE observed in eRisk-2018. These limitations are:

- The cost function for true positives ($lc_0(k)$) discussed in section 5.3.1 goes quickly to one based on the value of ($o \in \{5, 50\}$).
- Even for a perfect model that detects positive users from the first rounds does not get ERDE=0
- ERDE is not an interpretable measure.

The latency-weighted F1 combines both the effectiveness and the delay of the models' decisions. It multiplies the F1 score with a penalty factor based on the median delay for each model. The delay is defined by the number of processed user writings before a true positive decision. The median is computed using all positive users for each test set in both tasks (T1 and T2). The penalty function $penalty(k)$ is given by:

$$penalty(k) = -1 + \frac{2}{1 + e^{-p(k-1)}} \tag{5.6}$$

The value of p was set such that the penalty equals to 0.5 at the median number of posts of a user. The value is computed and fixed to $p = 0.0078$, for both tasks. Figure 5.5 shows how the penalty increases in a smooth way rather than the sudden behavior of ERDE (sigmoid). A perfect system will get F1 and weighted F1 equal to 1. These make this measure more interpretable than ERDE.

5.4.2 Proposed Models

We use the TMV architecture with DMEM discussed in section 5.3.2. The used temporal aspect is items rather than chunks. Each item is formed by the users'

writings available until then. In the second learning phase for TVM, we tried different machine learning, statistical models, and counting of successive positive writings. We tested using the Bayesian Variational Inference (BVI) model [220].

Bayesian Variational Inference (BVI)

We can represent the problem of classifying users from the already classified (observed) writings as a variant of independent Bayesian classifier combination [220]. Figure 5.6 shows the graphical model for the proposed BVI where the observed random variable W_i^k represents if the i^{th} writing for the k^{th} user if it is classified as positive or negative such that:

$$\begin{aligned} W_i^k &\sim \text{Bernoulli}(\pi_{u_k}) \\ \pi_i &\sim \text{Beta}(\lambda, \gamma) \end{aligned} \tag{5.7}$$

The hidden variable u_k represents if the user will be classified as at-risk (anorexia, self-harm) or not. So we can say:

$$\begin{aligned} u_k &\sim \text{Bernoulli}(\kappa) \\ \kappa &\sim \text{Beta}(\alpha, \beta) \end{aligned} \tag{5.8}$$

The variables λ, γ, α and β are the hyper-parameters reflecting our *a priori* belief about the proportion of positive and negative users.

We are interested in the *posterior* distribution of the random variable U_k, which defines if the user is positive or negative, which is unfortunately intractable. We use a variational inference approach to compute an approximation such as in [220]. The approximation is obtained by solving the following equation for all variables Z_i conditioned on the observed data X:

$$\log\, q_i(Z_i|X) = \mathbb{E}_{j\neq i}[\log\, p(Z, X)] + \text{const.} \tag{5.9}$$

So, we start from a number of positive and negative user writings (N^d) where $d \in \{+, -\}$ for positives and negatives respectively. More specifically:

$$N^+ = \sum_{k,i} 1[W_i^k = 1], \qquad N^- = \sum_{k,i} 1[W_i^k = 0] \tag{5.10}$$

Then, the expected number of positive and negative writings for positive users can be represented by N_1^+ and N_1^- respectively. The same for negative users is N_0^+ and N_0^-. These values are computed as:

$$N_r^d = \sum_{k,i} \mathbb{E}[1[u_k = d]].[1[w_i^k = r]], \quad d \in \{+, -\}, r \in \{0, 1\} \tag{5.11}$$

We can estimate the expectation of the log of the probability to observe positive writings independently of the user category as $\mathbb{E}[\ln(\kappa)]$ and for negative writings as $\mathbb{E}[\ln(1-\kappa)]$ such that:

$$\begin{aligned} \mathbb{E}[\ln(\kappa)] &= \psi(\alpha + N^+) - \psi(\alpha + \beta + N^+ + N^-) \\ \mathbb{E}[1 - \ln(\kappa)] &= \psi(\beta + N^-) - \psi(\alpha + \beta + N^+ + N^-) \end{aligned} \tag{5.12}$$

Where ψ is the digamma function defined as the logarithmic derivative of the gamma function. In addition, we can estimate the expectation of the log probability

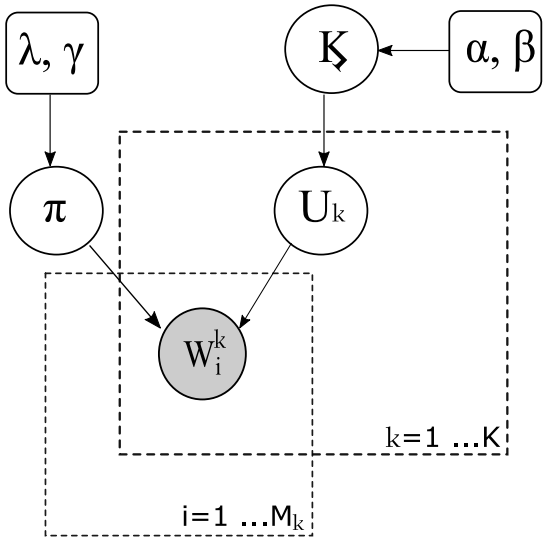


FIGURE 5.6: Graphical Model for BVI: The shaded node represents observed values, circular nodes are variables with a distribution and rectangular nodes are instantiated variables

for positive users to write positive writings as $\mathbb{E}[\ln(\pi_1)]$ and for negative users as $\mathbb{E}[\ln(\pi_0)]$ where:

$$\begin{aligned} \mathbb{E}[\ln(\pi_i)] &= \psi(\lambda + N_i^+) - \psi(\lambda + \gamma + N_i^+ + N_i^-) \\ \mathbb{E}[1 - \ln(\pi_i)] &= \psi(\gamma + N_i^-) - \psi(\lambda + \gamma + N_i^+ + N_i^-) \end{aligned} \tag{5.13}$$

So, the expectation of a user to be positive or negative can be obtained as:

$$\begin{aligned} \ln(\rho_j^k) = \sum_i^{M_k} W_i^k \, \mathbb{E}[\ln(\pi_j)] + (1 - W_i^k) \, \mathbb{E}[\ln(1 - \pi_j)] \\ + (\alpha - 1) \, \mathbb{E}[\ln(\kappa)] + (\beta - 1) \, \mathbb{E}[\ln(1 - \kappa)] \\ \mathbb{E}[1[U_k = j]] = \frac{\rho_i^k}{\sum_j \rho_i^k} \end{aligned} \tag{5.14}$$

Where $\mathbb{E}[1[U_k = j]]$ is a normalized value for the two types of users (at-risk or controlled). We can evaluate an optimal value for it iteratively by first initializing all factors, then updating each, in turn, using the expectations concerning the current values of the other factors [220].

5.4.3 Experimental Setup

For each task, each team could participate with five different runs. We create different variants of our proposed architecture. In this section, we will present all these variants, training procedures, and model hyperparameters.

Proposed Model Variants

All the proposed model variants for both tasks are based on two supervised learning phases (step 2 and step 3 in temporal mood variation model). For self-harm detection task (T2), as there is no training data, we train our models on the depression and anorexia datasets of eRisk-2018 [142]. We assumed that if a person with clear signs of

depression and/or anorexia could think about harming himself. We used the DMEM module as the first learning phase an all the variants and tried different machine learning and statistical methods as the second learning phase. Table 5.9 shows the used model for the second learning phase in all the runs for both tasks. MLP stands for Multi-Layer Perceptrons and RF is for Random Forest. All models that do not employ another learning phase are marked by dashes. In these runs, we used simple counting thresholds for successive positive classified writings.

	2^{nd} Learning Phase	
Model Name	T1	T2
LIRMM0	MLP	—
LIRMM1	RF	—
LIRMM2	—	MLP
LIRMM3	—	RF
LIRMM4	BVI	BVI

TABLE 5.9: Summary of the proposed model variants for eRisk-2019 tasks

Model Training and hyperparameters

We processed the training and testing streams of user writings by moving window concatenation of size (N). In other words, to give a decision about the current writing at time (t), we process all user writing starting from $(t - N + 1)$. This gives more information about the context of given writing and reduces the effect of noisy and irrelevant ones. Experiments show that $(N = 5)$ to be a reasonable choice for the window size. For DMEM, we used the same hyperparameter settings as for the chunk-based processing discussed in Section 5.3.3.

In the second learning phase, the used architecture of the MLP had two hidden layers with ten neurons each. Concerning the RF classifier, ten estimators were used. These models are used to classify time series of (N) points. For MLP, RF, and BVI models in T1, positive users were reported for those with classification probability higher than 0.8. This value increases to 0.9 in T2. We set both thresholds to 0.6 in the last rounds. For some model variants (LIRMM2 and LIRMM3 in T1 and LIRMM0 and LIRMM1 in T2), we apply counting of successive positive writings and give a decision after either 5 or 10 following writings respectively.

5.4.4 Results & Discussions

In eRisk-2019 two different types are used for model evaluation. The first one is decision-based evaluations; where the classical classification measures - precision (P), Recall (R), and (F1) - are computed for the positive (at-risk) user. In addition to these and due to the drawbacks of *ERDE* measure, a new latency weighted F1 measure is introduced [143]. The other complimentary evaluation is ranking-based. Besides the fired decision, scores are computed and used to build a ranking of users in decreasing estimation of risk. We participated only for decision-based evaluation. Tables 5.10 and 5.11 show the evaluation results of all our proposed variants for both tasks. Besides, we provide the results of the best participants' runs according to different metrics in

the first rows in each table. Using MLP for the second learning phase is the best choice for both tasks. However, the usage of a high threshold in T2 makes the models predict most of the positive users in late writings. Also, applying BVI gets more comparable results than the runs with simple counting of positive writings. But it needs a more precise choice of threshold for early detection in both tasks.

	P	R	F1	latency-weighted F1
CLaC [164]	0.64	**0.79**	**0.71**	**0.69**
INAOE-CIMAT [6]	0.67	0.68	0.68	0.63
UNSL [24]	0.42	0.78	0.55	0.55
LIRMM0	0.74	0.63	**0.68**	0.**63**
LIRMM1	**0.77**	0.60	0.68	0.62
LIRMM2	0.66	0.70	0.68	0.60
LIRMM3	0.74	0.42	0.54	0.48
LIRMM4	0.57	**0.75**	0.65	—

TABLE 5.10: Results of the proposed runs for eRisk-2019 anorexia task (T1)

	P	R	F1	latency-weighted F1
UNSL [24]	**0.71**	0.41	0.52	**0.52**
CAMH [143]	0.12	**1.0**	0.22	0.22
LIRMM0	**0.57**	0.29	0.39	**0.35**
LIRMM1	0.53	0.22	0.31	0.29
LIRMM2	0.48	**0.49**	**0.48**	—
LIRMM3	0.47	0.44	0.46	—
LIRMM4	0.52	0.41	0.46	—

TABLE 5.11: Results of the proposed runs for eRisk-2019 self-harm task (T2)

Tables 5.12 and 5.13 show some statistics of other participants runs compared to our proposed models. The ranks of the best run for each evaluation metric are also included. The statistics of the anorexia task are for 54 runs of 13 teams. The self-harm task statistics on results are for 33 runs of 8 teams. However the proposed architecture does not involve any hand-crafted features, it seems to be comparable with other contributions for both tasks. Also, combining anorexia and past eRisk depression training datasets for detecting signs of self-harm is very competitive.

	P	R	F1	latency-weighted F1
Max	0.77	0.99	0.71	0.69
Min	0.11	0.15	0.20	0.19
Average	0.45	0.63	0.48	0.46
Standard Deviation	0.17	0.24	0.17	0.15
Rank	1	14	5	5

TABLE 5.12: Statistics on 54 participating runs results and our ranks for eRisk-2019 anorexia task (T1)

	P	R	F1	latency-weighted F1
Max	0.71	1.00	0.52	0.52
Min	0.12	0.22	0.22	0.17
Average	0.29	0.73	0.32	0.29
Standard Deviation	0.18	0.29	0.11	0.10
Rank	3	17	3	4

TABLE 5.13: Statistics on 33 participating runs results and our ranks for eRisk-2019 self-harm task (T2)

5.4.5 Conclusions

In this section, we present our participation in the eRisk-2019 T1 and T2 tasks. Both tasks are for early detection of signs of anorexia and self-harm from users' posts on Reddit respectively. The datasets are released by item basis rather than the chunk-based release in the previous version of the competition. A new measure for penalty the late models has been used to overcome the limitaion of the ERDE metric. We propose to perform the classification through the two phases of supervised learning of TMV model using DMEM which utilizes modern transfer learning deep language modeling neural network. We tried different machine learning (MLP and RF) and statistical (BVI) models to temporal modeling step in TMV.

Furthermore, combining anorexia and previous eRisk depression datasets to detect early signs of self-harm (T2) is interesting and shows the correlation of such mental disorders. We proposed five different runs for each task and the results are interesting and comparable to other contributions. For both tasks, we ranked the second out of 13 teams according to the weighted-latency F1 - the new official measure used by the organizers. However, the proposed models need tuning of second learning phase classification thresholds for earlier risk detection.

However integrating NLP technologies alongside with understanding the nature of human mood variation over time was useful, we encourage more quantitative analysis of the variations associated with each mental disorder. In addition, incorporating a time-series forecasting model into the temporal modeling step in the TMV model could assist in earlier decisions for the models. On the other hand, the deep contextualized text vectorization model used in DMEM significantly improves the performance over the static shallow ones. This is consistent with the conclusions of recent Transformer-based models [60, 139, 260] that model size matters a lot. These models prove to

provide deeper language understating in different NLP tasks. It is interesting to adjust these models and test their behavior in the detection of other mental disorders including bi-polar disorder [19], schizophrenia [181], and suicide ideations [217].

Chapter 6

Mood II: Negatively Correlation Noisy Learners

"Fear does not prevent death, it prevents life."

Naguib Mahfouz

Contents

6.1 Introduction

Continuing the discussion of mental health problems raised in the previous chapter, suicide deems to be one of the serious problems in the global community. Suicide is a person's deliberate act of ending his/her own life. Suicide ideation reveals serious personal problems, but also often reflects a deterioration of social context in which an individual lives. The first alarming WHO (World Health Organisation) world suicide report highlighted the fact that one person dies of suicide every 40 seconds in the world - more than all the yearly victims of wars and natural disasters -, more than 1,100,000 per year [195]. The report points that there are as many as 90 attempts for every death with an approximate increase of 5% in the successful suicide attempts yearly [230]. Most suicide attempts are supported by hospital emergency units. It is considered as a major public health issue with strong socio-economic consequences [90]. However they may look similar, suicide and self-harm have two different types of risk profiles. Self-harm is the repetitive destruction or alteration of one's own body tissue but in the absence of intent to die or without suicidal aim.

Like other mental disorders, individuals suffering from suicidal thoughts may prefer to go and discuss their problems and feelings on social media platforms. They find there a sanctuary from the stigma, ignorance, prejudice, and fear. However, these have placed a responsibility on social media providers to promote a sense of community, provide social support, and ensure safety. Regularly, this is done by sequences of interactions between moderators and peer users through different moderation strategies. The process tends to be extremely complex and very sensitive to moderator experiences. Therefore, there is an increasing demand for tools and models for automatic detection of at-risk individuals that may need either help or moderation action [218]. These models should classify and assist the level of severity for at-risk users and perform this detection as early as possible.

Language could provide a natural eyepiece for the study and detection of such at-risk individuals through their writings on social media platforms. Some language indicators influence the discrimination of suicidal ideation risk from other risk factors including a higher rate of using violence and anger words [167], more references to death [176], and focus on the present tense [170]. Accordingly, the detection of suicidal thoughts and ideation is equally important with the previously discussed source of risks (depression, anorexia, and self-harm) in Section 5.1.

In this chapter, we address these problems by proposing a new model that introduces an ensemble method – Negatively Correlated Noisy Learners (NCNL). The model is designed to be back-bone independent and we examine it with the modern NLP and deep learning models. We obtained state-of-the-art results on five different tasks for the detection of at-risk individuals with clear signs of depression, anorexia, self-harm, and suicide ideations.

This chapter is organized as follows. Section 6.2 introduces the related work. Section 6.3 describes the datasets used and the relevant performance metrics. In section 6.4, we present the proposed models and their variants. The experimental setup including preprocessing and fine-tuning steps are introduced in Section 6.5. Then, Section 6.6 present the main results and discussions. Finally, we conclude the study and experiments in Section 6.7.

6.2 Related Works

6.2.1 Text Embedding

Text embedding has been revolutionized recently by the development of Neural Network Language Models (NNLM) [157]. The idea is to train a neural network in an unsupervised fashion for language modeling tasks and then extract the embeddings. Language Modeling (LM) which aims to predict the next word given a list of previous words or its context is a vital and important element in modern NLP applications. Not only because it tries to understand the long-term dependencies and hierarchical structure of the text, but for its open and free resources [100]. The model is pre-trained on vast amounts of textual data instead of training them on specific target datasets. There are two main types for NNLM-based embedding: static and contextualized. Static word embedding models are shallow neural network models that generate the same embedding for the same word in different contexts. It only leverages off the vectors for downstream tasks. Contextualized (Dynamic) words embedding capture word semantics in different contexts to address the issue of polysemous and the context-dependent nature of words. These models are deeper and output the pre-trained model not just embedding vectors. Contextualized embedding has been proven to achieve Computer Vision (CV)-like transfer learning for many NLP tasks [154]. Based on these developments, state-of-the-art results could be achieved by applying modern transfer learning methods like Universal Language Model Fine-tuning (ULMFiT) [100]. In 2017, a group of researchers in Google introduced the Transformer model [246]. This model is designed to handle ordered sequences of data, such as natural language, for various tasks such as machine translation. The model broke the domination of the recurrent deep models in NLP and became the basic building block of many state-of-the-art transfer learning models. Some of the revolutionary breakthrough models that make use of the Transformers are Bidirectional Encoder Representations from Transformers (BERT) proposed by Google AI [60], Robustly Optimized BERT Approach (RoBERTa) [139] proposed by Facebook AI and the generalized auto-regressive model (XLNet)[260] proposed by Google Brain.

In this context, the proposed models presented in this chapter are based on the state-of-the-art transfer learning NLP models by creating classifiers ensemble on top of each model for detecting at-risk individuals.

6.2.2 Negative Correlation Learning

Ensemble methods are one of the fundamental techniques in modern machine learning. The model is composed of a group of machine learning systems also called base learners. Each of them provides an estimate of the target output. These estimations are combined in some fashion to form the final decision of the overall model. Negative Correlation Learning (NCL) was introduced as a neural network ensemble technique [138]. It demonstrated significant performance improvements over a simple ensemble system, showing very competitive results with other techniques like mixtures of experts, bagging, and boosting [22]. It incorporates a measure of base learners diversity into the error function that should be back-propagated to the networks. Regression problems are one of the early applications that show empirical successes of applying NCL [23]. In a typical regression problems, let us assume that we have a number N of training samples $X = \{x_1, x_2, ..., x_N\}$ where x_i represents the high dimensional feature vector of the i^{th} sample. The target output $Y = \{y_1, y_2, ..., y_N\}$ is used to train a mapping function $G_\theta : x_i \rightarrow y_i$ parameterized by θ. The error loss of the mapping function can be approximated by:

$$e(G_\theta(X)) = L_\theta(X,Y) = \frac{1}{N}\sum_{i=1}^{N}(G_\theta(x_i) - y_i)^2 \tag{6.1}$$

For simplicity $G_\theta(X)$ can be written as G. The *bias-variance decomposition* theory [78] states that the mean square error of an estimator in a regression problem is equal to the biased square plus the variance.

$$\{(G-Y)^2\} = (\{G\}-Y)^2 + \{(G-\{G\}^2)\} \tag{6.2}$$

Considering the ensemble of M base regressors $\tilde{G} = \{G_1, G_2, ..., G_M\}$ where the ensemble output of the model $\bar{G}$ is the *arithmetic mean* of its individuals. This means that:

$$\bar{G} = \frac{1}{M}\sum_{m=1}^{M} G_m \tag{6.3}$$

The bias-variance decomposition, in this case, can be shown as:

$$\{(\bar{G}-Y)^2\} = (\{\bar{G}\}-Y)^2 + \{(\bar{G}-\{G\}^2)\} \tag{6.4}$$

In NCL, the error function of the individual base regressor becomes:

$$L_m(\tilde{G}) = \frac{1}{2}(G_m - Y) + \lambda P_m(\tilde{G}) \tag{6.5}$$

Where $P_m(\tilde{G})$ is a penalty term and defined as :

$$P_m(\tilde{G}) = (G_m - \bar{G})(\sum_{j\neq m}^{M}(G_j - \bar{G})) \tag{6.6}$$

This term balances the trade-off between those individual errors and the ensemble covariance [22]. This leads to a restatement of the NCL error function given the arithmetic mean way of averaging the model in equation (3) as:

$$L_{m,\lambda}(\tilde{G}) = \frac{1}{2}(G_m - Y)^2 - \lambda(G_m - \bar{G})^2 \tag{6.7}$$

It is clear that each base learner and regressor receives a lower error for moving its response closer to the target output and at the same time away from the final output of the ensemble. This trade-off is controlled by the strength parameter $0 \leqslant \lambda \leqslant 1$. Setting $\lambda = 0$ is exactly equivalent to independent training of base learners while increasing it will boost the ensemble diversity.

Aside from regression, NCL has been used for a wide range of applications in classification [48, 35] and time-series analysis and forecasting [11, 12]. After the significant evolutions of deep learning, NCL has been rediscovered again and used in many applications in image processing and computer vision [216, 172]. To the best of our knowledge, NCL has not yet been tried in any NLP application with deep learning models. Our intuition of using NCL learning for detecting at-risk individuals on social media is:

- The subjective nature of the problem may need to reinforce the diversity of the model.
- The great success of NCL deep models in similar subjective problems like personality detection and age estimation from images [265].

	Depression		Anorexia		Self-harm
	Training	Testing	Training	Testing	Testing
No. of Users (At-risk/Controlled)	886 (135/752)	820 (79/741)	472 (61/411)	815 (73/742)	340(41/299)
No. of Writings	531,394	544,447	253,341	570,510	170,698
Avg. No. of Writings/User	608.04	663.95	536.74	700.01	502.05
Avg. writing Size (words)	32.77	34.94	35.38	34.83	33.15
Vocabulary Size	234,181	222,201	117,090	210,763	105,448

TABLE 6.1: Statistics of eRisk-2018 (Depression) and eRisk-2019 (Anorexia and Self-harm) Datasets

6.3 Datasets

This section provides an overview of the datasets used in our experiments. It also provides the evaluation metrics for each classification task that would be used for comparison with previous state-of-the-art models.

6.3.1 eRisk Datasets

As a part of the Conference and Labs of the Evaluation Forum (CLEF), the eRisk workshop is organized to discuss and investigate the creation of reusable benchmarks for evaluating early risk detection algorithms relating to health and safety. The first pilot task in eRisk-2017 [141] was an exploratory task on early risk detection of depression using user posts on Reddit [1]. In eRisk-2018 [142], the extension of the study was planned to include detection of anorexia and the continuation of the depression task with new data. The main idea is to detect such problems from users posts as early as possible using the minimum amount of user writings (posts/comments). The release of eRisk-2018 depression datasets was on chunk-based style. This means that the training and testing datasets are divided into 10 chunks provided in chronological order. Each chunk contains 10% of the user's writings. The decisions on the users level have to be made after the processing of each chunk either to be positive (at-risk), negative (controlled) or to be postponed for future chunks. In eRisk-2019 [143], a continuation of the anorexia task with a new test set and a new task on early detection for signs of self-harm was proposed. The organizers decided not to release training data for self-harm and to move from chunk-based to item-by-item release of test data. In our experiments, we train and evaluate our models on the latest release of each dataset regarding each source of risk (depression, anorexia and self-harm). This means that we use the eRisk-2018 for depression and eRisk-2019 for anorexia and self-harm datasets. Table 6.1 summarizes the basic statistics for each of these datasets.

According to the evaluation metrics, the classical classification metrics – precision, recall and F1 – for positive users are used to measure the accuracy of predictive models. In addition, the Early Risk Detection Error ($ERDE$) defined in [140] was used for eRisk-2018 datasets. The $ERDE$ is an error measure that introduces a penalty for late correct decisions. The higher the number of user posts that had to be processed before the correct decision, the higher the penalty the model gets. The limitations of $ERDE$ discussed in Section 5.4.1 influenced the workshop organizers to look for alternative ways for evaluation. For eRisk-2019 tasks, a new evaluation

[1] Reddit is an open-source platform where community members (red-ditors) can submit content (posts, comments, or direct links), vote submissions, and the content entries are organized by areas of interests (subreddits).

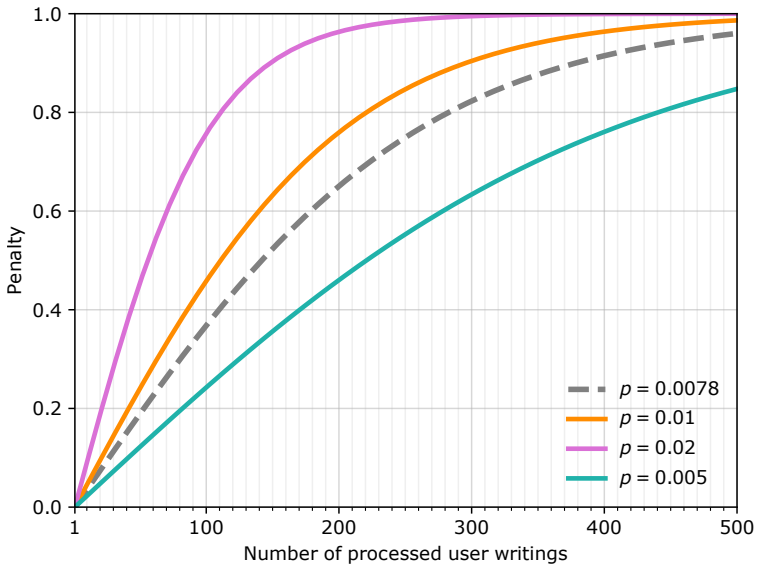


FIGURE 6.1: The penalty function changes with the value of p

measure ($F_{latency}$) proposed in [206] is used to measure the accuracy of early decisions. $F_{latency}$ combines the effectiveness of the decision ($F1$) and the latency only over the true positives. The following penalty function is computed for each true positive user u after processing k_u writings:

$$penalty(k_u) = -1 + \frac{2}{1 + \exp^{-p.(k_u - 1)}} \tag{6.8}$$

The value of p controls the speed of the penalty associated with true positives. Figure 6.1 shows that the lower p leads to a penalty function that increases slowly with the number of observed user writings. As recommended by the organizers, the value of p was set, such that the penalty equals 0.5 at the median number of posts. This value was first computed for eRisk-2017 [141] depression datasets and fixed to $p = 0.0078$ for all current and future experiments [143]. The overall model *speed* is computed for true positives as :

$$speed = 1 - median(penalty(k_u)) \qquad 0 < speed \leqslant 1 \tag{6.9}$$

The $F_{latency}$ is computed as a speed-weighting of F1 measure or simply $F_{latency} = F1.speed$.

6.3.2 University of Maryland Suicidality Dataset

The University of Maryland Suicidality Dataset (UMSD) is constructed using data from Reddit. The dataset originated from the 2015 *Full Reddit Submission Corpus*[2]. This data dump contains approximately 200 million submissions that have been collected using the Reddit API. The first version of UMSD (UMSD_V.1) described in [217] makes use of users posts in *r/SuicideWatch* subreddit to be annotated by experts and crowdsourcing. It defines four levels of categorization of risks defined in [43]. These levels are graded from *No*, *low*, *moderate* to *severe* risk. The dataset aims to promote research for identifying at-risk users that may ultimately help to prevent suicides. The dataset is considered to be the first demonstration of reliability in risk

[2] https://www.reddit.com/r/datasets/comments/3mg812/full_reddit_submission_corpus_now_available_2006/, last access on Mar. 15, 2020.

	Task-A		Task-B		Task-C	
	Training	Testing	Training	Testing	Training	Testing
No. of Users -Total-	496	125	993	125	993	249
- No Risk	127	32	127	32	127	32
- Low Risk	50	13	50	13	50	13
- Moderate Risk	113	28	113	28	113	28
- Severe Risk	206	52	206	52	206	52
No. of Writings	919	186	57,015	9,610	56,096	14,231
Avg. No. of Writings/User	1.85	1.49	57.42	76.88	56.49	75.39
Avg. Writing Size (words)	208.61	219.23	57.138	69.93	54.66	66.98
Vocabulary Size	18,317	6,506	226,140	66,873	222,025	65,386

TABLE 6.2: Statistics of University of Maryland Suicidality Dataset (UMSD_V.2)

assessment by clinicians based on social media postings. UMSD_V.1 has been updated for the shared tasks on predicting the degree of suicide risk from Reddit posts, run as part of the 2019 Computational Linguistics and Clinical Psychology Workshop (CLPsych-2019) held at the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL) [269]. These updates create the second version of the datasets (UMSD_V.2) by adding more de-identification steps, creating standard training/testing splits, performing data-cleaning steps to fix some encoding issues and filtering out some posts that contain Arabic content. The dataset annotated by experts is not used for the shared tasks in CLPsych-2019. In our experiments, we used the crowdsourcing training/testing splits in UMSD_V.2 as recommended by datasets owners to facilitate head-to-head comparisons of system performance.

Concerning the tasks, CLPsych-2019 organizers proposed three tasks for predicting individuals degree of suicide risk. The first task (*Task-A*), is a risk assessment task. It simulates the case where the user has reasonable evidence that he/she might need help. The dataset provided for this task contains only the users posts on *r/SuicideWatch* subreddit. The goal is to classify each user to the predefined four levels of risk. This task uses the smallest amount of data, only a few writings for each user. The Second task (*Task-B*) is similar to *task-A* with access to more user posts on other subreddits. This will enable the systems to understand the value of collecting more comprehensive information and its effect on the individual risk assessment and their mental state. The third task (*Task-C*) is about screening. The goal is to identify whether the individuals are at-risk, even if they have not explicitly participated in *r/SuicideWatch* subreddit. The categorization of the individuals is done only by accessing their writings on non-mental health-related subreddits. Table 6.2 shows a summary of the basic statistics that describe the datasets used for the three tasks. The training sets for *Task-B* and *Task-C* contain a controlled users group. These users have never posted to any mental-health-related subreddits. For both tasks, in testing sets, these users are considered as *No Risk* individuals.

Regarding the evaluation metrics, The official metric used in these shared tasks is the macro-averaged F1 score which treats all four classes as equally important to the overall system's performance. This helps avoid performance on a single class dominating the result in case of class imbalance. In addition, task organizers used two

other metrics *flagged-F1* and *urgent-F1*. These two metrics are derived from similar tasks [159] that used the same four-level classifications. These two metrics are practical in real-world use cases. *flagged-F1* measures the accuracy of the models in making binary decisions discriminating no risk with low, moderate and severe risks. A model with a good *flagged-F1* will save human efforts with processing no risk cases. *urgent-F1* measures the performance of the model in distinguishing between no and low risks with moderate and severe risks. A good model in identifying urgent cases (moderate and severe) will help with the selection process of cases that require immediate action or attention.

6.4 Negatively Correlated Noisy Learners (NCNL)

This section describes the proposed models used in our experiments. The main idea is to create negatively correlated base learners on top of state-of-the-art NLP deep learning models. In section 4.1, the proposed unity loss function is introduced. Different sources of noise are discussed in section 4.2. The main Model architecture is explained in section 4.3 followed by demonstrations of proposed model variations in section 4.4.

6.4.1 Unity Loss Function

As discussed in section 6.2.2, NCL creates an ensemble of base learners that train multiple models (base learners) taking into consideration the relationship between the individual error of each base learner and their interactions within the ensemble. The main idea is to regularize the correlations of these base learners. Traditional NCL models use the same loss function of equation (7) to update model parameters. This loss is applied to each of the base learners forming the ensemble. Let us assume that the output of the ensemble $\bar{G}$ has a constant value with respect to the output of each base learner G_m [21], i.e.

$$\frac{\partial \bar{G}}{\partial G_m} = 0 \tag{6.10}$$

Although this is a strong assumption, it permits us to derive a unique loss combining a classical error metric with an additional penalty term. In more details, using this assumption and the penalty term ($P_m(\tilde{G})$) defined in equation (6), the gradient of the individual base learner loss with respect to its output is:

$$\begin{aligned}\frac{\partial L_{m,\lambda}(\tilde{G})}{\partial G_M} &= (G_m - Y) + \lambda \sum_{j \neq m} (G_j - \bar{G}) \\ &= (G_m - Y) - \lambda(G_m - \bar{G}) \\ &= (1-\lambda)(G_m - Y) + \lambda(\bar{G} - Y)\end{aligned} \tag{6.11}$$

NCL extends the traditional neural networks backpropagation and gradient descent methods by adding the extra term of the form $(1-\lambda)(G_m - Y)$ to the weigh updates formula. In the proposed models, we use deep learning models that act like feature extractor layers followed by classification layers groups. This means that each base learner shares these lower layer feature extractors. We propose a unity loss function that incorporates all the atomic losses of each base learner into one loss function. This loss should exploit the explicit control of the base learner interactions to

encourage diversities. We define it as:

$$L_\lambda(\tilde{G}) = \ell_1(\bar{G}, Y) - \frac{\lambda}{M} \sum_{m=1}^{M} \ell_2(G_m, \bar{G}) \tag{6.12}$$

Because of the classification nature of the problem, we choose ℓ_1 to be the cross-entropy loss and ℓ_2 to be Kullback–Leibler (KL) divergence loss. In addition, we used the normalized geometric mean for averaging the ensemble outputs as recommended in [66]. This means that:

$$\bar{G}_\theta(x) = \prod_{m=1}^{M} G_m(x)^{\frac{1}{M}} \tag{6.13}$$

where x represents the input sequence representation which is the output of the features extractor part of the model (model encoder). The base learner output G_m is the probability-like output of the softmax.

6.4.2 Noise Sources

In training deep learning models, it is a challenging goal to train a model to perform well on unseen data, not just the training datasets. In other words, the goal is to enhance model generalization and reduce the effect of overfitting. However, the effect of introducing noise to deep models has never been systematically studied, it allows the model to generalize the observation of the training data that can be useful at test time [38]. One hypothesis is that relaxing model consistency by introducing noise, limits the memorization effect of deep neural networks [258].

In NCL, introducing different types of noises to the base learners will increase overall model diversity. We introduce two main types of noise to the proposed models. The first one uses different *dropout* [226] in classification layer groups. Simply, this is done by dropping the connection between some neurons which are chosen at random during the training phase. In the testing phase, all network activations are used. This shutting-down during training reduces the co-dependency amongst neurons. The dropout is defined by a value that represents the probability of each neuron to be off. In the proposed model, we used a different dropout for each base learner that form the ensemble.

The second source of noise is changing the input depth. This is similar to using *stochastic depth* [106] in computer vision. Since we use NLP models that are consisting of identical blocks (equivalent dimensionality) – (Transformer blocks) in BERT and RoBERTa and (Transformer-XL) in XLNet –, we can simply take the output from one layer and feed it directly to the base learners. This allows the models to learn from different abstraction levels of the input sequences.

6.4.3 Model Architectures

The recently proposed Transformer-based methods have been proven to outperforming the state-of-the-art NLP models on several tasks including language modelling and text classification [60, 139, 260]. Therefore, we proposed to use the feature extractor part of our models to be either BERT, RoBERTa or XLNet. Figure 6.2 shows our proposed models architectures used in our experiments. We compare two configurations; the single classifier and NCNL ensemble. For both configurations, the pre-trained and fine-tuned Transformers blocks encode the input sequence user writing. This represents each user writing in a multi-dimensional space that encode the representation of each word token within its context by each block. Suppose an input sequence of length N where each token is represented by the hidden size (H) of each individual

block. Each input sequence s_i is D dimensional such that $s_i \in \mathbb{R}^D, D = B \times N \times H$ where B denote the number of Transformer blocks. In a single classifier configuration, the sequence representation step takes the encoded sequence s_i and produces a one dimensional sentence (writing) level representation $x_i = \eta(s_i)$ where $x_i \in \mathbb{R}^H$. This is done by either applying block selection or combining multiple block representations by different pooling strategies according to the default configuration for each of the used models as discussed in section 6.4.4. The final decision for each input sequence is determined by passing x_i to the classification layer group G_θ. We used the cross-entropy as the loss function between the true labels Y and the predicted decision $G_\theta(x_i)$.

In NCNL configuration, the encoded input sequence s_i is passed to the noise generator that generates different sequence representations $\tilde{x_i}$ for the base learners G_m that form the ensemble $\tilde{G}$ such that $\tilde{x_i} = \{x_i^1, ..., x_i^M\}$ where $x_i^m = \eta^m(s_i)$ and η^m generate a noisy sequence representation by applying different dropout values and different Transformer block depths. Creating noisy inputs influences the development of noisy learners that leads to a more diversified ensemble. The output of each base learner $G_m = \{G_m^1, ..., G_m^R\}$ is computed by softmax which calculates the probabilities of each target class over all possible target classes.i.e.

$$G_m^j(x_i) = \frac{e^{\gamma_m^j(x_i)}}{\sum_{r=1}^{R} e^{\gamma_m^r(x_i)}} \tag{6.14}$$

Where G_m^j denotes $p(y = j|\gamma_m(x_i))$ which is the probability of an input sequence representation x_i to be classified as j by a given base learner (m). $\gamma_m(x_i) = \{\gamma_m^1(x_i), ...\gamma_m^R(x_i)\}$ stand for the output logits for the classification layer group for the same base-learner where R is the number of classes ($R = 2$ for eRisk datasets and $R = 4$ for UMSD tasks).

The overall ensemble output is calculated by taking the geometric mean of the individual base learners as given in equation (13). The unity loss function is used to calculate the loss for the overall model by incorporating the individual base learner losses and the final ensemble output.

In both configurations, the decision on the user level is done once a clear sign of risk is detected. The sign of risk in both datasets is completed by observing two consecutive user writings classified as risky. We relax this condition for UMSD Task-A dataset to be only one writing. This is because of the small number of writings for each user (average of 1.85) in this task.

6.4.4 Model Variations

In this section, we present in more detail the specifications of the main building blocks of the proposed models for single classifier and NCNL configurations. We used the pre-trained models for each of the Transformer-based feature extractor methods. The pre-training of these models make use of the bidirectional encoder blocks of the Transformer for certain language modelling tasks. We fine-tune these models based on the corresponding language modelling tasks using the target datasets used in our experiments.

BERT: Two main tasks are used for the pre-training of BERT - Masked Language Modeling (MLM) and Next Sentence Prediction (NSP) [60]. The model is pre-trained using 16 GB of textual data (3.3 billion words) from Google books corpus and Wikipedia. We fine-tune the language model for MLM task on the target datasets. The MLM task masks 15% of the total WordPiece tokens [256] at random and let the model predict them. We used the large version of BERT ($BERT_{LARGE}$)

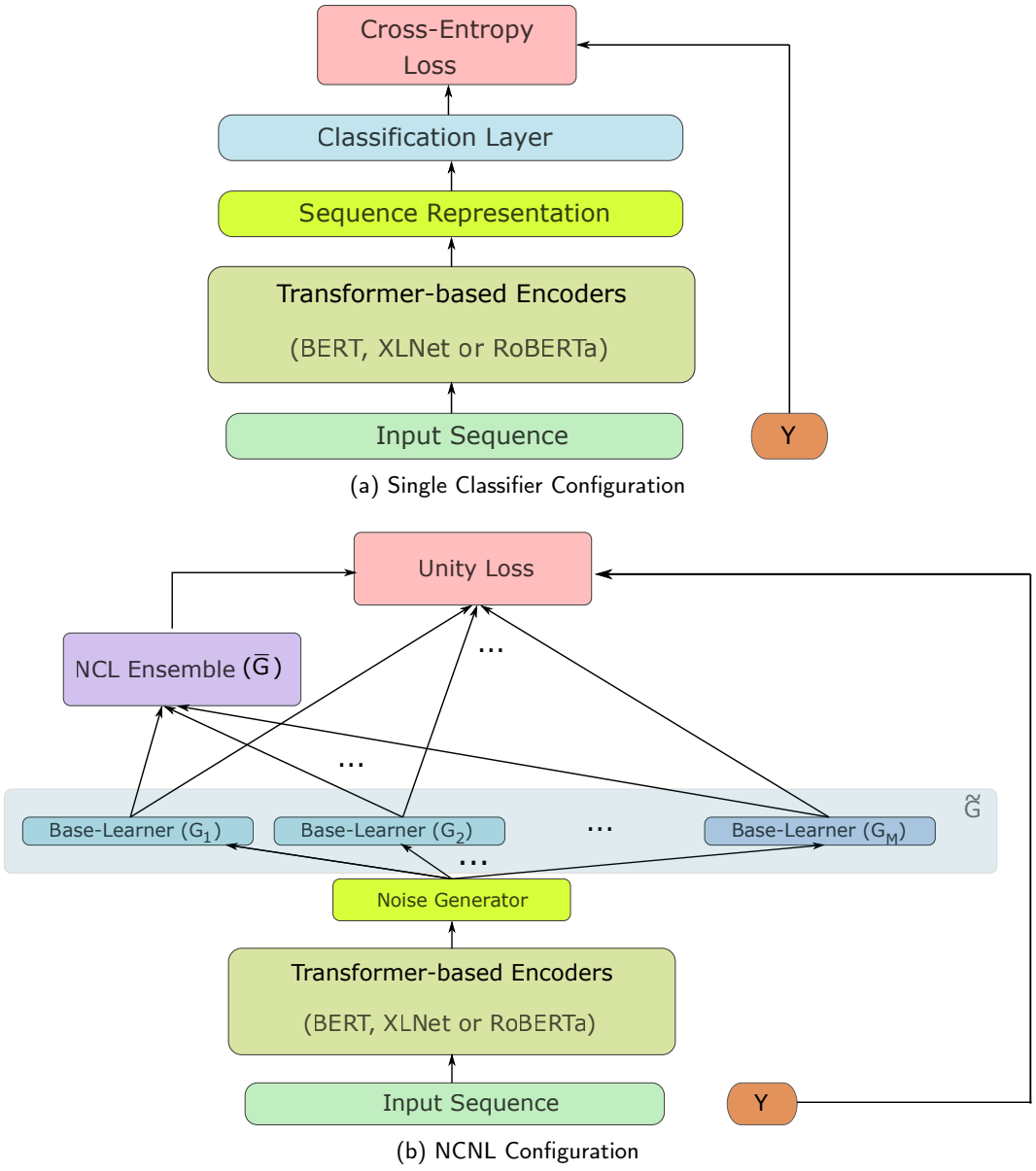


FIGURE 6.2: Proposed Model Architecture; Two configurations are used in the experiments. Single classifier configuration (a) where one classification layer group is used and Negatively Correlated Noisy Learners (NCNL) configuration (b) where a group of noisy base learners are used in NCL ensemble model

with a number of Transformer blocks (B = 24), hidden size (H = 1024) and maximum sequence length (N = 512). In single classifier configuration, BERT uses the first token representation of the last block and passes it to a fully connected linear (dense) layer with Hyperbolic Tangent (Tanh) activation function to get a sequence representation of the input writing. The classification is completed by another dense layer after applying dropout on the previous layer. In NCNL configuration, the sequence representation of each base learner is done by using the first token representation of either one of the last four blocks or a weighted sum of them at random. The output is passed to different dense layers for each base learner with Tanh activation to get the sequence level representation. The model uses different a dropout for each base learner (dense layer) before applying softmax and averaging to get the final ensemble output.

RoBERTa: This model is very similar to BERT in terms of architecture [139]. The model uses 160 GB of text for pre-training, including 16 GB of Books Corpus and English Wikipedia used in BERT. It removes the NSP task from BERT's pre-training and introduces dynamic masking so that the masked token changes during the training epochs. Rather than WordPiece tokens, RoBERTa uses Byte-Pair Encoding (BPE) based embedding [212]. In our experiments, we use $RoBERTa_{LARGE}$ with same model size as $BERT_{LARGE}$ (B = 24, H = 1024 and N = 512). In single classifier configuration, the sequence representation and classification steps are similar to BERT except that RoBERTa applies the dropout for both steps with the same value. In NCNL configuration, we apply the same NCNL as with BERT except that we use different dropout values for each base learner for each step to encourage diversity.

XLNet: The model generalizes the autoencoding language models used in BERT and RoBERTa where masked corrupted input text is used to reconstruct the original one. It uses autoregressive models by introducing permutation language modelling, where all tokens are predicted in random order [260]. The model is pre-trained on 113 GB of text (33 billion words) and fine-tuned on target used datasets. It uses SentencePiece [124] tokenization. The model uses Transformer-XL [47] as the base blocks rather than the original Transformer blocks used in BERT and RoBERTa. In our experiments, we use *XLNet-Large* model (B = 24, H = 1024 and N = 512). In single classifier configuration, unlike BERT and RoBERTa, the last token representation of the last Transforme-XL block is considered to represent the input sequence after passing it to a dense layer with Tanh activation. In NCNL configuration, we use the same base learners setup used in RoBERTa that uses different dropout values for sequence representation and classification.

6.5 Experimental Setup

In this section, we will describe the details of our experiments comparing the proposed Transformer based configurations of single classifier and NCNL. In all of our experiments, we used the Pytorch implementation of BERT, RoBERTa and XLNet[3] for fine-tuning the language models and downstream tasks training in both configurations of the proposed model. The models are trained and tested using two servers with a total capacity of six Nvidia GEFORCE GTX 1080 GPUs.

[3] https://github.com/huggingface/transformers, last access on Mar. 15, 2020.

6.5.1 Language Model Fine-Tuning

In our experiments, however, all the used Transformer based models are pre-trained on general-purpose datasets, fine-tuning the language models on in-domain (target) dataset can significantly boost its performance [100, 231]. We initialize the models with the pre-trained version and fine-tune them with the training datasets of all tasks. This will create a fine-tuned version of the language model for each task that acts as new pre-trained models for the classification downstream tasks. We used a clean version of user writings after removing all URLs, special tokens and characters. We remove short writings with less than three tokens. We primarily follow the same hyperparameter settings used in pre-training the original models and we point out the differences. We fine-tune the language models with mixed-precision floating point for better memory utilization and to speed up the training process.

With BERT, we initialize the model with the pre-trained $BERT_{LARGE}$ for fine-tuning on the MLM task. We apply a smaller learning rate of 5e-5 without warmup steps. We use a batch size of 1024 with a maximum sequence length of 512 (524,288 token/batch) trained for 10 epochs. For GPU memory considerations, we apply 128 gradient accumulation steps to simulate the large batch size. To speed up the pre-training, the default pre-training of BERT uses the full sequences for only the last 10% of the steps. As we are just fine-tuning the model, and the limited amount of text, we use the full sequence length over all training epochs.

Fine-tuning the RoBERTa model is similar to BERT with dynamic masking and a larger batch size. We use the same maximum sequence length of 512 over all training steps. We set the peak learning rate to 2e-4, without warming-up, with a batch size of 4K (≈ 2M token/batch) and trained for 5 epochs. We set the gradient accumulation steps to 500.

In XLNet, we fine-tune *XLNet-large* model with maximum sequence length of 512 and batch size of 4K (≈ 2M token/batch). We use a learning rate of 5e-6 for 5 epochs without warmup steps and 2000 gradient accumulation steps.

6.5.2 Downstream Task Training

The fine-tuned, pre-trained models are used for the downstream classification tasks for detecting individuals with different signs of risk. The classification layer weights in single classifier configurations and all the base learners weights in NCNL configuration are randomly initialized. We use the cleaned version of the datasets applying the same preprocessing steps used in language model fine-tuning. We apply the same hyperparameter settings in both configurations for all models. We use a batch size of 32 writings, learning rate of 2e-5 and train the model for 10 epochs. All other hyperparameters are fixed to the recommended values in the original papers. Because of the imbalanced nature of the problem that appears in the datasets, we use a weighted random sampler in creating the training batches. We set the weight for each class by its frequency in the imbalanced dataset without applying any sampling (oversampling or undersampling). We set each class weights (w_i) inverse proportionally to the class frequency ($|i|$) where $w_i = \frac{\sum_R |r|}{|i|}$.

For each task, except eRisk self-harm, we only use the corresponding training dataset without external sources or any data related to the other tasks. Since there is no available training dataset for eRisk self-harm task, we combined the training set for eRisk depression and anorexia and used them as the training set for detecting

signs of self-harm. In each proposed model, we process user writings sequentially in chronological order and flag at-risk individuals by detecting one or two consecutive risky writings (according to detection tasks). For eRisk tasks, we set the classification threshold for positive users to 0.8. For UMSD tasks, we apply a weighted version of the proposed unity loss function with the same weights used in creating training batches.

In NCNL configuration, the parameter λ controls the correlation between base learners and hence the diversity of the overall ensemble. We found setting $\lambda \in$ [1e-3, 1e-2] leads to good results for all models. We additionally found setting the number of base learners $M \in \{16, ..., 128\}$ obtains satisfactory results. In all our experiments, we report the results of NCNL models by setting λ = 5e-3 and M = 64.

6.6 Results Discussions

In this section, we show the main results on the different datasets compared to existing state-of-the-art reported results according to the official evaluation metrics mentioned in section 6.3. Besides, we discuss in more detail the ablation analysis of the default model and its variants. First, we apply different model sizes in the backbone Transformer-based model. Then, we show the effect of the two NCNL main parameters - diversity strength (λ) and the number of base learners (M). We conclude this section by measuring the dependency and diversity among the ensemble classifier members in different model configurations.

6.6.1 Results

Tables 6.3 and 6.4 show the results of the proposed models in both configurations of single classifier and NCNL ensemble. We report the average results for 3 different runs with random initialization seeds in downstream task training. We compare the obtained results with the existing state-of-the-art results for eRisk and UMSD_V.2 datasets and whole tasks.

For eRisk tasks, Table 6.3 shows the results of the F1 and $F_{latency}$ as the official metrics for models evaluation as discussed in section 6.3.1. The first three rows show the results of the best performing models for each task according to the result reported in [142] and [143]. No further improvements on these results have been reported after publishing the ground truth labels. The proposed models with both configurations attain good competitive results. NCNL ensemble elevates significantly the results of all Transformer based models and achieves new state-of-the-art F1 and $F_{latency}$ scores for all tasks. More precisely, RoBERTa-based models, especially with NCNL ensemble, get the best results for both evaluation metrics in depression and anorexia tasks. Also, merging anorexia and depression training sets to detect early signs of self-harm is interesting and reveals the correlation of such mental disorders.

In UMSD_V.2 results, the first four rows in Table 6.4 correspond to the best models in all the three tasks officially reported by task organizers in [269]. The next two rows show some improvement after releasing the ground truth labels. We report the results according to the official metrics (*macro*-F1, *flagged*-F1 and *urgent*-F1) described in section 6.3. Results of Task-C prove the challenging nature of the problem for detecting suicidal ideation risk level using only user writings in non-mental health-related topics. *macro*-F1 measures the performance of the models in whole classes of risks while the *flagged*-F1 and *urgent*-F1 do it by merging the sub-classes into binary super-classes fashion. This dissolves the misclassification between the sub-classes

Models	NCNL	Depression		Anorexia		Self-harm	
		F1	$F_{latency}$	F1	$F_{latency}$	F1	$F_{latency}$
FHDO-BCSGB [142]	-	0.64	0.52*	-	-	-	-
CLaC [143]	-	-	-	0.71	0.69	-	-
UNSL [143][142]	-	0.60	-	0.55	0.55	0.52	0.52
$BERT_{LARGE}$	✗	0.63	0.62	0.68	0.67	0.52	0.51
	✓	0.66	0.65	0.72	0.71	0.55	0.54
$RoBERTa_{LARGE}$	✗	0.65	0.63	0.72	0.70	0.54	0.53
	✓	**0.68**	**0.67**	**0.75**	**0.73**	0.56	**0.56**
XLNet-Large	✗	0.62	0.60	0.70	0.69	0.55	0.54
	✓	0.65	0.64	0.72	0.70	**0.57**	**0.56**

* *Reported $F_{latency}$ score on the chunk-based release of testing data*

TABLE 6.3: F1 and $F_{latency}$ Scores on the Testing Datasets of eRisk-2018 (Depression) and eRisk-2019 (Anorexia and Self-harm) Tasks

within the same super-class. This way of class merging explains the results of some models with significant lower *macro*-F1 compared to *flagged*-F1 and *urgent*-F1. None of either the proposed or previous best performing models dominates and reports best results for all tasks. However, NCNL ensembles gain significant improvement over single classifier configuration and achieve new state-of-the-art results for Task-A and Task-C in all evaluation metrics.

However, the UMSD_V.2 datasets are timestamped, all evaluation metrics do not incorporate a penalty for late decisions. We propose *macro*-F1 Latency (*m*-$F_{latency}$) an update of the $F_{latency}$ measure used for models evaluation in eRisk tasks. We apply the same penalty function in equation 6.8 and compute the *speed* of the model using the median penalty of all at-risk users (all risk level classes except "No Risk"). The final value is determined by weighting the *macro*-F1 value by *speed* such that: *m*-$F_{latency}$ =*macro*-F1.*speed*. Tabel 6.5 reports the *m*-$F_{latency}$ for all the proposed models in both configurations. NCNL models show improvements over the single classification configuration and $RoBERTa_{LARGE}$ gives the most accurate and fastest decisions in the three tasks.

6.6.2 Effect of Model Size

Each Transformer-based model comes in two main versions - *LARGE* and *BASE*. However, the best results come from the *LARGE* models, the *BASE* models are used for fair comparisons with similar model sizes. The NCNL model is backbone-independent and could be applied to different model types and sizes. As a case study, we try the *BASE* models of BERT, RoBERTa and XLNet with both configurations on the eRisk depression task. The results of F1 and $F_{latency}$ scores on the testing set are shown in Table 6.6.

Each of these models are initialized with the pre-trained model and fined-tuned with the corresponding language modeling tasks and hyperparameters for each model as discussed in Section 6.5.1. The *BASE* models are almost half the size of the *LARGE* ones, with number of Transformer blocks (B = 12), number of hidden sizes (H = 768) and the same sequence length (N = 512). Rather than using the last four blocks for

Models	NCNL	Task-A			Task-B			Task-C		
		macro-F1	*flagged*-F1	*urgent*-F1	*macro*-F1	*flagged*-F1	*urgent*-F1	*macro*-F1	*flagged*-F1	*urgent*-F1
CLaC [269]	-	0.48	0.92	0.78	0.34	0.84	0.72	0.27	0.67	0.63
SBU-HLAB [269]	-	0.46	0.84	0.84	0.46	0.82	**0.82**	0.18	0.59	0.55
CAMH [269]	-	0.44	0.90	0.78	0.41	**0.91**	0.81	0.23	0.67	0.60
Affective Computing [269]	-	0.38	0.92	0.86	-	-	-	-	-	-
CNN-RNN Ens-feat. [163]	-	0.53	0.92	0.84	0.38	0.82	0.73	0.24	0.67	0.61
DualContextBert [151]	-	0.50	-	-	**0.50**	-	-	-	-	-
$BERT_{LARGE}$	✗	0.50	0.89	0.85	0.41	0.80	0.81	0.29	0.69	0.68
	✓	0.53	0.93	0.84	0.44	0.82	0.79	0.31	**0.82**	**0.75**
$RoBERTa_{LARGE}$	✗	0.53	0.94	0.85	0.43	0.90	0.81	0.30	0.76	0.67
	✓	**0.56**	**0.95**	0.85	0.45	0.86	0.80	**0.34**	0.78	0.69
XLNet-Large	✗	0.49	0.94	0.85	0.41	0.84	0.80	0.29	0.71	0.70
	✓	0.54	**0.95**	**0.88**	0.44	0.90	**0.82**	0.33	0.78	0.71

TABLE 6.4: *macro*-F1, *flagged*-F1 and *urgent*-F1 on the Testing Datasets of UMSD_V.2 Tasks

Models	NCNL	Task-A	Task-B	Task-C
		$m\text{-}F_{latency}$	$m\text{-}F_{latency}$	$m\text{-}F_{latency}$
$BERT_{LARGE}$	✗	0.50	0.39	0.27
	✓	0.52	0.43	0.30
$RoBERTa_{LARGE}$	✗	0.53	0.42	0.28
	✓	**0.55**	**0.44**	**0.31**
XLNet-Large	✗	0.48	0.39	0.26
	✓	0.53	0.41	**0.31**

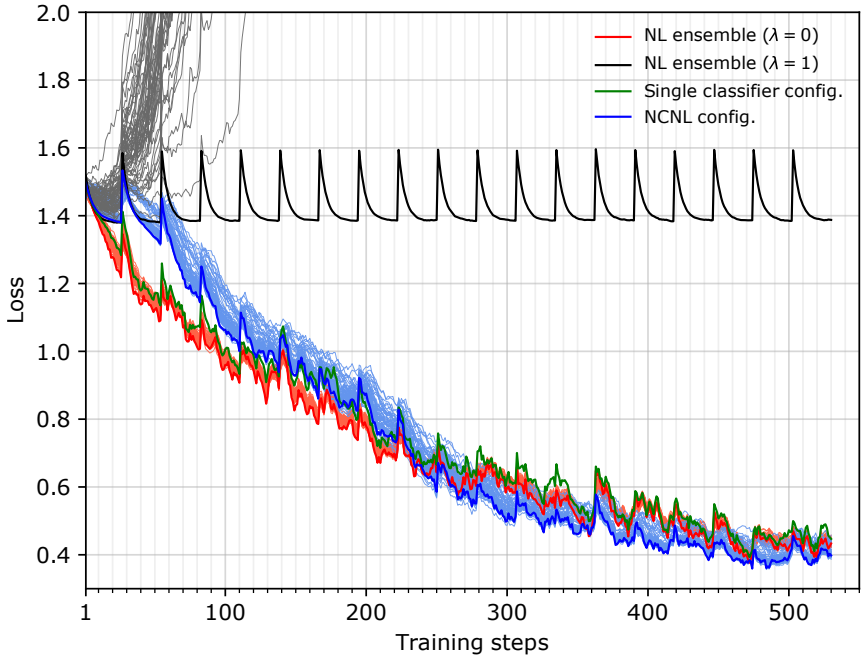
TABLE 6.5: *macro*-F1 Latency ($m\text{-}F_{latency}$) on the Testing Datasets of UMSDV.2 Tasks

creating the noise base learners in *LARGE* models, we use the last two blocks in the *BASE* ones. Additionally, we fix the number of base learners ($M = 32$). However, the *LARGE*-based models show better performance, NCNL configuration confirms the significant improvements over single classifier models for user-level results.

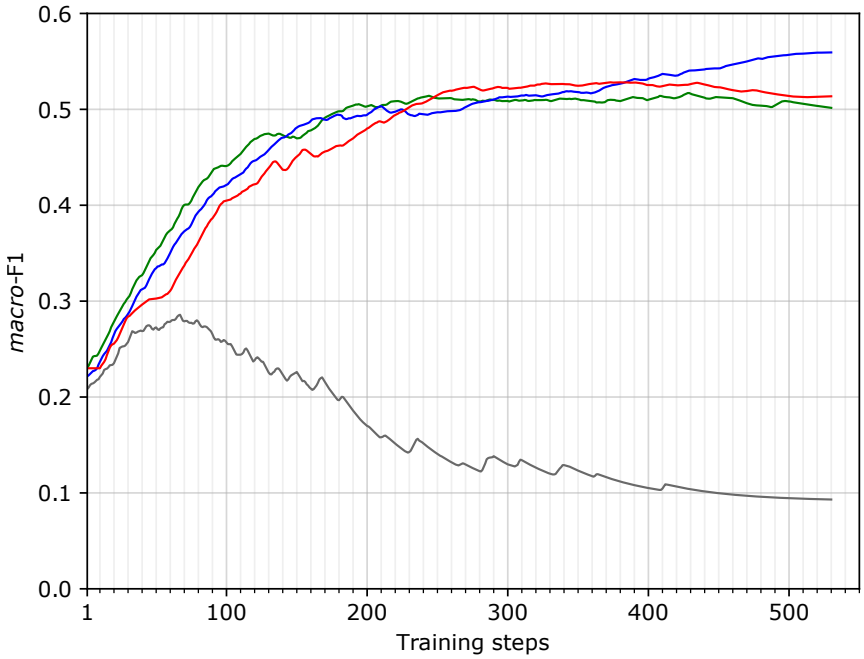
6.6.3 Effect of λ and M

The diversity strength parameter λ controls the relationship between each base learners' performance and the overall ensemble. The objective unity loss function – Equation 6.12 – considers the loss of the overall ensemble, while simultaneously maximizing the diversity of each base learner. Figure 6.3 compares the performance of $Roberta_{LARGE}$ model for UMSD_V.2 Task-A using Noisy Learner (NL) ensemble with the two extreme values of $\lambda \in \{0, 1\}$, the NCNL (λ = 5e-3) and the single classifier configuration. The individual base learners' losses (shadow lines in Figure 6.3 (a)) grows exponentially by setting λ=1 and leads to a bad ensemble. Setting λ=0 leads to very similar base learners which is clear by the intensive view of the individual losses. The NCNL configuration compromises this trade-off by setting a small value of λ which leads to more dispersed learners' losses and more accurate diverse ensemble.

Traditionally, ensemble size has a great impact on the accuracy of prediction [21] [22]. In the same vein, determining the number of base learners (M) in NCNL models



(a) Training loss



(b) Validation *macro*-F1

FIGURE 6.3: Training cross-entropy loss (a) and validation *macro*-F1 (b) by applying $RoBERTa_{LARGE}$ model on UMSD_V.2 Task-A with different model variants (Noisy Learner (NL) ensemble with $\lambda = 0$ (—), NL ensemble with $\lambda = 1$ (—), single classifier configuration (—) and NCNL configuration (—))

Models	NCNL	F1	$F_{latency}$
$BERT_{BASE}$	✗	0.63	0.60
	✓	0.65	0.63
$RoBERTa_{BASE}$	✗	0.64	0.62
	✓	**0.66**	**0.65**
XLNet-Base	✗	0.61	0.59
	✓	0.65	0.62

TABLE 6.6: F1 and $F_{latency}$ Scores on eRisk-2018 (Depression) Testing Set Using *Base* Transformer Models

$M \rightarrow$	4	16	32	64	128
macro-F1	0.29	0.30	0.32	0.33	0.33
flagged-F1	0.70	0.75	0.71	0.78	0.78
urgent-F1	0.68	0.67	0.75	0.71	0.72
m-$F_{latency}$	0.27	0.29	0.30	0.31	0.32

TABLE 6.7: Results of UMSDV.2 Task-C Testing Set Using NCNL *XLNet-Large* Models with Different M

is an essential decision. Through our experiments, we observe performance improvements by increasing M. This improvement saturates with large values of M. The main reason is that enlarging the ensemble size will lead to increasing the probability of getting very similar or even identical base learners. This is clear by inspecting the effect of the weighting value of $\frac{\lambda}{M}$ in Equation 6.12. Experimental results on UMSD_V.2 Task-C applying NCNL *XLNet-Large* model demonstrate this effect with different ensemble sizes presented in Table 6.7.

Diversity Measures	$\uparrow / \downarrow^*$	Classical Ens.	NL Ens. ($\lambda = 0$)	NCNL	NL Ens. ($\lambda = 1$)
ρ	↓	0.947	0.885	0.117	0.025
Q	↓	0.998	0.988	0.123	-0.468
κ_p	↓	0.011	0.006	-0.391	-0.554
$\ominus$	↓	0.237	0.221	0.049	0.108
KW	↑	0.013	0.029	0.201	0.192
F1	-	0.72	0.73	**0.75**	0.32
$F_{latency}$	-	0.71	0.71	**0.73**	0.30

* *Higher diversity if the measure is lower (↓)/ higher (↑)*

TABLE 6.8: Summary of Diversity Measures Applying Different $RoBERTa_{LARGE}$ Ensemble Models on eRisk Anorexia Task

6.6.4 Ensemble Diversity Measures

In this section, we present some empirical evidence in order to enhance the relationship between the ensemble diversity and its impact on performance. We show how NCNL models enhance diversity among individual base learners. We compare NCNL ensemble with the classical conventional ensemble and the two extreme NL ensemble setting $\lambda \in \{0, 1\}$. However, there is no generally accepted measure to assess ensemble diversity, a variety of measures exist in the literature [125]. In our study, we examine five diversity measures:

1. The correlation coefficient (ρ) [222] measures the correlation between two individual base learners and are averaged across the over-whole ensemble.

2. The Q-statistics (Q) [264] measures the association coefficient of two base learners outputs. The value of $Q \in [-1, 1]$ determines if the base learners' outputs are statistically independent ($Q = 0$) or tend to be either positively or negatively associated.

3. The Fliss' kappa (κ_p) [75] which determines the inter-rate reliability in ensemble classification members. This measure assesses the degree of classification correspondence unlike that which would be expected randomly.

4. The Kohavi-Wolpert variance (KW) [120] gives an expression of the variability of different predicted class labels over different base learners. Rather than averaging across all base learners, KW is averaged across all testing set labels. The higher KW value the better ensemble diversity.

5. The measure of difficulty ($\ominus$) [89] defines diverse ensemble to have testing data points that are difficult for some base learners and easy for others.

As a case study, we apply different ensemble models on the eRisk anorexia task using the $RoBERTa_{LARGE}$ backbone model. Table 6.8 reports the value of the five used measures for different ensembles and their performances. NL ensemble ($\lambda = 0$) improves the classical ensemble diversity by learning from different input while, NCNL boost ensemble diversity and the overall model performance.

In this regard, we verify these findings by computing the contingency probability matrix [125] for each model. It is a diagonal square matrix which reports the pairwise probability that the row base learner (i) makes the correct prediction, given that the column base learner (j) also predicts correctly ($p(G_i = Y|G_j = Y)$). Figure 6.4 shows a visualization of these matrices. We can easily recognize the diversity and variation of the different base learners in NCNL meanwhile, improving the performance reported in Table 6.8.

6.7 Conclusions

In this chapter, we studied the problems of detecting at-risk social media users. We focused on four sources of risks – depression, anorexia, self-harm and suicide. We considered the early detection of such mental disorders through the processing of user writings on social media. The proposed models introduce Negative Correlation Learning (NCL) in Natural Language Processing (NLP) with deep learning backbone models. The model improves the generalization capability by training a group of noisy base learners aimed at boosting ensemble diversity. However, the proposed Negative

Correlation Noisy Learners (NCNL) model is general and independent of the underlying systems, we used cutting-edge Transformer-based backbone models.

In comparison with the best performing reported models so far, the proposed models achieve new state-of-the-art results for five, out of six, different tasks for detecting depression, anorexia, self-harm, and suicide. Empirically, NCNL significantly outperforms the classical ensemble and single classifier models and shows to enhance model diversity.

All the datasets considered in this study are user writings to Reddit. However, we encourage testing the proposed architecture for shorter writings such as user tweets on Twitter. In addition, the model is tested for English datasets. However, NCNL could be applied to the multi-lingual pretrained Transformer-based models. Due to the lake of similar resources in other languages, incorporating data augmentation and neural translation models [259] into the training process may result in a multilingual at-risk detection model.

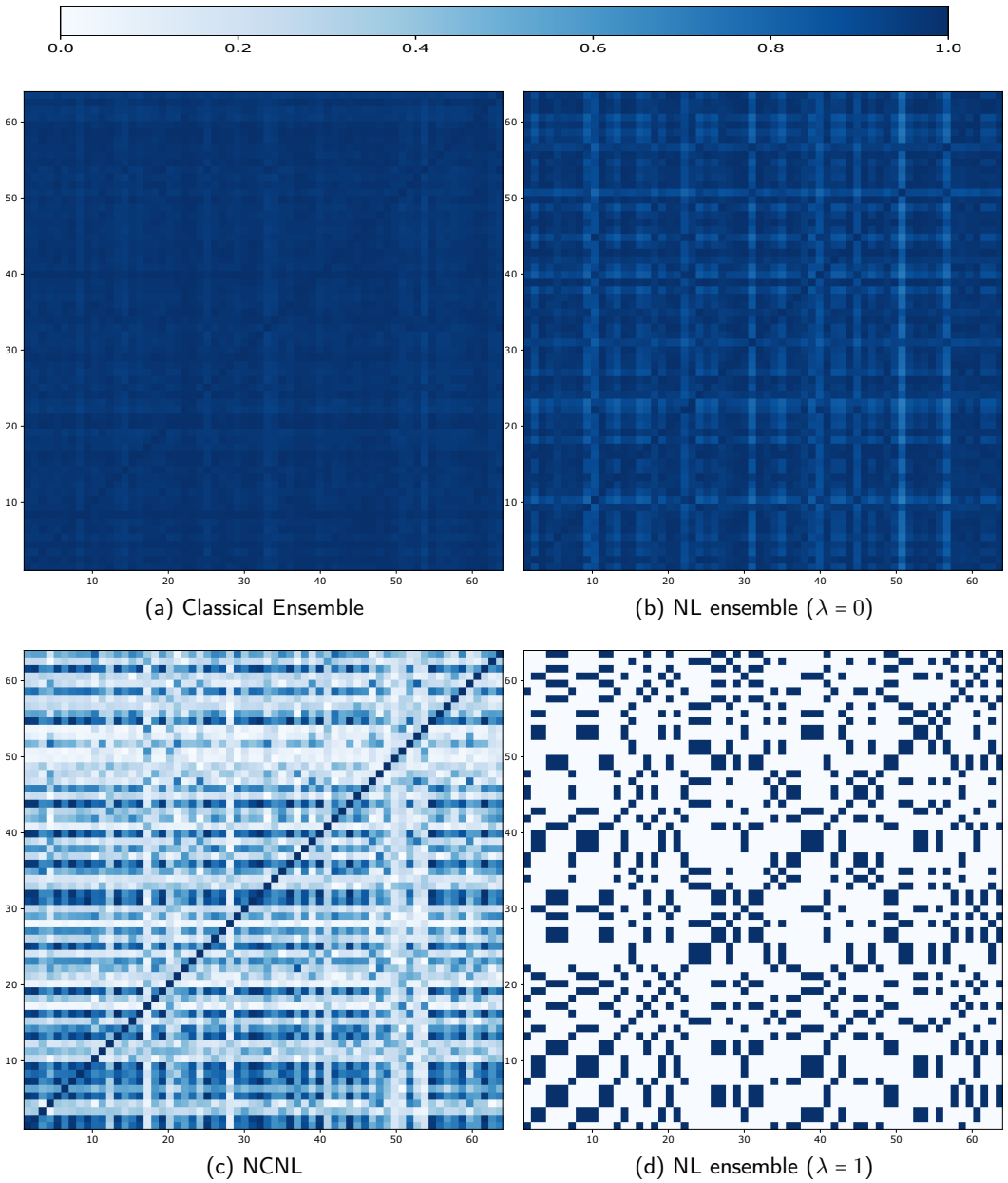


FIGURE 6.4: Visualization of the contingency probability matrices applying different $RoBERTa_{LARGE}$ ensemble models on eRisk anorexia task

Chapter 7

Conclusions and Future Work

> You can tell whether a man is clever by his answers. You can tell a man is wise by his questions.
>
> Naguib Mahfouz

Contents

7.1 Summary and Conclusions

In this thesis, we consider modeling human effect in social media text. We address the problem of detection and classification of the affective states of three of the most common affect-related subjectivity terms - sentiment, emotion, and the mood. The thesis provides practical and applicable solutions to the research questions raised in Chapter 1. Affective Computing (AC) is a strongly interdisciplinary research area encompassing psychologists, psychiatrists, and computer scientists. Therefore, feature engineering processes in many NLP models for AC involve many types of hand-crafted and lexical features. As discussed in Chapter 2, this engineering process is tedious, expensive, and almost requires domain experts. The contributions presented in the thesis do not involve any specific feature engineering processes for representing input text. Instead, We utilized and compared the classical static text vectorization methods with the contextual-deep models for word embedding. Besides, We entrusted to the attention mechanism for the self-guidance to focus on important parts of the text that influence model decisions. The proposed models in the thesis incorporate attention layer(s) in different configurations. Experiments and ablation analysis show improvements in the transfer learning capabilities and robustness Of the attentive versions of the corresponding models. We introduced different configuration of using self-attention layers.

In Chapter 3, we introduced the multi-level self-attention layers to the ULMFiT architecture for sentiment classification of user reviews. The model is trained on large scale sentiment classification datasets. This helps the attention layers to emphasize the significant segments in input sequences. We used three levels of attention in a recurrent deep architecture to model different abstraction levels of the input text. Further to the model accuracy, we evaluated model's explainability and user perceptions of the resulting attention scores. The results confirmed the advantages of proposing self-attention to the model. In Chapter 4, we propose self-attention to specific turns in textual conversations for tracking emotional states. Furthermore, we assessed the relevant words that acquire the attention of the model with an emotional lexicon (EmoLex). The attention layers were able to focus on the words that correspond to the considered emotional states. Moreover, chapter 5 proposed the Deep Mood Evaluation Module (DMEM) that applies a self-attention layer on the user writings through the multi-stages learning framework of the Temporal Mood Variation (TMV) architecture. However, in the models proposed in Chapter 6, we did not employ any explicit attention mechanisms, the Transformer-based models internally stacked different layers of multi-head attention. These enable the model to capture the relation between words more deeply than just using single attention mechanism[1].

On the other hand, as discussed in Chapter 1, bearing in mind the characteristics differences of the affect-related subjectivity terms improves the efficiency of the detection models. The core term of affect refers to an abstract concept that is perceived as the umbrella that covers the basic sense of feelings. The affect has a non-conscious behavior, making it difficult to be realized and detected. In this thesis, we focus on the sentiment, emotion, and mood as conscious experience for expressing the affect. The three affective concepts have different affective characteristics that we tried to consider them.

[1] We refer to [40] for further demonstration and analysis on the attention mechanisms applied in BERT as an example.

Regarding the sentiment, it is fairly enduring disposition as a response to affective states to a specified entity [46]. It is one of the most stable affective concepts. In chapter 3, We addressed the problem of sentiment analysis and proposed a deep learning model that applies transfer learning and multi-levels of self-attention layers to focus on the most important parts of the text that have a high influence on sentiments. The model is evaluated on several datasets and shows very competitive results. Furthermore, we evaluate the impact of attention mechanisms on the model's interpretability and user perceptions.

Concerning the emotion, it is a direct expression of affect and/or feelings. The emotion is considered as a brief (short-duration) and strong feeling as a response to a significant and immediate event [69]. In chapter 4, we tackle the problem of detection and classification of basic emotions in textual dialogues. We extend the basic model used for sentiment classification to model textual conversations and track the emotion over multiple turns. The model pays close attention to the instantaneous deflection of the last turns in the conversations that have been written by the same individual. We participate in the SemEval-2019 shared task on contextual emotion detection in text. The model shows very competitive results and ranked 9^{th} out of more than 150 participants.

Likewise the emotion, the mood as well is an affect/feeling reflection. However, the mood is not as intense as emotions and can have a less specific, immediate, or obvious cause [76]. It lasts significantly longer than emotion. However user mood can be classified into two main types - positive and negative mood, mood disturbances inflict various mental illnesses/disorders. In chapter 5, we consider the problem of early detection of depression, anorexia, and self-harm using users' writings on Reddit. Since the time factor plays a vital role in characterizing the mood, we proposed a new multi-stage architecture that models users' temporal mood variations (TMV). Two main learning phases were proposed. The first one builds a time series representing the mood swings and variation. The second learning phase is a classification model that learns patterns from the time series to detect early signs of such mental disorders. We participated in eRisk-2018 and eRisk-2019 tasks. The proposed models perform comparably to other contributions and ranked the 2^{nd} out of 13 teams in eRisk-2019.

In chapter 6, we reinforced the study of the mood consequences and include the problem of suicide thoughts detection. Therefore, we propose the Negative Correlation Noisy Learners (NCNL) as a novel backbone-independent model that uses state-of-the-art Transformer-based models through Negative Correlation Learning (NCL) configuration. The proposed model simulates the subjectivity in the detection problem and creates an ensemble of noisy weak learners that are boosted to be correct and different at the same time through training. We evaluate the model on different tasks for at-risk users detection. The models achieve significant improvements over the existing state-of-the-art results reported for five out of six tasks for the different risk sources.

In summary, considering language representation including the semantic and syntactic analysis is important for understanding and modeling human affect in NLP. However, the findings of this study point out that further consideration should be given to understanding the psychological differences and characteristics behind the expression of affect. The proposed models and presented results in this thesis highlighted two main concluding observations. The first one is regarding language representation. Our results provide confirmatory evidence that using modern contextualized word

embedding outperforms the classical shallow models. Besides that, transfer learning is highly affected by the model size and the data sources used for pretraining. Furthermore, attention mechanisms can provide proper guidance to the model and interpretation of the results. The second main observation is that it is beneficial to involve the affective properties in designing the predictive model.

Of course, there are some possible limitations in this study and much work remains to be done:

- In Chapter 4, we tried our model only on the Emocontext dataset [32] provided by the SemEval-2019 shared task on contextual emotion detection in text. The dataset is three turns conversation and the goal is to detect the emotion of the last turn. However, further experiments should investigate the performance of the model on other datasets that contains longer and multi-parties conversations -e.g. IEMOCAP [25] and DailyDialog [134]. Furthermore, we are interested to test and analyze the proposed instantaneous deflection modeling but in other modalities, for example, facial expression and voice tone changes.

- The TMV architecture presented in Chapter 5 transforms the detection of at-risk individuals to a time series classification problem. However, further time series analysis is recommended to validate the effect of mood instability on the mental health state of the individuals. Besides, incorporating a time-series forecasting model into the temporal modeling step in the TMV model could assist in earlier decisions for the predictive models.

- Concerning the DMEM that proposed to encapsulate the text vectorization and mood evaluation steps in the TMV architecture, it deserves further consideration to try the performance of the model with the Transformer-based models used in Chapter 6.

- It is difficult to have unified feature sets that can detect different sources of mental health risks on online forums [79]. However, as discussed in Section 5.4, combing the depression and anorexia datasets gives promising results in the detection of self-harm. This might indicate the implication of the correlation of such mental disorders. In this context, research questions could be raised regarding the possibility to have a general mental health well-being score.

- However, we extended the study of at-risk user detection to include suicidal ideation and thoughts as a risk factor in Chapter 6, it will be important to examine the proposed models with other problems and sources of risks. These could be worth inspecting even if it is not directly related to the mood (e.g. hate speech, abusive language, and cyberbullying)

- The work done in the thesis could be extended to other affective concepts and subjectivity terms, for example, opinions, personality traits, and temperaments.

7.2 Future Directions

In this section, we will present possible extensions and provide an outlook into the future directions to build upon this work.

7.2.1 Cross-lingual Data Augmentation

In modern machine/deep learning, the amount of available training data is the key factor that impacts model performance. Due to the digital language divide[2] exists on the internet, we want to ensure that non-English language speakers are not left behind. Cross-lingual text presentation can be considered as a transfer learning and domain adaptation tasks where different domains are corresponding to different languages. With the recent advances in unsupervised machine translation models [126, 7], there is a possibility to just translate the datasets from the source to the target languages. However this technique provides a strong baseline especially for low-resource languages [126], translation models may not be easily available and can be expensive to train in many languages. Moreover, naive translation of training data over languages do not fit for all tasks [137, 3] and struggle with topologically different language pairs and domain mismatches [87].

On the other hand, similarly to the success of moving from shallow word embedding to deep language models, deep cross-lingual embedding models are deemed to be a fruitful research direction. These models go beyond learning from sentence-level parallel data and incorporate non-parallel monolingual corpora. The most common example is the multilingual BERT (mBERT) [60] that is jointly trained on corpora contains 104 languages with a shared vocabulary of 110k subword tokens. The model train in a fully unsupervised manner. Along similar lines, the multilingual model XLM-RoBERTa (XLM-R) [41] jointly trained on 2.5 TB of data in 100 languages obtains state-of-the-art performance on cross-lingual classification, sequence labeling, and question answering.

The machine translation techniques and the cross-lingual embeddings are not odds and may be complementary to each other. With the lake of labeled data in many languages especially for affect-related tasks, data augmentation based on machine-translation models could be used to train deep cross-lingual models for multiple downstream tasks. The idea is similar to the recent work of Unsupervised Data Augmentation (UDA) [259] which utilizes back-translation methods for data augmentation. It refers to the procedure of translating existing examples in one language to another and then translating it back into the original one which obtains augmented diverse paraphrases of each example. UDA show to combine well with the English Transformer-based pretraining model and achieve competitive performances by training on a very small portion of the datasets (tens of examples). This work could be generalized for a multilingual setting through training unsupervised deep cross-lingual models weakly supervised by a machine translation model. Further research in this point could lead to cross-lingual models to detect affect states for different affective concepts.

7.2.2 Self-training and Continual Learning

Compared to the large amounts of textual data available, there is an obvious shortage of quality annotated data. The problem is very clear with affect-related tasks more specifically, the emotional and mental health (mood consequences) classification datasets. In machine learning, the lack of labeled data is well studied and one of the most promising solutions is to maximize the use of unlabeled data by using semi-supervised learning approaches. Self-training [152] is one of the proposed techniques

[2]Top ten languages of the internet world users, https://www.internetworldstats.com/stats7.htm

in which it leverages the model's own predictions (pseudo-labels) on unlabeled data to augment additional data points that can be used during training. Typically, the most confident predictions are taken and fed-back as additional training data for the model. The process is iterative and continues to maximize the model predictions on separate testing/validation sets. Selecting confident predictions in neural network models is not straightforward since it is known that output probabilities in neural networks are not calibrated [85]. consequently, modern self-training techniques rely on using model ensembles for more accurate selection and prediction. Some common examples are the co-training [17], tri-training [268], tri-training with disagreement [224], and multi-task tri-training [205]. These models show great success in different NLP tasks including conversation summarization [210], sentiment analysis [93], named entity recognition [39], and many other tasks. The main downside in self-training is that the model loses the ability to take corrective actions if the error accumulatively amplified.

Self-training models could be benefited from the self-reported annotation (self-diagnosis) [117, 4] datasets. These datasets are not manually labeled by experts rather, it imposes the labels based on clear markers in the text like twitter hashtags, Emoji in messages, phrases like "I have been diagnosed with ... " or "I feel ...". Self-annotating datasets may provide the self-training with more accurate and confident labels that could be used through the semi-supervised processes. The use of this massive amount of unlabeled or quasi-labeled data could be incorporated in an everlasting learning process that accommodates new knowledge and preserves previously learned ones which is referred to as continual learning or lifelong training [180]. This continual learning framework could be investigated by applying iterative and recursive use of unlabeled data to provide the model with the ability to learn online from a non-stationary and never-ending learning process from streams of data.

7.2.3 Integration with Active Learning Pipeline

As discussed, data labeling is considered to be the bottleneck in machine learning. Therefore, various learning approaches aim to maximize the use of unlabeled data that is freely available. Since training instances in a dataset do not contribute equally to the performance of the model, labeling subset of the available instance should be done carefully to reduce annotation effort and minimize the cost of manual labeling. In Active Learning [51], the learning algorithm is proactively select the subset of available instances that needed to be labeled from a pool of unlabeled examples. The basic idea behind the concept is that machine learning models could potentially obtain better accuracy using less labeled examples if the models were able to select the data used in training. Active learner models dynamically pose queries during the training process, usually in the form of unlabeled data instances to be labeled by an oracle, usually a human annotator. As such, active learning is one of the most powerful examples of the success of the Human-in-the-Loop paradigm.

Active learning is well studied with many classical machine learning models and proves to achieve interesting results in many NLP tasks including named entity recognition [214], sentiment analysis [123], emotions detection [8]. However, deep active learning [215, 219, 9] is still a growing area of research fueled by the impressive empirical results of deep learning models and the availability of unlabeled data.

Bibliography

[1] Jibril O. Abdulmalik et al. "Country Contextualization of the Mental Health Gap Action Programme Intervention Guide: A Case Study from Nigeria". In: *PLoS medicine* 10 (Aug. 2013). DOI: 10.1371/journal.pmed.1001501.

[2] Parag Agrawal and Anshuman Suri. "NELEC at SemEval-2019 task 3: think twice before going deep". In: *arXiv preprint arXiv:1904.03223* (2019).

[3] Alan Akbik and Roland Vollgraf. "ZAP: An Open-Source Multilingual Annotation Projection Framework". In: *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. Miyazaki, Japan: European Language Resources Association (ELRA), May 2018. URL: https://www.aclweb.org/anthology/L18-1344.

[4] Firoj Alam et al. "The social mood of news: self-reported annotations to design automatic mood detection systems". In: *Proceedings of the Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media (PEOPLES)*. 2016, pp. 143–152.

[5] Areej Alhothali and Jesse Hoey. "Good News or Bad News: Using Affect Control Theory to Analyze Readers' Reaction Towards News Articles". In: *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Denver, Colorado: Association for Computational Linguistics, 2015, pp. 1548–1558. DOI: 10.3115/v1/N15-1178. URL: https://www.aclweb.org/anthology/N15-1178.

[6] Mario Ezra Aragón, Adrián Pastor López-Monroy, and Manuel Montes-y Gómez. "INAOE-CIMAT at eRisk 2019: Detecting Signs of Anorexia using Fine-Grained Emotions." In: *CLEF (Working Notes)*. 2019.

[7] Mikel Artetxe, Gorka Labaka, and Eneko Agirre. "An Effective Approach to Unsupervised Machine Translation". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 194–203. DOI: 10.18653/v1/P19-1019. URL: https://www.aclweb.org/anthology/P19-1019.

[8] Mahim-Ul Asad et al. "Introducing active learning on text to emotion analyzer". In: *2014 17th International Conference on Computer and Information Technology (ICCIT)*. IEEE. 2014, pp. 35–40.

[9] Nabiha Asghar et al. "Deep active learning for dialogue generation". In: *arXiv preprint arXiv:1612.03929* (2016).

[10] Nabiha Asghar et al. "Generating Emotionally Aligned Responses in Dialogues using Affect Control Theory". In: *arXiv preprint arXiv:2003.03645* (2020).

[11] Waleed M. Azmy, Amir F. Atiya, and Hisham El-Shishiny. "Forecast Combination Strategies for Handling Structural Breaks for Time Series Forecasting". In: *Multiple Classifier Systems, 9th International Workshop, MCS 2010, Cairo, Egypt, April 7-9, 2010. Proceedings*. Vol. 5997. Lecture Notes in Computer Science. 2010, pp. 245–253. DOI: 10.1007/978-3-642-12127-2_25.

[12] Waleed M. Azmy et al. "MLP, Gaussian Processes and Negative Correlation Learning for Time Series Prediction". In: *Multiple Classifier Systems, 8th International Workshop, MCS 2009, Reykjavik, Iceland, June 10-12, 2009. Proceedings*. Vol. 5519. Lecture Notes in Computer Science. 2009, pp. 428–437. DOI: 10.1007/978-3-642-02326-2_43.

[13] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. "Neural Machine Translation by Jointly Learning to Align and Translate". In: *International Conference on Learning Representations (ICLR)*. Vol. abs/1409.0473. Sept. 2014.

[14] Marco Baroni, Georgiana Dinu, and Germán Kruszewski. "Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors". In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2014, pp. 238–247.

[15] Angelo Basile et al. "SymantoResearch at SemEval-2019 task 3: combined neural models for emotion classification in human-chatbot conversations". In: *Proceedings of the 13th International Workshop on Semantic Evaluation*. 2019, pp. 330–334.

[16] David M Blei, Andrew Y Ng, and Michael I Jordan. "Latent dirichlet allocation". In: *Journal of machine Learning research* 3.Jan (2003), pp. 993–1022.

[17] Avrim Blum and Tom Mitchell. "Combining labeled and unlabeled data with co-training". In: *Proceedings of the eleventh annual conference on Computational learning theory*. 1998, pp. 92–100.

[18] Piotr Bojanowski et al. "Enriching word vectors with subword information". In: *Transactions of the Association for Computational Linguistics* 5 (2017), pp. 135–146.

[19] M. B. Bonsall et al. "Nonlinear time-series approaches in characterizing mood stability and mood instability in bipolar disorder". In: *Proceedings of the Royal Society of London B: Biological Sciences*. 2011. eprint: http://rspb.royalsocietypublishing.org/content/early/2011/08/12/rspb.2011.1246.full.pdf.

[20] Hadjer Boubenna and Dohoon Lee. "Image-based emotion recognition using evolutionary algorithms". In: *Biologically Inspired Cognitive Architectures* 24 (2018), pp. 70 –76. ISSN: 2212-683X. DOI: https://doi.org/10.1016/j.bica.2018.04.008. URL: http://www.sciencedirect.com/science/article/pii/S2212683X18300185.

[21] Gavin Brown and Jeremy Wyatt. "Negative Correlation Learning and the Ambiguity Family of Ensemble Methods". In: *Multiple Classifier Systems*. Berlin, Heidelberg, 2003, pp. 266–275. ISBN: 978-3-540-44938-6.

[22] Gavin Brown, Jeremy L. Wyatt, and Peter Tiňo. "Managing Diversity in Regression Ensembles". In: *J. Mach. Learn. Res.* 6 (Dec. 2005). ISSN: 1532-4435.

[23] Gavin Brown and Xin Yao. "On the Effectiveness of Negative Correlation Learning". In: *PROCEEDINGS OF FIRST UK WORKSHOP ON COMPUTATIONAL INTELLIGENCE*. 2001.

[24] Sergio G Burdisso, Marcelo Errecalde, and Manuel Montes-y Gómez. "A text classification framework for simple and effective early depression detection over social media streams". In: *Expert Systems with Applications* 133 (2019), pp. 182–197.

[25] Carlos Busso et al. "IEMOCAP: Interactive emotional dyadic motion capture database". In: *Language resources and evaluation* 42.4 (2008), p. 335.

[26] Fidel Cacheda, Diego Fernández, and Francisco Nóvoa. "Artificial intelligence and social networks for early detection of depression (Preprint)". In: *Journal of Medical Internet Research* 21 (Oct. 2018). DOI: 10.2196/12554.

[27] Fidel Cacheda et al. "Analysis and Experiments on Early Detection of Depression." In: *CLEF (Working Notes)* 2125 (2018).

[28] Fabio Calefato, Filippo Lanubile, and Nicole Novielli. "EmoTxt: A toolkit for emotion recognition from text". In: *2017 Seventh International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*. IEEE. 2017, pp. 79–80.

[29] Rafael A Calvo and Sidney D'Mello. "Affect detection: An interdisciplinary review of models, methods, and their applications". In: *IEEE Transactions on affective computing* 1.1 (2010), pp. 18–37.

[30] Rafael A Calvo et al. "Natural language processing in mental health applications using non-clinical texts". In: *Natural Language Engineering* 23.5 (2017), pp. 649–685.

[31] Sabrina Cerini et al. "Micro-WNOp: A gold standard for the evaluation of automatically compiled lexical resources for opinion mining". In: *Language resources and linguistic theory: Typology, second language acquisition, English linguistics* (2007), pp. 200–210.

[32] Ankush Chatterjee et al. "SemEval-2019 Task 3: EmoContext: Contextual Emotion Detection in Text". In: *Proceedings of The 13th International Workshop on Semantic Evaluation (SemEval-2019)*. Minneapolis, Minnesota, 2019.

[33] Ankush Chatterjee et al. "Understanding Emotions in Text Using Deep Learning and Big Data". In: *Computers in Human Behavior* 93 (2019), pp. 309 –317. ISSN: 0747-5632.

[34] Ciprian Chelba et al. "One billion word benchmark for measuring progress in statistical language modeling". In: *arXiv preprint arXiv:1312.3005* (2013).

[35] H. Chen and X. Yao. "Multiobjective Neural Network Ensembles Based on Regularized Negative Correlation Learning". In: *IEEE Transactions on Knowledge and Data Engineering* 22.12 (2010), pp. 1738–1751. ISSN: 2326-3865. DOI: 10.1109/TKDE.2010.26.

[36] Jianbo Chen and Michael I. Jordan. "LS-Tree: Model Interpretation When the Data Are Linguistic". In: *CoRR* abs/1902.04187 (2019). arXiv: 1902.04187. URL: http://arxiv.org/abs/1902.04187.

[37] Sheng-Yeh Chen et al. "Emotionlines: An emotion corpus of multi-party conversations". In: *arXiv preprint arXiv:1802.08379* (2018).

[38] Trishul Chilimbi et al. "Project Adam: Building an Efficient and Scalable Deep Learning Training System". In: *11th USENIX Symposium on Operating Systems Design and Implementation (OSDI 14)*. Broomfield, CO: USENIX Association, Oct. 2014, pp. 571–582. ISBN: 978-1-931971-16-4.

[39] Massimiliano Ciaramita and Olivier Chapelle. "Adaptive parameters for entity recognition with perceptron HMMs". In: *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*. 2010, pp. 1–7.

[40] Kevin Clark et al. "What does bert look at? an analysis of bert's attention". In: *arXiv preprint arXiv:1906.04341* (2019).

[41] Alexis Conneau et al. "Unsupervised cross-lingual representation learning at scale". In: *arXiv preprint arXiv:1911.02116* (2019).

[42] Glen Coppersmith et al. "CLPsych 2015 shared task: Depression and PTSD on Twitter". In: *Proceedings of the 2nd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality*. 2015, pp. 31–39.

[43] Darcy Corbitt-Hall et al. "College Students' Responses to Suicidal Content on Social Networking Sites: An Examination Using a Simulated Facebook Newsfeed". In: *Suicide life-threatening behavior* 46 (Mar. 2016). DOI: 10.1111/sltb.12241.

[44] Lorenzo Coviello et al. "Detecting emotional contagion in massive social networks". In: *PloS one* 9.3 (2014), e90315.

[45] Kathleen Cumiskey and Richard Ling. "The Social Psychology of Mobile Communication". English. In: *The Handbook of Psychology of Communication Technology*. Ed. by S. Shyam Sundar. Wiley-Blackwell, 2015, pp. 228–246. ISBN: 978-1-118-41336-4.

[46] M. D. Munezero et al. "Are They Different? Affect, Feeling, Emotion, Sentiment, and Opinion Detection in Text". In: *IEEE Transactions on Affective Computing* 5.2 (2014), pp. 101–111. ISSN: 1949-3045. DOI: 10.1109/TAFFC.2014.2317187.

[47] Zihang Dai et al. "Transformer-XL: Attentive Language Models beyond a Fixed-Length Context". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Italy, July 2019, pp. 2978–2988. DOI: 10.18653/v1/P19-1285.

[48] H. H. Dam et al. "Neural-Based Learning Classifier Systems". In: *IEEE Transactions on Knowledge and Data Engineering* 20.1 (2008), pp. 26–39. ISSN: 2326-3865. DOI: 10.1109/TKDE.2007.190671.

[49] Charles Darwin, Mortimer Jerome Adler, and Robert Maynard Hutchins. *The origin of species by means of natural selection*. Vol. 49. Encyclopaedia Britannica, 1952.

[50] Himansu Das, Chattaranjan Pradhan, and Nilanjan Dey. *Deep Learning for Data Analytics : Foundations, Biomedical Applications, and Challenges*. Apr. 2020. ISBN: 9780128197646.

[51] Sanjoy Dasgupta, Adam Tauman Kalai, and Claire Monteleoni. "Analysis of perceptron-based active learning". In: *International conference on computational learning theory*. Springer. 2005, pp. 249–263.

[52] III Hal Daumé and Daniel Marcu. "Domain Adaptation for Statistical Classifiers." In: *Journal of Artificial Intelligence Research* 26 (2006), pp. 101–126.

[53] Jorge Carrillo De Albornoz, Laura Plaza, and Pablo Gervás. "SentiSense: An easily scalable concept-based affective lexicon for sentiment analysis." In: *LREC*. Vol. 12. 2012, pp. 3562–3567.

[54] Munmun De Choudhury. "Anorexia on Tumblr: A Characterization Study". In: *Proceedings of the 5th International Conference on Digital Health 2015*. DH '15. Florence, Italy: Association for Computing Machinery, 2015, 43–50. ISBN: 9781450334921. DOI: 10.1145/2750511.2750515.

[55] Munmun De Choudhury, Scott Counts, and Eric Horvitz. "Predicting postpartum changes in emotion and behavior via social media". In: *Proceedings of the SIGCHI conference on human factors in computing systems*. 2013, pp. 3267–3276.

[56] Munmun De Choudhury, Scott Counts, and Eric Horvitz. "Social media as a measurement tool of depression in populations". In: *Proceedings of the 5th Annual ACM Web Science Conference*. 2013, pp. 47–56.

[57] Munmun De Choudhury and Sushovan De. "Mental health discourse on reddit: Self-disclosure, social support, and anonymity". In: *Eighth international AAAI conference on weblogs and social media*. 2014.

[58] Munmun De Choudhury et al. "Predicting depression via social media". In: *Seventh international AAAI conference on weblogs and social media*. 2013.

[59] Scott Deerwester et al. "Indexing by latent semantic analysis". In: *Journal of the American society for information science* 41.6 (1990), pp. 391–407.

[60] Jacob Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *CoRR* abs/1810.04805 (2018).

[61] *Diagnostic and statistical manual of mental disorders : DSM-5*. English. 5th ed. American Psychiatric Association, 2013. ISBN: 089042554 0890425558 9780890425541 9780890425558.

[62] Louise Doyle, Margaret M Pearl Treacy, and Ann J. Sheridan. "Self-harm in young people: Prevalence, associated factors, and help-seeking in school-going adolescents." In: *International journal of mental health nursing* 24 6 (2015), pp. 485–94.

[63] Timothy Dozat and Christopher D. Manning. "Deep Biaffine Attention for Neural Dependency Parsing". In: vol. abs/1611.01734. 2017.

[64] Mauro Dragoni et al. "SMACk: An Argumentation Framework for Opinion Mining." In: *IJCAI*. 2016, pp. 4242–4243. ISBN: 978-1-57735-771-1.

[65] J. Du et al. "A convolutional attentional neural network for sentiment classification". In: *2017 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC)*. 2017, pp. 445–450.

[66] Anuvabh Dutt, Denis Pellerin, and Georges Quénot. "Coupled Ensembles of Neural Networks". In: *2018 International Conference on Content-Based Multimedia Indexing (CBMI)* (2018), pp. 1–6.

[67] To'Meisha Edwards and Nicholas Holtzman. "A Meta-Analysis of Correlations between Depression and First Person Singular Pronoun Use". In: *Journal of Research in Personality* 68 (Feb. 2017). DOI: 10.1016/j.jrp.2017.02.005.

[68] Johannes C. Eichstaedt et al. "Facebook language predicts depression in medical records". In: *Proceedings of the National Academy of Sciences* 44 (2018). ISSN: 0027-8424. DOI: 10.1073/pnas.1802331115.

[69] Panteleimon Ekkekakis. "Affect, mood, and emotion". In: *Measurement in sport and exercise psychology* 321 (2012).

[70] Paul Ekman. "An argument for basic emotions". In: *Cognition & emotion* 6.3-4 (1992), pp. 169–200.

[71] Andrea Esuli and Fabrizio Sebastiani. "SentiWordNet: a high-coverage lexical resource for opinion mining". In: *Evaluation* 17.1 (2007), p. 26.

[72] Arash Fazl, Stephen Grossberg, and Ennio Mingolla. "View-invariant object category learning, recognition, and search: how spatial and object attention are coordinated using surface-based attentional shrouds". In: *Cognitive psychology* 58.1 (2009), pp. 1–48.

[73] Bjarke Felbo et al. "Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm". In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, Sept. 2017, pp. 1615–1625. DOI: 10.18653/v1/D17-1169. URL: https://www.aclweb.org/anthology/D17-1169.

[74] E. Fersini. "Chapter 6 - Sentiment Analysis in Social Networks: A Machine Learning Perspective". In: *Sentiment Analysis in Social Networks*. Ed. by Federico Alberto Pozzi et al. Boston: Morgan Kaufmann, 2017, pp. 91 –111. ISBN: 978-0-12-804412-4. DOI: https://doi.org/10.1016/B978-0-12-804412-4.00006-1. URL: http://www.sciencedirect.com/science/article/pii/B9780128044124000061.

[75] J.L. Fleiss. *Statistical methods for rates and proportions Rates and proportions*. Wiley, 1973.

[76] Joseph P Forgas and Gordon H Bower. "Mood effects on person-perception judgments." In: *Journal of personality and social psychology* 53.1 (1987), p. 53.

[77] Dario G Funez et al. "UNSL's participation at eRisk 2018 Lab." In: *CLEF (Working Notes)*. 2018.

[78] Stuart Geman, Elie Bienenstock, and René Doursat. "Neural Networks and the Bias/Variance Dilemma". In: *Neural Computation* 4 (Jan. 1992). DOI: 10.1162/neco.1992.4.1.1.

[79] George Gkotsis et al. "The language of mental health problems in social media". In: *Proceedings of the Third Workshop on Computational Linguistics and Clinical Psychology*. San Diego, CA, USA: Association for Computational Linguistics, June 2016. DOI: 10.18653/v1/W16-0307.

[80] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. "Domain Adaptation for Large-scale Sentiment Classification: A Deep Learning Approach". In: *Proceedings of the 28th International Conference on International Conference on Machine Learning*. ICML'11. Bellevue, Washington, USA: Omnipress, 2011, pp. 513–520. ISBN: 978-1-4503-0619-5. URL: http://dl.acm.org/citation.cfm?id=3104482.3104547.

[81] Erving Goffman. *The Presentation of Self in Everyday Life*. Anchor, 1959. ISBN: 0385094027.

[82] Yoav Goldberg. "Neural network methods for natural language processing". In: *Synthesis Lectures on Human Language Technologies* 10.1 (2017), pp. 1–309.

[83] Harry F Gollob, Betty B Rossman, and Robert P Abelson. "Social inference as a function of the number of instances and consistency of information presented." In: *Journal of Personality and Social Psychology* 27.1 (1973), p. 19.

[84] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[85] Chuan Guo et al. "On Calibration of Modern Neural Networks". In: *International Conference on Machine Learning*. 2017, pp. 1321–1330.

[86] Michael U Gutmann and Aapo Hyvärinen. "Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics". In: *The journal of machine learning research* 13.1 (2012), pp. 307–361.

[87] Francisco Guzmán et al. "The FLORES Evaluation Datasets for Low-Resource Machine Translation: Nepali–English and Sinhala–English". 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: Association for Computational Linguistics, Nov. 2019, pp. 6098–6111. DOI: 10.18653/v1/D19-1632. URL: https://www.aclweb.org/anthology/D19-1632.

[88] Michael Hammond. "Introduction to the Mathematics of Language". In: *University of Arizona* (2007).

[89] L. K. Hansen and P. Salamon. "Neural Network Ensembles". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 12.10 (1990), 993–1001. ISSN: 0162-8828. DOI: 10.1109/34.58871.

[90] Keith Hawton, Daniel Zahl, and Rosamund Weatherall. "Suicide following deliberate self-harm: long-term follow-up of patients who presented to a general hospital". In: *British Journal of Psychiatry* 182.6 (2003), 537–542.

[91] Devamanyu Hazarika et al. "Conversational memory network for emotion recognition in dyadic dialogue videos". In: *Proceedings of the conference. Association for Computational Linguistics. North American Chapter. Meeting*. Vol. 2018. NIH Public Access. 2018, p. 2122.

[92] Devamanyu Hazarika et al. "Icon: Interactive conversational memory network for multimodal emotion detection". In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 2018, pp. 2594–2604.

[93] Yulan He and Deyu Zhou. "Self-training from labeled features for sentiment analysis". In: *Information Processing & Management* 47.4 (2011), pp. 606–616.

[94] David R. Heise. "Affect control theory: Concepts and model". In: *The Journal of Mathematical Sociology* 13.1-2 (1987), pp. 1–33. DOI: 10.1080/0022250X.1987.9990025. eprint: https://doi.org/10.1080/0022250X.1987.9990025. URL: https://doi.org/10.1080/0022250X.1987.9990025.

[95] David R Heise. *Expressive order: Confirming sentiments in social actions*. Springer Science & Business Media, 2007.

[96] David R Heise. *Understanding events : affect and the construction of social action*. English. Includes index. Cambridge ; New York : Cambridge University Press, 1979. ISBN: 0521225396. URL: http://www.loc.gov/catdir/enhancements/fy0909/78024177-t.html.

[97] Hans Hoek. "Review of the worldwide epidemiology of eating disorders". In: *Current Opinion in Psychiatry*. Vol. 29. 2016.

[98] Jesse Hoey, Tobias Schröder, and Areej Alhothali. "Affect control processes: Intelligent affective interaction using a partially observable Markov decision process". In: *Artificial Intelligence* 230 (2016), pp. 134–172.

[99] Jesse Hoey, Tobias Schroder, and Areej Alhothali. "Bayesian affect control theory". In: *2013 Humaine Association Conference on Affective Computing and Intelligent Interaction*. IEEE. 2013, pp. 166–172.

[100] Jeremy Howard and Sebastian Ruder. "Universal Language Model Fine-tuning for Text Classification". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia, 2018, pp. 328–339.

[101] Minqing Hu and Bing Liu. "Mining and summarizing customer reviews". In: *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. 2004, pp. 168–177.

[102] Binxuan Huang and Kathleen Carley. "Parameterized Convolutional Neural Networks for Aspect Level Sentiment Classification". In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, 2018, pp. 1091–1096. URL: https://www.aclweb.org/anthology/D18-1136.

[103] Chenyang Huang, Amine Trabelsi, and Osmar Zaïane. "ANA at SemEval-2019 Task 3: Contextual Emotion detection in Conversations through hierarchical LSTMs and BERT". In: *Proceedings of the 13th International Workshop on Semantic Evaluation*. Minneapolis, Minnesota, USA: Association for Computational Linguistics, June 2019, pp. 49–53. DOI: 10.18653/v1/S19-2006. URL: https://www.aclweb.org/anthology/S19-2006.

[104] Fei Huang and Alexander Yates. "Distributional representations for handling sparsity in supervised sequence-labeling". In: *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*. 2009, pp. 495–503.

[105] Fei Huang and Alexander Yates. "Open-domain semantic role labeling by modeling word spans". In: *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. 2010, pp. 968–978.

[106] Gao Huang et al. "Deep networks with stochastic depth". In: *European conference on computer vision*. Springer. 2016, pp. 646–661.

[107] Rola Jadayel, Karim Medlej, and Jinan Jennifer Jadayel. "Mental Disorders: A Glamorous Attraction on Social Media?" In: *Journal of Teaching and Education*. Vol. 07. 2017.

[108] R. Jenke, A. Peer, and M. Buss. "Feature Extraction and Selection for Emotion Recognition from EEG". In: *IEEE Transactions on Affective Computing* 5.3 (2014), pp. 327–339. ISSN: 1949-3045. DOI: 10.1109/TAFFC.2014.2339834.

[109] Rie Johnson and Tong Zhang. "Supervised and semi-supervised text categorization using LSTM for region embeddings". In: *arXiv preprint arXiv:1602.02373* (2016).

[110] Kenneth Joseph et al. "A social-event based approach to sentiment analysis of identities and behaviors in text". In: *The Journal of Mathematical Sociology* 40.3 (2016), pp. 137–166. DOI: 10.1080/0022250X.2016.1159206. eprint: https://doi.org/10.1080/0022250X.2016.1159206. URL: https://doi.org/10.1080/0022250X.2016.1159206.

[111] Diana Joyce and Michael L. Sulkowski. "The Diagnostic and Statistical Manual of Mental Disorders: Fifth Edition (DSM-5) Model of Impairment". In: *Assessing Impairment: From Theory to Practice*. 2016, pp. 167–189. ISBN: 978-1-4899-7994-0.

[112] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. "A Convolutional Neural Network for Modelling Sentences". In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*. Baltimore, USA, 2014.

[113] J.R. Kantor. *An objective psychology of grammar*. Indiana University publications: Science series. Indiana university, 1936.

[114] Andreas Kaplan and Michael Haenlein. "Siri, Siri, in my hand: Who's the fairest in the land? On the interpretations, illustrations, and implications of artificial intelligence". In: *Business Horizons* 62.1 (2019), pp. 15 –25. ISSN: 0007-6813. DOI: https://doi.org/10.1016/j.bushor.2018.08.004. URL: http://www.sciencedirect.com/science/article/pii/S0007681318301393.

[115] Andreas Kaplan and Michael Haenlein. "Users of the World, Unite! The Challenges and Opportunities of Social Media". In: *Business Horizons* 53 (Feb. 2010), pp. 59–68. DOI: 10.1016/j.bushor.2009.09.003.

[116] Alan E. Kazdin. "Addressing the treatment gap: A key challenge for extending evidence-based psychosocial interventions." In: *Behaviour research and therapy* 88 (2017), pp. 7–18.

[117] Mikhail Khodak, Nikunj Saunshi, and Kiran Vodrahalli. "A large self-annotated corpus for sarcasm". In: *arXiv preprint arXiv:1704.05579* (2017).

[118] Alan Kim. "Wilhelm Maximilian Wundt". In: *The Stanford Encyclopedia of Philosophy*. Ed. by Edward N. Zalta. Fall 2016. Metaphysics Research Lab, Stanford University, 2016.

[119] E David Klonsky. "The functions of deliberate self-injury: A review of the evidence". In: *Clinical psychology review* 27 (Apr. 2007), pp. 226–39. DOI: 10.1016/j.cpr.2006.08.002.

[120] Ron Kohavi and David Wolpert. "Bias plus Variance Decomposition for Zero-One Loss Functions". In: *Proceedings of the Thirteenth International Conference on International Conference on Machine Learning*. ICML'96. Bari, Italy, 1996, 275–283. ISBN: 1558604197.

[121] Marina Krakovsky. "Artificial (Emotional) Intelligence". In: *Commun. ACM* 61.4 (Mar. 2018), pp. 18–19. ISSN: 0001-0782. DOI: 10.1145/3185521. URL: http://doi.acm.org/10.1145/3185521.

[122] Adam DI Kramer, Jamie E Guillory, and Jeffrey T Hancock. "Experimental evidence of massive-scale emotional contagion through social networks". In: *Proceedings of the National Academy of Sciences* 111.24 (2014), pp. 8788–8790.

[123] Janez Kranjc et al. "Active learning for sentiment analysis on data streams: Methodology and workflow implementation in the ClowdFlows platform". In: *Information Processing Management* 51.2 (2015), pp. 187 –203. ISSN: 0306-4573. DOI: https://doi.org/10.1016/j.ipm.2014.04.001. URL: http://www.sciencedirect.com/science/article/pii/S0306457314000296.

[124] Taku Kudo and John Richardson. "SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing". In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Belgium, 2018. DOI: 10.18653/v1/D18-2012.

[125] Ludmila I. Kuncheva. *Combining Pattern Classifiers: Methods and Algorithms*. USA: Wiley-Interscience, 2004. ISBN: 0471210781.

[126] Guillaume Lample et al. "Phrase-Based & Neural Unsupervised Machine Translation". In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, 2018, pp. 5039–5049. DOI: 10.18653/v1/D18-1549. URL: https://www.aclweb.org/anthology/D18-1549.

[127] Hugo Larochelle and Geoffrey E Hinton. "Learning to combine foveal glimpses with a third-order Boltzmann machine". In: *Advances in Neural Information Processing Systems 23*. 2010, pp. 1243–1251.

[128] Quoc V. Le and Tomas Mikolov. "Distributed Representations of Sentences and Documents." In: *ICML*. Vol. 32. JMLR Workshop and Conference Proceedings. JMLR.org, 2014, pp. 1188–1196.

[129] Marianna Leite Barroso et al. "SOCIAL PANIC DISORDER AND ITS IMPACTS". In: *Amadeus International Multidisciplinary Journal*. Vol. 2. 2018, pp. 1–17.

[130] Gaël Letarte et al. "Importance of Self-Attention for Sentiment Analysis". In: *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Brussels, Belgium, 2018, pp. 267–275.

[131] Omer Levy and Yoav Goldberg. "Neural word embedding as implicit matrix factorization". In: *Advances in neural information processing systems*. 2014, pp. 2177–2185.

[132] Dawei Li, Jin Wang, and Xuejie Zhang. "YUN-HPCC at SemEval-2019 Task 3: Multi-Step Ensemble Neural Network for Sentiment Analysis in Textual Conversation". In: *Proceedings of the 13th International Workshop on Semantic Evaluation*. 2019, pp. 360–364.

[133] Xin Li et al. "Aspect Term Extraction with History Attention and Selective Transformation". In: *IJCAI*. ijcai.org, 2018, pp. 4194–4200.

[134] Yanran Li et al. "Dailydialog: A manually labelled multi-turn dialogue dataset". In: *arXiv preprint arXiv:1710.03957* (2017).

[135] Xihao Liang, Ye Ma, and Mingxing Xu. "THU-HCSI at SemEval-2019 Task 3: Hierarchical Ensemble Classification of Contextual Emotion in Conversation". In: *Proceedings of the 13th International Workshop on Semantic Evaluation*. Minneapolis, Minnesota, USA: Association for Computational Linguistics, June 2019, pp. 345–349. DOI: 10.18653/v1/S19-2060. URL: https://www.aclweb.org/anthology/S19-2060.

[136] Zhouhan Lin et al. "A Structured Self-Attentive Sentence Embedding". In: *International Conference on Learning Representations (ICLR)*. 2017.

[137] Jiahua Liu et al. "XQA: A Cross-lingual Open-domain Question Answering Dataset". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 2358–2368. DOI: 10.18653/v1/P19-1227. URL: https://www.aclweb.org/anthology/P19-1227.

[138] Y. Liu and X. Yao. "Ensemble Learning via Negative Correlation". In: *Neural Netw.* 12.10 (Dec. 1999), 1399–1404. ISSN: 0893-6080. DOI: 10.1016/S0893-6080(99)00073-8.

[139] Yinhan Liu et al. "RoBERTa: A Robustly Optimized BERT Pretraining Approach". In: *CoRR* abs/1907.11692 (2019). arXiv: 1907.11692.

[140] David E. Losada and Fabio Crestani. "A Test Collection for Research on Depression and Language use". In: *Conference Labs of the Evaluation Forum*. Springer, 2016, pp. 28–39. ISBN: 978-3-319-44563-2.

[141] David E. Losada, Fabio Crestani, and Javier Parapar. "eRISK 2017: CLEF Lab on Early Risk Prediction on the Internet: Experimental Foundations". In: *8th International Conference of the CLEF Association*. Springer Verlag, 2017, pp. 346–360. ISBN: 978-3-319-65812-4.

[142] David E. Losada, Fabio Crestani, and Javier Parapar. "Overview of eRisk – Early Risk Prediction on the Internet". In: *Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Ninth International Conference of the CLEF Association (CLEF 2018)*. Avignon, France, 2018.

[143] David E. Losada, Fabio Crestani, and Javier Parapar. "Overview of eRisk 2019: Early Risk Prediction on the Internet". In: *Experimental IR Meets Multilinguality, Multimodality, and Interaction. 10th International Conference of the CLEF Association, CLEF 2019*. Lugano, Switzerland: Springer International Publishing, 2019.

[144] Ling Luo et al. "Beyond Polarity: Interpretable Financial Sentiment Analysis with Hierarchical Query-driven Attention". In: *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI*. July 2018, pp. 4244–4250.

[145] Yukun Ma, Haiyun Peng, and Erik Cambria. "Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive LSTM". In: *Association for the Advancement of Artificial Intelligence (AAAI 2018)*. 2018.

[146] Andrew L. Maas et al. "Learning Word Vectors for Sentiment Analysis". In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*. HLT '11. Portland, Oregon: Association for Computational Linguistics, 2011, pp. 142–150. ISBN: 978-1-932432-87-9.

[147] Laurens Van der Maaten and Geoffrey Hinton. "Visualizing Data using t-SNE". In: *Journal of Machine Learning Research*. Vol. 9. 2008, pp. 2579–2605.

[148] Alhassan Mabrouk, Rebeca P Díaz Redondo, and Mohammed Kayed. "Deep Learning-Based Sentiment Classification: A Comparative Survey". In: *IEEE Access* 8 (2020), pp. 85616–85638.

[149] Navonil Majumder et al. "DialogueRNN: An Attentive RNN for Emotion Detection in Conversations". In: *CoRR* Association for the Advancement of Artificial Intelligence (AAAI 2019) (2018).

[150] Mika Viking Mäntylä, Daniel Graziotin, and Miikka Kuutila. "The Evolution of Sentiment Analysis - A Review of Research Topics, Venues, and Top Cited Papers". In: *Computer Science Review* 27 (2018), pp. 16–32.

[151] Matthew Matero et al. "Suicide Risk Assessment with Multi-level Dual-Context Language and BERT". In: *Proceedings of the Sixth Workshop on Computational Linguistics and Clinical Psychology*. Minneapolis, Minnesota, June 2019, pp. 39–44.

[152] David McClosky, Eugene Charniak, and Mark Johnson. "Effective self-training for parsing". In: *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*. 2006, pp. 152–159.

[153] Gary McKeown et al. "The semaine database: Annotated multimodal records of emotionally colored conversations between a person and a limited agent". In: *IEEE transactions on affective computing* 3.1 (2011), pp. 5–17.

[154] Stephen Merity, Nitish Shirish Keskar, and Richard Socher. "Regularizing and Optimizing LSTM Language Models". In: *International Conference on Learning Representations (ICLR)*. 2018.

[155] Stephen Merity et al. "Pointer Sentinel Mixture Models". In: *CoRR* abs/1609.07843 (2016). arXiv: 1609.07843. URL: http://arxiv.org/abs/1609.07843.

[156] Tomas Mikolov, Scott Wen-tau Yih, and Geoffrey Zweig. "Linguistic Regularities in Continuous Space Word Representations". In: *(NAACL-HLT-2013)*. 2013.

[157] Tomas Mikolov et al. "Distributed Representations of Words and Phrases and their Compositionality". In: *Advances in Neural Information Processing Systems 26*. Curran Associates, Inc., 2013, pp. 3111–3119.

[158] Tomas Mikolov et al. "Efficient Estimation of Word Representations in Vector Space". In: *CoRR*. Vol. abs/1301.3781. 2013. arXiv: 1301.3781.

[159] David N. Milne et al. "CLPsych 2016 Shared Task: Triaging content in online peer-support forums". In: *Proceedings of the Third Workshop on Computational Linguistics and Clinical Psychology*. San Diego, CA, USA: Association for Computational Linguistics, June 2016, pp. 118–127. DOI: 10.18653/v1/W16-0312.

[160] Volodymyr Mnih, Nicolas Heess, Alex Graves, et al. "Recurrent models of visual attention". In: *Advances in neural information processing systems*. 2014, pp. 2204–2212.

[161] Saif M. Mohammad. "Word Affect Intensities". In: *Proceedings of the 11th Edition of the Language Resources and Evaluation Conference (LREC-2018)*. Miyazaki, Japan, 2018.

[162] Saif M. Mohammad and Peter D. Turney. "Crowdsourcing a Word-Emotion Association Lexicon". In: *Computational Intelligence*. Vol. 29. 3. 2013, pp. 436–465.

[163] Elham Mohammadi, Hessam Amini, and Leila Kosseim. "CLaC at CLPsych 2019: Fusion of Neural Features and Predicted Class Probabilities for Suicide Risk Assessment Based on Online Posts". In: *Proceedings of the Sixth Workshop on Computational Linguistics and Clinical Psychology*. Minneapolis, Minnesota, 2019.

[164] Elham Mohammadi, Hessam Amini, and Leila Kosseim. "Quick and (maybe not so) Easy Detection of Anorexia in Social Media Posts". In: *Working Notes of CLEF 2019 - Conference and Labs of the Evaluation Forum, Lugano, Switzerland, September 9-12, 2019*. Vol. 2380. 2019.

[165] S. Mokhtari, T. Li, and N. Xie. "Context-Sensitive Neural Sentiment Classification". In: *2018 IEEE International Conference on Information Reuse and Integration (IRI)*. 2018, pp. 293–299.

[166] Christopher E Moody. "Mixing dirichlet topic models and word embeddings to make lda2vec". In: *arXiv preprint arXiv:1605.02019* (2016).

[167] Bilel Moulahi, Jérôme Azé, and Sandra Bringay. "DARE to Care: A Context-Aware Framework to Track Suicidal Ideation on Social Media." In: *Web Information Systems Engineering - WISE 2017.,Lecture Notes in Computer Science,. Springer, Cham*. 2017.

[168] David G Myers. "Theories of emotion". In: *Psychology: Seventh Edition, New York, NY: Worth Publishers* 500 (2004).

[169] Brendan O'Connor, Brandon M. Stewart, and Noah A. Smith. "Learning to Extract International Relations from Political Context". In: *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Sofia, Bulgaria: Association for Computational Linguistics, Aug. 2013, pp. 1094–1104. URL: https://www.aclweb.org/anthology/P13-1108.

[170] Bridianne O'Dea et al. "Detecting suicidality on Twitter". In: *Internet Interventions* 2.2 (2015), pp. 183 –188. ISSN: 2214-7829. DOI: https://doi.org/10.1016/j.invent.2015.03.005.

[171] Joseph Olive, Caitlin Christianson, and John McCary. *Handbook of natural language processing and machine translation: DARPA global autonomous language exploitation*. Springer Science & Business Media, 2011.

[172] Michael Opitz, Horst Possegger, and Horst Bischof. "Efficient Model Averaging for Deep Neural Networks". In: *Computer Vision – ACCV 2016*. Cham: Springer International Publishing, 2017, pp. 205–220. ISBN: 978-3-319-54184-6.

[173] World Health Organization. *Mental health action plan 2013-2020*. World Health Organization, 2013, 45 p.

[174] Rosa María Ortega-Mendoza, Delia Irazú Hernández Farías, and Manuel Montes-y-Gómez. "LTL-INAOE's Participation at eRisk 2019: Detecting Anorexia in Social Media through Shared Personal Information". In: *Working Notes of CLEF 2019 - Conference and Labs of the Evaluation Forum, Lugano, Switzerland, September 9-12, 2019*. Vol. 2380. 2019.

[175] Charles Egerton Osgood et al. *Cross-cultural universals of affective meaning*. Vol. 1. University of Illinois Press, 1975.

[176] Bridianne O'Dea et al. "A Linguistic Analysis of Suicide-Related Twitter Posts". In: *Crisis: The Journal of Crisis Intervention and Suicide Prevention* 38 (2017), 319–329.

[177] Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. "Unsupervised learning of sentence embeddings using compositional n-gram features". In: *arXiv preprint arXiv:1703.02507* (2017).

[178] F. Pallavicini, P. Cipresso, and F. Mantovani. "Chapter 2 - Beyond Sentiment: How Social Network Analytics Can Enhance Opinion Mining and Sentiment Analysis". In: *Sentiment Analysis in Social Networks*. Ed. by Federico Alberto Pozzi et al. Boston: Morgan Kaufmann, 2017, pp. 13 –29. ISBN: 978-0-12-804412-4. DOI: https://doi.org/10.1016/B978-0-12-804412-4.00002-4. URL: http://www.sciencedirect.com/science/article/pii/B9780128044124000024.

[179] Bo Pang and Lillian Lee. "Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales". In: *arXiv preprint cs/0506075* (2005).

[180] German I. Parisi et al. "Continual lifelong learning with neural networks: A review". In: *Neural Networks* 113 (2019), pp. 54 –71. ISSN: 0893-6080. DOI: https://doi.org/10.1016/j.neunet.2019.01.012. URL: http://www.sciencedirect.com/science/article/pii/S0893608019300231.

[181] Rashmi Patel et al. "Mood instability is a common feature of mental health disorders and is associated with poor clinical outcomes". In: *BMJ Open*. Vol. 5. 5. British Medical Journal Publishing Group, 2015. eprint: http://bmjopen.bmj.com/content/5/5/e007504.full.pdf.

[182] Michael J. Paul and Mark Dredze. "You Are What You Tweet: Analyzing Twitter for Public Health." In: *ICWSM*. 2011.

[183] Sayanta Paul, Sree Kalyani Jandhyala, and Tanmay Basu. "Early Detection of Signs of Anorexia and Depression Over Social Media using Effective Machine Learning Frameworks." In: *CLEF (Working Notes)*. 2018.

[184] James W Pennebaker. *Emotion, disclosure, & health*. American Psychological Association, 1995.

[185] Jeffrey Pennington, Richard Socher, and Christopher D Manning. "Glove: Global vectors for word representation". In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 2014, pp. 1532–1543.

[186] Matthew Peters et al. "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: Association for Computational Linguistics, June 2018, pp. 2227–2237. DOI: 10.18653/v1/N18-1202. URL: https://www.aclweb.org/anthology/N18-1202.

[187] Rosalind W Picard. *Affective computing*. MIT Technical Report 321, 1995.

[188] S Pinker. *Learnability and Cognition: The Acquisition of Argument Structure*. Cambridge, MA: MIT Press, 1989.

[189] Robert Plutchik. *The emotions*. University Press of America, 1991.

[190] Maria Pontiki et al. "Semeval-2015 task 12: Aspect based sentiment analysis". In: *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*. 2015, pp. 486–495.

[191] Maria Pontiki et al. "Semeval-2016 task 5: Aspect based sentiment analysis". In: *10th International Workshop on Semantic Evaluation (SemEval 2016)*. 2016.

[192] Soujanya Poria, Erik Cambria, and Alexander Gelbukh. "Deep convolutional neural network textual features and multiple kernel learning for utterance-level multimodal sentiment analysis". In: *Proceedings of the 2015 conference on empirical methods in natural language processing*. 2015, pp. 2539–2544.

[193] Soujanya Poria et al. "Context-dependent sentiment analysis in user-generated videos". In: *Proceedings of the 55th annual meeting of the association for computational linguistics (volume 1: Long papers)*. 2017, pp. 873–883.

[194] Soujanya Poria et al. "Meld: A multimodal multi-party dataset for emotion recognition in conversations". In: *arXiv preprint arXiv:1810.02508* (2018).

[195] "Preventing suicide: a global imperative". In: *WHO (World Health Organisation)* (2014), pp. 7–20.

[196] Alec Radford et al. "Improving language understanding by generative pre-training (2018)". In: *URL https://s3-us-west-2. amazonaws. com/openai-assets/research-covers/language-unsupervised/language_ understanding_ paper. pdf* (2018).

[197] Waleed Ragheb et al. "Temporal Mood Variation: at the CLEF eRisk-2018 Tasks for Early Risk Detection on The Internet". In: *Working Notes of CLEF 2018 - Conference and Labs of the Evaluation Forum, Avignon, France, September 10-14, 2018*. Vol. 2125. 2018.

[198] Diana Ramírez-Cifuentes, Marc Mayans, and Ana Freire. "Early Risk Detection of Anorexia on Social Media". In: *Internet Science - 5th International Conference, INSCI 2018, St. Petersburg, Russia, October 24-26, 2018, Proceedings*. Vol. 11193. Lecture Notes in Computer Science. Springer, 2018, pp. 3–14. DOI: 10.1007/978-3-030-01437-7_1.

[199] Adrian Rauchfleisch et al. "How journalists verify online sources during terrorist crises. Analyzing Twitter communication during the Brussels attacks". In: *Social Media and Society* 3.3 (2017), online. ISSN: 2056-3051. URL: https://doi.org/10.5167/uzh-148292.

[200] Kumar Ravi and Vadlamani Ravi. "A survey on opinion mining and sentiment analysis: tasks, approaches and applications". In: *Knowledge-Based Systems* 89 (2015), pp. 14–46.

[201] Tahilia Rebello et al. "Innovative strategies for closing the mental health treatment gap globally". In: *Current opinion in psychiatry* 27 (May 2014). DOI: 10.1097/YCO.0000000000000068.

[202] Ellen Riloff and Janyce Wiebe. "Learning extraction patterns for subjective expressions". In: *Proceedings of the 2003 conference on Empirical methods in natural language processing*. 2003, pp. 105–112.

[203] Stephen P Robbins and Tim Judge. *Essentials of organizational behavior*. Vol. 7. Prentice Hall Upper Saddle River, NJ, 2003.

[204] Sarah T Roberts. *Behind the screen: Content moderation in the shadows of social media*. Yale University Press, 2019.

[205] Sebastian Ruder and Barbara Plank. "Strong baselines for neural semi-supervised learning under domain shift". In: *arXiv preprint arXiv:1804.09530* (2018).

[206] Farig Sadeque, Dongfang Xu, and Steven Bethard. "Measuring the Latency of Depression Detection in Social Media". In: *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. WSDM '18. Marina Del Rey, CA, USA: Association for Computing Machinery, 2018, 495–503. ISBN: 9781450355810. DOI: 10.1145/3159652.3159725.

[207] Adam Sadilek et al. "Modeling fine-grained dynamics of mood at scale". In: *WSDM, Rome, Italy* (2013), pp. 3–6.

[208] Ivan A Sag et al. "Multiword expressions: A pain in the neck for NLP". In: *International conference on intelligent text processing and computational linguistics*. Springer. 2002, pp. 1–15.

[209] Gerard Salton and Christopher Buckley. "Term-weighting approaches in automatic text retrieval". In: *Information Processing Management* 24.5 (1988), pp. 513 –523. ISSN: 0306-4573. DOI: https://doi.org/10.1016/0306-4573(88)90021-0. URL: http://www.sciencedirect.com/science/article/pii/0306457388900210.

[210] Oana Sandu et al. "Domain adaptation to summarize human conversations". In: *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*. 2010, pp. 16–22.

[211] Elizabeth Seabrook et al. "Predicting Depression From Language-Based Emotion Dynamics: Longitudinal Analysis of Facebook and Twitter Status Updates". In: *Journal of Medical Internet Research* 20 (May 2018), e168. DOI: 10.2196/jmir.9267.

[212] Rico Sennrich, Barry Haddow, and Alexandra Birch. "Neural Machine Translation of Rare Words with Subword Units". In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*. Berlin, Germany, Aug. 2016, pp. 1715–1725. DOI: 10.18653/v1/P16-1162.

[213] Claude E Shannon. "A mathematical theory of communication". In: *The Bell system technical journal* 27.3 (1948), pp. 379–423.

[214] Dan Shen et al. "Multi-criteria-based active learning for named entity recognition". In: *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*. 2004, pp. 589–596.

[215] Yanyao Shen et al. "Deep active learning for named entity recognition". In: *arXiv preprint arXiv:1707.05928* (2017).

[216] Z. Shi et al. "Crowd Counting with Deep Negative Correlation Learning". In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, pp. 5382–5390. DOI: 10.1109/CVPR.2018.00564.

[217] Han-Chin Shing et al. "Expert, crowdsourced, and machine assessment of suicide risk via online postings". In: *Proceedings of the Fifth Workshop on Computational Linguistics and Clinical Psychology: From Keyboard to Clinic*. 2018, pp. 25–36.

[218] Hong-Han Shuai et al. "A Comprehensive Study on Social Network Mental Disorders Detection via Online Social Media Mining". In: *IEEE Transactions on Knowledge and Data Engineering* (Dec. 2017), pp. 1–1. DOI: 10.1109/TKDE.2017.2786695.

[219] Aditya Siddhant and Zachary C. Lipton. "Deep Bayesian Active Learning for Natural Language Processing: Results of a Large-Scale Empirical Study". In: *CoRR* abs/1808.05697 (2018). arXiv: 1808.05697. URL: http://arxiv.org/abs/1808.05697.

[220] Edwin Simpson et al. "Dynamic Bayesian Combination of Multiple Imperfect Classifiers". In: *Decision Making and Imperfection*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 1–35.

[221] Noah A Smith and Mark Johnson. "Weighted and probabilistic context-free grammars are equally expressive". In: *Computational Linguistics* 33.4 (2007), pp. 477–491.

[222] P.H.A. Sneath and R.R. Sokal. *Numerical Taxonomy. The Principles and Practice of Numerical Classification*. Freeman, 1973.

[223] Richard Socher et al. "Recursive deep models for semantic compositionality over a sentiment treebank". In: *Proceedings of the 2013 conference on empirical methods in natural language processing*. 2013, pp. 1631–1642.

[224] Anders Søgaard. "Simple semi-supervised training of part-of-speech taggers". In: *Proceedings of the ACL 2010 Conference Short Papers*. 2010, pp. 205–208.

[225] M Soheylizad and B Moeini. "Social media: An opportunity for developing countries to change healthy behaviors". In: *Health Education and Health Promotion* 7.2 (2019), pp. 57–58.

[226] Nitish Srivastava et al. "Dropout: A Simple Way to Prevent Neural Networks from Overfitting". In: *Journal of Machine Learning Research* 15.56 (2014), pp. 1929–1958.

[227] Philip J Stone and Earl B Hunt. "A computer approach to content analysis: studies using the general inquirer system". In: *Proceedings of the May 21-23, 1963, spring joint computer conference*. 1963, pp. 241–256.

[228] Carlo Strapparava, Alessandro Valitutti, et al. "Wordnet affect: an affective extension of wordnet." In: *Lrec*. Vol. 4. 1083-1086. Citeseer. 2004, p. 40.

[229] Jinsong Su et al. "A Hierarchy-to-Sequence Attentional Neural Machine Translation Model". In: *IEEE/ACM Trans. Audio, Speech & Language Processing*. Vol. 26. 3. 2018, pp. 623–632.

[230] "Suicide Fact sheet °398". In: *WHO (World Health Organisation)* (Mar. 2016).

[231] Chi Sun et al. "How to Fine-Tune BERT for Text Classification?" In: *CoRR* abs/1905.05583 (2019).

[232] Maite Taboada et al. "Lexicon-based methods for sentiment analysis". In: *Computational linguistics* 37.2 (2011), pp. 267–307.

[233] Matt Taddy. "Document Classification by Inversion of Distributed Language Representations". In: *CoRR*. Vol. abs/1504.07295. 2015. arXiv: 1504.07295.

[234] Duyu Tang, Bing Qin, and Ting Liu. "Document modeling with gated recurrent neural network for sentiment classification". In: *Proceedings of the 2015 conference on empirical methods in natural language processing*. 2015, pp. 1422–1432.

[235] Jianhua Tao and Tieniu Tan. "Affective Computing: A Review". In: *Affective Computing and Intelligent Interaction*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 981–995. ISBN: 978-3-540-32273-3.

[236] "The National Eating Disorders Association (NEDA).: Envisioning a world without eating disorders". In: *The newsletter of the National Eating Disorders Association. Issue 22*. 2009.

[237] Silvan S Tomkins and Robert McCarter. "What and where are the primary affects? Some evidence for a theory". In: *Perceptual and motor skills* 18.1 (1964), pp. 119–158.

[238] Antonio Torralba et al. "Contextual guidance of eye movements and attention in real-world scenes: the role of global features in object search." In: *Psychological review* 113.4 (2006), p. 766.

[239] Sebastian Trautmann, Jürgen Rehm, and Hans-Ulrich Wittchen. "The economic costs of mental disorders: Do our societies react appropriately to the burden of mental disorders?" In: *EMBO*. 2016.

[240] Marcel Trotzek, Sven Koitka, and Christoph Friedrich. "Linguistic Metadata Augmented Classifiers at the CLEF 2017 Task for Early Detection of Depression". In: *Working Notes of CLEF 2017 - Conference and Labs of the Evaluation Forum*. Vol. CEUR-WS 1866. 2017.

[241] Marcel Trotzek, Sven Koitka, and Christoph Friedrich. "Utilizing Neural Networks and Linguistic Metadata for Early Detection of Depression Indications in Text Sequences". In: *IEEE Transactions on Knowledge and Data Engineering* (2018).

[242] Marcel Trotzek, Sven Koitka, and Christoph M. Friedrich. "Utilizing Neural Networks and Linguistic Metadata for Early Detection of Depression Indications in Text Sequences". In: *IEEE Trans. Knowl. Data Eng.* 32.3 (2020), pp. 588–601.

[243] Marcel Trotzek, Sven Koitka, and Christoph M. Friedrich. "Word Embeddings and Linguistic Metadata at the CLEF 2018 Tasks for Early Detection of Depression and Anorexia". In: *Working Notes of CLEF 2018 - Conference and Labs of the Evaluation Forum, Avignon, France, September 10-14, 2018*. Vol. 2125. 2018.

[244] Sho Tsugawa et al. "Recognizing depression from twitter activity". In: *Proceedings of the 33rd annual ACM conference on human factors in computing systems*. 2015, pp. 3187–3196.

[245] Martin Tutek and Jan Šnajder. "Iterative Recursive Attention Model for Interpretable Sequence Classification". In: *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Brussels, Belgium, 2018, pp. 249–257.

[246] Ashish Vaswani et al. "Attention is All you Need". In: *Advances in Neural Information Processing Systems 30*. Curran Associates, Inc., 2017, pp. 5998–6008.

[247] Andreas Veglis. "Moderation Techniques for Social Media Content". In: *Social Computing and Social Media*. Ed. by Gabriele Meiselwitz. Cham: Springer International Publishing, 2014, pp. 137–148.

[248] Graham Vickery and Sacha Wunsch-Vincent. *Participative Web and User-Created Content: Web 2.0, Wikis and Social Networking*. 1st ed. OECD Publications. http://www.oecd.org/document/40/0,3746,en_2649_34223_39428648_1_1_1_1,00.html [Stand: 09.03. 2011]. Paris: OECDpublishing, 2007. ISBN: 978-92-64-03746-5. URL: http://www.oecd.org/document/40/0,3746,en_2649_34223_39428648_1_1_1_1,00.html.

[249] Daniel Vigo, Graham Thornicroft, and Rifat Atun. "Estimating the true global burden of mental illness". In: *The Lancet Psychiatry* 3 (Feb. 2016). DOI: 10.1016/S2215-0366(15)00505-2.

[250] Athanasios Voulodimos et al. "Deep Learning for Computer Vision: A Brief Review". In: *Computational Intelligence and Neuroscience*. Vol. 2018. Feb. 2018, pp. 1–13.

[251] Li Wan et al. "Regularization of Neural Networks using DropConnect". In: *Proceedings of the 30th International Conference on Machine Learning*. Ed. by Sanjoy Dasgupta and David McAllester. Vol. 28. Proceedings of Machine Learning Research 3. Atlanta, Georgia, USA: PMLR, 2013, pp. 1058–1066. URL: http://proceedings.mlr.press/v28/wan13.html.

[252] Shiyao Wang, Minlie Huang, and Zhidong Deng. "Densely Connected CNN with Multi-scale Feature Attention for Text Classification". In: *IJCAI*. 2018, pp. 4468–4474.

[253] Max L. Wilson, Susan Ali, and Michel F. Valstar. "Finding information about mental health in microblogging platforms: a case study of depression". In: *IIiX*. 2014.

[254] Genta Indra Winata et al. "CAiRE_HKUST at SemEval-2019 Task 3: Hierarchical Attention for Dialogue Emotion Classification". In: *Proceedings of the 13th International Workshop on Semantic Evaluation*. Minneapolis, Minnesota, USA: Association for Computational Linguistics, June 2019, pp. 142–147. DOI: 10.18653/v1/S19-2021. URL: https://www.aclweb.org/anthology/S19-2021.

[255] "World Health Organization.: Depression and other common mental disorders: global health estimates." In: *World Health Organization*. 2017.

[256] Yonghui Wu et al. "Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation". In: *ArXiv* abs/1609.08144 (2016).

[257] Joan Xiao. "Figure Eight at SemEval-2019 Task 3: Ensemble of Transfer Learning Methods for Contextual Emotion Detection". In: *Proceedings of the 13th International Workshop on Semantic Evaluation*. Minneapolis, Minnesota, USA: Association for Computational Linguistics, June 2019, pp. 220–224. DOI: 10.18653/v1/S19-2036. URL: https://www.aclweb.org/anthology/S19-2036.

[258] Tong Xiao et al. "Learning from massive noisy labeled data for image classification". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.

[259] Qizhe Xie et al. "Unsupervised data augmentation for consistency training". In: *arXiv preprint arXiv:1904.12848* (2019).

[260] Zhilin Yang et al. "XLNet: Generalized Autoregressive Pretraining for Language Understanding". In: *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019, pp. 5753–5763.

[261] Zichao Yang et al. "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*. Association for Computational Linguistics, 2016, pp. 1480–1489.

[262] Wenpeng Yin and Hinrich Schütze. "Multichannel variable-size convolution for sentence classification". In: *arXiv preprint arXiv:1603.04513* (2016).

[263] Tom Young et al. "Recent Trends in Deep Learning Based Natural Language Processing [Review Article]". In: *IEEE Computational Intelligence Magazine*. Vol. 13. 2018, pp. 55–75.

[264] G. Udny Yule. "On the Association of Attributes in Statistics: With Illustrations from the Material of the Childhood Society, c". In: *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character* 194 (1900), pp. 257–319. ISSN: 02643952.

[265] Le Zhang et al. "Nonlinear Regression via Deep Negative Correlation Learning". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PP (Sept. 2019), pp. 1–1. DOI: 10.1109/TPAMI.2019.2943860.

[266] Xiang Zhang, Junbo Zhao, and Yann LeCun. "Character-level Convolutional Networks for Text Classification". In: *Advances in Neural Information Processing Systems 28*. 2015, pp. 649–657.

[267] Hao Zhou et al. "Emotional Chatting Machine: Emotional Conversation Generation with Internal and External Memory". In: *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18)*. 2018, pp. 730–739.

[268] Zhi-Hua Zhou and Ming Li. "Tri-training: Exploiting unlabeled data using three classifiers". In: *IEEE Transactions on knowledge and Data Engineering* 17.11 (2005), pp. 1529–1541.

[269] Ayah Zirikly et al. "CLPsych 2019 Shared Task: Predicting the Degree of Suicide Risk in Reddit Posts". In: *Proceedings of the Sixth Workshop on Computational Linguistics and Clinical Psychology*. Minneapolis, 2019.