ICIMP 2020

The Fifteenth International Conference on Internet Monitoring and Protection

ISBN: 978-1-61208-804-4

September 27th – October 1st, 2020

ICIMP 2020 Editors

Sebastião Pais, University of Beira Interior, Covilhã, Portugal

Irfan Khan Tanoli, University of Beira Interior, Covilhã, Portugal

ICIMP 2020

Foreword

The Fifteenth International Conference on Internet Monitoring and Protection (ICIMP 2018), held between September 27 – October 1st, 2020, continued a series of special events targeting security, performance, vulnerabilities in Internet, as well as disaster prevention and recovery.

The design, implementation and deployment of large distributed systems are subject to conflicting or missing requirements leading to visible and/or hidden vulnerabilities. Vulnerability specification patterns and vulnerability assessment tools are used for discovering, predicting and/or bypassing known vulnerabilities.

Vulnerability self-assessment software tools have been developed to capture and report critical vulnerabilities. Some of vulnerabilities are fixed via patches, other are simply reported, while others are self-fixed by the system itself. Despite the advances in the last years, protocol vulnerabilities, domain-specific vulnerabilities and detection of critical vulnerabilities rely on the art and experience of the operators; sometimes this is fruit of hazard discovery and difficult to be reproduced and repaired.

System diagnosis represent a series of pre-deployment or post-deployment activities to identify feature interactions, service interactions, behavior that is not captured by the specifications, or abnormal behavior with respect to system specification. As systems grow in complexity, the need for reliable testing and diagnosis grows accordingly. The design of complex systems has been facilitated by CAD/CAE tools. Unfortunately, test engineering tools have not kept pace with design tools, and test engineers are having difficulty developing reliable procedures to satisfy the test requirements of modern systems. Therefore, rather than maintaining a single candidate system diagnosis, or a small set of possible diagnoses, anticipative and proactive mechanisms have been developed and experimented. In dealing with system diagnosis data overload is a generic and tremendously difficult problem that has only grown. Cognitive system diagnosis methods have been proposed to cope with volume and complexity.

Attacks against private and public networks have had a significant spreading in the last years. With simple or sophisticated behavior, the attacks tend to damage user confidence, cause huge privacy violations and enormous economic losses.

The CYBER-FRAUD track focuses on specific aspects related to attacks and counterattacks, public information, privacy and safety on cyber-attacks information. It also targets secure mechanisms to record, retrieve, share, interpret, prevent and post-analyze of cyber-crime attacks.

Current practice for engineering carrier grade IP networks suggests n-redundancy schema. From the operational perspective, complications are involved with multiple n-box PoP. It is not guaranteed that this n-redundancy provides the desired 99.999% uptime. Two complementary solutions promote (i) high availability, which enables network-wide protection by providing fast recovery from faults that may occur in any part of the network, and (ii) non-stop routing. Theory on robustness stays behind the attempts for improving system reliability with regard to emergency services and containing the damage through disaster prevention, diagnosis and recovery.

We take here the opportunity to warmly thank all the members of the ICIMP 2020 Technical Program Committee, as well as all of the reviewers. The creation of such a high quality conference program would not have been possible without their involvement. We also kindly thank all the authors who dedicated much of their time and efforts to contribute to ICIMP 2020. We truly believe that, thanks to all these efforts, the final conference program consisted of top quality contributions.

Also, this event could not have been a reality without the support of many individuals, organizations, and sponsors. We are grateful to the members of the ICIMP 2020 organizing committee for their help in handling the logistics and for their work to make this professional meeting a success.

We hope that ICIMP 2020 was a successful international forum for the exchange of ideas and results between academia and industry and for the promotion of progress in the field of Internet monitoring and protection.

ICIMP 2020 Chairs:

ICIMP 2020 Publicity Chair
Mar Parra, Universitat Politecnica de Valencia, Spain

ICIMP 2020

COMMITTEE

ICIMP 2020 Publicity Chair

Mar Parra, Universitat Politecnica de Valencia, Spain

ICIMP 2020 Technical Program Committee

Prashant Anantharaman, Dartmouth College, USA
Muhammad Ajmal Azad, University of Derby, UK
Lasse Berntzen, University of South-Eastern Norway, Norway
Francesco Buccafurri, Mediterranean University of Reggio Calabria, Italy
Paolina Centonze, Iona College, New York, USA
Nora Cuppens, IMT Atlantique, France
Paolo D'Arco, University of Salerno, Italy
Lorenzo De Carli, Worcester Polytechnic Institute, USA
Raffaele Della Corte, "Federico II" University of Naples, Germany
Parvez Faruki, Malaviya National Institute of Technology, India
Mathias Fischer, Universität Hamburg, Germany
Oliver Gasser, Max Planck Institute for Informatics in Saarbruecken, Germany
Kambiz Ghazinour. State University of New York in Canton, USA
Rita Girao-Silva, University of Coimbra &INESC Coimbra, Portugal
Zhen Huang, DePaul University, USA
Mikel Iturbe, Mondragon University, Spain
Hamid Jahankhani, Northumbria University London, UK
Terje Jensen, Telenor, Norway
Basel Katt, Norwegian University of Science and Technology (NTNU), Norway
Irfan Khan Tanoli, University of Beira Interior (UBI), Portugal
Vitaly Klyuev, University of Aizu, Japan
Pushpendra Kumar, Manipal University Jaipur, India
Aditya Kuppa, Tenable Inc. / University College Dublin, Ireland
Yuping Li, Pinterest, USA
Pooria Madani, York University, Toronto, Canada
Pradip Mainali, OneSpan, Belgium
Sathiamoorthy Manoharan, University of Auckland, New Zealand
Jims Marchang, Sheffield Hallam University, UK
Michael J. May, Kinneret Academic College, Israel
Anze Mihelic, University of Maribor, Slovenia
Aleksandra Mileva, University Goce Delcev in Stip, Republic of North Macedonia
Lorenzo Musarella, University Mediterranea of Reggio Calabria, Italy
Constantin Paleologu, University Politehnica of Bucharest, Romania
Antonio Pecchia, University of Sannio-Benevento, Italy
Eckhard Pfluegel, Kingston University, London, UK
Nikolaos Polatidis, University of Brighton, UK
Dumitru Popescu, University "Politehnica" of Bucharest, Romania

Danny Raz, Technion, Israel
Hamid Reza Ghaeini, CISPA - Helmholtz Center for Information Security, Germany
Antonia Russo, University Mediterranea of Reggio Calabria, Italy
Marco Antonio Sotelo Monge, Universidad de Lima, Peru
Moritz Steiner, Akamai, USA
Hung-Min Sun, National Tsing Hua University, Taiwan
Pengfei Sun, Shape Security, USA
Jani Suomalainen, VTT Technical Research Centre of Finland, Finland
Bernhard Tellenbach, Zurich University of Applied Sciences (ZHAW), Switzerland
Phani Vadrevu, University of New Orleans, USA
Rob van der Mei, Centre for Mathematics and Computer Science (CWI), Netherlands
Julien Vanegue, Bloomberg LP, USA
Miroslav N. Velev, Aries Design Automation, USA
Christian Wressnegger, Karlsruhe Institute of Technology (KIT), Germany
Zhen Xie, JD America Corp, USA
Apostolis Zarras, Maastricht University, Netherlands

Copyright Information

For your reference, this is the text governing the copyright release for material published by IARIA.

The copyright release is a transfer of publication rights, which allows IARIA and its partners to drive the dissemination of the published material. This allows IARIA to give articles increased visibility via distribution, inclusion in libraries, and arrangements for submission to indexes.

I, the undersigned, declare that the article is original, and that I represent the authors of this article in the copyright release matters. If this work has been done as work-for-hire, I have obtained all necessary clearances to execute a copyright release. I hereby irrevocably transfer exclusive copyright for this material to IARIA. I give IARIA permission or reproduce the work in any media format such as, but not limited to, print, digital, or electronic. I give IARIA permission to distribute the materials without restriction to any institutions or individuals. I give IARIA permission to submit the work for inclusion in article repositories as IARIA sees fit.

I, the undersigned, declare that to the best of my knowledge, the article is does not contain libelous or otherwise unlawful contents or invading the right of privacy or infringing on a proprietary right.

Following the copyright release, any circulated version of the article must bear the copyright notice and any header and footer information that IARIA applies to the published article.

IARIA grants royalty-free permission to the authors to disseminate the work, under the above provisions, for any academic, commercial, or industrial use. IARIA grants royalty-free permission to any individuals or institutions to make the article available electronically, online, or in print.

IARIA acknowledges that rights to any algorithm, process, procedure, apparatus, or articles of manufacture remain with the authors and their employers.

I, the undersigned, understand that IARIA will not be liable, in contract, tort (including, without limitation, negligence), pre-contract or other representations (other than fraudulent misrepresentations) or otherwise in connection with the publication of my work.

Exception to the above is made for work-for-hire performed while employed by the government. In that case, copyright to the material remains with the said government. The rightful owners (authors and government entity) grant unlimited and unrestricted permission to IARIA, IARIA's contractors, and IARIA's partners to further distribute the work.

Table of Contents

A Lexicon Based Approach to Detect Extreme Sentiments

Sebastião Pais
Computer Science Department
NOVA LINCS and UBI
Covilhã, Portugal
Email:sebastiao@di.ubi.pt

Irfan Khan Tanoli
Computer Science Department
University of Beira Interior
Covilhã, Portugal
Email:irfan.khan.tanoli@ubi.pt

Miguel Albardeiro
Computer Science Department
University of Beira Interior
Covilhã, Portugal
Email:miguel.albardeiro@ubi.pt

João Cordeiro
Computer Science Department
University of Beira Interior
Covilhã, Portugal
Email:jpaulo@di.ubi.pt

***Abstract*—Online social network platforms enable people freedom of expression to share their ideas, views, and emotions that could be negative or positive. Previous studies have investigated the user's sentiments on such platforms to study the behaviour of people for different scenarios and purposes. The mechanism to collect information on public views attracted researchers by analyzing data from social networks and automatically classifying the polarity of public opinion(s) due to the use of concise language in posts or tweets. However, each cluster of tweet messages or posts focusing on a burst topic may constitute a potential threat to society and people. In this paper, we propose an unsupervised approach for automatic detection of people's extreme sentiments on social networks. For this, our first task was automatically to build a standard lexicon consisting of extreme sentiments terms having high extreme positive and extreme negative polarity. With this new lexicon of extreme sentiments, our final task is to validate this lexicon, for which we developed an unsupervised approach for automatic detection of extreme sentiments, and we evaluated our performance on five different social networks and media datasets. This final task shows that, in these datasets, posts classified with negative sentiments, there are posts of extremely negative sentiments. On the other hand, in posts classified with positive sentiments, there are posts of extremely positive sentiments.**

Keywords–Sentiment Analysis; Extreme Sentiment Analysis; Social Media; Natural Language Processing; Extremism

I. INTRODUCTION

An unnatural way of sentiment analysis is to detect and classify extreme sentiment(s), which represent(s) the most negative and positive sentiments about a particular topic, an object or an individual. An extreme sentiment is the worst or the best view, judgment, or appraisal formed in one's mind about a particular matter or people. However, in this work, we consider extreme sentiment to be "a personal extreme positive or extreme negative feeling". We propose an interesting unsupervised and language-independent approach for detecting people's extreme sentiments on social platforms. Firstly, we analyze two standard corpora, i.e., SENTIWORDNET 3.0 [1] and SenticNet 5 [2] for extracting extreme words having a high negative and positive polarity, reflecting people's extreme sentiments. We design and develop a prototype system called *Extreme Sentiment Analyzer (ESA)* composed of two different components, i.e.,*Extreme Sentiment Generator (ESG)* and *Extreme Sentiment Classifier (ESC)*. *ESG* is based on statistical methods, and we apply it on SENTIWORDNET 3.0 and SenticNet to generate a standard lexical resource known as *ExtremeSentiLex* [3], that contains only extreme positive and extreme negative terms as discussed in Section III. Additionally, this lexical resource can be used by anti-extremism agencies to find an extreme opinion on social networks to counter violent extremism.

Next, we embed *ExtremeSentiLex* in the *ESC* and run on the compilation of five different datasets, which are constituted of social network and media posts as presented in Section IV. The purpose of this experimentation is to assess the accuracy of our tool, and this evaluation will validate our hypothesis that the *ESC* finds posts with extremely negative and positive sentiments in these datasets. To obtain more complete results, we use a confusion matrix to calculate adapt conventional performance measures, namely, recall, precision, f_1 score and accuracy to check the performance of the *ESC*.

II. RELATED WORK

Sentiment analysis and opinion mining, in the field of Natural Language Processing, is an active area of study that analyzes people's opinions, sentiments, evaluations, attitudes, and emotions via the computational treatment of subjectivity in text. We do provide some studies and techniques presenting the manifest extreme sentiments on social networks, social media, and on the digital web. Additionally, we also discuss the works regarding the sentimental lexicons and datasets that we exploit in our work.

A. Extreme Opinions

The fundamental task in Opinion Mining is the polarity classification, which occurs when a piece of text stating an opinion that is classified into a predefined set of polarity categories, e,g., positive, neutral, negative [4]. The authors of [5] investigate the effectiveness of the automatic construction of a sentiment lexicon using unsupervised machine learning classification to search for extreme opinions. The experiments are carried out using reviews on commercial products and movies. There are, at least, two types of strategies for sentiment analysis: Machine-Learning-Based and lexical-based. Machine learning strategies usually rely on supervised classification using lexical resources which tends to detect the sentiment in binary terms (i.e., positive or negative).

B. Detection and Classification of Social Media Extremist Affiliations

Sentiment Analysis (SA) is one of the prominent areas for researchers, particularly related to social networks activities. Generally, SA systems can be classified into two categories: knowledge-based and statistics-based systems. The earlier knowledge-based approaches were the most popular among researchers for sentiment polarity identification in texts. However, researchers have been progressively relying upon statistics based approaches with a keen focus on supervised statistical methods [2]. The authors work in [6] suggests a binary classification task to detect extremist affiliation. The

 ISBN: 978-1-61208-804-4

focus of the work is the use of machine learning classifier, i.e., Random Forest, Support Vector Machine, KN-Neighbors, Naive Bayes Classifiers, and Deep learning classifiers. The authors apply sentiment-based extremist classification technique based on user's tweets that operates in three modules: (i) user's tweet collection, (ii) pre-processing, and (iii) classification concerning extremist and non-extremist classes using different deep learning-based sentiment models, i.e., *Long Short Term Memory*, *Convolutional Neural Networks*, *FastText* and *Gated Recurrent Units (GRU)*.

C. Sentiment based Lexicons

SENTIWORDNET 3.0 is developed using the automatic annotation of all WORDNET synsets with the notions of 'positivity', negativity' and 'neutrality'. Each synset has three numerical scores, which indicate the terms as positive, negative, and objective (i.e., neutral). e.g., *majestic score: 0.75 (positive term), invalid 0.75 (negative)*. The study in [1] presents the use of SENTIWORDNET 3.0 as a base for the development of extremism lexical resource, an enhanced lexical resource to be used as support for sentiment classification and opinion mining applications [7].

SenticNet 5 [2] encodes the denotative and connotative information commonly associated with real-world objects, actions, events, and people. It steps away from blindly using keywords and word co-occurrence counts, and instead relies on the implicit meaning associated with common sense concepts. Superior to purely syntactic techniques, SenticNet 5 can detect subtly expressed sentiments by enabling the analysis of multi-word expressions that do not explicitly convey emotion but are instead related to concepts that do so. An example of SenticNet 5 datasets is: *favourite 0.87 (positive), worry -0.93 (negative)*.

D. Sentiment Analysis Datasets.

Sentiment analysis is a type of natural language processing algorithm that determines the polarity of a piece of text. That is, a sentiment analysis predicts whether the opinion given in a piece of text is positive, negative, or neutral. These analyses provide a powerful tool for gaining insights into large sets of opinion-based data, such as social media posts and product reviews. For example, a seller on the Amazon marketplace could use sentiment analysis to quickly assess thousands of reviews and gauge customer satisfaction of their goods. Sentiment analysis can also be used to predict the reviews for a new product by comparing product metadata to similar products and analyzing those products' reviews. Sentiment analysis requires large sets of labelled training data to develop and tune, also called a training sentiment analysis dataset. The first step in analysis development requires a sentiment analysis dataset of tens of thousands of statements that are already labelled as positive, negative, or neutral. Finding training data is difficult because a human expert must determine and label the polarity of each statement in the training data. Having a ready-made training dataset that is already significantly labelled reduces the time and effort needed to develop a sentiment analysis. In our work we use five dataset, Sentiment 140, Twitter for Sentiment Analysis (T4SA), RT-polarity, TurntoIslam and Ansar1.

To examine user's tweets for sentiment analysis, work in [8] utilizevSentiment 140 [9] and SentiStrength on a large representative set of research papers that specifically adopt few techniques to education articles distributed on Twitter. Sentiment 140 consists of two CVS files, one for test and another for training. Sentiment 140 provides one sentiment value per tweet on a scale from 0 (negative) to 4 (positive). For better comparison, values are converted to obtain three sentiment categories: positive, negative, and neutral. We select the test file for the evaluation of our system. The authors in [10] use of Twitter for Sentiment Analysis (T4SA) images dataset [11], that contains both textual and multimedia data for studying user's sentiment. The authors have gathered the twitter data using streaming crawler for six months and deploy for visual SA evaluation. The study concludes that the approach is useful for learning visual sentiment classifiers. T4SA dataset and the trained models are publicly released for future research and applications.

A user's opinion(s) despite positive or negative related to a specific topic has an impact on society and people. The study in [12], for detecting user's opinions on movie reviews using RT-polarity [13] lexicon, classified 2000 comments into two different categories. Generally, comment(s) mainly consist(s) sentence(s), the authors classify the user's sentiments at the sentence level and later classified overall comments as opinion. The obtained collection consists of two files, one for each set of 5331 positive opinions and negative opinions, containing one sentence per line, making it easy to process.

TurntoIslam [14] and Ansar1 [15] both having posts are organized into threads, which generally indicate topic under discussion and focus on extremist religious (e.g., jihadist) and general Islamic discussions. Each post includes detailed metadata, e.g., date, member name. As announced on the forum, this is an English language forum having a goal *"Correction of common misconceptions about Islam"*. Radical participants may occasionally display their support for fundamentalist militant groups as well. This two corpus will help us to understand if our approach has a good performance in the extremist religious (e.g., jihadist) and general Islamic discourse.

Although a vast number of existing approaches and few studies have offered an explicit comparison between sentiment analysis techniques. [16] shows the comparisons of eight popular sentiment analysis methods in terms of coverage and agreement. They develop a new method that combines existing approaches, providing the best coverage results and competitive agreement. [17] introduce a comparison of twenty-four popular sentiment analysis methods at the sentence-level, based on a benchmark of eighteen labelled datasets. The performance has been evaluated in two sentiment classification tasks: two classes, i.e., negative vs positive and three classes, i.e., negative, neutral and positive. However, these studies never compare the efficiency of sentiment analysis methods or sentiment lexicons in the specific task of identifying extreme sentiments, i.e., extreme positive and extreme negative.

III. Lexicon of Extreme Sentiments

In this section, we present a methodological approach to generate a lexicon of extreme positive and negative terms from SENTIWORDNET 3.0 and SenticNet.5. Our intention in this step is to collect a lexicon, using an automated approach without specific thresholds. In other words, our criterion for collecting terms can be adopted for any corpus input, because, their values of selection limits are defined by the average and

 ISBN: 978-1-61208-804-4

standard deviation of their scores. Figure 1 shows the overall process of extreme sentiment collection, where AVG is the average of positive and negative term scores, and SD is the standard deviation.

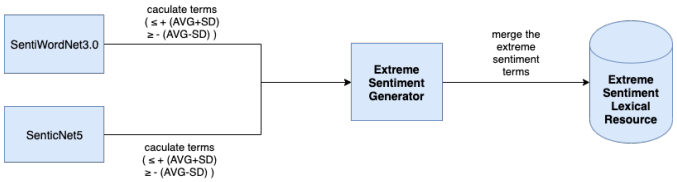

Figure 1. Extreme sentiment collection process.

A. Defining Extreme Polarity

The first phase of collecting extreme sentiments is to define the extreme polarity for the terms. The objective of this phase is to establish a metric to classify the terms that have extreme scores for both positive and negative. Referring to Figure 1, we develop a python application so-called *Extreme Sentiment Generator (ESG)* that performs certain operations, i.e., calculate the average and standard deviation of terms from the original lexical resources, filter and save it into a new lexical resource. We define two conditions in ESG to categorize both positive and negative terms, respectively. Since each dataset has different terms classification, we use either one condition or both to identify extreme positive and negative sentiments, whereas T_p refers as positive terms, and T_n as negative terms. The conditions are as follows:

if $T_p > Average + StandardDeviation$ **then**
The term is classified as *Extreme Positive*
end if
if $T_n < Average - StandardDeviation$ **then**
The term is classified as *Extreme Negative*
end if

Afterward, we process both data resources one by one as follows:

SENTIWORDNET 3.0: This dataset has three categories for terms: 'positive', 'negative' and 'neutral'. The score for both positive and negative terms are in a range of $[0, 1]$. First, we filter this lexical resource and obtain only positive and negative terms separately. Following we use the first condition for identifying extreme positive and negative terms. With the calculation using ESG, we obtained the following outputs:

```
Average for positive terms:             0.366
Standard Deviation for positive terms:  0.211
Extreme polarity for positive terms:    0.577
Average for negative terms:             0.412
Standard Deviation for negative terms:  0.230
Extreme polarity for negative terms:    0.642
```

The output shows that positive extreme polarity is 0.577 while negative extreme polarity is 0.642. To classify a term as positive or negative, consider the following examples output terms of SENTIWORDNET 3.0 generated by ESG:

```
ultrasonic  0.375 (non positive extreme)
selfless    0.875 (positive extreme)
thrash      0.125 (non negative extreme)
abduction   1     (negative extreme)
```

selfless is detected as a positive extreme since $0.577 < 0.875$ while *ultrasonic* is not. *Abduction* is a negative extreme $0.642 < 1$ and *thrash* is not. We discard all non-positive and non-negative extreme terms from our obtained lexicon and export the result in a CSV file.

SenticNet 5: In this dataset, to find the extremes, each term has one score in a single interval of $[-1, 1]$. To calculate the extreme polarities using *ESG*, the outputs are as follows:

```
Average for positive terms:             0.504
Standard Deviation for positive terms:  0.362
Extreme polarity for positive terms:    0.866
Average for negative terms:            -0.616
Standard Deviation for negative terms:  0.306
Extreme polarity for positive terms:   -0.922
```

Again only positive terms with intensity greater than 0.866 are considered as positive extremes, and negative terms with intensity lower than -0.922 are taken as negative extremes. Consider the following sample example output:

```
grace     0.79  (positive non extreme)
pioneer   0.97  (positive extreme)
anemic   -0.918 (negative non extreme)
traffic  -0.97  (negative extreme)
```

Again, all non-positive and non-negative extreme terms are discarded and export this result in another CSV file.

B. Generating Extreme Sentiments Lexicon

In this phase, we generate our final standard extreme sentiment lexicon. To achieve this, we merge both files obtained from SENTIWORDNET 3.0 and SenticNet 5. In SENTIWORDNET 3.0, positive and negative extremes lay in the range between $[0, 1]$ interval, while in SenticNet 5 the scores range -1 to 1, for negative (< 0) and positive (> 0) extremes. To uniform the scales, we multiply all the negative terms of SENTIWORDNET 3.0 by -1 to obtain a range in $[-1, 1]$. Then, we merge both files, remove all duplicate terms by considering the ones with the highest score and create the final CSV file refers as *ExtremSentiLex* [3], and shown in Figure 1. The final result is a text file with two columns: the term and its corresponding intensity. Below is a sample output of terms and their scores:

```
Term             Score
absolutely       +0.88
accept           +0.93
acknowledgeable  +0.95
acne             -0.96
actively         +0.95
adroitness       +0.88
agent            +0.91
agoraphobic      -0.95
alright          +0.88
amuse            +0.92
```

IV. Experimental Setup

We set up the experiment using *Extrem Sentiment Classifier (ESC)* having *ExtremeSentiLex* embed in it to check the accuracy of our system. We perform the experiments on three social media corpora, i.e., TurnToIslam [14], Ansar1 [15], RT-polaritydata [13], and two social network corpora, T4SA Images Dataset [10] and Sentiment 140 [9]. The main goal of this experimentation is to analyze whether *ESC* can identify the extreme positive and negative terms from these datasets or not. In other words, the focus is on detecting those posts that reflect extremely positive sentiments of users with current positive

 ISBN: 978-1-61208-804-4

polarity and detecting posts with extremely negative sentiments with current negative polarity. We further use confusion matrix to analyze the performance of our classification model by computing recall, precision for extreme positive, negative terms, the overall accuracy for measuring the results, and f1 score for extreme positive and negative terms.

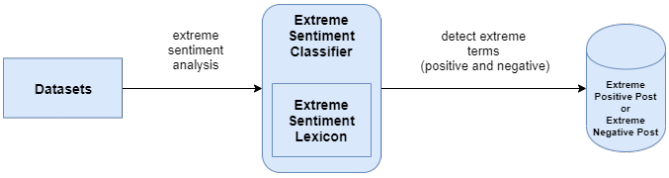

Figure 2. Performance testing of Extreme Sentiment Classifier.

Figure 2 depicts the overall process of experimentation. First, we apply ESC on datasets to detect only extreme posts (no polarity), i.e., ESC discovers posts that contain terms representing extreme sentiments. For this, we define the 1 to identify the posts containing extreme sentiments, and we consider only such post(s) as an extreme post(s) that satisfy the equation.

Whenever a positive or a negative term(s) is/are found, it is added and stored in a variable, i.e., $\sum T_{EP}$ refers to the total sum of all scores extreme positive terms while $\sum T_{EN}$ refers to the total sum of all scores extreme negative terms.

$$EXTREME: \ |\sum T_{EP} - |\sum T_{EN}|| > \frac{\sum T_{EP} + |\sum T_{EN}|}{2} \quad (1)$$

With 1, we detect extreme posts, but not their polarity, so, we hypothesize that an extreme post contains extreme sentiments, however, this post can contain extreme sentiments of only one polarity or both polarities. The next step, we determine the polarity of an extreme post, so, we define the three conditions that are applied on post polarity:

if $\sum T_{EP} > |\sum T_{EN}|$ && $EXTREME$ **then**
1. The post is classified as Extreme Positive
else if $\sum T_{EP} < |\sum T_{EN}|$ && $EXTREME$ **then**
2. The post is classified as Extreme Negative
else
3. The post is classified as Inconclusive
end if

Example1: Consider the following extreme positive example from Sentiment 140:

> *Since when does #alcohol equal **#happiness**? I know many people that started drinking; have been **happy** since.*

Where the terms and their scores in ExtremeSentiLex is:

happiness +1.0, happy +0.89

Above we see a tweet with two words that represent extreme positive sentiment, so we sum the scores and apply the algorithm:

$$|\sum T_{EP} - |\sum T_{EN}|| > \frac{\sum T_{EP}+|\sum T_{EN}|}{2} \Leftrightarrow$$
$$\Leftrightarrow |1.89 - 0| > \frac{1.89+0}{2} \Leftrightarrow 1.89 > 0.945$$

The condition $1.89 > 0.945$ is true, so the post is classified as *EXTREME*.

Now it is needed to check the polarity:

$$\sum T_{EP} > |\sum T_{EN}| \Leftrightarrow 1.89 > 0$$

The condition $1.89 > 0$ is true, so the post is classified as ***Extreme Positive***.

Example2: Consider the following negative extreme from TurnToIslam:

> *They will think all non-muslims are **sanguinary**, abominable monsters...! I want to ask you now, are they right?*

Where the term and their score in ExtremeSentiLex is:

sanguinary -0.93

Here, we can see a tweet with one word that represents negative sentiment. To testify this using our equation:

$$|\sum T_{EP} - |\sum T_{EN}|| > \frac{\sum T_{EP}+|\sum T_{EN}|}{2} \Leftrightarrow$$
$$\Leftrightarrow |0 - 0.93| > \frac{0+0.93}{2} \Leftrightarrow 0.93 > 0.465$$

The condition $0.93 > 0.465$ is true, so the post is *EXTREME*.

Now needs to check the polarity:

$$\sum T_{EP} < |\sum T_{EN}| \Leftrightarrow 0 < 0.93$$

The condition $0 < 0.93$ is true, so the post is classified as ***Extreme Negative***.

Example3: An example of the non extreme post from Ansar1:

Hustlers don't sleep, we nap!

There is no term detected as positive or negative. By analyzing using our equation:

$$|\sum T_{EP} - |\sum T_{EN}|| > \frac{\sum T_{EP}+|\sum T_{EN}|}{2}$$
$$\Leftrightarrow |0 - 0| > \frac{0+0}{2} \Leftrightarrow 0 > 0$$

The condition $0 = 0$, so the post is not *EXTREME*.

V. Results and Discussion

In this section, we present the results and analyze their efficiency, but this analysis takes into account that in the original datasets, the posts are classified with positive, negative, neutral polarity (except Ansar1 and TurntoIslam). However, the objective is to detect extreme posts, so we hypothesise that our methodology is capable of:

- **Detecting more extreme positive** posts and fewer negative extreme posts in the set of **original positive posts**;
- **Detecting more extreme negative posts** and fewer positive extreme posts in the set of **original negative posts**.

Table I shows the total number of the original posts, the total of the extreme posts detected, the total of the extremes positive

posts and total of the extremes negative posts. This information also reveals that our approach detected a few extreme posts in the datasets.

TABLE I. TOTAL NUMBER OF EXTREME POSTS DETECTED FROM ORIGINAL DATASETS

	RT-polarity	Sentiment 140	T4SA	Turnto Islam	Ansar1
Total of Extreme	1928 (≈ 18%)	45 (≈ 9%)	140987 (≈ 12%)	104038 (≈ 31%)	11022 (≈ 37%)
Extreme Positive	1646 (≈ 15%)	33 (≈ 7%)	130335 (≈ 11%)	97952 (≈ 29%)	9834 (≈ 33%)
Extreme Negative	282 (≈ 3%)	12 (≈ 2%)	10652 (≈ 1%)	6086 (≈ 2%)	1188 (≈ 4%)
Total	10662 (100%)	497 (100%)	1179957 (100%)	335328 (100%)	29492 (100%)

Tables II, III, IV, and V represent each dataset results individually; the organisations for these Tables are different, according to the each dataset itself original settings. For example, Ansar1 and TurnToIslam results only show the percentage of extreme posts, because the original dataset has not information about polarity.

For datasets, RT-polarity, Sentiment 140 and T4SA, we evaluate the results through the confusion matrix. A confusion matrix summarizes the classification performance of a classifier with respect to some test data. So, our case, **P** - Positive, **N** - Negative and **Neutral** are the original polarity of the posts, **EP** are posts classified as positive extremes, **EN** are classified posts as negative extremes and $\overline{E}$ **+ INC** are posts classified as non-extreme or inconclusive. We analyze the performance of our system by calculating adapt conventional performance measures as shown in Table VI.

TABLE II. RT-POLARITY RESULTS

	P	N	Total
EP	**971**	675	1646
EN	99	**183**	282
$\overline{E}$ + INC	4261	4473	8734
Total	5331	5331	10662

Table II shows that in RT-polarity , ESC detects 18% (True Positive (TP)) extreme positive posts from the set of original positives posts and 3% (True Negative (TN)) extreme negative posts. While ESC incorrectly classifies average 7% posts (False Positives (FP) + False Negatives (FN)). In a preliminary analysis, we can verify that our system has a good performance in the detection of extreme positive posts in this datasets. However, the results are not promising for the detection of extreme negative posts; the number of FN is greater to TN.

TABLE III. SENTIMENT140 RESULTS

	P	N	Neutral	Total
EP	**20**	11	2	33
EN	1	11	0	12
$\overline{E}$ + INC	160	155	137	452
Total	181	177	139	497

For Sentiment140 (Table III), ESC detects 11% (TP) extreme positive posts and 6% (TN) extreme negative posts from the set of original positives and negative posts. ESC also incorrectly classifies 3% posts (FP + FN).

TABLE IV. T4SA RESULTS

	P	N	Neutral	Total
EP	**82707**	10206	37422	130335
EN	1336	**8206**	1110	10652
$\overline{E}$ + INC	287298	160638	591034	1038970
Total	371341	179050	629566	1179957

For T4SA dataset (Table IV), ESC classifies 22% (TP) as extreme positive posts out of set of original positives posts, while, 4% (TN) as extreme negative posts.

TABLE V. TURNTOISLAM AND ANSAR1 RESULTS

Datasets	EP	EN	$\overline{E}$ + INC	Total
TurnTo Islam	97952 (≈ 29%)	6086 (≈ 2%)	231300 (≈ 69%)	335328 (≈ 100%)
Ansar1	9834 (≈ 33%)	1188 (≈ 4%)	18470 (≈ 63%)	29492 (≈ 100%)

Finally, the results in Table V show approximately 29% and 33% of extreme positive posts, which can indicate that ESC performs well on these two datasets to detect extreme positive polarity. Moreover, the total number of extreme positive posts is quite higher than the total number of extreme negative posts.

Analysis of our results, we concluded that our unsupervised and language-independent methodology presents good indicators for detecting extreme positive posts. In Table VI, we have a complete evaluation of our methodology as the objective is to detect extreme posts on original posts. The evaluation of our methodologies focuses on adapt conventional performance measures: **Recall** is the proportion of positive cases that were correctly detected, in our case, is the proportion of extreme posts that were correctly detected; **Precision** is the proportion of predicted positive cases that were correct, in our case, is the proportion of predicted extreme posts that were correct; $\mathbf{F}_1$ **score** is the harmonic mean of the precision and recall, where an F_1 score reaches its best value at 1 (perfect precision and recall) and worst at 0; **Accuracy** is the proportion of the total number of correct predictions, in our case, is the proportion of the total extreme posts of correct predictions.

TABLE VI. INDICATORS OF ALGORITHM EFFICIENCY

	RT-polarity	Sentiment 140	T4SA
Recall_{EP}	**91%**	**95%**	**98%**
Recall_{EN}	21%	50%	45%
Precision_{EP}	59%	65%	**89%**
Precision_{EN}	65%	**92%**	**86%**
$\text{F}_1\ \text{Score}_{EP}$	72%	**77%**	**93%**
$\text{F}_1\ \text{Score}_{EN}$	32%	65%	59%
Accuracy	60%	72%	**89%**

Table VI shows the overall status of acquired results are quite satisfactory, where in some evaluation measures, for certain datasets, we have more than 90%. The results of Sentiment 140 and T4SA are really prominent, where none of the values is less than 45%. However for RT-polarity, there appear some low values on negative terms, i.e., recall and F_1 score for EN. Besides, high precision for datasets may conclude choosing the correct polarity. The measure of accuracy for all data resources is equal to or greater than 60% indicating the overall performance of the approach is better. However, as we mentioned before, the results depict very good

 ISBN: 978-1-61208-804-4

status for detecting extreme positive posts, particularly in the case of T4SA dataset.

It is worth mentioning that we did not perform the calculation of recall, precision, f1 score and accuracy for Ansar1 and TurntoIslam due to these datasets' original settings. In these datasets, posts are organized as threads that include detailed metadata, e.g., name, age, date, etc. and also indicate topic under discussion on the forum. Since these datasets are directly referred to as 'Correction of common misconceptions about Islam', there is a possibility of radical participants may occasionally show their support for extremist fundamentalist militant groups. Hence, we select and perform the experiments on these two datasets due to the high probability of finding extreme sentiment posts.

We also identify a few issues and limitations during experimentation. One of the limitations with our system is not being able to distinguish an extreme positive term(s) expressed with negation, e.g.,*Dems **not Happy** with their nominee*. The system considers **happy** as an extreme positive term, but the presence of negation changes the meaning. Besides, long written posts with more positive and negative terms also impact our tool's performance due to sentence complexity as in the case of TurntoIslam and Ansar1 datasets. The appearance of emojis in posts appeared another issue, and the system can not handle this for now. These are specific issues which we will address in the future. Regardless, the preliminary results obtained from experiments appeared quite encouraging and satisfying for most of the datasets and our system able to detect extreme positive and negative terms having polarity.

VI. Conclusion and Future Work

In this paper, we demonstrated an unsupervised and language-independent approach for the detection of people's extreme sentiments on social media platforms. Our approach is based upon defining extreme polarity for terms and generating extreme sentiments lexicon by relying upon two standard lexical resources, i.e., *SENTIWORDNET 3.0* and *SenticNet 5*. We experimented with our system on five different social networks and media data lexicons to check the performance, effectiveness, and efficiency of the system. We provided a standard lexicon that can also be useful other researchers to exploit it for sentiment analysis studies as well as for anti-extremism authorities to identify people's extreme sentiments, e.g., on social networks and can prevent violent extremism.

As an extension of the research presented in this study, we want to improve and handle the issues and limitation identified during the experiment to make our system more efficient, for this we will apply linguistic tools in our approach, for example, to detect negation [18][19] (*he is happy is different from he is not happy*), to detect expressions with intensity [20] (*he likes it is different from the likes a lot*). Relatively in the context of intensity, we believe that it is also related to the expression of extreme feelings on the part of people. It is still our intention to apply word embeddings techniques to extend the lexical of extreme sentiments [21]. For future research, we are planning to enhance our system using natural language processing techniques to detect radical elements on social media and networks to predict a radical event(s).

Acknowledgment

This work is supported by National Founding from the FCT - Fundação para a Ciência e Tecnologia, through the MOVES Project - PTDC/EEI-AUT/28918/2017.

References

[1] S. Baccianella, A. Esuli, and F. Sebastiani, "Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining." in Lrec, vol. 10, 2010, pp. 2200–2204.

[2] E. Cambria, S. Poria, D. Hazarika, and K. Kwok, "Senticnet 5: Discovering conceptual primitives for sentiment analysis by means of context embeddings," in Thirty-Second AAAI Conference on Artificial Intelligence, 2018.

[3] MOVES, "Extreme Sentiment Lexicon," http://moves.di.ubi.pt/extremesentilex.html, retrieved: August, 2020.

[4] B. Liu, "Opinion mining and sentiment analysis," in Web Data Mining. Springer, 2011, pp. 459–526.

[5] S. Almatarneh and P. Gamallo, "A lexicon based method to search for extreme opinions," PloS one, vol. 13, no. 5, 2018.

[6] S. Ahmad, M. Z. Asghar, F. M. Alotaibi, and I. Awan, "Detection and classification of social media-based extremist affiliations using sentiment analysis techniques," Human-centric Computing and Information Sciences, vol. 9, no. 1, 2019, p. 24.

[7] B. Pang, L. Lee et al., "Opinion mining and sentiment analysis," Foundations and Trends® in Information Retrieval, vol. 2, no. 1–2, 2008, pp. 1–135.

[8] N. Friedrich, T. D. Bowman, W. G. Stock, and S. Haustein, "Adapting sentiment analysis for tweets linking to scientific papers," arXiv preprint arXiv:1507.01967, 2015.

[9] A. Go, R. Bhayani, and L. Huang, "Twitter sentiment classification using distant supervision," CS224N Project Report, Stanford, vol. 1, no. 12, 2009, p. 2009.

[10] L. Vadicamo, F. Carrara, A. Cimino, S. Cresci, F. Dell'Orletta, F. Falchi, and M. Tesconi, "Cross-media learning for image sentiment analysis in the wild," in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 308–317.

[11] "T4sa," http://www.t4sa.it/#dataset, retrieved: August, 2020.

[12] I. Smeureanu et al., "Applying supervised opinion mining techniques on online user reviews," Informatica Economică, vol. 16, no. 2, 2012, pp. 81–91.

[13] B. Pang and L. Lee, "Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales," in Proceedings of the ACL, 2005.

[14] U. o. A. Artificial Intelligence Lab, Management Information Systems Department, "Turn to islam forum dataset." University of Arizona Artificial Intelligence Lab, AZSecure-data, Director Hsinchun Chen, 2013.

[15] "Ansar1 forum dataset." University of Arizona, Artificial Intelligence Lab, AZSecure-data, Director Hsinchun Chen, 2013.

[16] P. Gonçalves, M. Araújo, F. Benevenuto, and M. Cha, "Comparing and combining sentiment analysis methods," in Proceedings of the first ACM conference on Online social networks, 2013, pp. 27–38.

[17] F. N. Ribeiro, M. Araújo, P. Gonçalves, M. A. Gonçalves, and F. Benevenuto, "Sentibench-a benchmark comparison of state-of-the-practice sentiment analysis methods," EPJ Data Science, vol. 5, no. 1, 2016, pp. 1–29.

[18] E. Blanco and D. Moldovan, "Some issues on detecting negation from text," in Twenty-Fourth International FLAIRS Conference, 2011.

[19] W. Sharif, N. A. Samsudin, M. M. Deris, and R. Naseem, "Effect of negation in sentiment analysis," in 2016 Sixth International Conference on Innovative Computing Technology (INTECH). IEEE, 2016, pp. 718–723.

[20] S. M. Mohammad and F. Bravo-Marquez, "Emotion intensities in tweets," arXiv preprint arXiv:1708.03696, 2017.

[21] K. W. Church, "Word2vec," Natural Language Engineering, vol. 23, no. 1, 2017, pp. 155–162.

 ISBN: 978-1-61208-804-4

Language-Independent Approaches to Detect Extremism and Collective Radicalisation Online

Sebastião Pais
Computer Science Department
NOVA LINCS and UBI
Covilhã, Portugal
Email:sebastiao@di.ubi.pt

Irfan Khan Tanoli
Computer Science Department
University of Beira Interior
Covilhã, Portugal
Email:irfan.khan.tanoli@ubi.pt

Miguel Albardeiro
Computer Science Department
University of Beira Interior
Covilhã, Portugal
Email:miguel.albardeiro@ubi.pt

João Cordeiro
Computer Science Department
University of Beira Interior
Covilhã, Portugal
Email:jpaulo@di.ubi.pt

***Abstract*—Due to lack of regulation, a lot of user-generated content reflects more closely the offline world than official news sources. Social media have become attractive platforms for anyone seeking independent information. Text mining and knowledge extraction are also crucial issues, in particular, directed toward social media and micro-blogging. The automatic identification of extremism and collective radicalisation require sophisticated Natural Language Processing (NLP) methods, text mining techniques, and resources, especially those dealing with opinions, emotions, or sentiment analysis. The area of understanding and detecting extremism and collective radicalism on social media has a connection with sentiment analysis and opinion mining. The main focus of this work is to provide the state-or-art to identify extremism and collective radicalisation on social networks based on user's sentiment analysis, and to develop an unsupervised and language-independent approach by relying on statistical and probabilistic methods. This paper discusses few important case studies related to the roots of radicalism, extremism detection, and terrorism detection using sentiment analysis and present machine learning models, and how these methodologies can be exploited to develop our desire system.**

Keywords–Natural Language Processing; Social Media; Extremism; Collective Radicalisation; Sentiment Analysis

I. INTRODUCTION

In the last few years, the advent of micro-blogging services has been impacting people's mind, communication, behavior, and activities conduct. It is due to several factors, including the use of convenience, and the lack of regulation, and the vast amounts of user-generated contents that reflect more closely the offline world than the official news source. Social media and network have become an attractive platform for anyone seeking independent information and eventually, more authentic news. Recently, we have assisted the news about the 'Yellow vests' or in French 'Gilets Jaunes' [1]. It began as a pacific manifestation, but later few extremist groups have joined the manifestations that made it a violent protest as was in the news: 'Absence of the progress of the movement, inexperience of the demonstrators, the action of extremist groups, the forces of higher duties' [2]. In Portugal, we have witnessed some radical events as few people were protesting against the actions taken by the police on a tough neighborhood referred as 'Bairro da Jamaica'; there were few from an extremist group protesting against the politicians, violently [3].

In social networks such as Facebook, Twitter, and Youtube, each cluster of posts, videos or tweets focus on a burst topic that may constitute a potential threat. However, the majority of clusters are harmless and represent casual, conventional or expressive crowds as well as noisy data [4]. To identify acting or protesting crowds on social networks, it is necessary to understand the tone of language usage, e.g., slang, abusive, jargon, formal, respectful etc., present in each cluster as well as its network activity. Ultimately, a crowd is characterized by its dominant emotions; it is the level of interaction and shared contents. The work in [5] discussed the technique that can be used to analyze the tweeter contents and detect the event related to the contents.

Users use social networks for various purposes. Unfortunately, few use it to spread distorted beliefs, negative opinion about things like spreading terrorism, extremism, and radicalism [6]. Since mid-2015 Twitter has already deleted more than 125,000 accounts that were somehow linked to terrorism [7]. Researchers focus Twitter for sentiment analysis due to few particular reasons: Twitter's popularity as enormous numbers of people continuously tweet on Twitter related to various topics. These topics could be political, about sports, religious, marketing, people's opinions or friend's conversations. Being an updated huge repository of facts, opinions banter and other minutiae, Twitter has received significant attention from business leaders, decision-makers, and politicians.

In this study, we aim to provide a theoretical review related to extremism and collective radicalisation detection. Extremism is a vague term that can be undermined in three different contexts [8]: Taking a political idea to its limits, regardless of unfortunate repercussions, impracticalities, arguments, and feelings to the contrary, and with the intention not only to confront but also to eliminate opposition; intolerance towards all views other than individual own; adoption of means to political ends which disregard accepted standards of conduct, in particular, which show disregard for the life, liberty and human rights of others. radicalisation is a process by which an individual or group comes to adopt increasingly extreme political, social, or religious ideals and aspirations.

With our understanding, it is clear that extremism and collective radicalisation has a direct connection with people's sentiments and opinions. There are many barriers to understand extremism and collective radicalisation on the social network. Among these challenges, one big challenge is to differentiate between the users commanding this process and the users talking about it. Hence, the main goal of this study is to propose an effective system to detect extremism and collective radicalisation on social media based on sentiment analysis. To do so, our focus is on statistical and probabilistic methods that can be used to develop an unsupervised and language-

 ISBN: 978-1-61208-804-4

independent system.

The main contribution of this study is significant for many reasons. First, it covers three different research areas, i.e., extremism, collective radicalisation, and sentiment analysis, and to provide a better understanding related to these areas. Second, instead of just providing brief details of different works for these areas, we analyzed three essential case studies in-depth to help readers understand different approaches that have been used for these fields. This angle could also help the researchers who are familiar with specific techniques dedicated to extremism and collective radicalisation, to exploit and choose the appropriate one for their work. Third, this study also present supervised, unsupervised, and language-independent approaches proposed for extremism, collective radicalisation, and sentiment analysis with brief details of the algorithms and their originating references. This can help us to develop an efficient unsupervised and language-independent system for extremism and collective radicalisation detection. Finally, the survey is enhanced with models related to everyday NLP tasks, and we discuss which one can be exploited for our desire system.

In the following sections, we deeply review three different works for extremism, radicalisation, and sentiment analysis detection. First, we present a work that discusses the roots of radicalism. Next, we analyze a work proposed for terrorism detection based on sentiment analysis. Finally, another work for sentiment detection on Twitter using hashtags. In section III, we present a few proposed methodology. Section IV overview few standard Machine Learning (ML) models. In the end, we provide a conclusion and a future direction for our ongoing work.

II. Machine Learning Classifiers for Extremism and Collective radicalisation

radicalisation involves a movement towards the support or representation of radical behavior(s). Radical behavior can be viewed as 'when it serves a specific purpose; it undermines other goals that are important to most people' [9]. At the same time, collective radicalisation is defined as a collective inter-group process. People are not radicalized on their own, but rather as part of a group and through the socially constructed reality of their group by gathering people on the streets to show their motive and protests against specific entity or entities. However, sometimes these protests become violent.

A. The roots of radicalism

Fernandez et Al [10] proposed an innovative NLP and Collaborative Filtering (CF) based approach for detecting radicalisation on social networks, The different roots of radicalisation, i.e., micro-roots, meso-roots, and macro-roots, are captured [11], and each user is represented through keyword-based vector description. The approach presented in [10], is sufficient enough to detect and predict radicalism. On social networks, the user either creates or posts the contents or shares other people's contents; the authors assumed that micro-roots or meso-roots are captured from the user's shared or created contents. While macro-roots are captured that are external to the given social network (links/URLs) and from other websites or other social networks, and videos, etc. [10].

In [10], the authors used keyword-based vectors that include the user's post(s). These vectors represent micro-roots and meso-roots influences over users, and they are transformed into n-grams (uni-grams, bi-grams, and tri-grams) [10]. Next, the value of each n-gram in the micro-root user's vector is computed as the frequency of the n-gram in the user's post, and normalized by the number of posts. In the macro-roots influences case, an automatic data scrapping over the URLs included on a macro-roots vector is performed by automatically parsing the HTML, and extracting the title and description of the websites. Giving the set of n-grams obtained after pre-processing, all the links defined the macro-roots of the user. The value of each word in a macro-roots of user's vector is computed as the frequency of the n-gram in all user's share URL entries and normalized the number of URL [10].

The authors in [10] further collected and integrated existing lexicons, i.e., ICT Glossary, Saffron Experts, Saffron Dabiq Magazines, Rowe, and Saif to create a single lexicon containing a more comprehensive set of terms and expression that shows radicalisation terminology. To mitigate lexicon merging issues, the authors first remove incorporated syntactic variances of each term, i.e., lowercase, uppercase, apostrophes and hyphens removal, diacritics removal. Then, if two terms are present in both lexicons, they are merged and added as one unique entry in the final lexicon. The final lexicon comprises 305 entries, including expression, terms, and variances [10].

The authors in [10] also compute the radicalisation influence of different roots over the user determining cosine similarity between the micro-roots and the meso-roots vectors and the generated lexicon. It is not possible to compute cosine similarity for macro vectors due to many sites were already disabled, and it was not possible to collect URLs information. Next Collaborative Filtering (CF) strategies are used to develop an automatic prediction about user's interests by collecting numerous user's preference information, using the following two steps: Search for such users that have a similar rating pattern to other users for whom the prediction is made; use the ratings of user found in the previous step to compute the predictions for the active user. Two publicly available datasets from the Kaggle Data Science Community are used to study radicalisation. One of the datasets contains 17,350 tweets from 112 pro-ISIS accounts. The second dataset is created as the opposite of the previous one. It contains 122,000 tweets from 95,725 users collected on different days.

The corresponding results in [10] show the effectiveness of the proposed algorithms for detecting and predicting the influence of radicalisation with up to 0.9 F-1 of the measurement for detection and between 0.7 and 0.8 precision is obtained for the prediction. The work concluded as the presentation of a computational approach to the detection and prediction of the influence of radicalisation to which a user is exposed, based on the concept of 'roots of radicalisation' identified in social science models.

Detecting radicalisation online faces several challenges. From an accuracy perspective, most of the 'ground truth' datasets used in different works are not reliably verified. Many of these datasets, e.g., [12], [13], [14], are collected using keyword sets, with users tweeting those words would be regarded as in the 'radicalized' set. It is also possible that users who use radicalisation terminology in their tweets may sometimes report on some event (e.g., 'Islamic State is hacking a Swedish radio station') or share harmless religious rhetoric (e.g., 'If you want to talk to Allah, pray, if you want Allah to

speak to you read the Quran').

There is still a need to use a gold standard dataset to train recognition models. This dataset must be manually checked by experts to ensure that the cases are real positives and/or real negatives not false positive and/or false negative. One source of manually identified radical accounts is Ctrl-Sec [15], where volunteers report ISIS propaganda on social media. This initiative claimed to have closed more than 200,000 Twitter accounts in three years [10]. While these are critical mechanisms to encounter radicalisation online, still the accounts are closed quickly once identified as radical means that the data cannot be further collected and analyzed to train automated methods.

From a policy perspective, radicalisation is not a crime. Radicals of all religions and ideologies can freely express their beliefs and practice their freedom of expression. However, adopting or preaching violent radicalisation is a crime [10]. Therefore, considering the above-presented work, our finding is that online radicalisation detection needs a multi-pronged approach(es). Researchers need to focus this research area and developed/proposed more constructive approaches to come up with the best and the most effective ones to prevent society from radicalisation.

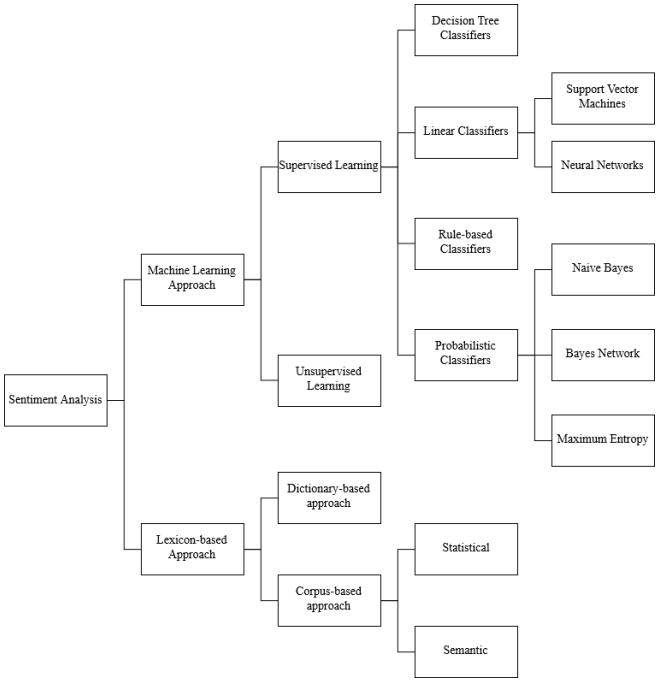

Figure 1. Sentiment Classification Techniques Used in SA [16]

B. Sentiment Analysis

The sentiment, polarity, and opinion mining or sentiment analysis deal with direction-based text analysis, i.e., text with opinions and emotions [17]. 'Sentiment Analysis or opinion mining is the computational study of people's opinions, attitudes, and emotions toward an entity. The entity can represent an individual, event, or topic [16]. Opinion Mining (OM) is not the same as Sentiment Analysis(SA). OM starts by extracting and analysing the opinion about something while SA is more about the sentiment that something causes on people, usually expressed in the text, like or share tweets or Facebook posts [16]. SA can also be observed as a type of text classification but deals with subjective statements that are harder to classify [18].

SA can be viewed as a classification process. It can be divided into three levels: document-level, sentence-level, and aspect-level. At the document-level, SA classifies an opinion document according to its polarity (negative or positive). The entire document should be considered as the primary unit of information. SA is an expressed feeling classified in each sentence at the sentence level. First of all, it must be determined whether the sentence is subjective or objective. If the sentence is subjective, SA determines the polarity of the opinion (positive or negative) [16]. However, using these two levels does not provide the necessary details on all aspects of the entity that are needed in many applications. The aspect-level classifies, taking into account the specific aspects of the entities. Firstly, it is required to identify the entities and their aspects. The opinion holders can give different opinions on different aspects of the same entity [16].

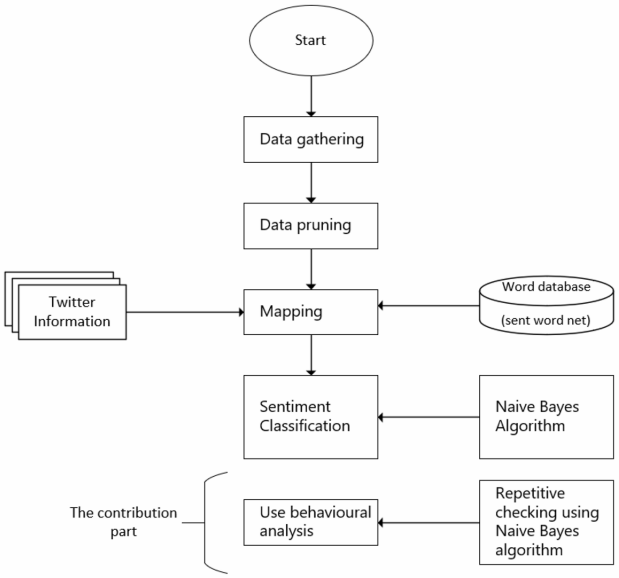

Figure 2. Proposed system development diagram [6]

C. Terrorism Detection using Sentiment Analysis and Machine Learning

In this section, we review an approach based on Sentiment Analysis (SA) for detecting terrorism on social networks. According to [6], social networks have recently been the most crucial channel for people to interact and share ideas. People choose to express their opinion(s) on a particular subject, news, or event due to the rapid spread of information on social media. For example, it is easier to reach more people online and influences the choices of potential users about the top trending topic on Twitter. Contrary, it is also easy for extremist groups and its members to recruit the people sharing the same ideology and views on social media and networks. In 2015 more the 250,000 accounts were linked to terrorism and later the accounts have been deleted and disabled [7].

Existing SA approaches aim to find a tweet that may or may not lead to an extremist user. These approaches are still not practical enough for specific reasons like ambiguity in

 ISBN: 978-1-61208-804-4

tweets, synonymy in tweets, use of emotions in tweets, etc. It is also quite familiar for humorists to make fun of people or even joke about terrorism on Twitter just for fun. A big challenge for existing approaches is to classify if a tweet is a real threat or not. The two most general approaches to SA are the lexical and Machine Learning (ML) approach [17]. These two approaches are further sub-classified into more approaches as shown in Figure 1.

The main objective of the proposed work in [6] is to present a system for improvising current techniques for SA through ML to detect terrorist acts on Twitter more accurately. The general structure of the system is shown in Figure 2. The novelty of this research is having divided the sentence into positive, negative, and neutral categories. Then all three categories are compared to the previous sentence for a given account holder based on the sentiment score for the latest and previous sentence. This means a specific account holder's tweet history in each of the categories is extracted, and the sentiment value is calculated. Later, the sentiment score from the above statement will be compared with the sentiment value of the latest identified sentiment. The system is based upon the ML approach rather lexical-based [6]. For better understanding, the functions of each component are presented below.

- **Data gathering:** The target source for data collection is Twitter due to its popularity, and even it is used for communication about terrorism. Even compared to Facebook and other popular blogs, Twitter has recorded more significant problems related to acts of terrorism. The data are gathered from the Twitter streaming API. For this work [6], the authors used keywords e.g., ISIS, Bomb, etc. to obtain data related to terrorism. If tweet(s) match(es) the user's criteria directly, these tweets are sent directly to the user in JSON format, a JavaScript object notation.
- **Data pruning:** After data collection, it is preprocessed for normalization. Removal of URLs, @tags, hashtags, uppercase and lowercase letters, misspellings, etc. are some examples of data pruning.
- **Mapping:** SentiWordNet [19] is used as a dataset for mapping. It is made up of thousands of English words that have a positive or negative score for each word. Tweets are compared and computed with SentiWordNet. Since the word alone is not enough to make a decision, the total score is calculated based on the sentence context.
- **Sentiment Classification:** Twitter sentences are classified into positive, negative, or neutral class for Sentiment Classification. Naïve Bayes is used because it is commonly used for SA. Bayes theorem is used to predict the probability that a given set of features will belong to a particular label. The Naïve Bayes classifies the statement as positive, negative, or neutral based on the result of the sentiment assessment.
- **User Behavioral Analysis:** User Behavioral Analysis is carried out using Snapbird tool [6] to track the previous tweets of a particular user. When tweets become classified to their polarity based on the sentiment score, all three classes are checked repetitively. For double-checking, tweets on each category are compared with tweets history. The purpose of this repetitive checking is to find better results on the understanding if tweets are leading towards terrorism or not [6].

If the score is negative after re-checking and results in the same class, it can be concluded that the account holder may lead to acts of terrorism. The purpose of reviewing the user's previous tweets is to analyze the user's tweet patterns. As mentioned above, the user can be a humorist or just joke about terrorism, so the pattern of user tweets can be related to jokes. However, if the user seriously discussing to support terrorism and wanted to convince or influence other readers about terrorism support, then that user is categorized in the terrorist category [6]. The use of Naïve Bayes has been proven and had the potential to be implemented [6]. Hence, Bayes theorem is being applied to predict a class for any giver text from tweets. The authors [6] applies Bayes theorem to predict the class of any tweet using the Equation 1.

$$P(label|features) = \frac{P(label)P(features|label)}{P(features)} \quad (1)$$

Where *P(label)* is the class (i.e., positive/negative/neutral) of the tweets while *P(features)* is the tweet. *P(label—features)* is the result of the application of the techapprichnique. By using 1, we get 2:

$$P(positive|tweet) = \frac{P(positive)P(tweet|positive)}{/P(tweet)} \quad (2)$$

The process has to be repeated for all three categories (positive/negative/neutral). Finally, the highest-ranked class is chosen to label the document [6]. The initial results show that there are more than 50 words indicated as terrorism keywords, e.g., jihad, bomb, radical Al-Qaeda etc. Among the top eight words in the list are terrorism, jihad, bomb, radical, Abu Sayyaf, ISIS and extremist [6].

To conclude it, Naïve Bayes algorithm-based system is proposed for terrorism detection on Twitter. Naive Bayes approach appears as a medium accuracy comparing with support vector machine and neural network. To further enhance the accuracy of Naive Bayes, the element of user behavioral analysis has been proposed to embed into the algorithm after sentiment classification process have been performed.

D. Learning with Hashtags

Here we present an interesting supervised approach [20] based upon for learning hashtags, hashtag patterns, and phrases associated with five emotions: AFFECTION, ANGER/RAGE, FEAR/ANXIETY, JOY, and SADNESS/DISAPPOINTMENT. It is usual for users to express an emotional state using hashtags (e.g., **#inlove, #hatemylife**) on Twitter. Few hashtags consist of a single word like **#Faith**, others composed of multiple words (e.g., **#FaithinhumanityRestored**), or it can even be a creative spelling, e.g.,**sk8** or **cantwait4tmrw**. As these hashtags are continuously created through various infinite open combinations, it is not easy to identify such hashtags using sentiments or emotions lexicons.

In [20], a bootstrapping framework for learning emotion hashtags is proposed as described above, which has been further improved to learn more general hashtag patterns. The emotion phrases are extracted from the hashtags and the hashtag patterns to classify contextual emotions. The first step is to find the common prefix in the hashtags. For example, **#Angryatlife** and **#Angryattheworld** have the same prefix

angry at that predicts ANGER emotion. As a result, certain hashtags generalize into hashtag patterns that match hashtag with the same prefix. A critical challenge here is to identify the same prefixes in hashtags with different emotions that could lead to incorrect emotion. For example, #anger pattern generally points out angry tweets. However, hashtag as #angrybirds refers to a game, not the emotion of the writer. AFFECTION can be determined as 'I love you' followed by the person (e.g., #loveyoufather). This can be related to JOY in other contexts (e.g., #loveyoulife). The authors use the probability estimates to determine certain hashtag patterns that are reliable indicators to an emotion [20]. Also, if there is a negation, it can toggle the polarity of the tweet (e.g., **not love life** can suggest SADNESS instead of JOY).

1) Learning Hashtags: The authors used and collapsed parrot emotion taxonomy [20] into only five emotions that occur more often in tweets and are easily distinguishable from each other, i.e., AFFECTION; ANGER/RAGE; FEAR/ANXIETY; JOY SADNESS/DISAPPOINTMENT. Adding another class, 'None of the above' that does not express any emotion. For each one of the five classes, the five common identified hashtags are strongly associated with the emotion and these hashtags are used as seeds.

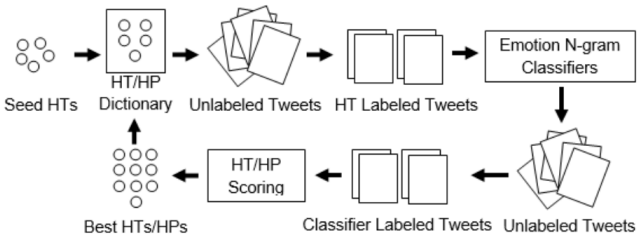

Figure 3. Bootstrapped Learning (HT = hashtag; HP = hashtag pattern) [20]

The general architecture of the framework is shown in Figure 3. The process starts with tweets containing the seed hashtags and marks with the appropriate emotion. There are 323,000 tweets received from at least one of the seed hashtags. Additionally, more than 2.3 million untagged tweets are collected using Twitter's streaming API that contains at least one hashtag (an average of 1.29 hashtags per tweet and 3.95 tweets per hashtag). Tweets are preprocessed with the CMU tokenizer and normalized against the case. The tagged tweet is then used to train a series of emotion classifiers. A logistic regression classifier is trained for each emotion class [20].

Each emotion classifier applies to unlabeled tweets. For each emotion e, the obtained tweets are classified as e, and the hashtags are extracted from such tweets to create a candidate pool of hashtags H_e for that emotion e. Next, the candidate hashtag h is assigned a score by calculating the average probability of the same emotion e obtained from the logistic regression classifier for the entire tweet containing the candidate hashtag h. From the untagged tweets, all tweets with one of the learned hashtags are then added to the training instance, and the process continues. To reduce the number of potential candidates, hashtags that appear less than ten times, those with a single character, and those that appear more than twenty times are discarded.

2) Learning Hashtag Patterns: In this phase, the hashtag is expanded into a sequence of words using an N-gram based word segmentation algorithm [21]. The Prefix Tree data structure is used for the representation of all possible prefixes of the expanded hashtag. Then, the tree is traversed, and all possible prefixes are considered as candidates of hashtag patterns. Later, each pattern is assigned a score as the way it is done with hashtags. The authors calculate the average probability of classifier, and for each emotion class, ten hashtag patterns with the highest scores are selected. For unlabeled tweets, all tweets with hashtags are added, which match one of the learned hashtag patterns to the training instances, and the bootstrapping process continues.

3) Creating Phrase-based Classifiers: The final type of emotion classifier aims to acquire is emotion phrases. Right at the end of the bootstrapping process, the word segmentation algorithm is applied to all hashtags and hashtag patterns to separate them into phrases (e.g., #lovemylife $\rightarrow$ 'love my life'). It is assumed that the obtained phrase has the same emotion as the original hashtag. Nevertheless, it will have low precision due to the presence of a phrase yields, and the surrounding context must also be taken into account [20]. Finally, a logistic regression classifier is trained for each emotion that classifies a tweet about its emotion based on the presence of learned phrases for the emotion, as well as a context window of size six around the obtained phrase, three for each side of the phrase.

The results in [20] show that the learned set of emotional indicators causes a substantial improvement in F-scores, ranging from + % 5 to + % 18 to basic classifiers, The result also showed that the combination of the emotion indicators learned with an N-gram classifier in a hybrid approach significantly improves performance in 5 emotion classes. This work [20] proposed three types of emotional indicators. The approach is categorized as weakly supervised monitored bootstrapping: hashtags, hashtag patterns, and phrases. Once the emotion indicators are trained using hashtags, and the hashtags can gain form in any language. Moreover, these indicators can also be applied to any language as a language-independent method since it does not depend on a specific corpus.

III. Supervised, Unsupervised and Language-Independent Approaches

In the previous Section II, we discuss three different studies in detail for extremism, collective radicalisation detection and detecting sentiment on Twitter using hashtags. The purpose of this study to create a state-of-the-art that aims to understand extremism and collective radicalisation, sentiment analysis, and to develop an unsupervised and language-independent system using machine learning models. To achieve it, we will rely on probabilistic approaches that can be applied to any language, or even in a mix of languages. On the unsupervised part, we aim to create a system that can detect extremist or radical tweets by itself without much human intervention. In this section, we overview a few NLP approach that we can use to achieve our desire goal.

A. Supervised Natural Language Processing Approach

Supervised machine learning involves labeling or commenting on a series of text documents with examples of what the machine is looking for and how it should interpret this aspect. Researchers use datasets to train a statistical model

that is then given unlabeled text for analysis. Later, more extensive or better datasets can be used to retrain the model as it learns more about the documents it analyzes. For example, one can deploy a supervised learning system to train a model on Twitter tweets and then use it for various purposes. Several methods have heretofore been used in a supervised approach, e.g., Support Vector Machines, Bayesian Networks, Maximum Entropy Conditional Random Field, Neural Networks/Deep Learning. An interesting supervised approach based on word embedding for the sentiment classification of Twitter is shown in [22]. We aim to develop a similar approach to detecting extremism and collective radicalisation based on sentiment analysis (SA). For SA, it is essential to examine them at different levels for extremism and radicalisation detection. A user may be talking on a topic that represents extremism but is not an extremist. For example, as ISIS became more active on social networks, some accounts unrelated to extremism groups were temporarily deleted from Twitter. Hence, it is essential to identify an extremist person or someone who is involved in a radicalisation process.

B. Unsupervised Natural Language Processing Approach

An unsupervised approach refers to a system where training inputs are not necessary to discover the target point of the learning. The system needs to train itself without human supervision and intervention or with human intervention only if there is a need to add or change the functionalities. Topic modelling is another core task of NLP. Let say you have a bunch of books; you want to categorize them according to (of course) the topics they talk about it, how you solve the challenge without reading all the books. An unsupervised approach discussed in [23], which uses matrix factorization to extract latent (or hidden) topics from the text; this approach is unsupervised as there is no model trained and tested, one just set the parameters in a trial and error to achieve the best results. The work discussed in subsection II-D, a weekly supervised system, and a supervised language-independent probabilistic is proposed in [24] for twitter sentiment analysis. Therefore, our focus is to use probabilistic methods to develop an unsupervised language-independent system for user's sentiment analysis.

C. Language-Independent Approach

A language-independent approach refers to a single system that is applied to different natural languages (e.g., English, Chinese, German, etc.), the results keep being satisfactory and the experiment values represent the reality in a viable way. Once an algorithm has been developed for a specific language, the question arises, it can be trivially extended to another language; All it is needed an adequate amount of training data for the new language. It is a virtue. However, the typical approach to developing language-independent systems is to avoid using any particular linguistic knowledge in their development. The approaches presented in [25][26][27] are a few examples of such an approach.

Hence, we aim to propose a language-independent system beside unsupervised. For example, if we collect tweets from the streaming API, having as a criterion the geo-localization as Portugal, the tweet would not only in Portuguese but probably on other languages. Therefore, we would like our system to analyse the tweet regarding the language it is written.

IV. MODELS

The current known models for solving NLP problems are based on Supervised Machine Learning (SML). The basic idea behind SML models is to follow automatically induces rules from training data. The most common ML models commonly used to resolve ambiguities in language knowledge with the main tasks of NLP are Hidden Markov Model (HMM), Conditional Random Fields (CRF), Maximum Entropy (MaxEnt), Support Vector Machines (SVM), Decision Trees (DT), Naïve Bays (NB) and Deep Learning (DL) [28]. Apart, the following models also explain possible ML techniques in [29] that can also be considered for the development of our desire system:

Naïve Bayesian: Naïve Bayesian (NB) classifier is constructed using Bayes' theorem with assumptions of independence between predictors. A NB model is easy to develop without complicated iterative parameter estimation, which makes it particularly useful for huge big datasets. Despite its simplicity, the NB classifier often does surprisingly well and is widely used because it often outperforms more sophisticated classification methods [29]. The classifier is stated as:

$$P(A \mid B) = \frac{P(B \mid A)\, P(A)}{P(B)} \tag{3}$$

Where $P(A)$ is the prior probability of B, $P(A|B)$ is the conditional probability of A, given B called the posterior probability, $P(B|A)$ is the conditional probability of B given A and $P(B)$ is the prior probability of $B4$.

The NB classifier is based on the assumption of conditional class independence. If conditional class independence is assumed, the effect of an attribute value on a particular class is independent other attributes values [28]. The contribution of Naïve Bayes technique in computational linguistic is minimal. Recently, few research works reported based on Naïve Bayes technique for NLP tasks are [30][31][32] respectively.

Neural Networks: The biological neurons of brain structures inspire Neural Networks (NNs). Individual neuron models can be combined into several networks made up of many individual nodes, each with their variables. These networks have an input layer, an output layer, and one or more hidden layers. Hidden levels provide connectivity between entrances and exits. The network can also receive feedback using the result variables as input to the pre-processing nodes [29]. A network of interconnected functional elements, each with several inputs/one output as specified in equation 4:

$$y(x_1, \ldots, x_n) = f(w_1x_1 + w_2x_2 + \ldots + w_nx_n) \tag{4}$$

w_n, x_n are parameters of equation, f is the activation function of equation 4, crucial for learning that *addition* is used for integrating the inputs.

K-Nearest Neighbor: In the K-Nearest Neighbor (KNN) model, there is no learning phase as the training set is used every time a classification is performed. The NN search, also known as an approximate search, similarity search, or closest point search, is an optimization problem to find the closest points in metric spaces. K nearest neighbor is used to simulate daily precipitation and other meteorological variables [29].

Decision Trees: The Decision Tree (DT) is one of the standard classification algorithms currently used in ML. The DT is a

new field of ML that involves the algorithmic acquisition of structured knowledge in forms such as concepts, decision trees, and discrimination networks or production rules [29].

Each of these models has its pros and cons. The NB model is quick to train and classify, but also it is assumed to be independent of features approach [29]. For NN, they are not sensitive to irrelevant properties in contrast to NB. NNs are manufactured as specialized hardware systems. This is also advantageous for network learning. On the contrary, this is too large a black box technique, and it is not probabilistic [29].

KNN is an appropriate model once we collected data from Twitter. Since the data can be quite noisy, this model is robust for noisy training data, even for a large amount of training data. On the other hand, the KNN value needs to be determined, which is not easy to identify. Furthermore, it has a high computation cost [29]. Finally, the TD that offers an easy way to understand and interpret calculations, and it can always be used with other decision techniques. As mentioned before, these techniques are supervised, but this can be an initial point, and to exploit one of the model to develop an unsupervised system.

Native unsupervised approaches generally *Lexicon based, Dictionary-based, and Corpus-based approaches*. *Lexicon-based approaches* use insights obtained on the ground of words polarity composing a sentence. With this approach, one can create a categorical polarity (Positive, Neutral, Negative), or one can calculate a score [33]. Two most famous lexicons are: Sentiwordnet [34] and SenticNet [35]. *Dictionary-based approaches* follow two main steps: a small amount of manually collected opinion words with known instructions; expand this set by searching the WordNet dictionary [36] for synonyms and antonyms. The newly found words are added to the seed list, and the next iteration starts. The iterative process stops when no more new words are found [37]. *Corpus-based approaches* rely on syntactic patterns in large corpora. This approach can generate opinion words with relatively high accuracy. Most of the corpus-based approaches need extensive labeled training data. This approach has a significant advantage compared to dictionary-based approaches as it can help find domain-specific opinion words and their orientations [38].

V. Conclusion

Social media have a significant role in the process of extreme ideas dissemination all over the world. People have the dissemination of similar information, which can lead to collective radicalisation and extremism. In this study, we discussed three different research areas, i.e., collective radicalisation, extremism, and sentiment analysis, and analysed three different approaches for their detection. Our aim of this study was to provide the state-of-art to construct an unsupervised and language-independent system for collective radicalisation and extremism detection using SA. To do this, we also presented a few supervised NLP models and discussed lexicon, dictionary and corpus based approaches that can be integrated to achieve this goal. The area of extremism and/or radicalisation does not have much previous work. However, there are few works based on SA classification. With our knowledge, this study is the first attempt to provide a depth review related to these research areas.

Furthermore, this study paper gave a generic structure and guidelines for developing a new unsupervised language independent-system for addressing radicalisation and extremism issue. This study intended to cover supervised, unsupervised and language-independent techniques in the context of NLP tasks to develop an efficient and effective system. Hopefully, this study will also guide students and researchers with essential resources, both to learn what is necessary to know and to advance further the integration of supervised and language-independent techniques with different machine learning models.

Acknowledgment

This work was supported by National Founding from the FCT- Fundacão para a Ciência e a Tecnologia, through the MOVES Project- PTDC/EEI-AUT/28918/2017 and by operation Centro-01-0145-FEDER-000019-C4 - Centro de Competências em Cloud Computing, co-financed by the European Regional Development Fund (ERDF) through the Programa Operacional Regional do Centro (Centro 2020), in the scope of the Sistema de Apoio à Investigação Científica e Tecnológica.

References

[1] L. Williamson, "Gilets jaunes: Anger of yellow vests still grips france a year on," 2019, URL: https://www.bbc.com/news/world-europe-50424469#:~:text=But%20three%2Dquarters%20of%20French,sausages%20over%20a%20small%20fire. [accessed: 2019-09-18].

[2] X. Crettiez, ""gilets jaunes": la violence, l'arme des bavards, est aussi celle des silencieux"," 2018, URL: https://www.bbc.com/news/world-europe-50424469#:~:text=But%20three%2Dquarters%20of%20French,sausages%20over%20a%20small%20fire. [accessed: 2018-12-04].

[3] C. Reis, "Polícia não vai deixar manifestação da extrema-direita chegar à sede do bloco de esquerda," 2019, URL: www.dn.pt/pais/policia-nao-vai-deixar-manifestacao-da-extrema-direita-chegar-a-sede-do-bloco-de-esquerda-10487434.html [accessed: 2019-01-25].

[4] H. Becker, M. Naaman, L. Gravano et al., "Selecting quality twitter content for events." ICWSM, vol. 11, 2011, pp. 442–445.

[5] J. S. Krumm, "Influence of social media on crowd behavior and the operational environment," Army Command and General Staff College Fort Leavenworth Ks School of Advanced Military Studies, Tech. Rep., 2013.

[6] B. S. Iskandar, "Terrorism detection based on sentiment analysis using machine learning," Journal of Engineering and Applied Sciences, vol. 12, no. 3, 2017, pp. 691–698.

[7] D. Yadron, "Twitter deletes 125,000 isis accounts and expands anti-terror teams," 2016, URL: https://www.theguardian.com/technology/2016/feb/05/twitter-deletes-isis-accounts-terrorism-online [accessed: 2016-02-05].

[8] R. Scruton, The Palgrave Macmillan dictionary of political thought. Springer, 2007.

[9] A. W. Kruglanski, M. J. Gelfand, J. J. Bélanger, A. Sheveland, M. Hetiarachchi, and R. Gunaratna, "The psychology of radicalization and deradicalization: How significance quest impacts violent extremism," Political Psychology, vol. 35, 2014, pp. 69–93.

[10] M. Fernandez, M. Asif, and H. Alani, "Understanding the roots of radicalisation on twitter," in Proceedings of the 10th ACM Conference on Web Science, 2018, pp. 1–10.

[11] A. P. Schmid, "Radicalisation, de-radicalisation, counter-radicalisation: A conceptual discussion and literature review," ICCT Research Paper, vol. 97, no. 1, 2013, p. 22.

[12] S. Agarwal and A. Sureka, "Using knn and svm based one-class classifier for detecting online radicalization on twitter," in International Conference on Distributed Computing and Internet Technology. Springer, 2015, pp. 431–442.

[13] M. Rowe and H. Saif, "Mining pro-isis radicalisation signals from social media users," in Proceedings of the tenth international AAAI conference on web and social media (ICWSM 2016), 2016, pp. 329–338.

 ISBN: 978-1-61208-804-4

[14] M. Ashcroft, A. Fisher, L. Kaati, E. Omer, and N. Prucha, "Detecting jihadist messages on twitter," in 2015 European Intelligence and Security Informatics Conference. IEEE, 2015, pp. 161–164.

[15] Twitter, "Ctrlsec," 2020, URL: https://www.bbc.com/news/world-europe-50424469#:~:text=But%20three%2Dquarters%20of%20French,sausages%20over%20a%20small%20fire.[accessed: 2019-09-18] [accessed: 2020-09-18].

[16] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," Ain Shams Engineering Journal, vol. 5, no. 4, 2014, pp. 1093–1113.

[17] H. Ismail, S. Harous, and B. Belkhouche, "A comparative analysis of machine learning classifiers for twitter sentiment analysis." Res. Comput. Sci., vol. 110, 2016, pp. 71–83.

[18] M. T. Khan, M. Durrani, A. Ali, I. Inayat, S. Khalid, and K. H. Khan, "Sentiment analysis and the complex natural language," Complex Adaptive Systems Modeling, vol. 4, no. 1, 2016, pp. 1–19.

[19] A. Esuli and F. Sebastiani, "Sentiwordnet: A publicly available lexical resource for opinion mining." in LREC, vol. 6. Citeseer, 2006, pp. 417–422.

[20] A. Qadir and E. Riloff, "Learning emotion indicators from tweets: Hashtags, hashtag patterns, and phrases," in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1203–1209.

[21] P. Norvig, "Natural language corpus data: Beautiful data," 2020, URL: http://norvig.com/ngram [accessed: 2020-09-18].

[22] D. Tang, F. Wei, N. Yang, M. Zhou, T. Liu, and B. Qin, "Learning sentiment-specific word embedding for twitter sentiment classification," in Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), vol. 1, 2014, pp. 1555–1565.

[23] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," Journal of machine Learning research, vol. 3, no. Jan, 2003, pp. 993–1022.

[24] S. Narr, M. Hulfenhaus, and S. Albayrak, "Language-independent twitter sentiment analysis," Knowledge discovery and machine learning (KDML), LWA, 2012, pp. 12–14.

[25] O. Araque and C. A. Iglesias, "An approach for radicalization detection based on emotion signals and semantic similarity," IEEE Access, vol. 8, 2020, pp. 17 877–17 891.

[26] M. Nouh, R. J. Nurse, and M. Goldsmith, "Understanding the radical mind: Identifying signals to detect extremist content on twitter," in 2019 IEEE International Conference on Intelligence and Security Informatics (ISI). IEEE, 2019, pp. 98–103.

[27] L. Povoda, R. Burget, and M. K. Dutta, "Sentiment analysis based on support vector machine and big data," in 2016 39th International Conference on Telecommunications and Signal Processing (TSP). IEEE, 2016, pp. 543–545.

[28] W. Khan, A. Daud, J. A. Nasir, and T. Amjad, "A survey on the state-of-the-art machine learning models in the context of nlp," Kuwait journal of Science, vol. 43, no. 4, 2016.

[29] K. Suresh and R. Dillibabu, "Designing a machine learning based software risk assessment model using naïve bayes algorithm," TAGA Journal, vol. 14, 2018, pp. 3141–3147.

[30] V. K. Jain, S. Kumar, and S. L. Fernandes, "Extraction of emotions from multilingual text using intelligent text processing and computational linguistics," Journal of computational science, vol. 21, 2017, pp. 316–326.

[31] V. Malik and A. Kumar, "Sentiment analysis of twitter data using naive bayes algorithm," International Journal on Recent and Innovation Trends in Computing and Communication, vol. 6, no. 4, 2018, pp. 120–125.

[32] M. S. Mubarok, Adiwijaya, and M. D. Aldhi, "Aspect-based sentiment analysis to review products using naïve bayes," in AIP Conference Proceedings, vol. 1867, no. 1. AIP Publishing LLC, 2017, p. 020060.

[33] R. S. Jagdale, V. S. Shirsat, and S. N. Deshmukh, "Sentiment analysis of events from twitter using open source tool," IJCSMC, vol. 5, no. 4, 2016, pp. 475–485.

[34] S. Baccianella, A. Esuli, and F. Sebastiani, "Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining." in Lrec, vol. 10, 2010, pp. 2200–2204.

[35] E. Cambria, D. Olsher, and D. Rajagopal, "Senticnet 3: a common and common-sense knowledge base for cognition-driven sentiment analysis," in Proceedings of the twenty-eighth AAAI conference on artificial intelligence, 2014, pp. 1515–1521.

[36] G. A. Miller, "Wordnet: a lexical database for english," Communications of the ACM, vol. 38, no. 11, 1995, pp. 39–41.

[37] S. Vohra and J. Teraiya, "A comparative study of sentiment analysis techniques," Journal JIKRCE, vol. 2, no. 2, 2013, pp. 313–317.

[38] O. Nasraoui, "Web data mining: Exploring hyperlinks, contents, and usage data," ACM SIGKDD Explorations Newsletter, vol. 10, no. 2, 2008, pp. 23–25.

 ISBN: 978-1-61208-804-4

Modeling Natural Language Policies into Controlled Natural Language: A Twitter Case Study

Irfan Khan Tanoli

Computer Science Department
University of Beira Interior
Covilha, Portugal, 6201-001
Email: irfan.khan.tanoli@ubi.pt

Sebastião Pais

Computer Science Department
NOVA LINCS and UBI
Covilha, Portugal, 6201-001
Email: sebastiao@di.ubi.pt

Abstract—**Social network providers usually describe the terms of data storage, usage, and sharing, by adopting natural languages. To automatically evaluate such terms of use, to understand, analyse, and enforce rights and obligations over the user's data, it is of uttermost importance to translate them in a machine-readable format. Natural Languages (NLs) are the most prominent form of knowledge representation for humans. However, due to NLs complexities, it is quite burdensome to process their sentences by machines in a seamless and standardised way. Controlled Natural Languages (CNLs) are subsets of NLs that are obtained by restricting the grammar and vocabulary, to minimize - or even eliminate - ambiguity and complexity of NL. These languages hold two major characteristics: they look informal and easy to read and write by humans, quite like natural languages, but they can be easily transited into machine-readable forms. In this paper, we study some policy-oriented CNLs. We adopt them as source languages for translating sample Twitter policies. Then, we assess the value of the different languages, according to the difficulties of the translation, its readability, and other compelling properties to find which CNL is more suitable for NL translation.**

Keywords–Natural Language; Controlled Natural Languages; Social Networks; Natural Language Processing; Data Policies.

I. INTRODUCTION

Social Networks (SNs) have a great impact on our everyday life. Users increasingly rely on SNs to share their opinions, plan activities, exchange information, and establish social relationships. SNs interactions usually require the exchange of users' data for a variety of purposes, including the provisioning of services. The collection, usage, and sharing of user's data is usually regulated by social networks (e.g., Facebook data [1], Twitter privacy policies [2], Google privacy policies [3]). Usually publish in English NL, the policies describe the terms and condition under which the provider will manage the data in terms of e.g., authorised, obliged, or denied. Although the use of Natural Language (NL) enables end users to read and understand the authorised (or obliged, or denied) operations on their data, a key issue relies on the fact that NLs are not machine readable, and automatic controls on how the data are actually going to be used and processed by the entities that operate on them is not feasible.

In particular, NLs cannot be used as the input language for a policy-based software infrastructure to be used for policy management. In fact, both automated policy analysis (the process to assure the lack of conflicting data policies, see, e.g., [4] [5]) and policy enforcement (the actual application of the data policies, whenever a data access request takes place) require inputs in a machine readable form, like, e.g., the *de facto* standard eXtensible Access Control Markup Language (XACML) [6] [7]. With the aim of moving in the direction of managing and enforcing access policies automatically, in this paper we consider a selection of different machine-oriented, English-based CNLs, originally designed within different contexts, and we investigate their effectiveness in expressing data policies as specified on a popular SN site.

CNLs are a subset of NLs, specifically conceived to make machine processing simpler. A CNL is, in essence, a developed language that is based on NL, but it is more restrictive in terms of lexicon, syntax, semantics, while at the same time retaining most of its natural properties [8]. CNLs have more contrived representation, in terms of grammar and vocabulary, and they thus reduce the ambiguity and complexity of a complete language [9], e.g., English, Spanish, French, Swedish, Mandarin, etc. [10]. CNLs have been proved to be effective in mitigating linguistic ambiguity challenges, as they can easily be translated into a formal language such as, first-order logic or different version of description logic, automatically and mostly deterministically [9].

CNLs can be roughly classified into two broad classes: human-oriented and machine-oriented. Human-oriented CNLs mostly used for improvement of technical documentation readability and comprehensibility. Machine-oriented CNLs are purposely dedicated to refine the translation of complex and technical documents [11], for knowledge presentation or processing [12], and for the Semantic Web [13]. Machine-oriented CNLs can also support translation of large texts, e.g., in English, into first-order logic, to automatically map their expressiveness into a small subset of expressions [13]. CNLs can be developed for specific scenarios and application domains [9]. For example, the Attempto Controlled English (ACE) [12] [14] has been designed with an expressive

 ISBN: 978-1-61208-804-4

knowledge representation that is easy to learn, read and write for domain experts.

The variety of CNLs attributes suggests that it is difficult to identify their general properties. First, CNLs are defined for different areas, (e.g., academia and industry), and for different fields, (e.g., computer science, mathematics, engineering, linguistics, etc.) Secondly, even if CNLs usually share common properties, there can be either CNLs that are inherently ambiguous, or precise as formal logic. Some are quite natural, others are closer to programming languages or to logic-based formalisms, or just defined with simple grammar rules, others are more complex and their syntax and semantics are not easy to define and/or understand [8]. Due to such variations, it is difficult to define fundamental properties to be used for comparing different CNLs.

Here, we consider samples of real Twitter data policies for translating from their original form in natural language to each of the selected controlled languages. Google [3], Facebook [1] and Twitter [2] data policies express the same actions like how user's data is regulated. The reason for choosing Twitter as a sample case study for translation is arbitrary, although Facebook and Google policies can also be translated in the same way. The translations are evaluated with respect to key properties defined in the so-called *Precision, Expressiveness, Naturalness, Simplicity* (PENS) classification scheme [8], having one new property, namely *policy enforcement*. The evaluation will help researchers to choose the most appropriate CNL and to automatically process the terms and conditions under which user's data are accessed, stored, and used for machine readability. The main contribution of this study is to provide an understanding related to CNLs and the need for NL translation into CNL for machine understandability. This study help us finding a certain CNL for NL policy translation. After our finding, we are motivated for development of an automated system that can translates CNL into NL.

The rest of the paper is organized as follows: Section II presents an overview of the three CNLs. Section III describes the key properties of the PENS scheme with a new general one. Section IV introduces some sample Twitter policies and their translations into each of the targeted CNLs. By relying on the translations, Section V presents an assessment and comparison of the considered CNLs. The final section VI outlines directions for future work and draws the conclusions.

II. Controlled Natural Languages

CNLs are general-purpose languages designed to facilitate domain experts in expressively representing knowledge. On the one side, they are easy to learn, write and read, but, on the other side, they are meant to be fully machine-readable (or, at least, designed in a way that makes possible their automatic translation into a machine-readable language). In this section, we consider three different machine-oriented policy-based languages. For our study, we include both general-purpose and domain-specific controlled languages, originally targeted at different contexts, e.g., knowledge representation, and policy authoring and enforcement.

Throughout the section, we present three sample policies in natural language and we translate them in each of the three languages. The sample policies we will consider are the following:

- **E1:** *User can log into system with valid Id and Password.*
- **E2:** *Bob must send documents to Alex, when Alex requests to Bob.*
- **E3:** *Ryan cannot share Paulo's data, if Paulo disallows Ryan.*

A. Attempto Controlled English

Attempto Controlled English (ACE) [12] [14] is a CNL developed for an automatic and unambiguous translation into a first-order logic. It was initially designed as a specification language, but the language has been improved over the years in various ways, gradually shifting towards knowledge representation and applications for the Semantic Web [8]. ACE has a few small set of construction and set of interpretation rules. The former explains its syntax and the latter makes the constructs clear which are vague in full English. ACE has a vocabulary which consists of some function words (conjunctions, pronouns), fixed phrases (there is), and content words (nouns, verbs, adjectives). Definitive Clause Grammar (DCG) is used to write grammars upon which the language processor relies. DCGs are equipped with certain structures that convert declarative and interrogative sentences into first-order logic. Once the discourse representation structure is created only then can anaphoric references be resolved. ACE also provides support for active and passive words, subject and object relative clauses [9].

ACE is intended for researchers who wish to use formal notation and formal methods even though they are not familiar or expert with them [15]. Notable features of this controlled language include the capability to express complex noun phrases, plurals, anaphoric references, subordinated clauses, modality, and questions. In ACE, the previously introduced sample policies can be expressed as:

- **E1:** `A user has a valid ID and PASSWORD to log into system and system validates ID and PASSWORD.`
- **E2:** `If Alex requests Bob THEN Bob must send documents to Alex.`
- **E3:** `If Ryan disallows Paulo then Paulo cannot share Ryan's data.`

There exists other CNLs similar to ACE for knowledge representation: as an example, Processable English (PENG) [16], Computer Processable Language (CPL) [17], Common Logic Controlled English (CLCE) [18], and Formalized English [19]. The comparison amongst this group of languages has been already presented [11]. Here, we decided to consider ACE because of its generality and its features, that render it more expressive, both syntactically and semantically [12].

B. Protune

The Protune (Provisional Trust Negotiation) policy language [4] is based on logic programming and, is

designed for policy evaluation, enforcement, and negotiation [5]. The language is based on standard logic rules of the form $A \leftarrow L_1,....,L_i$ where A is a standard logical atom (called the head of the rule) and $L_1,....,L_i$ (the body of the rule) are literals (that is L_i equals B_i or $\neg B_i$ for some logical atom B).

The format of Protune policy rules is as follows:

```
allow(action) ← condition_1....condition_n
  condition ← condition_1....condition_n
```

An action is allowed if all the conditions are satisfied. The rendering in Protune of the three sample policies is the following:

- **E1:** `allow(loginsystem) ← user(userid=U, password=P):'valid'`
- **E2:** `allow (send(Bob,Alex,documents)) ← request(Alex,Bob)`
- **E3:** `allow (share#Not(Ryan,Paulo,Data)) ← disallow(Paulo,Ryan,Data)`

C. Logic Based Policy Analysis Framework

A logic-based policy analysis language for policy specifications is presented in [20], which comes with a policy analyser providing also diagnostic information about detected conflicts, separation of duty, coverage gaps, behavioural simulation and policy comparison. vLBPAF is developed using the abductive constraint logic programming (ACLP) system as basis for algorithm analysis, and on the Event calculus [21] to represents how events and actions happening that affect states of the system, leading to circumstances in which a given policy rule is applicable and the information is an output of the analysis. The language uses a number of sorted first-order logic predicates, and discriminates between policy language and domain description language. The policy language representation $\mathcal{L}^{\pi}$ consists of sorts for subjects *Sub*, actions *Act*, and targets *Tar*, together with a sort for time *T*, represented using the non-negative reals.

The three $\mathcal{L}^{\pi}$ predicates, referred as 'regulatory predicates', are as follows:

- **Input Regulatory:**
  `req(Sub,Tar,Act,T)`
- **Output Regulatory:**
  `do(Sub,Tar,Act,T)`
  `deny(Sub,Tar,Act,T)`
- **State Regulatory:**
  `permitted(Sub,Tar,Act,T)`
  `denied(Sub,Tar,Act,T),`
  `obl(Sub,Tar,Act,T_s,T_e,T)`
  `fulfilled(Sub,Tar,Act,T_s,T_e,T)`
  `violate(Sub, Tar, Act, T_s, T_e, T)`
  `cease_obl(Sub, Tar, Act, T_init, T_s, T_e,T)`

The input regulatory predicate represents a request for *Sub* to perform *Act* on *Tar*, at time *T*. The output regulatory predicates indicate whether an *Act* is permitted or denied, for *Sub* to *Tar*, at time *T*. The state regulatory predicates indicate different situations concerned with the permitted and denied actions, the fact that an obligation exists, the fact that obligation has been actually fulfilled, violated, or expired. *T* indicates the actual time, while the pair T_s, T_e represent the interval time for the existence of an obligation. As a matter of fact, there exist translations of LPBAF to Ponder [22] and XACML. Both the target languages are enforceable, meaning, they serve as input to a standard policy enforcement infrastructure *a la* XACML. Again, let us see how the three sample properties are rendered in LPBAF:

- **E1:** The action login is permitted by the user 'U' on the system 'S', at time 'T', whenever at time 'T' the user has a valid Id and Password P (holdsAt is based on Event Calculus):
  `permitted(U,S,login,T) ←`
  `holdsAt(U,(Id,P),valid,T)`
- **E2:** In the language notation, 'B' (Bob) is obliged to send to 'A' (Alex) the documents 'D', at time 'T', when 'A' requests to 'B', at time 'T'.
  `obl(B,A,D,send,T) ← do(A,B,request,T)`
- **E3:** Considering Ryan 'R' cannot share Paulo 'P' Data 'D'. The prohibition is enabled if 'P' prohibits 'R' to share it. 'T', a variable rather than a fixed time, signals the beginning of the prohibition.
  `denied(R,P,D,share,T) ←`
  `do(P,R,disallow,T)`

III. Properties for Controlled Natural Languages (CNLs)

A well-established classification scheme, known as *Precision, Expressiveness, Naturalness, Simplicity (PENS)*, has been presented in [8] to support CNL comparison and classification.

A. The PENS Classification Scheme

A standard classification scheme is the better approach for controlled natural languages analysis to determine whether a language fulfills specific characteristics. The Precision, Expressiveness, Naturalness, Simplicity (PENS) scheme [8] was defined following the intuition that CNLs place themselves in between natural and formal languages. In general, CNLs are quite structured and constrained (thus, closer to pure formal languages), still, their syntax is close to natural terms. Furthermore, to establish a general, but, at the same time, restricted classification, the PENS scheme considers English as a natural language and propositional logic as a formal language.

To develop a base classification scheme, it is essential to put the properties under a few dimensions, to avoid as much as possible dependence between each other [8]. The PENS classification scheme considers only four properties *Precision, Expressiveness, Naturalness, Simplicity*, to condense under those umbrellas, the highest number of possible characteristics. For example, attributes like ambiguity in the text, formal definition of language, and capability to transform the language into a propositional logic can be merged under the Precision dimension. Natural writing, natural feeling and understanding of the language can be put under the Naturalness dimension. Instead, Simplicity measures the non-complexity of the language. The expressiveness of

a language is a measure of the variety of lexical and grammatical constructions, it allows (irrespective of the reader).

In the following, we will consider such four properties as the standard base for our comparison, plus one more property *Policy enforcement*, which is discussed later in this section. Each of the PENS dimensions is measured through five classes, ranging over the interval 1, ..., 5. Each of the five classes presents a one-dimensional area between the two extremes, i.e., English at one end and propositional logic on the other one. The decision to assign a language to one of the five classes, for each dimension, is left arbitrary. Considering Simplicity and Precision, English is at the bottom, i.e., S^1 and P^1, while propositional logic is at the top, S^5 and P^5. Conversely, for Expressiveness and Naturalness, English is at the top: E^5 and N^5 while propositional logic is at the bottom: E^1 and N^1The complete details are available in [8]. The five classes for each dimension are described in a vast scope and cover a wide range of CNLs. Therefore, to make a simple, but effective comparison among the languages as described in (Sect. II), we select only one class for each dimension (usually, a class in the middle).

1) Precision: Precision is referred to as the degree to which the meaning of a text can be directly understood and recovered from its textual form in a particular language, i.e., the sequence of linguistic symbols [8]. The ambiguity in the meaning, predictability, and formality of the definition can be combined with precision. Formal logic languages are highly precise because the meaning of the text is strictly defined based on the possible sequences of the symbols of the language, as compared to NLs which are, according to the property definition, imprecise and ambiguous.

The precision classes are defined as: Imprecise languages, Less imprecise languages, Reliably interpretable languages, Deterministically interpretable languages, Languages with fixed semantics., we select **'Deterministically Interpretable Languages (DIL)'** as the reference class: this class includes languages that are entirely formal at the *syntactic* level. Texts in this language can be deterministically translated into a logical representation that defines the meaning of sentences. However, any sensitive deduction may require additional background axioms, external or heuristic resources [8].

2) Expressiveness: Expressiveness is related to the range of propositions that a language is capable of expressing. For example, language 'Y' is more expressive than language 'Z' if 'Y' can describe all that 'Z' can, but 'Z' cannot do the same w.r.t. 'Y'. This relationship does not necessarily induce a total order. For example, given two languages, it might be that none of them is more expressive than the other one. This makes it hard, or even unfeasible, to objectively rank in a linear order a set of languages, in terms of expressiveness [8].

PENS consider the following characteristics of expressiveness:

1) universal quantification over individuals, i.e., the presence in the language syntax of the logical predicate ∀, 'given any' or 'for all'.
2) relations of arity greater than one, i.e., languages which functions/predicates are taking as input more than one argument.
3) general rule structures, e.g., if-then-else conditions.
4) negation (failure or strong negation).
5) second-order (extension of first-order logic) universal quantification over concepts and relations [23].

By considering the above characteristics, it is possible to categorize languages according to five different classes of expressiveness: inexpressive languages, languages with low expressiveness, languages with medium expressiveness, languages with high expressiveness and languages with maximal expressiveness, we focus on **'Languages with Medium Expressiveness (LwME)'**, i.e., languages with all the characteristics of expressiveness as above, except second-order universal quantification.

3) Naturalness: The dimension of naturalness defines how a language is 'natural' in terms of reading and understanding from the user standpoint. Linguistic properties such as modification of grammar, comprehensibility, and natural reading and writing can be considered elements of naturalness. CNLs retains most of the natural properties of native languages, so that native language users can, quite effortlessly, understand texts without the need of language experts. The five naturalness classes are: unnatural languages, languages with dominant unnatural elements, languages with dominant natural elements, languages with natural sentences, languages with natural texts. This study considers **'Languages with Dominant Natural Elements (LwDNE)'** as point of reference, this study considers *'Languages with Dominant Natural Elements (LwDNE)'* as a point of reference.

With these types of languages, natural elements of languages dominate unnatural elements, and the overall grammar structure corresponds to the grammar of the natural language. However, due to the rest of natural elements or combination of unnatural elements, these languages cannot be considered valid natural sentences. Natural language speakers cannot easily recognize the sentences statements and cannot understand their essence without any guidance or instructions but still intuitively understand the language to a substantial degree [8].

4) Simplicity: Simplicity is consider as how simple (resp., complex) is to describe the language accurately and comprehensively, covering syntax and semantics. These 'exact and comprehensive descriptions should define all syntactic and semantic properties of the language using accepted grammar notations to define the syntax and accepted mathematical or logical notations to define the semantics. Concerning the PENS classification scheme, the indicator of simplicity is the number of natural language pages needed to describe the language accurately and comprehensively, consisting in the definition of all the syntactic and semantic properties of the language. Page counting should be done considering a single-column format, with a maximum of 700 words per page. The language descriptions do not require to include vocabularies [8].

 ISBN: 978-1-61208-804-4

From the following five properties of simplicity, i.e., very complex languages, languages without exhaustive descriptions, languages with lengthy descriptions, languages with short descriptions and languages with very short descriptions, we consider **'Languages with Short Descriptions (LwSD)'** as the term of comparison: a language considered to be simple enough to be described in more than a single page but less than ten pages.

B. Policy Enforcement

A standard architecture for the application (technically, 'enforcement') of privacy policies is as follows. Consider a generic subject 'S' that tries to access the object 'O' (a medical report, a picture published on a social network, etc.) to, e.g., modify or delete it or share it with third parties. This sketched architecture is adopted by the most common and tested authorization and control systems, such as the one implemented in the authorization infrastructure associated with XACML [6] [7]. We will thus consider a further property, Policy Enforcement (PE), taking into accounts if the CNLs under investigation are enforceable, or not. In other words, we will consider if they serve as input to standard tools for policy enforcement.

IV. TANSLATION OF TWITTER POLICIES

In this section, we consider real Twitter policies and present their translation into each of the three selected CNLs. The outcome will be evaluated in Section V, to assess the relative merits of the considered CNLs with respect to Precision, Expressiveness, Naturalness, Simplicity (PENS), and amenability to Policy Enforcement (PE).

A. Twitter Data Policies

The Twitter Data Policies [2], describe the kind of information collected by the social network and how such information is used and shared. Hereafter, we consider the following sample policies.

- **Contact Information and Address Books:**
  P1: You can choose to upload and sync your address book on Twitter so that we can help you find and connect with people[...].
- **Twitter for Web Data:**
  P2: When you view our content on third-party websites that integrate Twitter content such as embedded timelines or Tweet buttons, we may receive Log Data that includes the web page you visited.
- **Developers**
  P3: If you access our APIs or developer portal, we process your personal data to provide our services..
- **Object, Restrict, or Withdraw Consent**
  P4: When you are logged into your Twitter account, you can manage your privacy settings and other account features here at any time.
- **Accessing or Rectifying Your Personal Data**
  P5: If you have registered an account on Twitter, we provide you with tools and account settings [...].

B. From natural to controlled natural languages

Below, we show examples of translations of the Twitter policies listed above to the CNLs described in (Sect. II). Here, we consider P1, P2, P3, P4, P5.

1) Attempto Controlled English:

P1 in ACE:

```
IF You can choose to upload and sync your
address book on Twitter THEN we can help you
find and connect with people.
```

P2 in ACE:

```
IF you view our content on third-party
websites that integrate Twitter content such
as embedded timelines or Tweet buttons THEN we
may receive log data that includes the web
page you visited.
```

P3 in ACE:

```
IF you access our APIs or developer portal
THEN we process your personal data to provide
our services.
```

P4 in ACE:

```
IF you are logged into your Twitter account
THEN you can manage your privacy settings and
other account features here at any time.
```

P5 in ACE:

```
IF you have registered an account on Twitter
THEN we provide you with tools and account
settings.
```

2) Protune (PROvisional TrUst NEgotiation):

P1 in Protune:

```
allow (help(we,you,(FindandConnect(people))))
← ChoosetoUpload (you,address book,Twitter),
  ChoosetoSync (you,address book,Twitter)
```

P2 in Protune:

```
allow (receive(We,LogData) ←
  visit (you,web page), view
(our,content,third-party website),
  integrate(twitter,content),
content:timeline,tweet buttons.
```

P3 in Protune:

```
allow (process(your,personal
data,(provide(our,services))) ←
access (you,our (API developer portal))
```

P4 in Protune:

```
allow (manage#atanyTime (your,privacy
                settings,
other account features))← log (you,twitter
                account)
```

P5 in Protune:

```
allow (provide(we,you,tools,account
settings))← register (you,twitter account)
```

3) Logic Based Policy Analysis Framework:

P1 in LBPAF:

If you 'Y' choose to upload and sync Address Book 'AB' on Twitter 'TW' THEN we 'W' can help 'Y' find and connect with people 'P' in Time 'T'. 'T' in the head of the rule is a variable rather than a fixed time and it has been inserted since required by the syntax of LBPAF.

```
permitted(W,Y,help(P,find,connect,T) ←
   do(Y,C,AB,TW,ChoosetoUpload,T), do
            (Y,AB,TW,sync,T)
```

P2 in LBPAF:

If you 'Y' view our content 'C' on third-party website 'TPW' that integrate Twitter content 'TC' such as embedded timelines 'ET' or Twitter buttons 'TB' THEN we 'W' may receive that Log Data 'LD' that inluded page 'P', 'Y' visited in Time 'T'. 'happens' is based on Event Calculus.

```
        permitted(W,LD,receive,T) ←
do(Y,TC,TPW,view,T),holdAt(TC,(ET,TB,T),
   integrate,T),happens(Y,P,visited,T)
```

P3 in LBPAF:

```
permitted(W,YP,D,process,(TW,provide),T) ←
              do(Y,TA,access,T)
```

P4 in LBPAF:

If you 'Y' logged' into your Twitter account 'TA', you 'Y' can manage your privacy settings 'PS' and other account feature 'OAF' in Time 'T'.

```
permitted(Y,PS,manage,T) ← do(Y,TA,log,T)
```

P5 into LBPAF:

If you 'Y' register an account 'A' on Twitter 'T' THEN We 'W' provide you'Y' with tools 'TO' and account settings 'AS' in Time 'T'.

```
permitted(W,Y,TO,AS,provide,T) ←
       do(Y,A,T,register,T)
```

V. EVALUATION

In this section, we consider the three languages discussed in Section II and we evaluate to which degree they fulfil the properties introduced in Section III based on the translation presented in Section IV.

A. ACE

ACE (Sect. II-A) is a precise language, according to the definition of *precision* (Sect. III-A1) and, in particular, it can be classified as a *Deterministically Interpretable Language* (completely formal at syntactic level). In terms of *expressiveness* (Sect. III-A2), ACE can be classified as a *Language with Medium Expressiveness*. As noticed in [8], it has general rule structures, negation, arity relation greater than one and universal quantification over individuals [12]. In terms of *naturalness*, ACE cannot be considered as a *Language with Dominant Natural Elements* as discussed in [8]. The Twitter policies in Sect. IV-B1 can be easily understood by a general audience without external guidance. For *simplicity*, authors in [8] define ACE as *Language with Lengthy Descriptions* [12], [24]. ACE is also not a *policy-enforceable* language (Sect. III-B, being not associated to any policy enforcement architecture [12].

TABLE I. COMPARISON OF CONTROLLED NATURAL LANGUAGES

	ACE	Protune	LBPAF
Precision (DLI)	Yes	Yes	Yes
Expressiveness (LwME)	Yes	Yes	Yes
Naturalness (LwDNE)	No	No	No
Simplicity (LwSD)	No	Yes	Yes
Policy Enforcement	No	Yes	Yes

B. Protune

Being equipped with a formal syntax, Protune (Sect. IV-B2) holds the *precision* property, with degree *Deterministically Interpretable Languages*. Protune meets all the four features needed for being classified as a *Language with Medium Expressiveness* (Sect. III-A2): general rules structure, negation, universal quantification over individuals, and relations of arity grater than one [25]. Protune features a mixture of natural and unnatural terms and its grammar structure does not correspond to that of a natural language (Sect. IV-B2).

Proper guidance is needed to adopt Protune; users fail to intuitively understand the respective statements [8]. Therefore, our opinion is that it cannot be classified as a *Language with dominant natural elements* (Sect. III-A3). Protune is described with exact and comprehensive syntax and semantics and the language description is more than a single page but less than 10 pages [25]; hence, it can be categorized as a *Language with Short Descriptions* (Sect. III-A4). Finally, Protune supports *policy enforcement* [15] [24].

C. Logic Based Policy Analysis Framework

Logic Based Policy Analysis Framework (LBPAF) is a *precise* language (Sect. III-A1), fully formal and fully specified both at the syntactic and at the semantic level. The language is a *Deterministically Interpretable Language*. LBPAF is an *expressive* language for policy definition, in particular, it enjoys the four properties needed for being a *Language with Medium Expressiveness* [20]. Non expert people need proper guidance for using the language. Moreover, as shown in Sect. IV-B3, the unnatural elements are dominant with respect to the natural ones. Therefore, we cannot classify LBPAF as a *Language with Dominant Natural Elements*. Regarding *simplicity*, the language description is such that it takes more than a single page but less than 10 pages [20]. Therefore, this language can be classified as a *Language with Short Descriptions*. Finally, it can be translated into the enforceable language Ponder [22], fulfilling, even if indirectly, the property of *policy-enforcement*.

D. Summary

Our analysis is summarised in Table I, where rows indicate the policy languages and columns indicate the properties. Intuitively, cells are marked with 'Yes' or 'No', according to whether or not a language satisfies a

 ISBN: 978-1-61208-804-4

certain property. The evaluation shows that Protune and LBPAF fulfils the highest number of properties. The two languages are formal at the syntactic level and have an associated formal semantics; their description is concise, thus fulfilling the simplicity property at level of languages with shorts descriptions; they were not defined with a specific vocabulary associated and all have a policy enforcement infrastructure associated. Protune and LBPAF enjoy the property of medium expressiveness, and none of the language appears to be a language with dominant natural elements.

VI. Conclusions

In this study, we considered three Controlled Natural Languages and we evaluated them according to a set of standard properties defined in the literature. The evaluation is carried out based on the translation of a Twitter policies into the analysed CNLs. Findings are that, according to the PENS scheme, all languages are formal at the syntactic level (remarkably, all but ACE also have a precise semantics associated). The three languages feature different degrees of expressiveness (in terms of expressible logical operators and functions), presence of natural elements, and simplicity of their descriptions. Finally, two out of three i.e., Protune and LBPAF serve as input to a standard policy enforcement infrastructure *a la* XACML.

Notably, each of the investigated languages is capable of expressing data privacy policies. Aiming at choosing a CNL as the target language to automatically translate NL social network(s) data policies, the outcome of our evaluation helps us towards Protune and LBPAF. However, both languages are rigorous at the syntactic and semantics level, expressive enough, and they do not need a huge effort in terms of learning of use. A notable remark is that they come with devoted toolkits for policy authoring, analysis and enforcement. For future work, we aim at designing a CNL (or adapting an existing one, possibly among the ones investigated in this work) easily understandable and sufficiently expressive to be used directly by the managers of social network sites, to describe the use they make of the data that users provide them.

Acknowledgment

This work was supported by National Founding from the FCT- Fundacão para a Ciência e a Tecnologia, through the MOVES Project - PTDC/EEI-AUT/28918/2017 and by operation Centro-01-0145-FEDER-000019-C4 - Centro de Competências em Cloud Computing, co-financed by the ERDF through the Centro 2020, in the scope of the Sistema de Apoio à Investigação Científica e Tecnológica.

References

[1] "Data Policy," 2020, URL: https://www.facebook.com/full_data_use_policy [accessed: 2020-09-15].

[2] "Twitter Privay Policy," 2020, URL: https://twitter.com/en/privacy [accessed: 2020-09-15].

[3] "Google Privacy and Terms," 2020, URL: https://policies.google.com/privacy [accessed: 2020-09-15].

[4] J. L. De Coi, D. Olmedilla, P. A. Bonatti, and L. Sauro, "Protune: A framework for semantic web policies." in International Semantic Web Conference (Posters & Demos), vol. 401, 2008, p. 128.

[5] P. Bonatti, J. L. De Coi, D. Olmedilla, and L. Sauro, "A rule-based trust negotiation system," IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 11, 2010, pp. 1507–1520.

[6] B. Parducci, H. Lockhart, and E. Rissanen, "Extensible access control markup language (xacml) version 3.0," OASIS Standard, 2013, pp. 1–154.

[7] D. Ferraiolo, R. Chandramouli, R. Kuhn, and V. Hu, "Extensible access control markup language (xacml) and next generation access control (ngac)," in Proceedings of the 2016 ACM International Workshop on Attribute Based Access Control, 2016, pp. 13–24.

[8] T. Kuhn, "A survey and classification of controlled natural languages," Computational Linguistics, vol. 40, no. 1, 2014, pp. 121–170.

[9] T. Gao, "Controlled natural languages for knowledge representation and reasoning," in Technical Communications of the 32nd International Conference on Logic Programming (ICLP 2016). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2016, p. 21.

[10] T. Khun et al, "On controlled natural languages: Properties and prospects," in International Workshop on Controlled Natural Language. Springer, 2009, pp. 281–289.

[11] R. Schwitter, "Controlled natural languages for knowledge representation," in Coling 2010: Posters, 2010, pp. 1113–1121.

[12] N. Fuchs, K. Kaljurand, and T. Kuhn, "Attempto Controlled English for knowledge representation," Reasoning Web, 2008, pp. 104–124.

[13] H. Safwat and B. Davis, "Cnls for the semantic web: a state of the art," Language Resources and Evaluation, vol. 51, no. 1, 2017, pp. 191–220.

[14] N. E. Fuchs, "Understanding texts in attempto controlled english." in CNL, 2018, pp. 75–84.

[15] J. De Coi, P. Kärger, D. Olmedilla, and S. Zerr, "Using natural language policies for privacy control in social platforms." CEUR Workshop Proceedings, ISSN 1613-0073, 2009.

[16] C. White and R. Schwitter, "An update on PENG light," in ALTA, vol. 7, 2009, pp. 80–88.

[17] P. Clark, W. R. Murray, P. Harrison, and J. Thompson, "Naturalness vs. predictability: A key debate in controlled languages," in Controlled Natural Language. Springer, 2009, pp. 65–81.

[18] J. F. Sowa, "Common Logic Controlled English," URL: http://www.jfsowa.com/clce/specs.htm [accessed: 2020-09-15].

[19] P. Martin, "Knowledge representation in CGLF, CGIF, KIF, frame-CG and Formalized-English," in Conceptual Structures: Integration and Interfaces. Springer, 2002, pp. 77–91.

[20] R. Craven et al, "Expressive policy analysis with enhanced system dynamicity," in Symposium on Information, Computer, and Communications Security. ACM, 2009, pp. 239–250.

[21] R. Kowalski and M. Sergot, "A logic-based calculus of events," New Generation Computing, vol. 4, no. 1, Mar 1986, pp. 67–95.

[22] G. Russello, C. Dong, and N. Dulay, "Authorisation and conflict resolution for hierarchical domains," in 8th IEEE International Workshop on Policies for Distributed Systems and Networks, 2007, pp. 201–210.

[23] Stanford Encyclopedia of Philosophy, "Quantifiers and quantification," 2018, URL: https://plato.stanford.edu/entries/quantification/#SecOrdQua[accessed: 2020-09-15].

[24] J. De Coi, N. E. Fuchs, K. Kaljurand, and T. Kuhn, "Controlled English for reasoning on the Semantic Web," in Semantic techniques for the web. Springer, 2009, pp. 276–308.

[25] P. Bonatti and D. Olmedilla, "Driving and monitoring provisional trust negotiation with metapolicies," in Policies for Distributed Systems and Networks. IEEE, 2005, pp. 14–23.

 ISBN: 978-1-61208-804-4