Bilingual Sentiment Analysis of Tweets Using Lexicon

Farwa Iqbal[1], Amber Ayoub[2], Jaweria Manzoor[3], Rida Hijab Basit[4]
Computer Science Department
Kinnaird College for Women Lahore
{farwaiqbal786, ember.ayub}@gmail.com[1,2], {jaweria.manzoor, rida.basit}@kinnaird.edu.pk[3,4]

Abstract

Sentiment Analysis determines the emotions, attitude, feelings or behavior of people towards an event, topic, or a product. The advent and growth of social media platforms have given people opportunity to voice their opinions, reviews and share experiences. User Generated Content when analyzed for its sentiments can be helpful for various reasons such as predictive analysis, summarization of reviews, measuring popularity, acceptance of products, and much more. In this regard, various researches and studies exist, but all these studies focus on resource-rich languages like English, Chinese and Arabic. In this paper, we focus on Roman Urdu language. Around 30 million people across the world speak Urdu and mostly use Roman Urdu written in Roman script to express their views, feelings or experiences over the Internet. Keeping this in view, an approach has been proposed that performs sentiment analysis of bilingual data (English and Roman Urdu), using Lexicon based approach. In order to create domain specific lexicon, political tweets related to 2018 Elections held in Pakistan have been collected and analyzed for the sentiments expressed in them.

Keywords – Sentiment Analysis, Text, Opinion Mining, User Generated Content, Pakistan Election 2018

1. Introduction

With advancements in technology and internet, the use of mobiles and laptops to access social media accounts has increased [1]. People now openly give their reviews and opinions about anything, thus making it necessary to analyze the content generated by them. Substantial amount of work related to sentiment analysis on structured languages like English, Chinese and Arabic exist, but limited work has been done on Roman Urdu or Urdu languages [2]. Majority of the people in subcontinent are not much well versed in English language and use Urdu to express their sentiments on social platforms. People tend to express their opinions in Urdu mostly using Roman script known as Roman Urdu also due to the limited availability of keyboards in Urdu; thereby emphasizing upon the need of sentiment analysis in Roman Urdu. The importance of research in Roman Urdu has also been highlighted in other researches [2], [3] and [4]**.** Many techniques and methods for sentiment analysis exist but we have focused on the development of a domain specific lexicon of 3900 words that consists of Roman Urdu and English Language. Generally, sentiment analysis has been carried out using adjectives, but sentiment analysis approach presented in this paper makes use of lexicon built using adjectives, verbs, adverbs and nouns for improved sentiment analysis. The analysis has been performed on the data collected from Twitter using several hash tags based on election, from different pages of political parties, anchors etc. For detailed experimentation and to improve results two different datasets with 5031 tweets and 4177 tweets are considered. This paper is divided into five sections. Section 1 gives an introduction, Section 2 focuses on literature survey, Section 3 discusses the methodology, Section 4 compiles the results and Section 5 concludes the paper.

2. Related Work

Various existing systems that have performed sentiment analysis using lexicon approach have been studied. Some of those papers depicting similar work are discussed below.

The concept of bilingual sentiment analysis using lexicon approach on Roman Urdu has been presented by [4]. The purpose of this approach is to analyze the bilingual data from twitter. Tweets are collected based on the keywords related to four main political parties. SentiStrength is used for extracting the sentiments for the English language but for Roman Urdu a new lexicon is created to provide sentiment strength to the Roman Urdu words. SentiStrength along with English to Roman Urdu dictionary are utilized to create bilingual sentiment repository which provides 3900 Roman Urdu words and 1673 English words. The results depict that PTI dominates other parties in general whereas in Lahore, public opinion is mostly in favor of PML-N.

In [5], analysis of the sentiments expressed in news comments has been performed using lexicon based approach to get users' opinions about a certain topic. The problem of comments oriented sentiment analysis is that the user may express his/her own opinion, which is different from the original focus of discussion. This system has used a lexicon based approach to analyze the opinions by extracting comments. The lexicon used in this system is a manually created lexicon that contains 250 news items. Objective expressions have been stored in the lexicon for identifying the focus. Objects and its features are arranged in a taxonomy-based structure. Lexicon's knowledge and the user generated content are then preprocessed by using NLP techniques. The overall sentiment of the entire document is also computed by using certain weights assigned to positive, negative and neutral comments. The experiments provide an accuracy of 0.89.

In a similar study [6], the salience for an entity in the news corpus or lexicon and the polarity of each salience as positive or negative has been calculated. The system builds the news corpus from different websites including DAWN news, ARY news, Nawai Wakt and BBC Urdu news. Corpus is further divided into chunks and POS tagging is performed to create tag list. Tag list is fed into entity finding module, which selects entities meeting a certain criterion and weight for each entity is calculated. Every salience is provided with polarity using the manual polarity tagger. After assigning the polarity the corpus is searched for intensifiers like

(1) Shadid (ʃaḏiḏ, Extremely)
(2) Bohut (bɔhət̪, Lots, Intense)

Polarity of the salience with intensifier gets double for example from -5 to -10. The accuracy achieved is 84.5%.

A different approach has been discussed by [7] to find the subjectivity and polarity of the tweets using lexicon based approach. Tweets have been analyzed to predict the results of elections about a certain candidate providing a comparison on the various candidates based on sentiments expressed in them. 10,000 labeled tweets have been collected, preprocessed, imported for sentiment analysis for determining the subject to overall polarity. The sentences are classified by first assigning polarities to individual words: +1 for positive words, -1 for negative words and 0 for neutral words. Then polarity of a sentence is calculated by adding polarities of occurring words and classified as positive, neutral or negative. The subjectivity (users personal view about a candidate) of the tweets has been represented by 1 whereas the objectivity by 0. After the calculation of average polarity and subjectivity, percentage of positive, negative and neutral tweets is calculated. The experiments show that candidate Hillary has received a greater number of positive tweets whereas Trump received highest number of negative tweets.

Based on the limited research in Roman Urdu sentiment analysis, we propose a Lexicon based approach to detect sentiments depicted in tweets using Roman Urdu language as it is used by many people around the world to express their opinions on social media. In this approach, we have created a domain specific lexicon containing different parts of speech like nouns, verbs, adjective and adverbs to depict sentiments.

3. Methodology

The approach presented in this paper aims on performing bilingual sentiment analysis using lexicon. The novelty of our work lies in creation of a domain specific lexicon that contains both English and Roman Urdu words containing adjectives, nouns, verbs and adverbs. For the purpose of creating domain specific lexicon, tweets related to Pakistan Elections 2018 have been collected from Twitter using Twitter APIs. The collected data is then preprocessed, cleaned (noisy data and hashtags are removed) and tokenized (stop words removal). After formation of the tokens each word is assigned a polarity by using the lexicon ranging from -1 to + 1. Polarity of the sentence is then calculated by summing up polarities of all occurring words in the sentence. Based on the polarity, each sentence is classified as positive, negative or neutral. Furthermore, results in the form of accuracy, precision, recall and F-measure are calculated with the help of confusion matrix.

3.1. Data Preparation

3.1.1. Data Collection. The data has been collected using Twitter APIs with the help of Python. Making use of the twitter developer account, tweets from 2018 elections of Pakistan have been gathered. A total of 5031 tweets are extracted using multiple political hash tags and official pages of different anchors and politicians. A total of 2673 positive tweets, 1923 negative and 426 neutral tweets have been collected. These tweets have been stored in .json file after extraction.

Table 1: Hashtags and keywords used to extract data

#PTI	#PMLN	#MQM	#corruptleader	#nayapakistan
#imrankhan	#votekoizzatdo	#nawazsharif	#maryamnawaz	#absirfimrankhan
#jiyebhutto	#maryammeriawaz	#miansaab	#tabdeeli	#shehbazsharif

3.1.2. Data Preprocessing. Extracted data has been cleaned by removing all the unnecessary characters and symbols [8].

3.1.3. Translation of Urdu Tweets. At this stage, the extracted data consists of tweets in three different languages namely English, Urdu and Roman Urdu. For our proposed approach, Roman Urdu and English tweets are to be considered so Urdu tweets have been translated into Roman Urdu using online translator iJunoon.com.

3.1.4. Labeling of Tweets. After the translation, data set now contains only English and Roman Urdu tweets. As the next step, data labeling of tweets as positive, negative or neutral has been performed for each tweet by a single resource person. However, to remove partiality while labeling data for the dataset with 4177 tweets, data labeling has been performed using crowdsourcing technique. The tweets are labelled according to the sentiments in them like

(1) IK buhat acha politician hai

IK bɔhə̪t aʧʰa politician hæ
"IK is a great politician" is labeled as positive

(2) Zardari ek corrupt insaan hai
zərd̪ari æk corrupt ɪnsan hæ
"Zardari is a corrupt person" is labeled as negative

(3) PTI or PMLN jo bhi jeete, hume vote aur elections ko izzat deni chahiye

PTI ɔr PMLN ʤo bhi ʤit̪e həme vote ɔr elections ko ɪzət̪ d̪eni ʧahie
"PTI or PMLN whoever wins, we should respect vote and elections" is labeled as neutral

3.1.5. Normalization and Tokenization of Tweets. The tweets are then normalized, where the stop words are removed from each tweet so that the meaningless words (words that play no part in sentiment analysis) like 'wo (vo, "they"), hum (həm, "we"), are, is, it' etc. are eliminated. Unlike English, Roman Urdu is not a structured language and does not have list for stop words. Therefore, we created a list translating Urdu stop words to Roman Urdu using iJunoon.com and used this with built-in Python Normalization function to remove stop words from Roman Urdu tweets.

Roman Urdu Stop Words	English Stop Words
aa (a, "come")	where
ab (əb, "now")	at
abb (əb, "now")	it
aagai (agəi, "arrived")	when
aagaya (agəja, "arrived")	which
aain (ajæ̃, "come")	why
aaj (aʤ , "today")	am
aaja (aʤa, "come")	he
aakar (akər, "in terms of coming")	she
aakr (akər, "in terms of coming")	than
hum (həm, "we")	then
tum (təm, "you")	that
apka (apka, "yours")	so
nay (ne, "connector in Urdu")	who

Figure 2: List of some stop words

3.2. Lexicon based Approach

Sentiment analysis approach presented in this paper make use of lexicon of 3900 words built using adjectives, verbs, adverbs and nouns for improved sentiment analysis [4]. The reason of creating a lexicon with different parts of speech is the morphological richness of Urdu Language and Roman Urdu. Basically, Roman Urdu is Urdu language written using Roman Script where an adjective can inflect from noun or other parts of speech like

(1) daftari,

d̪əft̪əri
"official"

(2) kagazi

kɑɣəzi
“Thin, scariose”
etc. [9]. Moreover, the sentiment of a complete sentence may not be depicted by adjective only like

(1) Mujhe ye phone buhat acha aur sasta lagta hai.

mʊʤʰe je phone bɔhət̪ aʧʰa ɔr səst̪a ləgt̪a hæ
“I think this phone is very good and inexpensive”

For the analysis of this sentence we also must take intensifier ‘buhat (bɔhət̪, “very”)’ into consideration along with adjectives ‘acha (aʧʰa, “good”)’ and ‘sasta (səst̪a, “inexpensive”)’.

The lexicon is created using SentiWordNet (lexical resource for Sentiment analysis and opinion mining). The sentiment scores for English words are directly taken from SentiWordNet whereas the Roman Urdu words are first translated into Urdu and then into English and then those English translations are searched in SentiWordNet for the sentiment scores. The scores assigned range from -1 to +1.

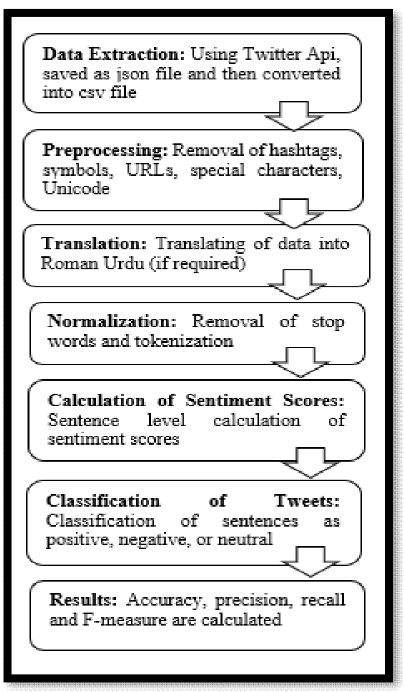

Figure 2: Lexicon based Approach

3.2.1. Sentence Level Classification. Each tweet is normalized and converted into tokens. The java code specifically built for this purpose then assigns sentiment score (polarity) to individual tokens in each tweet using the lexicon. Then the sentiment score of the whole sentence is calculated by summing up the scores of all occurring tokens or words in the sentence similar to the approach used by [7] for sentence level sentiment classification. The classification of a sentence is based on the following conditions:

- If the resultant value is greater than 0, the sentence is classified as positive
- If the resultant value is less than 0, the sentence is classified as negative
- If the resultant value is equals to 0, the sentence is classified as neutral

4. Results and Discussion

This section discusses the results of the experiment carried out in the light of objective of this study.

Experiments are performed initially by considering dataset with 5031 tweets consisting of tweets including negation words like ‘nae, ni, nahi (no)’ etc. and mixed sentiments expressed in them like:

(1) Imran Khan buhat acha insaan hai, a great cricketer lakin buhat hi bura leader aur politician hai is ko kuch nahi ata pata ni umeed krni chaiye ya ni

Imran Khan bɔhət̪ aʧʰa ɪnsan hæ, a great cricket lekɪn bɔhət̪ hi bura leader ɔr politician hæ ɪs ko kuʧʰ nəhi at̪a pət̪a ni umid̪ kərni ʧahie ja ni
“Imran Khan is a good person, he is a great leader but not a good politician, he knows nothing, I don’t know we should hope or not.”

Better results are achieved by eliminating tweets with different sentiments and negation words. Removal of such tweets reduced the dataset to 4177 tweets only. Although there is a difference between negation words and words that depict negative sentiment like corrupt, evil, bad. We have kept the tweets like

(1) IK ek corrupt insaan hai

IK æk corrupt ɪnsan hæ
“IK is a corrupt person”

But have removed sentences like

(2) IK ek acha leader ni hai

IK æk aʧʰa leader ni hæ
“IK is not a good leader”

Furthermore, different measures have been calculated using confusion matrix for all three classes (positive, negative and neutral) separately. Confusion matrix is a form of a table that is used to represent a classification model. In the confusion matrix below, n depicts total number of tweets. Actual values of three classes depict the tweets labeled as positive, negative and neutral through human labeling. Predicted values of three classes depict the tweets labeled as positive, negative and neutral using domain specific lexicon that is created in this paper. Furthermore, accuracy, precision, recall and F-measure with respect to each class are calculated separately.

4.1. Dataset with 5031 Tweets containing Positive, Negative and Neutral Classes

4.1.1. Positive Class. Here the confusion matrix is created by taking positive class in consideration and the accuracy, recall, precision, F-measure are calculated. Our lexicon based approach of sentiment analysis predicts positive class with 81% accuracy. For positive class, TP (true positive) is the intersection of actual positive and predicted positive. FP (false positive) is the sum of values in the corresponding column, whereas FN (false negative) is the sum of values in the corresponding row excluding value of TP in both cases. TN (true negative) is the sum of all the values excluding the row and column containing positive class. The column matrix with respect to positive class is given below.

Table 2: Confusion matrix for positive class

n = 5031	Predicted Negative	Predicted Positive	Predicted Neutral	
Actual Negative	TN = 739	FP = 0	TN = 1193	1932
Actual Positive	FN = 0	TP =1734	FN = 939	2673
Actual Neutral	TN = 0	FP = 0	TN = 426	426
	739	1734	2558	

Measures calculated for positive class are as follow:
Accuracy = 0.81 = 81%
Precision = 1
Recall = 0.649
F Measure = 0.787

4.1.2. Negative class. Here the confusion matrix is created by taking negative class in consideration and the accuracy, recall, precision, F-measure are calculated. Our lexicon based approach predicts negative class with 76.7% accuracy. For negative class, TP (true positive) is the intersection of actual negative and predicted negative. FP (false positive) is the sum of the values in the corresponding column, whereas FN (false negative) is the sum of values in the corresponding row excluding value of TP in both cases. TN (true negative) is the sum of all the values excluding the row and column containing negative class. The column matrix with respect to negative class is depicted below.

Table 3: Confusion matrix for negative class

n = 5031	Predicted Negative	Predicted Positive	Predicted Neutral	
Actual Negative	TP = 739	FN = 0	FN = 1193	1932
Actual Positive	FP = 0	TN =1734	TN = 939	2673
Actual Neutral	FP = 0	TN = 0	TN = 426	426
	739	1734	2558	

Measures calculated for negative class are as follow:
Accuracy = 0.763 = 76.3%
Precision = 1
Recall = 0.383
F Measure = 0.55

4.1.3. Neutral class. Here the confusion matrix is created by taking neutral class in consideration and the accuracy, recall, precision, F-measure are computed. Our lexicon based approach of sentiment analysis predicts neutral class with 57.6% accuracy. For neutral class, TP (true positive) is the intersection of actual neutral and predicted neutral. FP (false positive) is the sum of the values in the corresponding column, whereas FN (false negative) is the sum of values in the corresponding row excluding value of TP in both cases. TN (true negative) is the sum of all the values excluding the row and column containing neutral class. The column matrix with respect to neutral class is depicted below.

Table 4: Confusion matrix for neutral class

n = 5031	Predicted Negative	Predicted Positive	Predicted Neutral	
Actual Negative	TN = 739	TN = 0	FP = 1193	1932
Actual Positive	TN = 0	TN =1734	FP = 939	2673
Actual Neutral	FN = 0	FN = 0	TP = 426	426
	739	1734	2558	

Measures calculated for neutral class are as follow:
Accuracy = 0.576 = 57.6%
Precision = 0.167
Recall = 1

F Measure = 0.286

4.1.4. Results with 5031 tweets. The results with this dataset show that positive class is predicted by lexicon more accurately as compared to negative and neutral classes. The accuracies of positive, negative and neutral classes are not encouraging because there is a huge difference in their actual and predicted values. The reason of bad performance is that the lexicon created in our approach does not handle negation which are words like 'not' in English and 'nahi, nai, nae, nayi, ni (nəhi, nəi, nəi, nəi, ni" not/no")' in Roman Urdu. Therefore, lexicon based approach is unable to predict tweets containing mixed sentiments correctly and labels them as neutral. Moreover, for this dataset, labeling has been performed by only one person contributing to the poor performance to some extent.

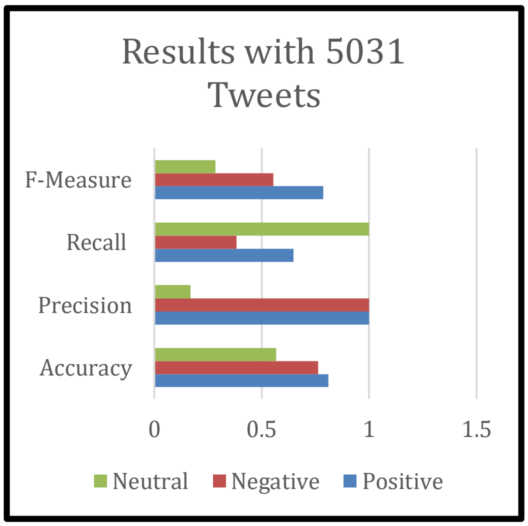

Figure 2: Results with 5031 tweets

4.2. Dataset with 4177 tweets containing positive, negative and neutral classes

To improve the performance of our proposed approach the tweets containing mixed sentiments and negation words discussed earlier are removed. Moreover, for this experiment crowdsourcing technique has been used to perform data labeling of tweets. In this technique labeling from more than one person is taken into consideration and one final labeling is deduced from those labeled tweets. The confusion matrix is then built using these manually labeled tweets along with tweets where sentiment score has been computed using domain specific lexicon.

4.2.1. Positive class. Here, the confusion matrix is created by taking positive class in consideration and the accuracy, recall, precision, F-measure are calculated. The lexicon based approach predicts positive class with 98% accuracy. Confusion matrix for this class is depicted below which is constructed like positive class with 5031 tweets given in the previous section.

Table 5: Confusion matrix for positive class

n = 4177	**Predicted Negative**	**Predicted Positive**	**Predicted Neutral**	
Actual Negative	TN = 417	FP = 0	TN = 0	417
Actual Positive	FN = 81	TP = 1411	FN = 0	1492
Actual Neutral	TN = 150	FP = 0	TN = 2118	2268
	648	1411	2118	

Measures calculated for positive class are as follow:
Accuracy = 0.98 = 98%
Precision = 1
Recall = 0.946
F Measure = 0.972

4.2.2. Negative class. Here, the confusion matrix is created by taking negative class in consideration and the accuracy, recall, precision, F-measure are calculated. Our lexicon based approach predicts negative class with 94% accuracy. Confusion matrix for this class is depicted below which is constructed like negative class with 5031 tweets given in the previous section.

Table 6: Confusion matrix for negative class

n = 4177	**Predicted Negative**	**Predicted Positive**	**Predicted Neutral**	
Actual Negative	TP = 417	FN = 0	FN = 0	417
Actual Positive	FP = 81	TN = 1411	TN = 0	1492
Actual Neutral	FP = 150	TN = 0	TN = 2118	2268
	648	1411	2118	

Measures calculated for negative class are as follow:
Accuracy = 0.94 = 94%
Precision = 0.64
Recall = 1
F Measure = 0.783

4.2.3. Neutral class. Here, the confusion matrix is created by taking neutral class in consideration and the accuracy, recall, precision, F-measure are calculated. Our lexicon based system predicts neutral class with 96% accuracy. Confusion matrix for this class is depicted below which is constructed like neutral class with 5031 tweets given in the previous section.

Table 7: Confusion matrix for neutral class

n = 4177	Predicted Negative	Predicted Positive	Predicted Neutral	
Actual Negative	TN = 417	TN = 0	FP = 0	417
Actual Positive	TN = 81	TN = 1411	FP = 0	1492
Actual Neutral	FN = 150	FN = 0	TP = 2118	2268
	648	1411	2118	

Measures calculated for neutral class are as follow:
Accuracy = 0.96 = 96%
Precision = 1
Recall = 0.934
F Measure = 0.966

4.2.4. Results with 4177 tweets. Results show improved accuracies in all three classes on this reduced dataset as compared to the previous dataset, with highest accuracy achieved for positive class. Accuracy for positive, negative and neutral classes is better with this dataset because there is less difference in their actual and predicted values due to elimination of tweets with negation words and mixed sentiments. Moreover, crowd sourcing has been performed for manual labeling instead of using a single resource person for labeling to remove any bias and impact the result positively.

.

4.3. Comparison of proposed approach with related work

In related work highest accuracy of 89% has been achieved by [5] that propose an approach to analyze sentiments depicted in news comments using domain specific lexicon consisting of 250 words. In the approach presented here, accuracy of 98% for positive class, 94% for negative class and 96% for neutral class is achieved with dataset containing 4177 tweets. This dataset includes positive, negative and neutral tweets and excludes tweets expressing mixed sentiments or tweets including negation. However, for the larger dataset of 5031 tweets with all kinds of sentiments expressed, the results show accuracy of 81%, 76.3% and 57.6% for positive, negative and neutral class respectively.

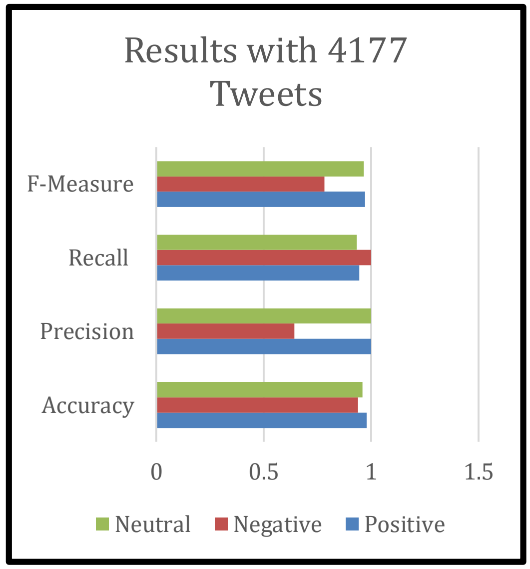

Figure 3: Results with 4177 Tweets

5. Conclusion and Future Work

People in the subcontinent mostly use Urdu language but due to the unavailability of Urdu keyboards, Roman script is used to write Urdu language which is known as Roman Urdu. In this study, bilingual sentiments expressed in tweets including English and Roman Urdu are analyzed. The analysis has been performed by building domain specific bilingual lexicon to assign sentiment scores. Extensive experimentation has been carried out by considering different kinds of tweets. Better results are achieved with tweets dataset containing specific sentiments i.e. positive, negative or neutral as compared to mixed sentiments or sentiments making use of negation. We now aim to further build our lexicon making it more generic and applying the proposed approach to a larger dataset for further validation. We are already in the process of using hybrid technique combining Machine Learning and lexicon for identifying sentiments expressed in bilingual user generated content. Work can also be done to handle negation and to analyze those sentences which include multiple kinds of sentiments.

6. References

[1] M. Kamran and K. Moin , "Sentiment Classification of Customer's Reviews About Automobiles in Roman

Urdu," in Future of Information and Communication Conference, 2018, pp. 630--640.
[2] G. Saqib Muhammad and S. Tariq Rahim, "Twitter and Urdu," in 2018 International Conference on Computing, Mathematics and Engineering Technologies – iCoMET 2018, 2018.
[3] M. Khawar, E. Daryl and S. Kamran, "Sentiment Analysis System for Roman Urdu," in Intelligent Computing, vol. AISC 858, 2018, pp. 29--42.
[4] Javed and A. Hammad, "Opinion Analysis of Bi-Lingual Event Data from Social Networks," in ESSEM@ AI * IA, 2013, pp. 164--172.
[5] M. Alejandro, R. M, C. JL and Z. Jose Manuel, "Lexicon-based comments-oriented news sentiment analyzer system," Expert Systems with Applications, vol. 39, pp. 9166--9180, 2012.
[6] S. A. ALi , M. D. Noor, M. A. Javed , M. M. Aslam and O. A. Khan, "Salience Analysis of NEWS Corpus using Heuristic Appoach in Urdu Language," International Journal of Computer Science and Network Security, vol. 16(4), p. 28, 2016.
[7] N. Farah and H. B. Sayyada, "Sentiment Analysis to predict election results using Python," in 2018 2nd International Conference on Inventive Systems and Control (ICISC), 2018, pp. 1259--1262.
[8] B. Muhammad, I. Huma, S. Muhammad and K. Amin, "Sentiment classification of Roman-Urdu opinions using Naive Bayesian, Decision Tree and KNN classification techniques," Journal of King Saud University-Computer and Information Sciences, vol. 28(3), pp. 330--344, 2016.
[9] S. Afraz Z., A. Muhammad and M.-E. Ana Maria, "Lexicon Based Sentiment Analysis of Urdu Text Using SentiUnits," in Mexican International Conference on Artificial Intelligence, springer, 2010, pp. 32--43.
[10] H. Ahmed , K. Hoda and M. Walaa, "Sentiment analysis algorithms and applications: A survey," Ain Shams engineering journal, vol. 5(4), pp. 1093--1113, 2014.