Filipino Tweets Sentiment Analysis using Convolutional Neural Network with Distant Supervision

Aleimar Villabrille
Department of Mathematics, Physics and Computer Science
University of the Philippines Mindanao
apvillabrille@up.edu.ph

Maureen Agrazamendez
Department of Mathematics, Physics and Computer Science
University of the Philippines Mindanao
mdagrazamendez@up.edu.ph

***Abstract*— Philippines being labelled as the social media capital of the world means that Filipinos spend their time in the internet for recreation and entertainment purposes. Social media is the main avenue of people to share stories and express their sentiments in various topics ranging from politics to media personalities. All these sentiments suggest that there is a huge amount of Filipino sentiment data out there waiting to be converted into valuable information. Current approaches used in natural language processing for Filipino language is the classic machine learning algorithms such as Support Vector Machine and Naïve Bayes which yielded low accuracy or use imbalanced data set for their training. In this paper, the researcher explored Convolutional Neural Networks by analyzing sentiments of Filipino Tweets. The network was based performs binary classification and initially labelled their tweets based on their emoticons. For the architecture to accept tweets, before training, the tweets were converted to vectors using word2vec. The network was trained to 7,741 tweets and tested to 1,936 tweets. The model yielded with accuracy of 74.68% accuracy and was compared to Naïve Bayes that yielded 69.49% by the same dataset. This small margin between the classic machine learning algorithm means that there is a potential in exploring CNN architecture for various natural language processing problems for Filipino languages.**

Keywords—convolutional neural network, deep learning, Filipino tweets, sentiment analysis

I. Introduction

Sentiment analysis aims to give computers ability of understanding opinions or emotions expressed through a written text. Sentiment analysis also portray a role in understanding how human being deal with text data [1]. When used properly, sentiment analysis can be use in a wide spectrum of application ranging from academic, scientific, business, economics or even politics. For example, analyzing customers' sentiment through online product reviews [2]. Sentiment analysis was also used by [3] for correlating twitter responses on companies, with the movements of prices in stock market. Twitter was one of the major social networking accounts that's currently used by 9.5 million Filipino users [4] making the Philippines in the eighth for the most popular twitter users globally [5]. This makes Filipino tweets a vast opportunity for research including sentiment analysis. Currently, there have been very few studies that analyzes emotions of Filipino tweets. Some studies conducted produces either use of imbalanced data for training or low accuracy of sentiment prediction [6]. Thus, the reason this study was conceptualized.

II. Objectives

This study aimed to apply Convolutional Neural Network designed by [7] to be applied in Filipino language using Filipino tweets. This study made use of distant supervised learning of data set proposed by [8] since there is no publicly available corpus for Filipino tweets sentiment analysis.

The study tested accuracy of convolutional neural networks applied in Filipino language. This accuracy was also compared to the classic Naïve Bayes Model. The study specifically aimed to:

1. use convolutional neural network (CNN) proposed by [7] in classifying positive and negative sentiments of Filipino tweets;
2. use distant supervision for training data of tweets proposed by [8] that uses emoticons for labelling i.e. ":)" in a tweet indicates positive sentiment while ":(" indicates negative sentiment;
3. determine accuracy of CNN approach in sentiment analysis of Filipino tweets;
4. compare Convolutional Neural Network approach to novel Naïve Bayes Classifier approach in terms of accuracy.

The study was limited only to Filipino tweets and also code-mixed language like Filipino-English tweets since these types of language is widely used by Filipino twitter users. The study did not test performance of CNN in other Filipino language like Cebuano, Ilonggo and others. Future studies will be conducted for addressing CNN performance in these languages.

The study also focused on analyzing polarity of sentiments of tweets. This means that only sentiments, either positive or negative, was analyzed. The study did not detect neutral tweets nor any form of sarcasm or any complexity behavior of human language i.e. figurative languages.

This study aimed to provide overview of convolutional neural network performance applied to Filipino language and further studies could be developed to the Natural Language

Processing community in the whole country. The architecture of the network could then be modified to address the specifics of Filipino language compared to English and other languages.

Lastly, businessman, economists, politicians could use the model to analyze sentiments and reaction of the Filipino masses in twitter community regarding their products, stock market and support on certain matters. The study was purely an introduction of CNN applied to the Filipino language thus, future studies for development could be applied in the model.

III. Review of Related Literature

A. Twitter Data

Twitter is a popular social networking site that allows users to post anything and follow trending topics locally and even worldwide. In average, twitter makes a big source of data because of its 500 million tweets per day [9]. In Philippines alone, there is approximately 9.5 million twitter users [4]. Looking at these numbers, this makes up a vast data resource for studies and analysis in text data specifically for Filipino languages. This huge amount of data was used by [10] in analyzing twitter themes during election season. Different themes are generated in their study regarding election related tweets such as experience after voting, questions and explanations, encouragement and campaign, names of candidate, voting process, humor, fantasies, organizations and criticisms on personal appearance. This potential of Twitter as a major social media influencer draws attraction to sentiment analysis research globally.

B. Sentiment Analysis in the Filipino Tweets

Various studies used Filipino tweets as their primary source of data, however, there exists very few studies in sentiment analysis Filipino tweets. [11] used Filipino tweets to analyse behaviour of Filipinos during a disaster by differentiating direct victims and observers during a calamity. [12] used Filipino tweets to identify tourism related tweets while generating a Mapbox for visualizations of tweets' geolocation. [12] also used Support Vector Machines in classifying polarity of emotions in these tourism-related tweets. In terms of solving sentiment analysis problem using Filipino tweets, existing approaches yields low accuracy or use imbalanced data set for their training [6].

C. Distant Supervision Sentiment Classification

[8] used distant supervision for twitter sentiment classification. The study stripped emoticons thus forcing the classifier to learn from other features. Labelling through emoticons was done in way that ":)", ":-)", ":)", ":D", "=)" were labelled as positive and ":(", ":-(", ": (" emoticons were labelled as negative. Their study showed that using emoticons as noisy labels for training data in sentiment classification, yields effectivity and achieve high accuracy to machine learning algorithms such as Naïve Bayes, Maximum Entropy Classification and Support Vector Machine. Traditional machine learning techniques in sentiment analysis works well only when test data matched with the topic. [13] used emoticons to identify sentiment classification and demonstrated that data labelled with emoticons has the potential of being independent from domain, topic and time.

D. Neural Network Approach

Neural Networks mimics how human brain function. It uses cascade of multiple non-linear processing units to extract features and transformation. Each successive layer output of previous layer as input which is biologically inspired from behavior of neurons in human brains. Recent studies show impressive performance of deep learning in vision [14], speech recognition [15] and other applications. In sentiment analysis, researchers are now using deep learning model for better accuracy of sentiment prediction. More importantly neural networks address the weakness of Naïve Bayes Classifier as it shows connections between sequential such as text data. [16] stated that neural networks for natural language processing revolutionized the whole field. Through series of works by [17][18][19][20] obtained syntactic parsing through a fully connected feed forward network. A lot more success stories in natural language processing are achieved with the help of neural networks, this includes document classification [21], short text categorization [22], sentiment classification [23] [24], relation type classification between entities [25], event detection [26], paraphrase identification [27], semantic role labelling [28], question answering [29], predicting box-office revenues of movie based on critic reviews [30], modelling text interestingness [31], modeling relations between character-sequences and part-of-speech tagging [32].

IV. Methodology

The Convolutional Neural Network was applied to create a model for sentiment analysis of Filipino tweets. This study went to different stages from tweet gathering, tweet labelling, tweet preprocessing, transformation of words to vectors, training, and testing stages Figure 1.

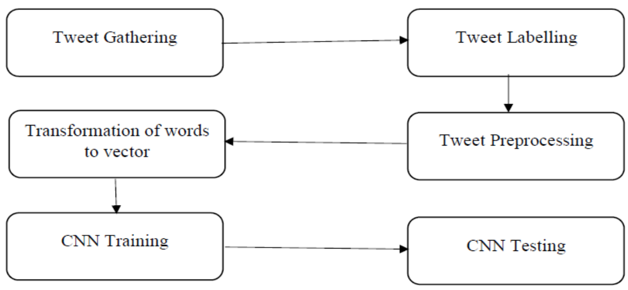

Figure I. General Flow of Methodology.

A. Tweet Gathering

Tweets are gathered through Python Script built with Tweepy. Tweepy is an easy-to-use Python library for accessing Twitter API. To stream tweet, the study requested credentials from twitter to allow data gathering for research purposes. Tweets streamed on the Philippines were temporarily save to a JSON file for other filtration and preprocessing. After streaming tweets during a span of 7 – 9 days as suggested by [34], the study gathered 362,247 English tweets and 711,068 Filipino tweets. The whole data was saved to a JSON file which approximately used 4GB of memory. For this research, English tweets were used for further processes.

B. *Tweet Labelling*

There is no Filipino tweet corpus to train the CNN architecture thus the study used distant supervision for training data set. The script searched for tweets with emoticon and classified them based on it. Tweets with smileys ':)', ':-)', ':D' were labelled as 1 (positive) and ':(', ':-(', ':[' as 0 (negative). After extracting tweets with emoticons, the dataset was filtered from 362,247 unlabeled tweets to 34,536 labelled tweets. The dataset contains 24,860 negative tweets and 9,676 positive tweets.

C. *Tweet Preprocessing*

Before the tweets were converted into vectors through the process of word2vec, the data was cleaned through removing unnecessary elements in the tweets. This data cleansing process included removal of URLs in tweet, hashtags, emoticons, usernames and lowercasing the tweets as this was the standard preprocessing procedure that was also used by [7] for Italian tweets sentiment analysis. Sample of preprocessed tweet is shown in Table 1.

TABLE I. SAMPLE PREPOCESSED TWEET

Raw Tweet	***Preprocessed Tweet***
@xxgrei I'll wait for that :)	i ll wait for that
@aaaaahsatan te kakahalf day lang namin kahapon... :-(pagisipan namin kung love ka talaga naming	te kakahalf day lang namin kahapon pagisipan naming kung love ka talaga namin
@loeyyy1485 @weareoneEXO thank you kanina!! :)	thank you kanina

D. *Word2Vec*

Word2vec is an open source, follows the Apache License 2.0 open source license, tool proposed by Google in 2013. Word2vec translates text to vectors that CNN can understand. Word2vec creates features without human intervention thus given enough data, usage and context, it can make highly accurate guess about a word's meaning. One advantage of word2vec is that it runs fast even in big dataset. The model used skipgram-model implementation with window-size of 5 and d=52 dimensions of word vectors.

Tweets in the dataset have at most 19 length, and dimension for vectorization is set to 52 as [7] implemented the same. Gensim package was used to implement the skipgram model. Then, the study created a 2-dimensional Principal Component Analysis (PCA) using scikit-learn. The resulting projection was plotted using matplotlib Figure 2 which demonstrated that the token "love" is closer to "sakit" than the "happy" token.

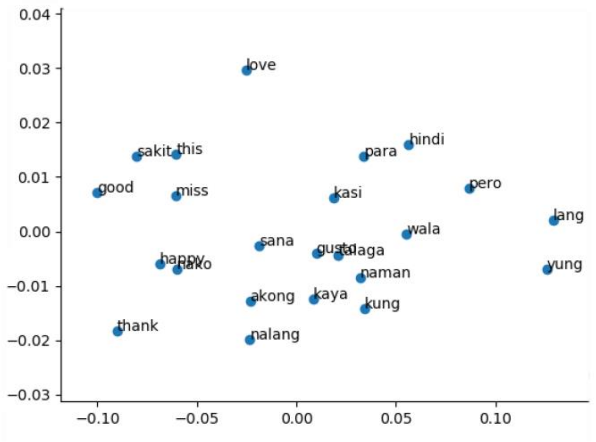

Figure II. Scatter Plot of PCA Projection of Word2Vec Model..

V. CONVOLUTIONAL NEURAL NETWORK

Convolutional Neural Network (CNN) basically contains several layers of convolution for feature extraction. Unlike Recurrent Neural Networks, CNN has lesser parameters and connections which makes it easier to train. Convolutional Neural Network was primarily used for computer vision and gives major breakthroughs in image analysis and classification. CNN was then applied two Natural Language Processing (NLP) problems such as sentiment analysis and yield successful results [24]. CNN was also applied to various languages for sentiment analysis and showed promising accuracy results.

The Convolutional Neural Network architecture used in the study is a 2-layer CNN inspired by [7] architecture. This was demonstrated in Figure 3 and is described as follows:

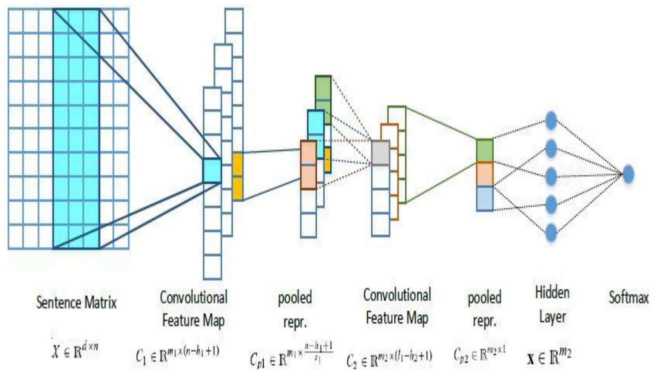

Figure III. Convolutional Neural Network Architecture.

A. Sentence Representation

The input of our model are tweets associated to a d-dimensional vector representation. Each tweet was represented through concatenation of n constituent words. The matrix formed by this is $T \in \mathbb{R}^{d \times n}$ that serves as the input of the convolutional neural network. Then the first layer of this network will be a looked up in a word embedding $X \in \mathbb{R}^{d \times |V|}$ formed by concatenating embeddings of all words in vocabulary V. This simply means that the i-th column of X represents the i-th word in vocabulary V.

B. Convolutional Layer

The goal of this layer is extracting patterns found within the training phase. Formally, a set of m filters is applied to a sliding window of length h over each tweet. The concatenation of word vectors s_i to s_{i+h} was then be denoted as $S_{[i:i+h]}$. A feature ci was generated for a given filter F by:

$$ci := \Sigma(S_{[i:i+h]})k,j * \boldsymbol{F}_{k,j} \qquad (1)$$

The concatenation of vectors in a sentence produced a feature vector $c \in \mathbb{R}^{n-h+1}$. Combining all vectors in c with m filters would then yield a map matrix $C \in \mathbb{R}^{m \times (n-h+1)}$. These features was then learned on the next phase of the neural network.

C. Max Pooling

To enable the learning of non-linear decision boundaries, each convolutional layer was typically followed by a non-linear activation function before entering the pooling layer. This was through combining vector elements by taking maximum over a fixed set if non-overlapping intervals. The resulting pooled feature map matrix was in the form:

$$cpooled \in \mathbb{R}^{m \times n - h + 1/s} \qquad (2)$$

D. Hidden Layer

A fully connected hidden layer computes the transformation of $\alpha(W * x + b)$ where $W \in \mathbb{R}^{m \times m}$ is the weight of the matrix, $b \in \mathbb{R}^m$ for bias, and α the rectified linear unit (ReLU) function defined max(0, x) to ensure feature maps are always positive.

E. Softmax

Lastly, the outputs of hidden layer $x \in \mathbb{R}m$ are connected to softmax regression layer which $\hat{y} \in [1, K]$ with largest probability,

$$\hat{y} := \arg max_j((e^{xTwj+aj}) / \Sigma_{k=1}[e^{xTwk+ak}]) \qquad (3)$$

where w_j denotes the weights vector of class j and a_j the bias of class j.

F. Dropout

To overcome overfitting, the study used dropout regularization method [35]. On training stage, it randomly drops neurons with probability p, reduced neural network is updated and dropped nodes are re-inserted. The model applied dropout to hidden layer and output layer with p=0.5.

G. Convolutional Neural Network architectures

The study explored different hyperparamaters of the main convolutional neural network. It implemented and compared accuracy of [24], [7], two different variants of 1-layer CNN and 2-layer CNN. Table 2 shows different hyperparameters in different architectures.

TABLE II. SUMMARY OF HYPERPARAMETERS

Architecture	*Number of Layers*	*Number of Filters*	*Filter window size (h)*	*Size of max-pooling intervals and striding*
Yoon (2014)	1	100	h=[23,4]	None
Deriu (2015)	2	200	h=5, h=5	w=3, st=2
L1A	1	300	h=5	None
L2A	1	200	h=5	None
L1B	2	200	h=4, h=3	w=4, st=2
L2B	2	200	h=6, h=4	w=6, st=2

VI. RESULTS AND DISCUSSION

The Convolutional Neural Network was used for sentiment analysis of Filipino tweets. From 34, 536 labeled tweets using distant supervision, the model used only 19,353 tweets. Since positive tweets are only 9,767 tweets, the researcher balanced the number of negative tweets by randomly picking 9,767 negatively labelled tweets. Thus, the study used the 19,353 tweets. Then, the researcher split the dataset into training and testing data. 80% or 7,741 tweets were used for training and 20% or 1936 tweets for testing. The model only accepted a sentence matrix therefore before training the researcher created word embeddings for each tweet through loading the pre-trained word2vec model. Each unique word has an equivalent vector value which was used to transform text to sequences of numbers. Since the model can't use sequences of words with different length, the study searched the maximum length of sequence from the training data. The maximum length of sequences resulted to 19, then the researcher padded zeroes to sequences below 19 until they all have equal length. Though, there are only two classifications for the model, the study followed [7] architecture which uses a categorical cross entropy for the loss function then connected to an output layer with softmax activation.

A. Training Architectures

The researcher used mini-batch gradient descent of 32 for all CNN model variations and plot loss and accuracy of training and validation. This was demonstrated through Figure 4 for Yoon (2014), Figure 5 for L1A, Figure 6 for L1B, Figure 7 for L2A, Figure 8 for L2B and Figure 9 for Deriu (2015).

Figure IV. Training and validation accuracy and loss of Yoon (2014) Architecture using Batch Size 32.

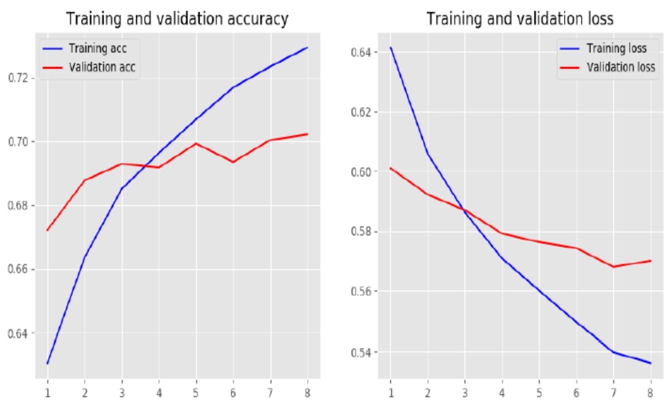

Figure V. Training and validation accuracy and loss of L1A Architecture using Batch Size 32.

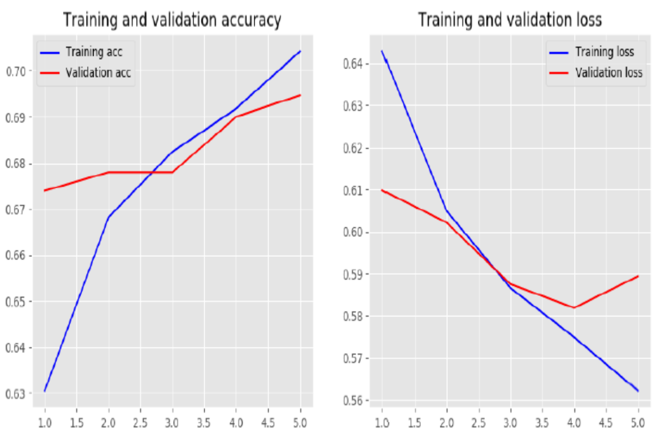

Figure VI. Training and validation accuracy and loss of L1B Architecture using Batch Size 32.

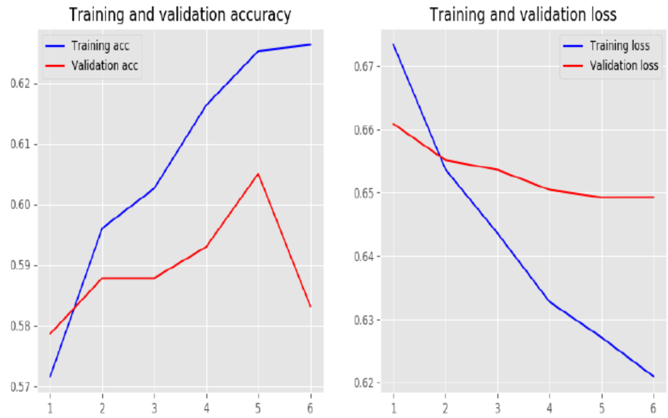

Figure VII. Training and validation accuracy and loss of L2A Architecture using Batch Size 32.

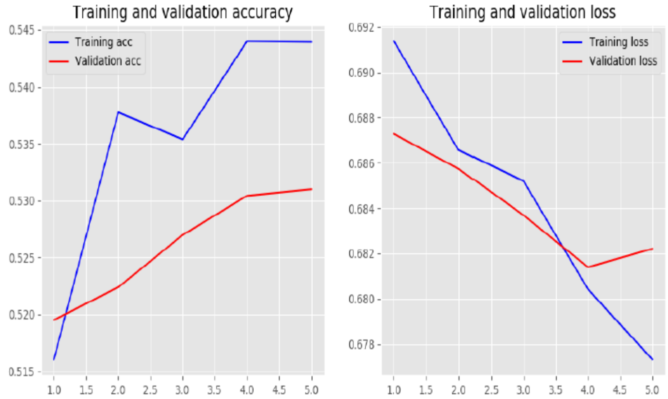

Figure VIII. Training and validation accuracy and loss of L2B Architecture using Batch Size 32.

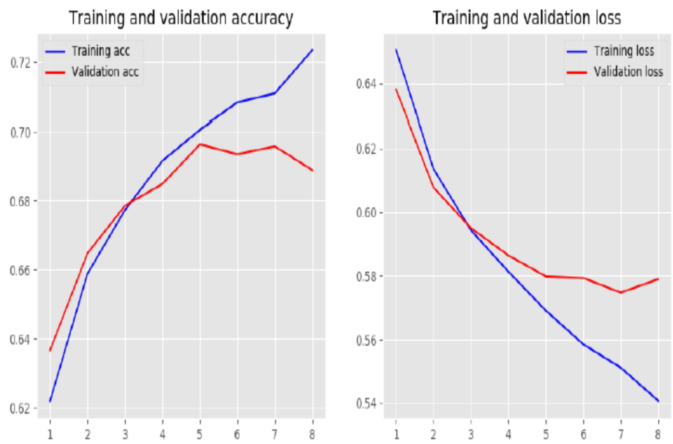

Figure IX. Training and validation accuracy and loss of Deriu (2015) Architecture using Batch Size 32.

The researcher computed the accuracy of each model through the testing set. Yoon's (2014) model performed poorly with only 0.5018 accuracy. L1A model stood-out with accuracy of 0.7636 followed by Deriu (2015) model of 0.7468 accuracy. L1B model followed third by 0.7378 accuracy, then by L2A model with 0.6487 then L2B model with 0.5643 accuracies. Meanwhile, a Unigram Naïve Bayes Classifier was implemented through sci-kit learn and produced accuracy of 0.6949 accuracy using the same training and testing data set. The researcher observed on Table 3 that the accuracy of negative tweets peaked at L1B model with 71.26% accuracy followed by L1A model of 69.93% accuracy then by Deriu (2015) model of 69.54%. For positive tweets, L1A model yielded 71.03% accuracy followed by Deriu (2015) model of 70.80%. L1A model gave the highest mean accuracy with 70.48% followed by Deriu (2015) model with 70.17%. Table 3 also demonstrated that some CNN architecture performed low accuracy compared to the classic Naïve Bayes method. It was also observed that in high performing CNN architectures, like L1A, L1B and Deriu (2015), there is a little difference in negative and positive accuracy compared to Naïve Bayes model. Specifically, there is only a margin of 1.42% accuracy in negative and 0.76% accuracy on positive labels compared to Naïve Bayes model. Table 4 shows tweets misclassified as negative by Deriu (2015) model while 5 shows tweets misclassified as positive by the same architecture.

TABLE III. SENTIMENT ANALYSIS PERCENT ACCURACY OF CNN AND UNIGRAM NAÏVE BAYES.

Model	*Negative*	*Positive*	*Mean*
Naïve Bayes	69.84	70.27	70.06
Yoon (2014)	56.53	44.29	50.41
Deriu (2015)	69.54	70.80	70.17
L1A	69.93	71.03	70.48
L1B	71.26	66.08	68.67
L2A	61.44	58.61	60.03
L2B	63.31	43.21	53.26

TABLE IV. SAMPLE OF NEGATIVE TWEETS CLASSIFIED AS POSITIVE.

Number	*Tweet*
1	blessed thanks
2	good night
3	Just home alone walang pero thanks loyal dogs feeling secured naman

TABLE V. SAMPLE OF POSITIVE TWEETS CLASSIFIED AS NEGATIVE.

Number	*Tweet*
1	plus sadly dangerous
2	miss your chats
3	promises meant broken

We observed that not all CNN model outperformed CNN specifically the architecture of Yoon (2014). We also observed that CNN models do have a potential in beating accuracy of current approaches such as Naïve Bayes particularly the L1A architecture and Deriu (2015) architecture. A margin of 6.87% is a significant value to explore CNN architectures.

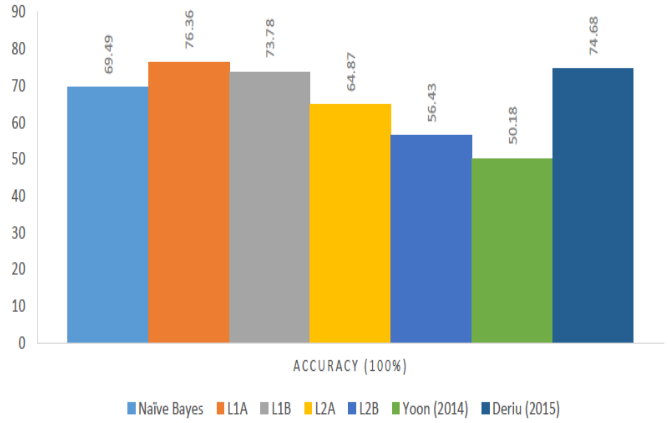

Figure X. Summary of accuracies of different CNN-model and Naïve Bayes.

VII. CONCLUSION

Sentiment analysis aimed to use computers in analyzing human emotions or sentiment. With the Philippines labelled as the social media capital of the world, there rests a massive amount of Filipino sentiment that could be used to understand culture, preferences and improve natural language processing for Filipino language. With recent advancement of technology and artificial intelligence, studies like Natural Language Processing for Filipino language seems to have a big gap. Current approaches of sentiment analysis of Filipino language was still through classic machine learning algorithms like Support Vector Machine and Naïve Bayes. Recently, [6] explored Recurrent Neural Network for sentiment analysis of Filipino tweets but a more prestigious neural network architecture, the Convolutional Neural Network, was the latest approach for sentiment analysis in different languages globally.

This study used convolutional neural network used by [7] for sentiment analysis in Italian tweets and applied it to Filipino tweets. Distant supervision was used to label tweets for training and testing as proposed by [8] which simply used emoticons as indicator of positive or negative sentiment. After using CNN for sentiment analysis on different number of layers and hyperparameters, the study observed that while not all CNN model beat the classic Naïve Bayes approach for sentiment analysis, CNN architectures like Deriu and a 1-layered CNN implemented in the study topped Naïve Bayes for 6.78%. This small margin could be an investment for future studies in improving the CNN architectures presented and could make possible breakthrough in sentiment analysis for Filipino languages.

VIII. Future Directions

In the future, it is suggested to observe dropout values and other hyperparameters to construct the best CNN architecture for Filipino language. Instead of distant supervision, it is more ideal to manually label tweets since emoticons are not common in all tweets. It is also suggested that instead of early stopping to monitor the model, a cross-validation would be a more reliable way to detect if the model is overfitting. Also, a standardized Filipino tweet corpus should be made to benchmark performances of different machine learning and neural network approaches in sentiment analysis. Lastly, it could be a promising field to expand the number of classifications into more specific emotions.

Acknowledgment

I would like to thank my adviser for letting me explore my limits through studying neural networks, machine learning and natural language processing. My family for their continuous support. And God the creator who gave me strength to overcome struggles and difficulties throughout my undergraduate study.

References

[1] Ouyang, X., Zhou, P., Hua Li, C., Liu, L., 2015. Sentiment Analysis using Convolutional Neural Network, in: International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing, Liverpool, UK, pp. 2359 - 2364.

[2] Rambocas, M., Gama, J., 2013. Marketing research: The role of sentiment analysis, Universidade do Porto, Portugal.

[3] Pagolu, V., Challa, K., Panda, G., Majhi, B., 2016. Sentiment Analysis for Twitter Data for Predicting Stock Market Movement, in: International Conference on Signal Processing, Communication, Power and Embedded Systems, India, pp. 1345 – 1350.

[4] Montecillo, P., 2012. Philippines has 9.5M twitter users, ranks 10th [WWW Document]. Philippine Inquirer. URL https://technology.inquirer.net/15189/philippines-has-9-5m-twitter-users-ranks-10th (accessed 10.17.2018).

[5] Russell, J., 2011. Philippines named social networking capital of the world [WWW Document]. Asian Correspondent. URL https://asiancorrespondent/54475/philippines-named-the-socialnetworking-capital-of-the-world-indonesia-malaysia-amongst-top-10/ (accessed 10.17.2018).

[6] Mina, R.J.L., Atienza, R., 2017. Sentiment Analysis of Filipino Tweets Using Recurrent Neural Network. University of Philippines Diliman, Philippines.

[7] Derui, J., 2016. Sentiment Analysis using Deep Convolutional Neural Network, Department of Computer Science, Eidgenössische Technische Hochschule Zürich, Zürich, Swirzerland.

[8] Go, A., Bhayani, R. & Huang, L., 2009. 'Twitter Sentiment Classification using Distant Supervision', Processing, 1 – 6.

[9] Krikorian, R., 2013. New Tweets per second record, and how! [WWW Document]. Twitter. URL https://blog.twitter.com/2013/new-tweets-per-second-record-and-how/ (accessed 10.17.2018).

[10] Cheng, C. K., Roxas, R., Pablo, Z. , Roldan, M. Z. & Oco, N. . (2015). Paggamit ng Natural Language Processing bilang Gabay sa Pagtuklas at Pagsiyasat ng Tema sa mga Tweet tuwing Halalan / Using Natural Language Processing in the Discovery and Analysis of Themes of Tweets during Elections. MALAY. 27, 91 – 101.

[11] Lee, J.B., Ybañez, M., De Leon, M., & Estuar, M. R. (2013). Understanding the Behavior of Filipino twitter User during Disaster.

[12] Menchavez, J. C., Espinosa, K. J., 2015. Fun in the Philippines: Automatic Identification and Sentiment Analysis of Tourism-related Tweets, in: 15th International Conference on Data Mining Workshops, New Jersey, USA, pp. 660 - 667.

[13] Read, J., 2005. Using emoticons to reduce dependency in machine learning techniques for sentiment classification, in: ACLstudent '05 Proceedings of the ACL Student Research Workshop, Ann Arbor, Michigan, pp. 43 – 48.

[14] Lee, A., 2016. Comparing Deep Neural Networks and Traditional Vision Algorithms in Mobile Robotics.

[15] Gevaert, W., Tsenov, G., Mladenov, V., 2010. Neural Networks used for Speech Recognition. Journal of Automatic Control. 20, pp. 1 – 7.

[16] Goldberg, Y., 2017. Neural Network Methods for Natural Language Processing. Morgan and Claypool (Synthesis Lectures on Human Language Technologies). Volume 44, Issue 1, pp. 0–2.

[17] Chen, D., Manning, C., 2014. A Fast and Accurate Dependency Parser using Neural Networks, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, pp. 740 – 750.

[18] Durrett, G., Klein, D., 2015. Neural CRF Parsing, in: Proceedings of the 53rd Annual Meeting of the Association of Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Beijing, China, pp. 302 – 312.

[19] Pei, W., Ge, T., Chang, B., 2015. An effective Neural Network Model for Graphbased Dependency Parsing, in: Proceedings of the 53rd Annual Meeting of the Association of Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Beijing, China, pp. 313 – 322.

[20] Weiss, D., Alberti, C., Collins, M., Petrov, S., 2015. Structured Training for Neural Network Transition-Based Parsing, in: Proceedings of the 53rd Annual Meeting of the Association of Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Beijing, China, pp. 323 – 333.

[21] Johnson, R., Zhang, T., 2015. Effective use of Word Order for Text Categorization with Convolutional Neural Network, in: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, pp. 103 – 112.

[22] Wang, P., Xu, J., Xu, B., Liu, C., Zhang, H., Wang, F., Hao, H., 2015. Semantic Clustering and Convolutional Neural Network for Short Text Categorization, in: Proceedings of the 53rd Annual Meeting of the Association of Computational Linguistic and the 7th International Joint Conference on Natural Language Processing, Beijing, China, pp. 352 – 357.

[23] Kalchbrenner, N., Grefenstette, E., Blunsom, P., 2014. A Convolutional Neural Network for Modelling Sentences, in: Proceedings of the 52nd Annual Meeting of the Association of Computational Linguistic, Baltimore, Maryland, pp. 655 – 665.

[24] Yoon K., 2014. Convolutional Neural Networks for Sentence Classification, in: EMNLP 2014 – Empirical Methods in Natural Language Processing, Doha, Qatar, pp. 1746 – 1751.

[25] dos Santos, C., Xiang, B., Zhou, B., 2015. Classifying Relations by Ranking with Convolutional Neural Networks, in: Proceedings of the 53rd Annual Meeting of the Association of Computational Linguistic

and the 7th International Joint Conference on Natural Language Processing, Beijing, China, pp. 626 – 634.

[26] Chen, Y., Xu, L., Liu, K., Zeng, D., Zhao, J., 2015. Event Extraction Via Dynamic Multi-Pooling Convolutional Neural Networks, in: Proceedings of the 53rd Annual Meeting of the Association of Computational Linguistic and the 7th International Joint Conference on Natural Language Processing, Beijing, China, pp. 167 – 176.

[27] Yin, W., Schutze, H., 2015. Convolutional Neural Network for Paraphrase Identification, in: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, pp. 901 – 911.

[28] Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P., 2011. Natural Language Processing (almost) from scratch, J. Mach. Learn. Res. 12, 2493–2537.

[29] Dong, L., Wei, F., Zhou, M., Xu, K., 2015. Question Answering over Freebase with Multi-Column Convolutional Neural Networks, in: Proceedings of the 53rd Annual Meeting of the Association of Computational Linguistic and the 7th International Joint Conference on Natural Language Processing, Beijing, China, pp. 260 – 269.

[30] Bitvai, Z., Cohn, T., 2015. Non-Linear Text Regression with a Deep Convolutional Neural Network, in: Proceedings of the 53rd Annual Meeting of the Association of Computational Linguistic and the 7th International Joint Conference on Natural Language Processing, Beijing, China, pp. 180 – 185.

[31] Gao, J., Pantel, P., Gamon, M., He, X., Deng, L., 2014. Modelling Interestingness with Deep Convolutional Neural Network, in: EMNLP 2014 – Empirical Methods in Natural Language Processing, Doha, Qatar, pp. 2 – 13.

[32] dos Santos, C., Zadrozny, B., 2014. Learning Character-Level Representations for Part-of-Speech Tagging, in: 31st International Conference on Machine Learning (ICML), Beijing, China, pp. 1818 – 1826.

[33] Mikolov, T., Sutskever, I, Chen, K., Corrado, G., Dean, J., 20013. Distributed representation of words and phrases and their compositionality, in: NIPS'13 Proceedings of the 26th International Conference on Neural Information Processing Systems, Lake Tahoe, Nevada, pp. 3111 – 3119.

[34] Littman, J., 2017. Where to get Twitter data for academic research [WWW Document]. Social Feed Manager. URL https://gwu-libraries.github.io/sfm-ui/posts/2017-09-14-twitter-data (accessed 11.14.2018).

[35] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. J Mach Learn Res. 15, 1929 – 1958.