Philippine e-Journal for Applied Research and Development 9(2019), 27-39 ISSN 2449-3694 (Online)
http://pejard.slu.edu.ph/vol.9/2019.10.01.pdf

A Method of Semi-Supervised Learning using Siamese Neural Network for Disaster Monitoring on Philippine Social Media

Andrew T. Marges[1,*] and Jaime M. Samaniego[2]

[1]Graduate School, University of the Philippines Los Baños, 4031 Los Baños, Laguna; [2]Institute of Computer Science, College of Arts and Sciences, University of the Philippines Los Baños, 4031 Los Baños, Laguna

* *Corresponding author (atmarges@up.edu.ph)*

Received, 23 July 2019; Accepted, 04 September 2019; Published, 01 October 2019

Abstract

Studies have shown the potential of social media as a valuable source of real-time information for disaster monitoring through machine learning. However, the scarcity of labeled data used for training, especially during the onset of a disaster event, leads to poor machine learning output. This study introduced a method of improving the performance of machine learning classifiers by developing proxy labels for unlabeled datasets to increase the amount of training data. The design framework applied different concepts in machine learning including semi-supervised learning, Siamese neural network and transfer learning. The resulting models, together with traditional and deep neural network classifiers, were subjected to a comparative analysis experiment using disaster-related tweets collected within the Philippines as benchmark datasets. Results showed that the proposed method produced better F1-scores as compared to traditional machine learning and deep neural network approaches. Integration of this method to disaster monitoring systems can drastically lessen the need for a large workforce to provide manual labelling on collected social media data, thus, saving on cost, time and resources.

Keywords: Disaster monitoring on social media, machine learning techniques for text classification, transfer learning, word embeddings

Introduction

Social media has come to play an important role in times of emergencies, as people use this to seek help from a wide variety of audiences during disasters and life-threatening situations. Rapid analysis of messages posted on social media platforms such as Twitter provides help for the government and humanitarian organizations to gain situational awareness, learn about urgent needs of affected people, critical infrastructure damage, and medical emergencies [1]. However, manually processing large amounts of real-time information posted on social media networks during times of disasters can cause information overload. This has been one of the major challenges for humanitarians who keep track of social media for the purpose of improving disaster response [2]. As a solution, machine learning techniques designed for text classification, which produced notable results in other fields such as web security, bio-medicine and company resource planning [3], have been explored and have shown promising results in automating such processes to extract useful information regarding a disaster event [4].

Machine learning is a field of study concerned with developing methods that can automatically

detect patterns in data, which can be used to uncover patterns for predicting future data and other outcomes of interests. Tasks in machine learning can be categorized into two main types: supervised and unsupervised [5]. The task of learning a function that maps an input to an output based on a set of input-output training set is called supervised learning [6]. This type of machine learning allows a model to predict future outcomes after training using past labeled data. Unsupervised learning, on the other hand, is the kind of learning that focuses in finding interesting patterns in data. Unlike supervised learning, unsupervised learning approach uses sets of unlabeled data in discovering patterns that can be proven useful upon analysis by an expert [5].

To provide the government and humanitarian organizations specific information needed for disaster monitoring, the ideal approach is the use of supervised learning. However, supervised machine learning algorithms rely heavily on labeled data available for training. Despite the availability of plain unlabeled social media data in huge quantities, the scarcity of labeled data, especially during the earlier stages of a disaster event, hinders machine learning tasks which can cause delay to disaster response [7]. The limited availability of labeled data for training leads to machine learning models with poor performance. Traditional machine learning techniques have shown promising results in automating the process of identifying useful, relevant and trustworthy information in big crisis data [4]. However, majority of studies on utilizing social media for disaster monitoring put little emphasis on the fact that a high frequency of labeled data is not always readily available for training, especially during the earlier stages of a disaster.

Some of the approaches used in machine learning to solve the problem of insufficient availability of training data are semi-supervised learning and transfer learning. Semi-supervised learning takes advantage of the numerous amounts of unlabeled data to augment the knowledge from the supervision information gained using limited sets of labeled data to produce better machine learning models [8]. Its most straightforward way of implementation is called self-training which makes use of proxy labels for unlabeled sets of data in place of ground-truth labels to enable a traditional supervised learning approach. Transfer learning, on the other hand, is a technique used in machine learning which utilizes acquired knowledge gained from one task to solve different but related ones [9].

This study aims to introduce a method of producing proxy labels that contain more information for unlabeled tweets using a type of neural network architecture commonly used for comparison tasks called Siamese neural network to improve the performance of classification models in a semi-supervised learning setup. The goal is to develop an improved classification model that can outperform other machine learning classifiers given a limited set of training data which can be used for disaster monitoring based on Philippine social media. The design framework will utilize the combination of semi-supervised learning, Siamese neural network and transfer learning to produce a new approach towards tackling the problem of scarcity of training data during the onset of a disaster. The resulting models will then be subjected to a comparative analysis experiment using disaster-related tweets collected within the Philippines as benchmark datasets. Non-parametric statistical evaluation tests, specifically Friedman test and Wilcoxon signed rank test, will be used to evaluate the significance of the results. By enabling the construction of more accurate classification models using smaller frequencies of labeled training data, this study shows great potential towards the development of low-cost and improved disaster monitoring systems which can provide the government and humanitarian organizations vital information needed for faster and effective response in aiding the public during dire situations.

Methods

Model development

The core of the proposed method revolved around model development which was composed of a series of stages. For its first stage, the study had three types of input, namely a labeled text dataset $\mathcal{D}_l$, an unlabeled text dataset $\mathcal{D}_u$ and a word embedding model *emb*. The expected model employed a semi-supervised learning approach

to produce better classification models, thus, the need for both labeled and unlabeled data. The word embedding model acted as a means of implementing transfer learning. The model produced basically served as a mapping $emb(w_i)$ of the collection of words $\{w_1,w_2,\ldots,w_1n\}$ to vectors of real numbers where semantically similar words were located near each other. This allowed new classification models to learn vast amount of semantic information that were not available from small training sets.

For the second stage, the input data underwent an initial step of preprocessing involving word filtering based on the collection of words from the word embedding model. This was followed by the proxy labelling stage which mainly revolved around a semi-supervised learning procedure, which learns a classification model from both labeled $\mathcal{D}_l$ and unlabeled dataset $\mathcal{D}_u$ using proxy labels. However, rather than leveraging the classification model trained from the available labeled data to produce weak labels, this study utilized an architecture that learns distance metrics to produce a more informative proxy label. This study built up from the idea that providing proxy labels that contained more information could improve the performance of classifiers in a semi-supervised learning setup. Basically, this study defined a proxy label of an unlabeled input data as its proximity measure in reference to other data containing real labels. With this, we used a neural network architecture that learned distance metrics from pairs of input data called Siamese neural network [10].

Training the Siamese neural network required mapping the existing classification problem into a binary classification task whose goal was to classify if the two inputs were of the same class or not. This required a set of labeled data pairs $S = \{x_{ai}, x_{bi}, y_i\}_{i=1}^{N}$ where x_{ai} and x_{bi} were the data pairs and y_i was the label. Using the labeled dataset $\mathcal{D}_l$, the initial task was to create another dataset S by randomly selecting samples from $\mathcal{D}_l$ producing a unique dataset of size N^2 at maximum. For this study, the labels $y_i \in \{0, 1\}$ were set to 1 for pairs belonging to the same class, else 0, indicating that the Siamese neural network was tasked to learn similarity metrics.

Figure 1 shows the structure of a Siamese neural network that makes use of a convolutional neural network (CNN) (Figure 2) as its sub-networks. The Siamese neural network was trained on a binary classification task to produce a model that can evaluate the similarity between two input pair. Training the network also allowed the CNN sub-networks to learn a representation of the input data guided by the available labeled data.

The set of proxy labels were produced by computing the similarity of input data to established reference points. The data samples from labeled dataset $\mathcal{D}_l$ that contained labels based from ground truth were used as reference points. Given that the amount of input data was small, all samples from $\mathcal{D}_l$ were utilized as reference points. For larger set of data, clustering techniques can be applied. Samples

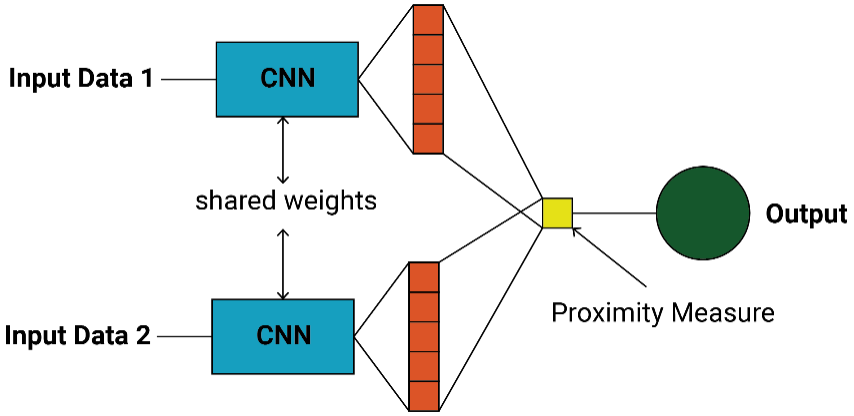

Figure 1. Structure of a Siamese neural network

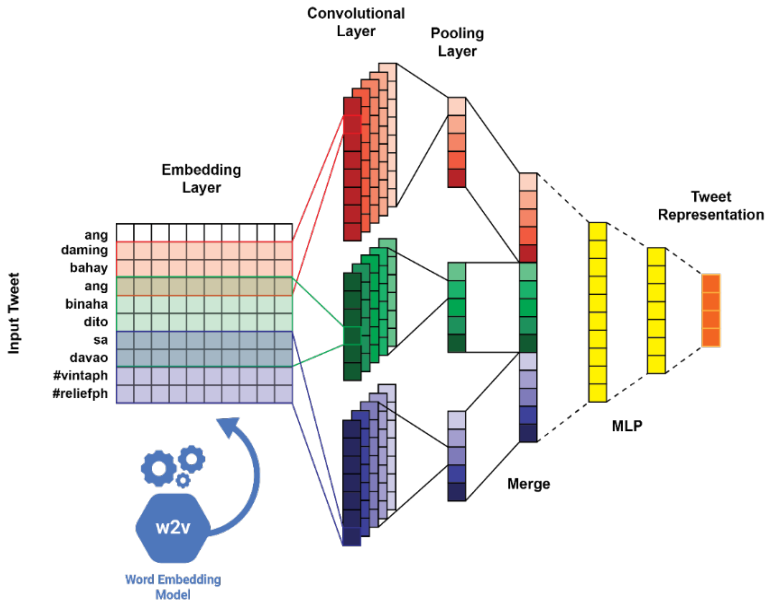

Figure 2. Structure of a convolutional neural network

nearest from the centroid can be selected as the reference points. The distances of each sample from the unlabeled dataset $\mathcal{D}_u$ to every reference point were determined using the trained Siamese neural network model. The distance to reference points belonging to the same class were then averaged to produce the final set of proxy labels. To visualize the proposed proxy label, a dimensionality reduction technique called principal component analysis (PCA) was used. The labeled tweets, together with an unlabeled tweet sample, which were represented by a 50-dimension vector, were reduced into a smaller vector with two dimensions represented by the two principal component values produced by PCA. Figure 3 shows a 2-dimensional (2-D) visualization of the proposed proxy label. The produced proxy labels can be described as the

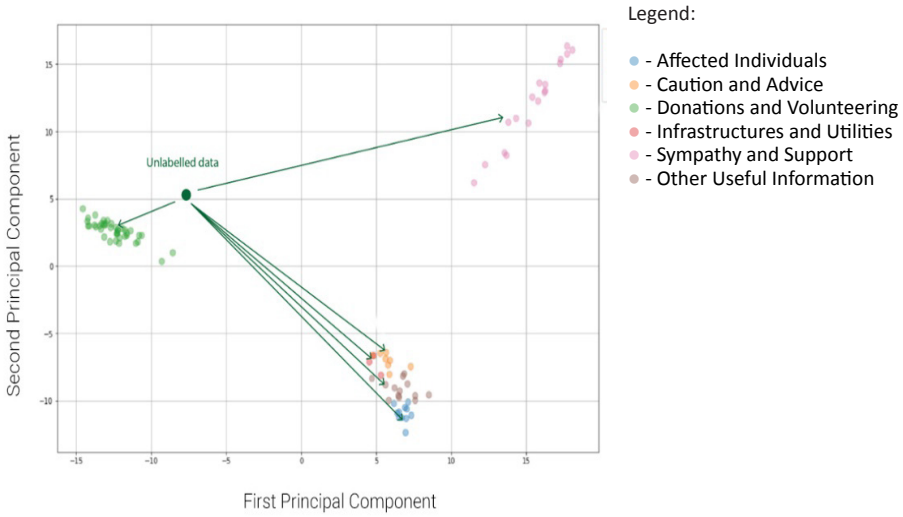

Figure 3. 2-D visualization of the proposed proxy label using distance measure learned from Siamese neural networks using PCA

average distances of an input data to the set of classes available.

The set of proxy labels allowed utilization of the set of unlabeled data $\mathcal{D}_u$ for a supervised learning approach. However, unlike regular proxy labels produced through self-training which directly state a specific class for every unlabeled input, the proxy labels produced from the proposed approach were vectors of distances from reference points. To accommodate the change in proxy label structure, the task was converted from a classification task that learns probability distribution on classes to a regression task that learns the relation (proximity measure) of an input to every classes available.

Gathering of Data for Experiment

Two types of datasets were collected for the experiment. The first one was a collection of labeled tweet datasets collected during disasters needed for benchmarking. This study made use of the datasets by Olteanu et al. available online called CrisisLexT26, a collection of ~28,000 labeled dataset containing tweets posted during a series of 26 different disaster events from around the world in 2012 – 2013 [11]. From this collection, only the five datasets that involved disasters that happened within the Philippines were used for the experiment.

The second dataset used was a large collection of raw texts from tweets posted within the Philippines. This dataset was used to train a word embedding model specifically designed for natural language processing (NLP) tasks on Philippine social media. A dedicated program that can fetch continuous data from Twitter was built.

Data Preprocessing

The collected datasets were preprocessed before the experiment. For instance, the benchmark datasets went through data cleaning, including fixing entries occupying multiple lines of texts and filtering out entries with no labels. Fields that were not relevant to the study were excluded from the final datasets.

Word count was a huge factor that was considered during the development of the text corpuses. Due to memory limitation, a restriction to the number of words was implemented. The text data were preprocessed into word tokens allowing the removal of unnecessary input such as usernames and URLs. Data preprocessing implemented filtering schemes using regular expressions.

Development of Word Embedding Model

Part of the conceptual design was to enable transfer learning through the use of trained word embeddings. With this, a word embedding model specifically designed for NLP tasks on Philippine social media was created.

Using the large text corpus obtained from data gathering as training data, a continuous skip-gram model architecture [12] was implemented to learn a distributed representation of words. The model used, known as word2vec, allowed learning word embeddings by maximizing the classification of a word based on the surrounding words. Using each word found in the corpus as a token, the task was to read every word and use it as input to a log-linear classifier with continuous projection layer to predict the nearby words contained in a pre-defined window size before and after the current word. Gensim, a python NLP library that provides an efficient python implementation of the continuous bag-of-words and skip-gram architectures of word2vec for computing vector representations of words, was used to produce the word embedding model.

Default settings from Gensim were used as parameters for training the word2vec model. Modification was made to limit the number of words learned by setting a word filter with a minimum frequency of 10. The word2vec model with a context window size of 5 and k = 5 negative samples was trained using a text corpus containing 66,598,374 lines of tweets. A total of 679,352,628 raw words was collected from the text corpus. A minimum frequency threshold was used to reduce the total number of words to 374,118. A word2vec model with 374,118 words and 50 dimensions was produced upon training for 20 epochs.

Evaluation of the created word2vec model was performed by utilizing the presence of emojis within the model. A study demonstrated that training a good word2vec model allowed the creation of emoji vectors that showed semantic

information expressed by emojis. Unlike words, the meaning of emojis can easily be identified though their appearance [13]. Thus, emojis which express their meanings visually were used to evaluate the created word2vec model. 435 emojis with frequencies more than 5000 were used for the evaluation. K-means clustering was used on the feature vectors to produce 30 clusters. A dimensionality reduction method called Principal Component Analysis (PCA) was used to plot the vectors into a 2-dimensional space. A visual evaluation was then performed to analyze each cluster and their relation to other clusters within their neighborhood.

Comparative Analysis Experiment

Performance evaluation was conducted using a set of traditional and deep neural network classification algorithms as baseline models. Test evaluations required the use of statistical tools. With the limitation on the number of available datasets, data resampling was conducted. The final performance evaluation was conducted using the new resampled datasets.

1. Data Resampling

Data resampling was conducted using a modified version of the 5x2 fold cross-validation, a method that involves five replications of 2-fold cross validation. Statistical tests conducted on samples based on this resampling method showed that it produced acceptable type I error or false positives [14]. This meant that tests on this resampling method minimize the case of suggesting a significant difference when no such difference exists.

A python script was written to perform the modified version of the 5x2 fold cross-validation on the available benchmark datasets. For each dataset, the data were partitioned into two equally-sized sets (f_a,f_b) using stratified random sampling. The left fold f_a was used as the training set while the right fold f_b was used as the testing set. The training set was then further divided into two folds (f_{a1},f_{b2}). f_{a1} was used as the labeled training set while the labels for f_{b2} were dropped to produce the unlabeled dataset. The same procedure was repeated using the right fold f_b as the training set and the left fold f_a as the testing set. Five replications of this procedure were performed on each dataset using different sets of seeding value to produce the final resampled datasets.

2. Baseline Classifiers

To evaluate the performance of the proposed approach, a non-exhaustive list of traditional and deep neural network classification algorithms was created for baseline comparison. The list of traditional classification algorithms included the following: Ada Boost, Decision Trees, k Nearest Neighbor, Logistic Regression, Naïve Bayes, Random Forest and Support Vector Machine. For deep neural network, two architectures were selected, namely, multi-layered perceptron and convolutional neural network.

Word representation as input for traditional classifiers used TF-IDF features. For MLP, the average of the vectors of the words contained within the input tweets based on the constructed word embedding model was used to represent the input features. On the other hand, CNN models made use of embedding layers with word embedding weights to automatically learn latent feature representations as distributed dense vectors.

3. Performance Evaluation

F1-score was used as the evaluation metric for all experimental tests conducted. It is a measure of a test's accuracy that represents the harmonic mean between precision P and R recall [15] following the formula:

$$F = \frac{2PR}{P + R}$$

Statistical tests were used to evaluate the validity of the results from the experiment. An omnibus test to detect if at least one of the classifiers performs differently from the others was used, specifically the Friedman nonparametric test. This is a non-parametric statistical test similar to the parametric repeated measures ANOVA.

The Wilcoxon signed rank test was used for the follow-up tests to perform a more in-depth pairwise analysis between the performance of the

evaluated classifiers. This is a non-parametric statistical hypothesis test equivalent to the dependent t-test. This is used to compare two related samples, matched samples, or repeated measurements on a single sample to assess whether their population mean ranks differ [16].

Results And Discussion

Evaluation of Word Embedding Model

This study used word2vec to implement the word embedding model for Philippine social media NLP. Evaluation on clusters of emojis selected from the word2vec model showed that the emojis contained within each cluster represented similar meanings. Figure 4 shows a 2-dimensional plot of the selected emojis and their respective clusters.

Evaluation of the clusters showed that emojis containing the same or related meaning were grouped together. Analysis of the positions of the clustered emojis, when plotted on a 2-dimensional space, showed that the clusters that were related were positioned near one another. Such results showed that the word2vec model produced was able to learn semantic information from training on a collection of unlabeled tweets.

Comparative Analysis Experiment Results

Evaluation on the performance of the proposed semi-supervised learning model was conducted using a comparative analysis experiment. The experiment was divided into a series of tests. Table 1 shows the list of codes of classification models used in the discussion of the experiment results.

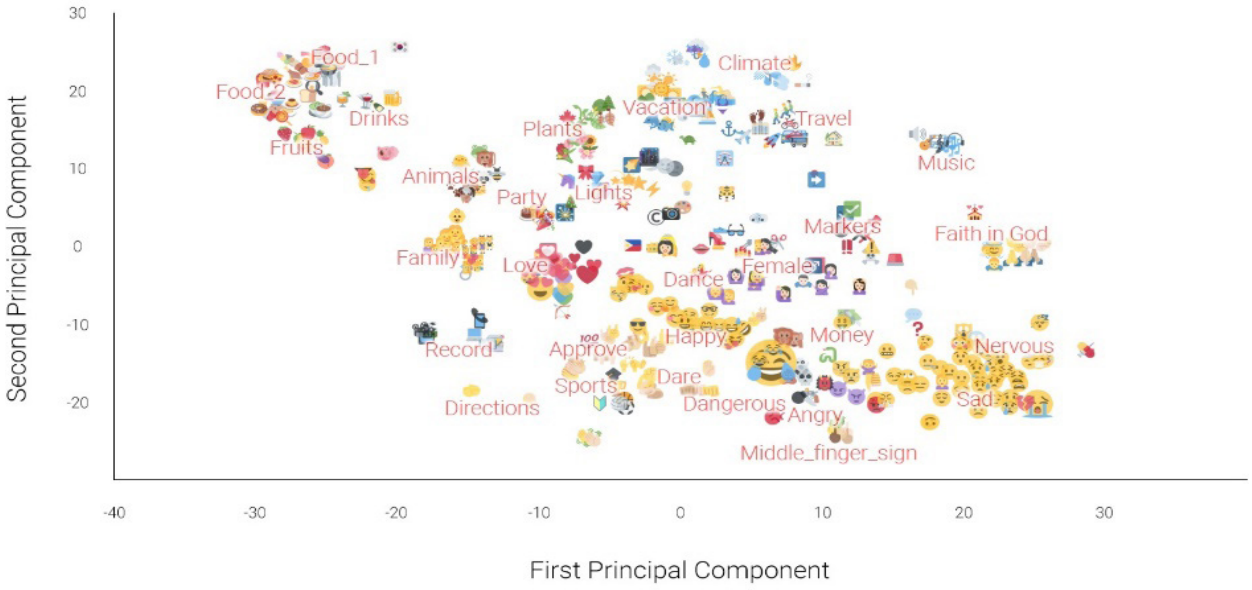

Figure 4. 2-D visualization of emojis from the word2vec model using PCA

Table 1. List of classification models and their code

Code	Classification Model
AdaB	Adaptive Boosting
DT	Decision Trees
KNN	k Nearest Neighbor
LR	Logistic Regression
NB	Naïve Bayes
RF	Random Forest
SVM	Support Vector Machine
MLP-ST	Multi-Layered Perceptron using Self-Training
CNN-ST	Convolutional Neural Network using Self-Training
MLP-Siam	Multi-Layered Perceptron using the proposed SSL method
CNN-Siam	Convolutional Neural Network using the proposed SSL method
MLP-Siam-OE	Multi-Layered Perceptron using the proposed SSL method trained from separate event dataset (out- of-event model)
CNN-Siam-OE	Convolutional Neural Network using the proposed SSL method trained from separate event dataset (out- of-event model)
MLP-Siam-EE	Multi-Layered Perceptron using the proposed SSL method trained from the same event dataset (event-exclusive model)
CNN-Siam-EE	Convolutional Neural Network using the proposed SSL method trained from the same event dataset (event-exclusive model)

1. Assessment of the effects of using semi-supervised learning on the performance of machine learning models on classifying disaster-related tweet datasets

The first test aimed to evaluate the effect of using semi-supervised learning as compared to plain traditional approaches of machine learning on the performance of classification models. The initial task for this test was to determine the best traditional classifiers to be compared to deep neural network models and semi-supervised learning approaches. Evaluation on the listed traditional classifiers showed that the performance of KNN produced the best average F1-score of 0.5369. This was followed by SVM with an average F1-score of 0.5038. AdaB got the lowest average score of 0.3450. Statistical evaluation showed that the pairs LR and NB as well as DT and RF produced comparable performance while the rest of the classifiers produced results that were statistically significant.

The top 2 traditional classifiers (i.e., KNN and SVM) were then compared to deep neural network classifiers (MLP and CNN) and their semi-supervised learning counterparts using self-training (i.e., MLP-ST and CNN-ST). Results showed that deep neural network classifiers significantly outperformed traditional classifiers. An increase in the performance of deep neural network classifiers was also observed when self-training was used to produce proxy labels for the unlabeled training data. However, given the small size of the datasets, statistical evaluation showed that such increase was not significant. Figure 5 shows the comparison of average F1-scores of traditional classifiers, deep neural network classifiers, and classifiers using self-training approach for proxy labelling.

The same test was conducted using the proposed semi-supervised learning approach. The same results were observed as classification models produced by the proposed approach outperformed their deep neural network

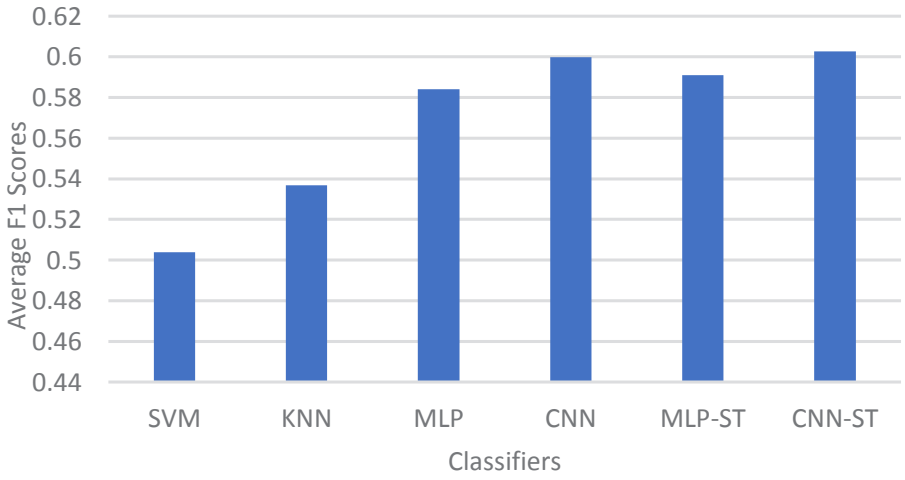

Figure 5. Comparison of average F1-scores of traditional classifiers, deep neural network classifiers, and classifiers using self-training approach for proxy labelling

counterparts. However, statistical evaluation showed two notable observations different from simple average F1 score comparisons. First, the tests showed that using the proposed SSL approach instead of the self-training method to produce proxy labels can significantly increase the performance scores of a classifier even for pairs with the same neural network architectures. Second, the test revealed that using the proposed approach allowed MLP classifiers to produce results on-par with CNN classifiers. Figure 6 shows the comparison of average F1-scores of traditional classifiers, deep neural network classifiers, and classifiers using the proposed semi-supervised learning approach.

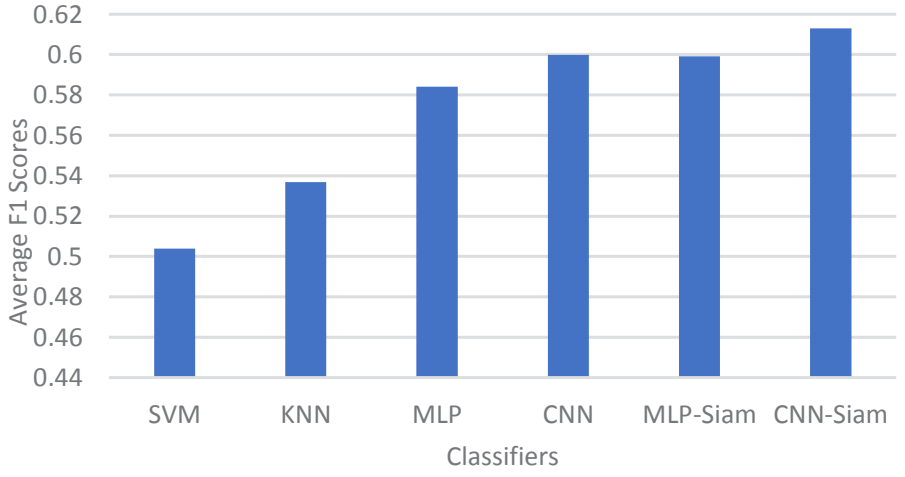

Figure 6. Comparison of average F1-scores of traditional classifiers, deep neural network classifiers, and classifiers using the proposed semi-supervised learning approach

2. Comparison on the performance of the proposed semi-supervised learning using Siamese neural network against classic semi-supervised learning - proxy labelling approach specifically self-training

The goal of this test was to directly compare the performance of the proposed approach against self-training method. Results revealed a significant difference between the two approach. Figure 7 shows the comparison of the average F1 scores produced by the two approaches on MLP and CNN architectures. Statistical evaluation results showed that MLP-Siam significantly outperformed MLP-ST with a mean difference of 0.0080 and a p-value of 0.0291. The same was true with the CNN architecture where CNN-Siam significantly outperformed CNN-ST with mean difference of 0.0104 and a p-value of 0.0221.

3. Comparison on the performance of using out-of-event models against event-exclusive models

Test 3 focused on evaluating the performance of the proposed semi-supervised learning method using training samples from different events. Comparison of the F1-scores of MLP-Siam and CNN-Siam using training samples from previous disaster event (out-of-event models) and training samples from the same event as the testing set (event-exclusive models) was conducted. A large difference in labeled data frequency was used for the experiment setup to reflect real-world scenarios with out of event datasets being 10 times larger than the event-exclusive datasets. Figure 8 shows a comparison of the average F1-scores of the evaluated classifiers. Analysis showed that event-exclusive models outperformed the out-of-event models on both MLP and CNN

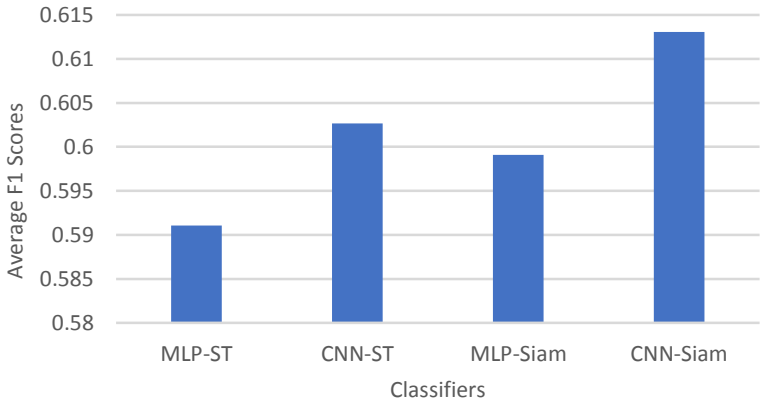

Figure 7. Comparison of average F1-scores of deep neural network classifiers using self-training method and the proposed semi-supervised learning approach

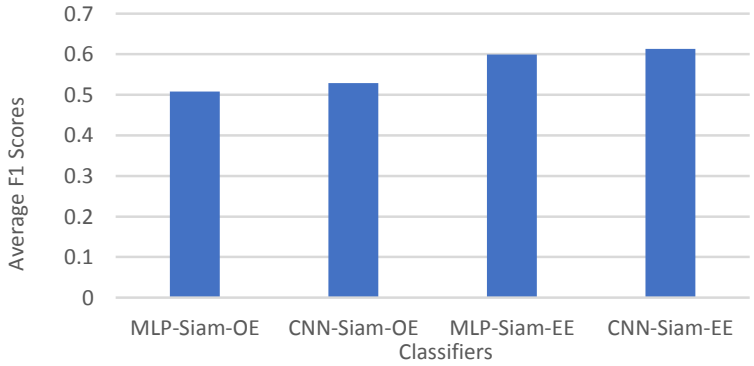

Figure 8. Comparison of average F1-scores of out-of-event models and event-exclusive models

architectures even with a large difference in training sizes. This was supported by the results of the Wilcoxon signed ranked test which showed significant differences between the performances of the models. Such results may be attributed to the use of discrete words that varied across the datasets collected from different events.

4. Evaluation of the effect of frequency of labeled data on the performance of the proposed semi-supervised learning using Siamese neural network

The final test was designed to evaluate the effect of frequency of training data to the performance of classification models using the proposed approach. Increments of 10% or roughly around ~100 tweet samples per setup were used for the labeled training set. The convolutional neural network architecture implementing the proposed method (CNN-Siam) was used for the final test.

Analysis of the average F1-scores showed that classifiers trained with more labeled data led to the production of better models. Figure 9 provides a plot showing the increasing performance of classifiers in accordance with the increase of labeled data for training. This was supported by the follow-up Wilcoxon signed rank tests which showed that the increase in F1-scores brought by the increase in frequency of labeled data were statistically significant at α=0.05. However, results also showed that the effect of increasing the frequency of labeled data towards performance scores of classifiers using fixed 10% increments of labeled data diminishes as the frequency increases. This was evident on the mean differences between the pairs of frequencies {(10%, 20%), (20%, 30%), (30%, 40%)}, which got mean difference scores of 0.0365, 0.0132 and 0.0095, respectively.

Overall, the tests revealed a number of notable results. First, the test results indicated that using self-training to implement semi-supervised learning on small datasets did not have a significant effect on neural network models of the same architecture. On the contrary, results showed that using the proposed approach to implement a semi-supervised learning on the same datasets produced a significant increase on the performance scores of the classifiers. Comparing the self-training method and the proposed approach showed that the latter significantly outperformed the first.

Results also showed that the choice of classification algorithm plays a significant role on the performance of the resulting models. For instance, the KNN classifier significantly outperformed the other traditional classification algorithms. On the other hand, the deep neural network models trained significantly outperformed KNN. Results showed that

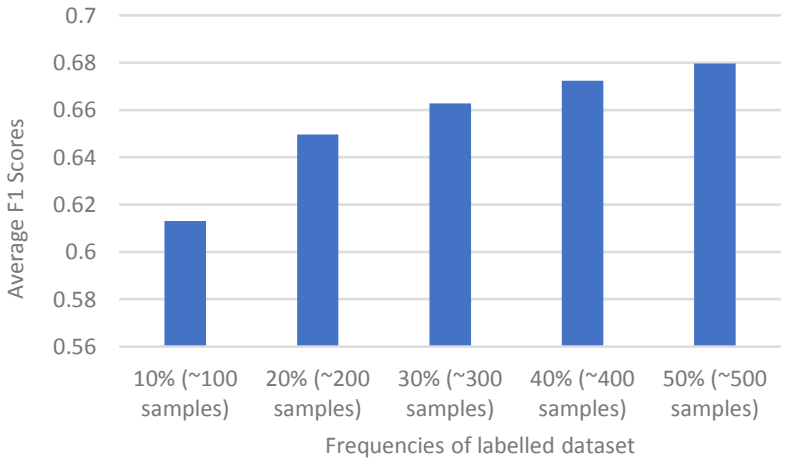

Figure 9. Comparison of average F1-scores of CNN-Siam with varying frequencies of labeled data

CNN performed significantly better than MLP. However, using the proposed approach significantly increased the performance of MLP to a point where it obtained F1-scores that were on-par with the much complex CNN.

Evaluation of the performance of the proposed SSL approach in relation to the frequency of the labeled training data was also conducted. Results showed that an increase in the number of labeled data can significantly improve the performance of a classifier. However, the improvements brought about by increasing the labeled data frequency diminish as the frequency increase. Also, the tests revealed that the quality of training data is much more important than the quantity. Specifically, models trained using labeled data from the same event (event-exclusive models) significantly outperformed models trained using data from a separate event (out-of-event models) despite having an absolute advantage in the amount of training data.

Conclusion

The main objective of the study was to provide a method of improving the performance of machine learning classifiers designed for disaster monitoring on Philippine social media. To alleviate the impact of labeled data scarcity during the earlier stages of a disaster event, utilization of the numerous unlabeled data available by learning proxy labels using Siamese neural network to implementing a semi-supervised learning approach was implemented. The resulting model outperformed traditional machine learning classifiers including AdaB, KNN, LR, NB, RF, and SVM and deep learning approaches including MLP and CNN in terms of F1-scores given a limited set of labeled training data. Integrating such method to train core machine learning models of disaster monitoring systems can drastically lessen the need for a large workforce to provide manual labelling on the collected social media data, thus, saving on cost, time and resources. Future iteration of this study will involve the application of the proposed method to existing online systems that harvest real-time tweet data to evaluate its performance on live datasets.

References

[1] Nguyen, T.D., Joty, S.R., Imran, M., Sajjad, H., and Mitra, P. (2016). Applications of Online Deep Learning for Crisis Response Using Social Media Information. In *Proceedings of the 4th International Workshop on Social Web for Disaster Management (SWDM'16), 4*. Retrieved from https://arxiv.org/pdf/1610.01030.pdf.

[2] Imran, M., Castillo, C., Lucas, J., Meier, P., and Vieweg, S. (2014). AIDR: Artificial Intelligence for Disaster Response. In *Proceedings of the 23rd International Conference on World Wide Web*, 159-162. Retrieved from https://doi.org/10.1145/2567948.2577034.

[3] Patel, F., and Soni, N. (2012). Text mining: A Brief survey. *International Journal of Advanced Computer Research, 2*(6), 243-248. Retrieved from https://pdfs.semanticscholar.org/11c4/6d00a0e136e8e4e27aa15fbb8c9111cdee75.pdf.

[4] Qadir, J., Ali, A., Rasool, R. U., Zwitter, A., Sathiaseelan, A., and Crowcroft, J. (2016). Crisis Analytics: Big Data Driven Crisis Response. *Journal of International Humanitarian Action*, *1*(1), 1-21. Retrieved from https://doi.org/10.1186/s41018-016-0013-9

[5] Murphy, K. (2012). *Machine Learning: A Probabilistic Perspective*. Cambridge, Massachusetts: The Massachusetts Institute of Technology (MIT) Press. ISBN: 9780262018029.

[6] Russell, S., and Norvig, P. (2009). *Artificial Intelligence: A Modern Approach* (3rd ed.). Upper Saddle River, New Jersey: Prentice Hall Press. ISBN: 9780136042594.

[7] Alam, F., Joty, S.R., and Imran, M. (2018). Graph Based Semi-supervised Learning with Convolution Neural Networks to Classify Crisis Related Tweets. In *Proceedings of the 12th International AAAI Conference on Web and Social Media*, *12*, 556-559. Retrieved from https://arxiv.org/pdf/1805.06289v1.pdf.

[8] Chapelle, O., Scholkopf, B., and Zien, A., Eds. (2006). *Semi-Supervised Learning*. Cambridge, Massachusetts: The Massachusetts Institute of Technology

(MIT) Press. ISBN: 9780262033589.

[9] Pan, S. J., and Yang, Q. A. (2010). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering, 22*(10), 1345–1359. Retrieved from https://doi.org/10.1109/TKDE.2009.191.

[10] Neculoiu, P., Versteegh, M. and Rotaru, M. (2016). Learning Text Similarity with Siamese Recurrent Networks. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, 148-157. Retrieved from https://doi.org/10.18653/v1/W16-1617.

[11] Olteanu, A., Castillo, C., and Vieweg, S. (2015). What to Expect When the Unexpected Happens: Social Media Communications Across Crises. In *Proceedings of the ACM 2015 Conference on Computer Supported Cooperative Work and Social Computing (CSCW '15)*, 994-1009. Vancouver, BC, Canada: ACM. Retrieved from https://doi.org/10.1145/2675133.2675242.

[12] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems*, *2*, 3111-3119. Retrieved from https://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf.

[13] Barbieri, F., Ronzano, F., and Saggion, H. (2016). What does this Emoji Mean? A Vector Space Skip-Gram Model for Twitter Emojis. In *Proceedings of the 10th International Conference on Language Resources and Evaluation, 10,* 3967-3972. Retrieved from https://www.aclweb.org/anthology/L16-1626.pdf.

[14] Dietterich, T.G. (1998) Approximate statistical test for comparing supervised classification learning algorithms. *Neural Computation. Neural Computation, 10*(7), 1895–1923. Retrieved from https://doi.org/10.1162/089976698300017197.

[15] Sasaki, Y. (2007). The truth of the F-measure. *Teach Tutor Mater, 1*(5), 1-5. Retrieved from https://www.researchgate.net/publication/268185911_The_truth_of_the_F-measure.

[16] Hollander, M., and Wolfe, D. (2013). *Nonparametric Statistical Methods* (3rd ed.). New York: Wiley Interscience Publication. ISBN: 9781118553299.