CNN-BASED FEATURES FOR FILTERING OF CRISIS RELATED SOCIAL MEDIA MESSAGES

Dao Nam Anh, Nguyen Quynh Anh*, Le Manh Hung
Electric Power University

ARTICLE INFO	ABSTRACT

Received: 07/6/2022
Revised: 03/8/2022
Published: 04/8/2022

KEYWORDS

Image Patten
Filter
NLP
CNN
Social media message

Analysis of the likelihood of attributes like real or false awareness, given a series of message from social media, is a common problem in natural language processing (NLP). This paper presents a reliable method for categorizing emergency of messages in Tweeter. We rely on representation of text features by image patterns instead of using original features extracted from text message. The initial text features were extracted with morphological segmentation and statistical analysis of appearance of keywords in messages by NLP techniques. In order to increase the classification accuracy image patterns-based approach was implemented. The transformation of text features into image allows applying convolution operations for patterns detection. This opens the way to combinations of NLP and image analysis where the powers of both are preserved. Convolutional neural networks were performed with image patterns for the final social media sentence classification. Pros and cons of the method were discussed along with comprehensive report of performance.

SỬ DỤNG MẠNG CNN TRÍCH RÚT ĐẶC TRƯNG LIÊN QUAN ĐẾN CÁC TIN NHẮN KHẨN CẤP TRÊN MẠNG XÃ HỘI

Đào Nam Anh, Nguyễn Quỳnh Anh*, Lê Mạnh Hùng
Trường Đại học Điện lực

THÔNG TIN BÀI BÁO	TÓM TẮT

Ngày nhận bài: 07/6/2022
Ngày hoàn thiện: 03/8/2022
Ngày đăng: 04/8/2022

TỪ KHÓA

Đặc trưng ảnh
Trích rút đặc trưng
Xử lý ngôn ngữ
Mạng CNN
Mạng xã hội

Từ các thông tin trên các trang mạng xã hội, bài toán phân tích xác định nội dung là thật hay giả là một vấn đề cần nghiên cứu trong xử lý ngôn ngữ tự nhiên (NLP). Bài báo trình bày một phương pháp để phân loại trường hợp cấp thiết trong các tin nhắn trên Tweeter. Nhóm nghiên cứu dựa vào biểu diễn các đặc trưng văn bản bằng các mẫu hình ảnh thay vì sử dụng các đặc trưng text được trích xuất trực tiếp từ tin nhắn văn bản. Trong các kỹ thuật xử lý ngôn ngữ tự nhiên, các đặc trưng text thường trích chọn dựa trên việc phân đoạn hoặc phân tích thống kê tần suất xuất hiện của các từ khóa trong các tin nhắn văn bản. Để làm tăng độ chính xác của việc phân lớp nhóm nghiên cứu đã cài đặt một phương pháp dựa trên nhận dạng các mẫu ảnh. Việc chuyển từ đặc trưng text thành ảnh cho phép áp dụng các phép toán tích chập để nhận dạng các mẫu. Điều này mở ra một sự kết hợp giữa NLP và phân tích hình ảnh. Bài báo sử dụng mạng nơ ron tích chập (CNN) thực hiện với các mẫu ảnh để phân lớp các câu. Nghiên cứu cũng được so sánh với các phương pháp khác để đánh giá trong phần mô phỏng so sánh của nghiên cứu đề xuất.

DOI: https://doi.org/10.34238/tnu-jst.6133

* Corresponding author. *Email: anhnq@epu.edu.vn*

1. Introduction

There is nowadays a robust demand for automated argument mining systems which can infer or understand more complex argumentative structures. In particular enabling extraction of domain specific information for disaster monitoring and risk management is an essential problem in natural language processing. Its horizon of applications includes but not limited to information retrieval [1], outbreak detection [2], hazard estimation [3], and damage assessment [4], evacuation behavior study [5] health and disease analysis and propagation detection [6], quantifying controversial information [7], and sentiment analysis [8], [9]. The latter puts forward the significant motivation for this work, which is related to monitoring emergency situations in social media by learning patterns of natural language messages in order to identify real disaster events. These events are of substantial interest in monitoring intention and ignoring false disturbances in social media.

For instance, some natural disaster like earthquake taking place, communication among people in social media could give valuable information for evacuation, rescue, and donation. However, while the use of social network seems appealing, the rise of the likelihood of improper or incomplete information sharing is remarkably observed.

It is just a strong demand of natural language processing used to improve the classification of information. This makes the assessment of discussion on social media available for monitoring disaster, in such a way that inappropriate awareness can be detected and reliability of message processing can be enhanced.

In this work we focus especially on sentiment analysis in social media sentences by learning patterns of texts and implementing CNN over image patterns that represent the features of the texts. The objective is to qualify awareness of disasters noticed in social messages. To the best of our knowledge this work is one of the first attempts to interpret social messages patterns by composing images from extracted features allowing implementation of CNN for image patterns. The results with a disasterrelated Tweeter message benchmark database show the effectiveness of the proposed method.

A number of researchers have attempted to deal with sentiment analysis and the classification of social messages by searching methods for enhancing reliability of text reprocessing and classification in the presence of various grammatical nuances, cultural variations, slang and misspellings. Aiming to review related work we look into two groups of interest: (I) the works focused on major linguistical analysis and (II) application of learning techniques for the field of advantage of the sentiment analysis.

(I) To facilitate analysis of text corpora that describe long-term recovery, Lin et al. [10] employed a statistical syntax-based semantic matching model for a standard, publicly available training dataset. The method can be useful for an appropriate news article corpus and, potentially, large corpora in general. A disaster-related news corpus was a successful stud case in the scope of the paper. Verma et al. [11] showed that a classifier based on low-level linguistic features performs well at identifying tweets that related to situational awareness. Then, linguistically motivated features including subjectivity, personal versus impersonal style, and register are proved to substantially improve system performance. Selecting key features of user behavior can aid in predicting whether an individual tweet will contain tactical information.

Compared to these in the literature, their focus on linguistical features is always significantly dominant for the natural language processing (NLP) problem. We do not focus on the term in the work. However, the linguistical methods proved to be robust under a considerable amount of noise for getting linguistical features applied in our text preparation task. Then, the features are processed further by deep learning.

(II) Li et al. [12] proposed to apply a domain adaptation approach, which learns classifiers from unlabeled target data, in addition to source labeled data. Naive Bayes classifier, together with an iterative self-training strategy were implemented in their experiments which used a self-training iterative strategy to incorporate labeled data from a source disaster and unlabeled data from an emerging target disaster into a classifier for the target disaster. Stowe et al. [13] addressed classifying disaster-related tweets with Twitter data generated before, during, and after Hurricane Sandy in the fall of 2012. Here, baseline features are the counts of uni-grams in tweets, after pre-processing to remove capitalization, punctuation and stop words. Different classification models including parameter optimization like SVM regularization and feature selection methods were experimented using uni-grams for relevance classification. Then the best-performing approach was selected.

A rich set of features that include Bag-of-Words, text-based, and user-based features for traditional models were used in BERT-based models for the informative tweet classification problem by Joao [14]. Machine learning methods for automatically identifying informative tweets among those that are relevant to a target event were studied to propose a hybrid model that leverages both the handcrafted features and the automatically learned ones.

Long Short-Term Memory (LSTM) was proposed by Hochreiter et al. [15] to deal with the vanishing gradient problem. The initial version of the LSTM block included cells, input and output gates. A deep learning model combining attention based Bi-directional Long Short-Term Memory (BLSTM) and Convolutional Neural Network (CNN) was used by Kabir et al. [16] to classify the tweets under different categories. Pre-trained crisis word vectors and global vectors for word representation were implemented for capturing semantic meaning from tweets. Feature engineering then is used to create an auxiliary feature map.

In this work, we focus especially on a novel variational approach that integrates several of the above-mentioned concepts including preprocessing to remove capitalization, punctuation and stop words, linguistical feature engineering with BERT-based models. It is further shown that presentation of text features by image allows implementing different CNN models. This has two major effects: Firstly, it becomes feasible to unite the CNN technique, which was image originally motivated, into an NLP domain. Secondly, it shows a theoretically sound way of how a particular tweet messages classification problem can be solved with an effective pattern recognition technique.

2. The proposed method

The following summarizes the method for classifying social media messages. Given a message s, a class c can be associated with the message. To describe the learning process in our method, we use Bayes' Rule [17] that expresses conditional probability for message sample s and class c.

$$\mathrm{p}(c|s) = \mathrm{p}(s|c)\mathrm{p}(c)/\mathrm{p}(s) \tag{1}$$

From any query message sample s, the maximum a posteriori (MAP) most likely class c, appropriate for s, can be determined by a Bayesian decision where C is the set of classes.

$$c_{MAP} = argmax_{c\in C}\, p(c|s) \tag{2}$$

Here, Bayes' Rule (1) enables to show the most likely class c

$$c_{MAP} = argmax_{c\in C} \frac{p(s|c)p(c)}{p(s)} \tag{3}$$

Then, the denominator $p(s)$ can be dropped

$$c_{MAP} = argmax_{c\in C}\, p(s|c)p(c) \tag{4}$$

The arrow in Figure 1a clearly shows that at the classification for message s is based on direct relationship between message s and class c, i.e. most judgments are based on text-form of the original message. However, this was not always observed.

Figure 1. *a. Relation between sample s and class c; b. Image f is determined by s, and then convolution operation on f allows having g.*

The fact that representation of encoded text message by 2D image allows us implementing convolutional techniques and extracting CNN based features. In our model, words can be split from any message sample, and then encoded by tokenization, which refers to lexical analysis [18] for converting a sequence of characters into a sequence of tokens. In addition, tokens are strings with an assigned and thus associated meaning. Thus, a text message sample *s* can be encoded into a vector of real numbers. Note that the vector can be normalized so that values of each vector member belong to interval [0, 1]. Using five integers in interval [0, 255] a real value in the interval [0, 1] is represented now by one of $256 * 5 = 1280$ integers. The vector is then reshaped into 2D array. As a gray image is a 2D matrix of pixels which have discrete values in the interval [0, 255] the input message is converted to a gray image. We mark *tran* function for the task of transforming a text message *s* to an image *f*:

$$f = tran(s), f \in R^2 \quad (5)$$

Given that the gray image *f* contains encoded features of the original message and a kernel *h*, a convolution operation can be performed to get presentation g for *s*:

$$g(x, y) = h * f(x, y) = \sum_{dx=-a}^{a} \sum_{dy=-a}^{a} h(dx, dy). f(x + dx, y + dy) \quad (6)$$

where x, y are location of a pixel in the image, while (2a+ 1)∗(2a+ 1) is the size of the convolution kernel. At this point, convolution is an important application of integration. We implemented CNN for processing images derived from text messages. It is important to emphasize in our case study for analysis of the social message with assistance of VGG16, GoogleLeNet, Inception V3 and ResNet101.

VGG16 [19] is CNN designed for images of fixed size of 224*224 and outputs a vector of 1000 values. GoogleLeNet (or Inception V1) [20] was proposed by research at Google with the architectural decisions that is based on the Hebbian principle and the intuition of multi-scale processing. Inception V3 [21] is a convolutional neural network for assisting in image analysis and object detection, and got its start as a module for GoogleLeNet. Input image has a size of 299*299. ResNet101 [22] is Residual CNNs for image classification tasks with constructed 101 layers for input image of a size of 224*224. All above CNN networks output a vector of 1000 real values, which are formally denoted by mentioned symbol *g*. Figure 1b illustrates the introduction of the image-based representation by f and the output of CNN by *g* for the original relation between the text message *s* and the class *c*. When a message *g* needs classification, using (3-6) a most likely class *c* can be estimated:

$$c_{MAP} = argmax_{c \in C} \frac{p(\hat{g}|c)p(c)}{p(\hat{g})} \quad (7)$$

By dropping $p(\hat{g})$ from (7) classification for test message $g\hat{}$ is derived as follows:

$$c_{MAP} = argmax_{c \in C}\, p(\hat{g}|c)p(c) \quad (8)$$

It’s clear that $p(c)$ from (8) can be estimated by training data with appearance of pairs of messages and assigned class:

$$p(c) = \sum_g p(c|g). p(g) \quad (9)$$

Since a number of messages *g* is available for learning, the similarity of the test message $\hat{g}$ with the training message can be measured. If modeled as a Gaussian function, we can express mathematically the similarity by a likelihood function.

$$p(\hat{g}|g) = \frac{1}{\sqrt{2\pi|\sigma|}} \exp(-\frac{(\hat{g}-g)^2}{\sigma^2}) \quad (10)$$

The function involves correlation using a multivariate Gaussian, where σ is the covariance. So, the availability of measure (10) allow us to have formula (11):

$$p(\hat{g}|c) = \sum_s p(\hat{g}\,|g).p(g|c) \quad (11)$$

When data training has been trained, class estimation for c by (8) is fulfilled with assistance of (9-11). In addition, accuracy is the performance metric for our case study. Its definition is based on true positive (TP), true negative (TN), false negative (FN), and false positive (FP)

$$Acc = (TP + TN)/(TP + TN + FP + FN) \quad (12)$$

To demonstrate that the combination of the linguistical features extraction with the CNN is needed to be performed in training stage in Figure 2, two essential tasks were included. In the first, the *ImageTransform(s),* that creates an image from a message's text features. In the second, a CNN model is built by CNN (*f*). These tasks are seen in the test stage where test data is the object to apply.

ALGORITHM 1. Image Pattern for Filtering of Crisis Related Social Media Messages
Input: training set of text message s,c;
test set of text message $\hat{s}$, class $\hat{c}$
Output: the prediction c;

```
//Text feature extraction
Data Cleaning (s)
Tokenization (s)
f =Image Transform (s)
for each CNN model in VGG16, ResNet101, GoogleLeNet, Inception V do
    //CNN feature extraction
    g =CNN (f)
    //Training
    model = learning (g,c)
    //Test
    c = classify (ĝ,model)
    //Performance
    accuracy=compare(ĉ, c)
end for
```

Figure 2. *Primary path of CNN based method*

3. Experimental results

To evaluate the performance of the method, a set of social media messages from Kaggle [24] was used. In this database 10,000 tweets were hand classified. As for the message classification, the data analysis and image analysis by CNN were taken into consideration in our experiment. So far, the experiments perform classification of tweet messages into real disaster and not disaster message following described method in section 3. It runs into two main stages: (I) Data Analysis covering mainly linguistic operations; (II) Image Pattern Analysis by providing CNN.

(I) Data Analysis

The Bayesian approach [18] from section 3 has set-up around the use of the data to search relation of text messages with classes. A message shown on a tweeter application, based on measured color coordinates, is checked initially by data cleaning process, where symbols and numbers are removed. By collecting and analyzing the length of messages for two classes including Not Disaster and Real Disaster, Figure 3 shows that the number of texts having length under 120 for real disaster appeared lower for the other class. The contradiction is not shown for text with longer length.

This is due the fact that people who are being in real case of disaster have not much time for writing or sharing information. The most intention at that moment is given for tasks to deal with actual dangerous situation. This enables us to determine the feature of message length with high interest in learning. As such, a study on distribution of number of words appeared in a message can show undoubtable distinction for the class of real disaster. Figure 4 displays the distribution of number of words for both classes, showing low level for the class of real disaster for text having 10 to 22 words. This notation yields principled features based on the number of words in a tweet.

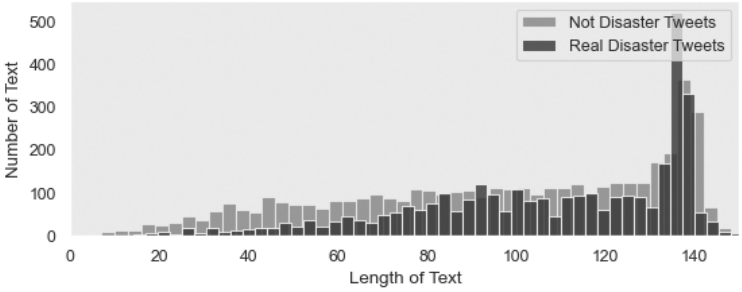

Figure 3. *Length of tweets vs. number of tweets*

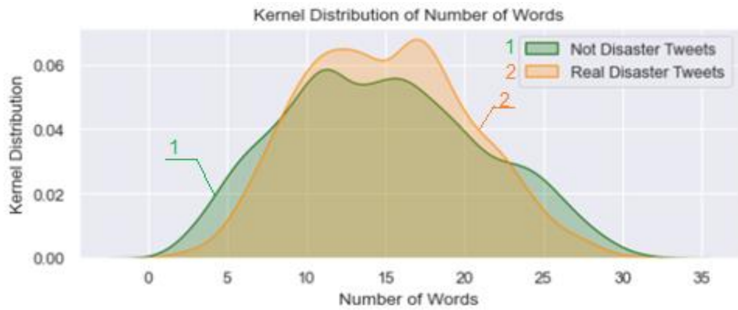

Figure 4. *Kernel distribution of number of words*

In terms of the length of words, the curve of distribution for real disaster is allocated in the left to tell that messages of real disaster are usually shorter than other class. Figure 5 also draws red curve for fake disaster higher than real cases. Likewise, we addressed to the kernel distribution by creating features based on the length of words for each tweet data sample. By using statistics for words from the Tweeter database, frequency of each word can be estimated. If a font size is set to be proportional with the frequency, then a picture can be shown by Figure 6. The most frequently used words are via, new, people, storm, don't, day, weapon and go. Using the analysis of statistics for words as a basis, specific features were created for our text samples. To analyze the similarity of words we use additional data set of Google News by BERT-based models. What is interesting is the database allows us to find similar words for a given word, and the distinction between words is also estimated. Figure 7 demonstrates a map where distance show level of distinction of words. Given a set of words showed in the right column of the Figure 7, the words and their similar are located in the map with colors. Therefore, the similarity of words in message with other words can be evaluated and this supported to create corresponding features for tweet messages.

Applying data analysis for the text messages, we discovered different distributions between two classes and a set of features were created for original text database. However, the distinction is not clear enough for classifying the tweeter messages, and we continue the study in the next session with assistance of image patterns.

(II) Image Pattern Analysis

Many NLP systems use tokens to represent text message by array of numbers. We transformed each tweet message to a vector using a tokenization centric approach, which is based on number of occurrences for each word in the whole data set. Thus, the more a word turns out in the database, the less it has bias as a feature.

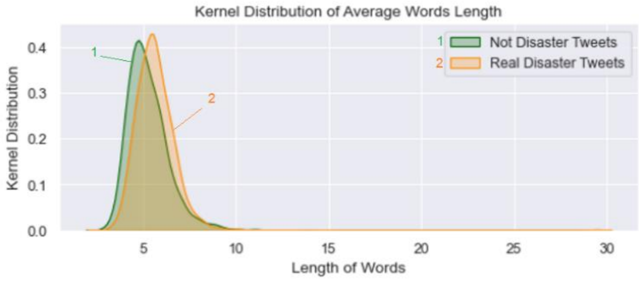

Figure 5. *Kernel distribution of average words length*

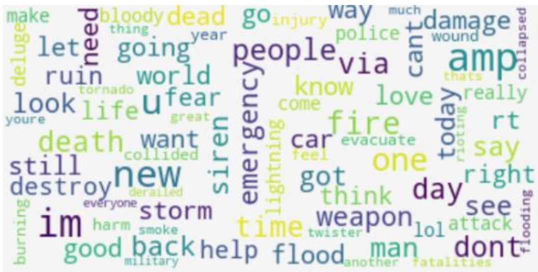

Figure 6. *WordCloud of train tweets*

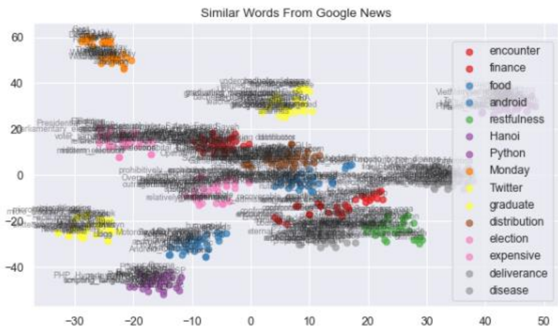

Figure 7. *Visualizing similar words from Google news*

Figure 8. *Image patterns examples*

Array of real value attained by the tokenization then enriched by features mentioned above. Once the numerical feature array has been identified, its members can be normalized to real values in the interval [0,1] for converting to image later. Note that, normalization for features is conducted for training data and test features separately. As value interval of the two set of features are different, we transform training features to interval of [ϵ : 1-ϵ], ϵ = .05 but not to [0:1] in order to reserve space for values of test features which are out of value domain of the training data. The final feature result is used for creating image *f* mentioned in formula (5).

Figure 8 shows images which are results of transformation from feature arrays of tweets. Actually, these images are darker. However, we make the images lighter for better printing. Notice that the patterns of images are totally different each other and it is easy to recognize this under normal lighting condition. The transformation gives us 10,000 images of size 224*224 and other 10,000 images of size 299*299. The size of 224*224 is used for VGG16 [19], ResNet101 [23], and GoogleLeNet [20]. The Inception V3 [21] uses image of size of 299*299 for its input [22].

Here are examples of conversion of a text message to an image.

A further implementation of CNN for the database of images allows us to get final feature for each image. This is a vector of 1,000 real values, which are used for classification. The feature was named as *g* in previous section and the learning process was explained by formulas (7-11). Our experiments applied cross validation with five splits, each split has 70% number of messages for training and 30% for testing. Evaluating each split by accuracy metric described by formula (12) gives us possibility to get averaged accuracy from the cross validation.

(III) Performance Evaluation.

The classification for the tweeter messages database with support of image patterns and CNN VGG16 [19] provided accuracy of 73.70%. Learning by GoogleLeNet [20] offered 75.71% of accuracy, with 2% higher than VGG16. By applying ResNet101 [23] for the image database, one gains accuracy of 77.65%, again having 2% higher the GoogleLeNet. By using input image of fixed size of 299*299 and specific CNN conFigureuration of Inception V3 [21], 82.42 is the best accuracy rate that we have achieved from the experiments. Table 1 shows the accuracy results by test splits and the averaged scores.

In particular, NLP analysis including the tokenization techniques, kernel distributions analysis and image patterns with CNN are joined in solving the tweet disaster classification problem. Based on a range of experimental CNNs, the Inception V3 have shown that this is the most suitable solution for the tweet message database.

Table 1. *Accuracy results by testing in 5 splits*

Method/split	1	2	3	4	5	Average
VGG16 [19]	62.30%	76.67%	76.47%	71.61%	81.45%	73.70%
GoogleLeNet [20]	81.45%	81.45%	76.67%	71.88%	67.08%	75.71%
Inception V3 [21]	86.25%	76.67%	81.45%	86.25%	81.45%	82.42%
ResNet101 [23]	76.67%	81.45%	67.08%	89.13%	73.90%	77.65%

In related works other methods were implemented for the same NLP domain. Table 2 lists results by accuracy for reference. To address tweets classification problem in disaster management field, Ma G. [25] applied BERT architecture for transfer learning. The standard BERT and other customized BERT architectures were trained to compare with the baseline bidirectional LSTM with pretrained Glove Twitter embeddings. The BERT and BERT-based LSTM were reported with outperforming the baseline model in the experiment. Muhammed et al. [26] have employed LSTM networks for the classification considering the whole text structure using long-term semantic word and feature dependencies.

Bernhard et al. [27] addressed social media feeds to detect emergencies and extract significant information to support rescue operations. The proposed stream filter consists of posts analysis, facts extraction through natural language processing. The stream filter and event clustering allowed extracting event information from post texts. It is interesting to analyze and exercise text mining on twitter messages dividing tweets into 2 categories covering disaster related and not disaster related. Goswami et al. [28] used Decision Tree CART algorithm for the classification task. The degree of accuracy can depend on many factors. The data clearance and initial statistics analysis for text messages are the first remarkable tasks for removing noise and selecting suitable method. Each text database has its characters and appropriate method need to be explored.

We have reported experiments for tweet disaster message classification. The study case shown that, after tokenization stage and feature extraction from kernel distributions, the feature array of a text message was processed to be transferred to images to apply CCN methods designed for images. The approach could be failed if the vectorized text data appeared short vectors, which provide insufficient number of features for CNN works. Thus, the combination of NLP method and image pattern CNN needs strong linguistic analysis and kernel distribution extraction in the initial stage of learning.

Table 2. *Results for reference*

Method	Database	Accuracy (%)
BERT [25]	CrisisLexT26	67.00
LSTM CNN [26]	Hurricane Irma	74.78
NLP [27]	Hurricane Irma	81.89
Decision Tree CART [28]	Hurricane Irma	71.50
VGG16 [19] (our)	Kaggle Tweeter	73.70
ResNet101 [23] (our)	Kaggle Tweeter	75.71
GoogleLeNet [20] (our)	Kaggle Tweeter	77.65
Inception V3 [21] (our)	Kaggle Tweeter	82.42

4. Conclusion

The article presented an image pattern-based method for an NLP problem. The CNN method groups image patterns together with NLP tokenization and feature engineering to perform classification for tweet disaster messages. The resulting class represent true or fake news that need to be detected. Consequently, this method is potentially very valuable for text message classification with assistance of image patterns. Within the early stage, a range of text leaning and text analysis is essential to remove noise and to create new features based on kernel distribution. The transformation of the feature set to image allows selecting suitable learning method for implementation.

The variety of the CNN methods grants a set of solutions to select one. To facilitate differentiation between methods, text data is processed by the same preparation to get representation by images. It can be seen from experimental results that image patterns of the text database were classified the best by Inception V3. This is caused by the suitability of the CNN method with the database. Results from experiments have so far have given confidence, setting an opening base for carrying out further study into implementation of various image patterns techniques for text classification. Future research will cover categorizing tweet messages by searching other CNN methods for performance improvement.

REFERENCES

[1] J. R. Finkel, T. Grenager, and C. Manning, “Incorporating non-local information into information extraction systems by Gibbs sampling,” *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics* (ACL’05), 2005, pp. 363–370.

[2] Y. Kryvasheyeu, H. Chen, E. Moro, P. V. Hentenryck, and M. Cebrian, “Performance of social network sensors during Hurricane Sandy,” *PloS ONE*, vol. 10, no. 2, 2015, Art. no. e0117288, doi: 10.1371/journal.pone.0117288.

[3] B. Herfort, J. P. Albuquerque, S. J. Schelhorn, and A. Zipf, “Does the spatiotemporal distribution of tweets match the spatiotemporal distribution of flood phenomena? A study about the River Elbe Flood in June 2013, Twitter Analysis of River Elbe Flood,” *Proceedings of the 11th International ISCRAM Conference*, May 2014, pp. 1-6.

[4] B. Resch, F. Uslander, and C. Havas. “Combining machine-learning topic models and spatiotemporal analysis of social media data for disaster footprint and damage assessment,” *Proceeding of the Cartography and Geographic Information Science 45.4,* 2018, pp. 362-376.

[5] K. Stowe, J. Anderson, M. Palmer, L. Palen, and K. Anderson, “Improving Classification of Twitter Behavior During Hurricane Events,” *Workshop on Natural Language Processing for Social Media*, 2018, pp. 67-75.

[6] M. Park, Y. Sun, and M. L. McLaughlin, “Social media propagation of content promoting risky health behavior,” *Proceeding conference Cyberpsychology, Behavior, and Social Networking*, 2017, pp. 278-285.

[7] K. Garimella, G. D. F. Morales, A. Gionis, and M. Mathioudakis, “Quantifying controversy on social media,” *ACM Transactions on Social Computing*, vol. 1, no. 1, pp. 1-27, 2018.

[8] V. Vasilis, “The importance of Neutral Class in Sentiment Analysis,” 2013. [Online]. Available: https://blog.datumbox.com/the-importance-of-neutral-class-in-sentiment-analysis/. [Accessed February 10, 2022].

[9] C. E. Schuller, B. Xia, and Y. Havasi, “New avenues in opinion mining and sentiment analysis,” *Conference IEEE Intelligent Systems*, vol. 28, no. 2, pp. 15-21, 2013.

[10] L. H. Lin, S. B. Miles, and N. A. Smith, “Natural Language Processing for Analyzing Disaster Recovery Trends Expressed in Large Text corpora,” *2018 IEEE Global Humanitarian Technology Conf.*, October 2018, pp. 1-8.

[11] S. Verma, S. Vieweg, W. J. Corvey, L. Palen, J. H. Martin, M. Palmer, A. Schram, and K. M. Anderson, “Natural Language Processing to the Rescue? Extracting Situational Awareness Tweets During Mass Emergency,” *Proceedings of the Fifth International Conference on Weblogs and Social Media*, 2011, pp. 545-554.

[12] S. H. Li, D. Caragea, C. Caragea, and N. Herndon, “Disaster Response Aided by Tweet Classification with a Domain Adaptation Approach,” *Journal of Contingencies and Crisis Management (JCCM)*, Special Issue on HCI in Critical Systems, pp. 1-20, 2017.

[13] K. Stowe, M. Paul, M. Palmer, L. Palen, and K. Anderson, “Identifying and Categorizing Disaster-Related Tweets, Inter,” *Workshop on Natural Language Processing for Social Media*, 2016, pp. 1-6.

[14] R. S. Joao, “On Informative Tweet Identification for Tracking Mass Events,” *Proceedings of the 13th International Conference on Agents and Artificial Intelligence*, vol. 2, 2021, pp. 1226- 1273, doi: 10.5220/0010392712661273

[15] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735- 1780, 1997, doi:10.1162/neco.1997.9.8.1735.

[16] M. Y. Kabir and S. Madria, “A Deep Learning Approach for Tweet Classification and Rescue Scheduling for Effective Disaster Management,” *Proceedings of the 27th ACM Sigspatial International Conference on Advances in Geographic Information*, 2019, pp. 269-278.

[17] Trim, “The Art of Tokenization,” *IBM Developer Works*, 2013. [Online]. Available: https://trimc-nlp.blogspot.com/2020/11/the-art-of-tokenization.html. [Accessed February 10, 2022].

[18] D. Barber, *Bayesian Reasoning and Machine Learning*. Cambridge University Press, 2007.

[19] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” *International Conference on Learning Representations*, 2015, pp. 1-14.

[20] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A.Rabinovich, “Going Deeper with Convolutions,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1-9.

[21] A. G. Howard, “Some improvements on deep convolutional neural network based image classification,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 11-20.

[22] X. T. Dang and N. A. Dao, “Deep Learning-Based Imbalanced Data Classification for Chest X-Ray Image Analysis,” *The International Conference on Intelligent Systems & Networks*, vol. 243, Springer, 2021, pp. 109-115.

[23] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770-778.

[24] Kaggle, “Natural Language Processing with Disaster Tweets,” Aug. 2015, [Online]. Available: https://www.kaggle.com/c/nlp-getting-started/overview. [Accessed February 10, 2022].

[25] G. Ma, “Tweets Classification with BERT in the Field of Disaster Management,” *Workshop Department of Civil Engineering 2019*, Stanford University, 2019, pp. 1-15.

[26] M. A. Sit, C. Koylu, and I. Demir, "Identifying disaster related tweets and their semantic, spatial and temporal context using deep learning, natural language processing and spatial analysis: a case study of Hurricane Irma," *International Journal of Digital Earth,* vol. 12, no. 11, pp. 1205-1229, 2019.
[27] B. Klein, F. Castanedo, I. Elejalde, D. L. de-Ipina, and A. P. Nespral, "Lecture Notes in Computer Science," *Ubiquitous Computing and Ambient Intelligence. Context-Awareness and Context-Driven Interaction.* vol. 8276. Springer, Cham - LNISA, 2013, pp. 239-246.
[28] G. Shriya, and R. Debaditya, "Identification of Disaster-Related Tweets Using Natural Language," *Inter. Conf. on Recent Trends in AI, IOT, Smart Cities & App.*, 2020, pp. 28-36.