Flood Detection in Twitter Using a Novel Learning Method for Neural Networks

Rabiul Islam Jony
School of Computer Science
Queensland University of Technology
Brisbane, Queensland, Australia
r.jony@qut.edu.au

Alan Woodley
School of Computer Science
Queensland University of Technology
Brisbane, Queensland, Australia
a.woodley@qut.edu.au

ABSTRACT

In this paper we use a novel backpropagation technique, Direct Backpropagation (DBP), to train a neural network and use it to detect flooding in Twitter posts. We use the textual information from the tweets and the visual features from the associated images to classify the posts into two categories, flood (1) and no-flood (0). We also fuse these two modes using fusion methods for the classification. For the classification task we employ a neural network that we train using our proposed method instead of typical backpropagation method. This work has been done in the context of the MediaEval 2020 Flood-Related Multimedia Task.

1 INTRODUCTION

Satellite images have been used for flood detection for decades [5–7]. However, with the worldwide dominance of social media usage, they are becoming popular for a similar application [3]. The Flood-related Multimedia Task aims to detect flooding relevancy of social media data [1]. Here, The textual features and the visual features of a Twitter dataset are used separately and then fused for this purpose. We test both feature-level and decision fusion approaches for data fusion and use the training dataset to train our neural network that employs Direct Backpropagation (DBP) method proposed in [4] for the learning.

The results show that among all of our runs, feature-level fusion of textual information performs best and achieves the highest F-Score of 0.41, which was around 6% higher than the average of all participants. Our visual information only run also achieves better than average F-Score. However, the fusion of visual and textual information, both feature-level and decision fusion performs poorly compared to the average.

2 METHODOLOGY

2.1 Textual

For the textual approach we adapt the Bag-of-Words (BoW) method for text feature extraction. The textual data were provided as in full text and hashtags. However, the full texts, called the description, also contained the hashtags in them. Therefore, we removed the keywords from them and collected the keywords separately. We also remove the URLs, any punctuation, usernames and alphanumeric symbols from the full text. The keywords of a tweet, called tags, were provided as separate words. We joined them together to create a single string containing all keywords of a tweet.

We extract both uni-gram and bi-gram features from the description and tags and then calculate their term frequency inverse document frequency. Then we employ the chi-squared method to select the best 2048 features from the description and tags to make sure the visual features and the textual features have similar number of features in the fusion process.

2.2 Visual

For the visual features we employed Xception[2] pre-trained on ImageNet dataset, that extracted 2048 feature from each image. We prepossessed the images by resizing them to 299X299 dimensions and converted the bands to RGB (Red-Green-Blue). We also employed InceptionV3 [8] for extracting another set of visual features. However, they were not included in the submitted run due to their poor performance on validation set.

2.3 Direct backpropagation

We used the described textual and visual features for the classification task, where we employed a neural network. This neural network uses direct backpropagation (DBP) technique instead of typical backporpagation (BP) method. This technique sends the cost function calculated in the last layer back to every layer to calculate their gradients which removes the dependency on previous layer in the learning process and hence reduces the processing time and cost. The direct backpropagation method is described more elaborately in [4].

Equation (1) and (2) show the hidden layer update direction for BP, and DBP respectively. Here, considering a three layered network, δa_2 and δa_1 are the second and first layer gradients respectively, $f'()$ is the derivative of the non-linearity, $\odot$ is an element-wise multiplication operator, B are random feedbacks, e is the gradient at the last layer, known as the cost function, W are the forward weights and W^T are the symmetric weights. Our proposed learning method can be presented by equation (2).

$$\delta a_2 = (W_3^T e) \odot f'(a_2), \delta a_1 = (W_2^T \delta a_2) \odot f'(a_1) \quad (1)$$

$$\delta a_2 = (W_3^T e) \odot f'(a_2), \delta a_1 = (W_2^T e) \odot f'(a_1) \quad (2)$$

2.4 Fusion

Here, we used two types of fusion approaches, feature-level fusion and decision fusion. In feature-level fusion, the features from different modes, such as visual or textual are concatenated together and used as input feature of the neural network. In decision fusion,

MediaEval'20, December 14-15 2020, Online

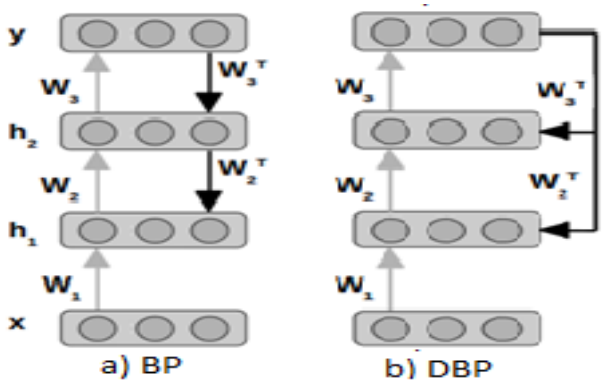

Figure 1: Overview of Different Learning Methods

the probability values of relevancy to flooding generated by each mode separately are averaged together to calculate the relevancy of a post.

3 RESULTS

3.1 Run 1

As instructed in the task, run 1 uses fusion of textual and visual data. We used a feature-level fusion of description, tags and Xception for this run.

3.2 Run 2

Run 2 uses textual information only. However, we also implemented a fusion method by adding a feature-level fusion of description and tags.

3.3 Run 3

This is a visual information only run. We used visual features extracted by Xception for this run.

3.4 Run 4

In this run we took textual information only similar to run 2. However, here the description and the tags were fused in a decision fusion manner.

3.5 Run 5

Run 5 is also a decision fusion run. Here we used description, tags, Xception features in decision fusion manner.

The neural network was trained with a 10-fold cross validation for every experiment.

3.6 Discussion

Table 1 shows the F-score results generated by the organizers on our submitted runs and the average of all participants. It shows that our textual only, visual only and fusion runs perform better than average. However, the visual features perform poorly compared to the textual information, both in our submission and in average. This is because of the nature of the dataset. As described in the task description, the tweets were retrieved by using the keywords as

Table 1: Results Achieved Using Different Modes

Run	Modes	F-score	Average F-score
Run 1	Description, Tags, Xception	0.1478	0.1415
Run 2	Description, Tags	0.4158	0.3938
Run 3	Xception	0.1436	0.1318
Run 4	Description, Tags	0.1402	0.1562
Run 5	Description, Tags, Xception	0.0546	0.1991

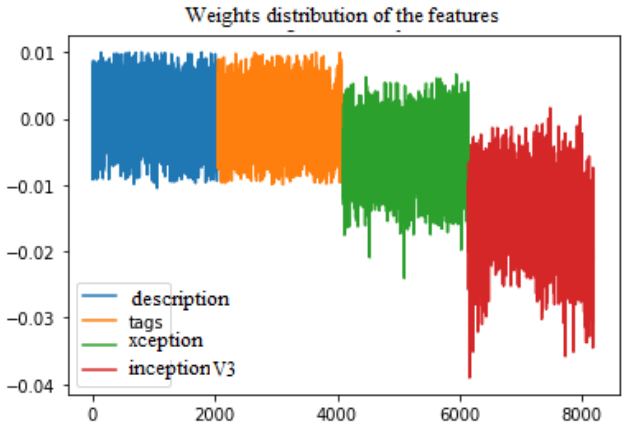

Figure 2: Weight Values Distribution of Each Mode

search criteria and then the tweets were annotated, not the associated images. Manual inspection of the associated images showed that many of the flood relevant tweets did not contain any flooding evidence in the images. Our results and the average scores also show that fusion of the visual information with textual information degraded the result compared to textual only, which also identifies the visual information as noise. We have also inspected the weights generated by the neural network for each features. To evaluate each mode's credibility we inspected the weight values generated by the neural network for each feature and presented in Figure 2. Here, we used four modes namely, description, tags, Xception and InceptionV3. It shows that the visual features have average of negative weights, that means they have a negative impact in the classification process. InceptionV3 had the lowest weight values with an average of -0.02 and therefore, performed very poorly when it was used for the classification. That is why we did not include it in any of the submitted runs.

4 CONCLUSION

We illustrated our approaches for the MediaEval 2020 Flood-Related Multimedia Task. Our approaches contained five runs, where we used the textual information and the visual information separately and also in fused manner. The average results of all participants were not promising (highest average F-score was 0.3571). The highest F-score we achieved was 0.4158, using textual information only, where we fused the tweet text and the keywords. The visual features performed poorly and degraded the overall performance when fused with textual information.

REFERENCES

[1] Stelios Andreadis, Ilias Gialampoukidis, Anastasios Karakostas, Stefanos Vrochidis, Ioannis Kompatsiaris, Roberto Fiorin, Daniele Norbiato, and Michele Ferri. 2020. The Flood-related Multimedia Task at MediaEval 2020. (2020).

[2] François Chollet. 2017. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1251–1258.

[3] Rabiul Islam Jony, Alan Woodley, and Dimitri Perrin. 2019. Flood Detection in Social Media Images using Visual Features and Metadata. In *2019 Digital Image Computing: Techniques and Applications (DICTA)*. IEEE, 1–8.

[4] Rabiul Islam Jony, Alan Woodley, and Dimitri Perrin. 2020. Fusing Visual Features and Metadata to Detect Flooding in Flickr Images. In *2020 Digital Image Computing: Techniques and Applications (DICTA)*. IEEE, 1–8.

[5] Rabiul Islam Jony, Alan Woodley, Aishvarya Raj, and Dimitri Perrin. 2018. Ensemble Classification Technique for Water Detection in Satellite Images. In *2018 Digital Image Computing: Techniques and Applications (DICTA)*. IEEE, 1–8.

[6] Chandrama Sarker, Luis Mejias Alvarez, and Alan Woodley. 2016. Integrating recursive Bayesian estimation with support vector machine to map probability of flooding from multispectral Landsat data. In *2016 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*. IEEE, 1–8.

[7] Chandrama Sarker, Luis Mejias, Frederic Maire, and Alan Woodley. 2019. Flood mapping with convolutional neural networks using spatio-contextual pixel information. *Remote Sensing* 11, 19 (2019), 2331.

[8] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2818–2826.