Model Fusion for Efficient Flood-Related Twitter Posts Detection

Georgia Christodoulou[1], Stelios Kontogiannis[1], Konstantinos Avgerinakis[1]

[1] Catalink Limited, Nicosia, Cyprus

Abstract
The evolution of Artificial Intelligence has impacted many aspects of our lives. Recently, the aim has been placed on building intelligent and autonomous systems, able to assist the first responders to plan and manage their actions faster and more efficiently in emergency situations. For emergency events, data can be usually gathered from many sources, however one of the most valuable can be considered the collection of real-time data from social media. Real-time reporting through social media can be leveraged to monitor the progress of a critical situation, and then be exploited to improve the monitoring, organisation and response of the event from the responsible departments. In the presented work, we developed two modules, including a Natural Language Processing model recognising which social media posts may refer to 'floods', and a Computer Vision model distinguishing between the image posts that depict a 'flood' situation, from those that do not. While both models achieved remarkably accurate results, we decided to fuse their prediction scores to see if we can improve their performance, leading finally to the introduction of two low level AI representations and one higher level that leverages the former to provide its decision. Their comparison and valuable insights are presented in the following paragraphs.

1 INTRODUCTION

Flood is one of the most common natural disasters, happening when an excess of water submerges normally dry ground. Floods are frequently caused by prolonged periods of heavy rain, quick melting of the snow, storm surges from tropical cyclones or tsunamis in coastal zones. Floods can wreak havoc across a large area, killing people, damaging private property, and destroying vital public health facilities. More than 2 billion people worldwide were impacted by floods between 1998 and 2017 [15]. The most vulnerable people to floods are those who reside on floodplains, in non-flood proof structures, or who lack warning systems and awareness of the risk of flooding. In essence, there are three primary forms of floods: flash floods, river floods, and coastal floods. All three of them are becoming more frequent and intense, and climate change is projected to contribute to this trend [16].

Throughout history, numerous technological advancements have shielded human lives from danger. In our times, the widespread use of smartphones allows individuals to report problems in real time. For instance, Twitter has become a vital tool for emergency communication, leading more and more organisations to consider automating Twitter monitoring for satisfying their purposes. If someone tweeted about an emergency or impending tragedy and it was promptly detected by NLP models, we would be able to react more quickly than usual, potentially saving lives. The goal of this competition based on the overview paper [1] is to develop a machine learning model that can determine which Tweets are about actual disasters and which ones are not, potentially contributing to the concept of using Twitter for actual natural disasters.

2 RELATED WORK

On the one hand, researchers mainly focus on exploring **computer vision methods** for flood monitoring, flood inundation mapping, debris flow estimation and post-flood damage estimation [2]. Since Convolutional Neural Networks (CNNs) are best suited for processing images, they have been employed for this kind of tasks and have demonstrated great success [6]. Szara et al. [5] proposed a method that initially extracts handcrafted features from the data, and then trains conventional machine learning and deep learning models (VGG-16) for floodwater detection on roadways. Kang et al. [7] proposed a Fully Connected Network (FCN) for flood mapping on satellite images, while Gebrehiwot et al. [3] introduced a similar CNN architecture that is trained to extract flooded areas from UAV imagery. Thirumarai Selvi et al. [4] suggested an AlexNet-based model, for mapping flooded regions on remote sensing satellite images. Several works investigate encoder-decoder model schemes for satellite image segmentation and classification of flooded and non-flooded areas, such as Nemni et al. [8], Hashemi-Beni and Gebrehiwot [9] and Liu et al. [10].

Another interesting piece of work that inspired us, was [11], which employs and trains CNNs for real-time fire detection. Two state-of-the-art, powerful and compact CNNs, NasNet and ShuffleNetV2, are developed and trained in order to perform fire detection in real-time. As we explain in Section 3.1, we worked with ShuffleNetV2 and we re-purposed it for flood detection.

On the other hand, **Natural language processing (NLP)** has embraced transfer learning as a standard practice since word-embedding [24], sentence-embedding [23], and more recently BERT [20] based models have demonstrated considerable improvements in downstream tasks. Despite the existence of some pre-trained language embeddings, performing natural language processing in non-English languages—such as the Italian tweets used in the competition—is challenging. Even though multilingual natural language processing is still an ongoing research area, it is currently possible to achieve great results employing the following three methods: leveraging multilingual models [18], applying English translation for non-English content [19] and, lastly, the technique we deployed, which involved working with a particular non-English model [20].

MediaEval'22, 12-13 January 2023, Bergen, Norway and Online

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073

3 APPROACH

3.1 Computer Vision Model

The computer vision algorithm was developed based on ShuffleNetV2 [12]. The reason that we chose the particular model to work with was that it can be easily deployed on IoT devices, since the architecture is lightweight, and so it can be easily employed for flood detection applications. ShuffleNetV2 is specifically designed for mobile vision applications and it is an improved version of ShuffleNetV1 [13].

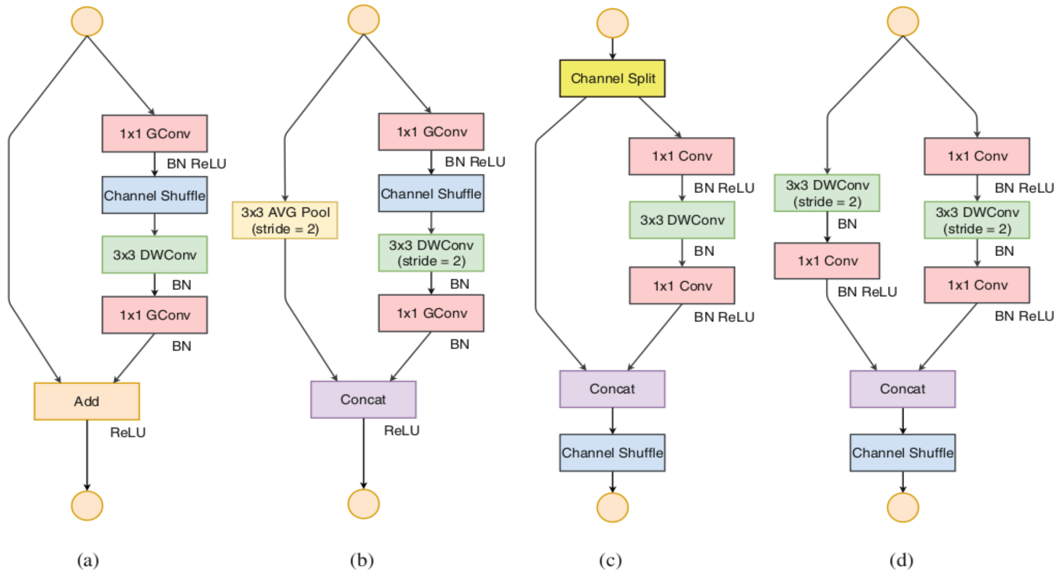

Fig 1: The fundamental units of the new architecture are the re-designed building blocks of ShuffleNetV1. (a) the basic ShuffleNetV1 unit; (b) the ShuffleNetV1 unit for spatial down sampling (2x); (c) ShuffleNetV2 Normal cell; (d) ShuffleNetV2 Reduction cell (for spatial down sampling, 2x).
DWConv: depthwise convolution. Gconv: group convolution
Figure source: [12]

More specifically, ShuffleNetV1 has been re-designed in such a way so that it satisfies some specific guidelines that the authors propose and explain in their work [12]. For example, they suggest to have equal number of input and output channels in the convolution layers, and carefully choosing the group number of convolutions, to minimise the MAC[1]. ShuffleNetV2 introduces the *channel-split* operator and changes the design of the building blocks of ShuffleNetV1 (Fig. 1(a) and 1(b)). The new building blocks are the Normal Cell and the Reduction Cell. The Normal Cell (Fig.1(c)), splits the input feature channels into two branches. One branch remains as identity, while the other branch contains three convolutions with the same input and output number of channels. After convolution, the two branches are concatenated so the number of input and output channels is the same. The *channel-shuffle* operation is also incorporated in order to enable information communication between the two branches. The Reduction Cell (Fig.1(d)) on the other hand, is used for spatial downsampling. The channel split operator is removed, so the number of output channels is doubled. The building blocks are stacked to build the whole network. The overall architecture of ShuffleNetV2 is summarised in Table 1.

The number of channels in each building block is scaled to generate four networks of different complexities. For the task of flood detection, we used the simplest architecture, i.e. the one with the 0.5x level of complexity, since it has the least number of trainable parameters (1.4 million) and executes the least number of FLOPs[2] (41 million).
For the model training, we used the Transfer Learning technique, where a pre-trained model can be repurposed on a new problem. For our training, we used as a starting point the ShuffleNetV2, that has been pre-trained on ImageNet dataset [14]. Then, we removed the fully-connected layers and we added new ones, while all the convolutional layers were set as non-trainable. The fully-connected layers were re-trained, but this time on the flood detection dataset, which is a combination of images collected from different sources. Some data were distributed from the MediaEval contest, while we completed the dataset with freely available data found from Unsplash [25], Pixabay [26], Flickr [27]. The rest of the data sources can be found at the references section[28][29][30][31][32].

1 Memory Access Cost
2 Float-Point Operations, i.e. the number of multiply-adds

Layer	Output size	KSize	Stride	Repeat	Output channels			
					0.5×	1×	1.5×	2×
Image	224×224				3	3	3	3
Conv1	112×112	3×3	2	1	24	24	24	24
MaxPool	56×56	3×3	2					
Stage2	28×28		2	1	48	116	176	244
	28×28		1	3				
Stage3	14×14		2	1	96	232	352	488
	14×14		1	7				
Stage4	7×7		2	1	192	464	704	976
	7×7		1	3				
Conv5	7×7	1×1	1	1	1024	1024	1024	2048
GlobalPool	1×1	7×7						
FC					1000	1000	1000	1000
FLOPs					41M	146M	299M	591M
# of Weights					1.4M	2.3M	3.5M	7.4M

Table 1: Overall architecture of ShuffleNetV2, for four different levels of complexities. For our work, we used the 0.5x level of complexity, since it contains the least number of weights and requires the least FLOPs. Table source: [12]

3.2 NLP Model

Initially, any noise that might make it difficult to classify the tweet's text had to be carefully removed. We used regular expressions techniques Regex[3] to remove any links, users, hashtags, or audio/video tags and more specifically, the model was given lowercase text without square brackets, links, punctuation, words with numbers, and emojis. Using tokens is, for the most part, a necessary evil since recent methods have demonstrated the possibility of removing them altogether and instead operating on the raw text directly [21]. Given the non-English text, we chose a huggingface[4] model, more precisely the *bert-base-italian-cased*[5], whose source data includes a recent Wikipedia dump and other texts from the OPUS[6] corpus collection.

4 RESULTS AND ANALYSIS

4.1 Computer Vision Model Experiments

The model was trained to distinguish images that contain floods from those that do not. To find the optimal model, different experiments were conducted, trying different number of channels at the last convolutional layer (conv5), several hyper-parameter combinations and different optimization algorithms (Stochastic Gradient Descent and Adam optimization). For the model training, we split the dataset into training (85%) and validation (15%) dataset. The first was used for the model training, while the second was used to validate the performance of the model and to tune the hyper-parameters accordingly. Some of the experiments that we executed are summarised in Table 2. Due to the fact that our dataset was imbalanced, we measured the performance of the models using F1-score. The model that scored the best performance was model 7, with F1-score = 0.81.

One difficult part of the whole procedure was to gather appropriate negative data (i.e. images that did not contain floods). It is straightforward to gather data that contain floods, but for the negative class, it was not that easy since we had to include images from rivers, the sea and other water bodies in general. In that way, we trained our model with the indent to be able to distinguish between images that contained water bodies, and those that contained floods.

	Optimizer	Loss function	Batch size	Epochs	Learning rate	Output channels	F1-score
Model 1	SGD	Cross entropy	64	30	0.0001	[24,48,96,192,1024]	0.68
Model 2	Adam	Cross entropy	64	30	0.001	[24,48,96,192,64]	0.77
Model 3	Adam	Cross entropy	64	40	0.001	[24,48,96,192,128]	0.79
Model 4	Adam	Cross entropy	64	40	0.001	[24,48,96,192,256]	0.78
Model 5	Adam	Cross entropy	64	40	0.001	[24,48,96,192,128]	0.77
Model 6	Adam	Cross entropy	64	50	0.001	[24,48,96,192,128]	0.78
Model 7	Adam	Cross entropy	64	100	0.001	[24,48,96,192,128]	0.81

Table 2: Summary of some of the experiments conducted in order to find the optimal model.

3 https://docs.python.org/3/library/re.html

4 https://huggingface.co/

5 https://huggingface.co/dbmdz/bert-base-italian-cased

6 https://opus.nlpl.eu/

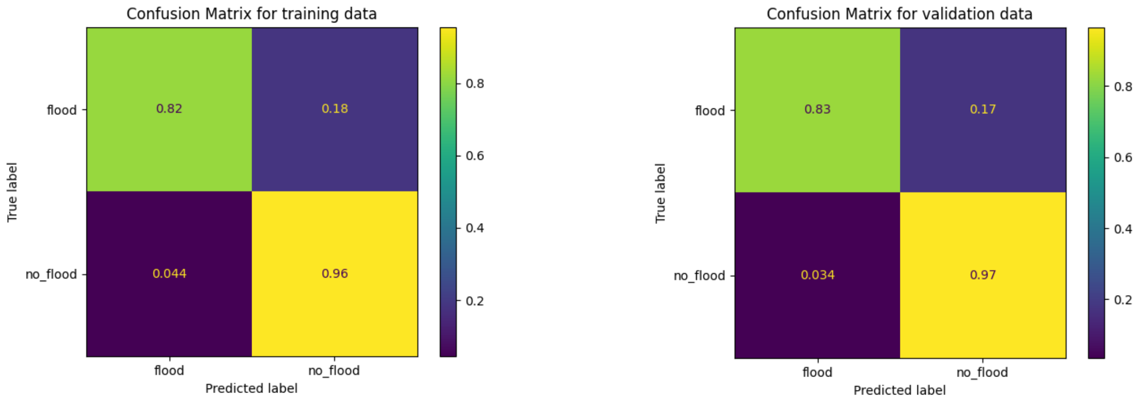

Fig.2 Normalized confusion matrices for the training and validation dataset. The flood class scored 82% and 83% classification accuracy for training and validation, while the no_flood class scored 96% and 97% respectively.

4.2 NLP Model Experiments

The algorithm was also trained to distinguish between tweet texts that reference floods and those that do not. In an effort to find the best model, numerous tests were run using different combinations of hyper-parameters. We utilised K-Fold cross validation to divide the dataset into 5 consecutive folds for model training, and we determined the f1 score for each validation-train split. The first was used to train the model, and the second to evaluate the model's performance and modify the hyper-parameters as needed. The average F1-score of our validation sets was around 94% even from the first trials. As a result, we made minor model changes while still attempting to significantly raise the high F1 score. Having to confirm that our model did not overfit the data was a challenging aspect of the entire process. We therefore tested multiple K-fold splittings with varying K values and many training epochs, but the average F1 score results remained consistent at around 94%. Our search for assurance of not overfitting was satisfied by results on the hidden test set in the following table. Given that *False Positive* (FP) and *False Negative* (FN) make up a minor part of the *True Positive* (TP) and *True Negative* (TN) tweets, there is little evidence of overfitting, as can be seen.

Run	# Correct IDs	# Missing IDs	# Extra IDs	TP	FP	TN	FN	Precision	Recall	F-Score
001	1.315	0	0	485	46	748	36	0,91337	0,93090	0,92205

Fig.3 Brief results; Run type: automated using textual information only.

4.3 Model fusion

The two models mentioned above were integrated in this section in an attempt to make a combined prediction for the test set. We chose to lean more heavily on the CNN model because we were not certain whether the NLP model was overfitting or not. Additionally, since the majority of the predictions made by our NLP model had a confidence level of above 94%, we decided against using the average of the two models and instead moved forward with some manual criteria. For each tweet where the CNN model scored at least 80% confidence score towards the positive or the negative class, we therefore allocated the respective value, while for the remaining tweets, we took the average. As we can see in the table below, this technique performed at about 85%, which is slightly inferior to the NLP model prediction.

Run	# Correct IDs	# Missing IDs	# Extra IDs	TP	FP	TN	FN	Precision	Recall	F-Score
002	1.315	0	0	433	61	733	88	0,87651	0,83109	0,85320

Fig.4 Brief results; Run type: automated using fused textual and visual information (including using image data from external sources).

5 CONCLUSIONS

This research presents three methods for categorising tweets related to flood catastrophes, which may enable further use of Twitter to mitigate the effects of severe natural disasters. It has been proven that both non-English text and images from tweet posts can be used effectively. Based on the findings, we may infer that the NLP model outperforms the computer vision model mostly due to the data itself, as images frequently lack acuity when describing the context of a tweet. Future work will focus on ways to enhance the fused version of our model, such as utilising OCR[7], weighting more text when an image lacks pertinent context or when we detect a human being in the bulk of the image, along with expanding the use of NLP models in non-English languages like German, French, and others.

Acknowledgement

This work has been performed in the context of the SILVANUS project, which has received funding from the European Community’s Horizon 2020 research and innovation programme under the grant agreement No 101037247.

[7] https://web.archive.org/web/20160415060125/https://dev.havenondemand.com/apis/ocrdocument

REFERENCES

[1] **Task overview paper:**
Maria Authorsen and Jacques de Coauthor. Cool Task: Challenges, Dataset and Evaluation. Proc. of the MediaEval 2022 Workshop, Bergen, Norway and Online, 12-13 January 2023

[2] Arshad B, Ogie R, Barthelemy J, Pradhan B, Verstaevel N, Perez P. Computer Vision and IoT-Based Sensors in Flood Monitoring and Mapping: A Systematic Review. Sensors (Basel). 2019 Nov 16;19(22):5012.

[3] Gebrehiwot A, Hashemi-Beni L, Thompson G, Kordjamshidi P, Langan TE. Deep Convolutional Neural Network for Flood Extent Mapping Using Unmanned Aerial Vehicles Data. Sensors (Basel). 2019 Mar 27;19(7):1486.

[4] Thirumarai Selvi C., and Kalieswari S., “Convolutional Neural Network Based Flood Detection Using Remote Sensing images”, Easychari preprint no. 2235, https://easychair.org/publications/preprint/CSNF

[5] Sazara C., Cetin M. and Iftekharuddin K. M., "Detecting floodwater on roadways from image data with handcrafted features and deep transfer learning," 2019 IEEE Intelligent Transportation Systems Conference (ITSC), 2019, pp. 804-809

[6] Bentivoglio, R., Isufi, E., Jonkman, S. N., and Taormina, R.: Deep Learning Methods for Flood Mapping: A Review of Existing Applications and Future Research Directions, Hydrol. Earth Syst. Sci. Discuss.

[7] Kang, W., Xiang, Y., Wang, F., Wan, L., and You, H.: Flood Detection in Gaofen-3 SAR Images via Fully Convolutional Networks, Sensors, 18, https://doi.org/10.3390/s18092915, 2018

[8] Nemni, E., Bullock, J., Belabbes, S., and Bromley, L.: Fully Convolutional Neural Network for Rapid Flood Segmentation in Synthetic Aperture Radar Imagery, Remote Sensing, 12

[9] Hashemi-Beni, L. and Gebrehiwot, A.: Flood Extent Mapping: An Integrated Method Using Deep Learning and Region Growing Using UAV Optical Data, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 14, 2127–2135, 2021.

[10] Liu B., Li X., and Zheng G.: Coastal Inundation Mapping From Bitemporal and Dual-Polarization SAR Imagery Based on Deep Convolutional Neural Networks, Journal of Geophysical Research: Oceans, 124, 9101–9113, 2019.

[11] Thomson W., Bhowmik N. and Breckon T. P., "Efficient and Compact Convolutional Neural Network Architectures for Non-temporal Real-time Fire Detection," 2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA), 2020, pp. 136-141

[12] Ma N., Zhang X., Zheng H. T., Sun J., “ShuffleNetV2: Practical Guidelines for Efficient CNN Architecture Design”, Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 116-131

[13] Zhang X., Zhou X., Lin M., Sun J., “ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices”

[14] Deng W. Dong R. Socher L. -J. Li, Kai Li and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248-255

[15] World Health Organisation, “Floods.”, www.who.int, https://www.who.int/health-topics/floods#

[16] Zurich, “ Avoiding Atlantis: Will rising seas and coastal flooding wipe out some cities?”, www.zurich.com, June 2022, https://www.zurich.com/en/knowledge/topics/global-risks/can-coastal-cities-hold-back-the-sea

[17] Mapfre, “Artificial Intelligence will save us from natural disasters.”, www.mapfre.com, Sept. 2021, https://www.mapfre.com/en/insights/sustainability/natural-disasters-artificial-intelligence/

[18] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems. 3111–3119

[19] Alexis Conneau, Douwe Kiela, Holger Schwenk, Loic Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. arXiv preprint arXiv:1705.02364 (2017).

[20] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018).

[21] Sagen, Markus, Peltarion, “Towards a token-free future in NLP”, Jan., 2022, https://peltarion.com/blog/data-science/towards-a-token-free-future-in-nlp

[22] Andreadis S., Bozas A., Gialampoukidis I., Moumtzidou A., Fiorin R., Lombardo F., Mavropoulos T., Norbiato D., Vrochidis S., Ferri M., Kompatsiaris I., DisasterMM: Multimedia Analysis of Disaster-Related Social Media Data Task at MediaEval 2022, in: Proceedings of the MediaEval 2022 Workshop, Bergen, Norway and Online, 2023.

[23] Heidenreich H., Evaluation of sentence embeddings in downstream and linguistic probing tasks, Towards Data Science,Aug.,2018,https://towardsdatascience.com/paper-summary-evaluation-of-sentence-embeddings-in-downstream-and-linguistic-probing-tasks-5e6a8c63aab1

[24] Heidenreich H., Introduction to Word Embeddings, Towards Data Science, Aug.,2018 https://towardsdatascience.com/introduction-to-word-embeddings-4cf857b12edc

Datasets:

[25] Unsplash, https://unsplash.com/

[26] Pixabay, https://pixabay.com/

[27] Flickr, https://www.flickr.com/

[28] Roadway flooding Image dataset, https://www.kaggle.com/datasets/saurabhshahane/roadway-flood-dataset

[29] European Flood 2013 Dataset (v1.0), https://github.com/cvjena/eu-flood-dataset

[30] Alamy, https://www.alamy.com/

[31] iStockphoto, https://www.istockphoto.com/

[32] Stock Adobe, https://stock.adobe.com/