A Domain-based Late-Fusion for Disaster Image Retrieval from Social Media

ELEDIA@UTB and The 2017 Multimedia Satellite Task: Emergency Response for Flooding Events

Minh-Son Dao[1], Quang-Nhat-Minh Pham[2], Duc-Tien Dang-Nguyen[3]
[1]ELEDIA@UTB Lab., Universiti Teknologi Brunei, Brunei
[2]FPT Technology Research Institute, FPT University, Hanoi, Vietnam
[3]Dublin City University, Dublin, Ireland
minh_son@utb.edu.bn,minhpqn2@fe.edu.vn,duc-tien.dang-nguyen@dcu.ie

ABSTRACT

We introduce a domain-specific and late-fusion algorithm to cope with the challenge raised in The MediaEval 2017 Multimedia Satellite Task. Several known techniques are integrated based on domain-specific criteria such as late fusion, tuning, ensemble learning, object detection using deep learning, and temporal-spatial-based event confirmation. Experimental results show that the proposed algorithm can overcome the main challenges of the proper discrimination of the water levels in different areas as well as the consideration of different types of flooding events.

1 INTRODUCTION

This paper presents a method that is specially built to meet the subtask 1 of the MediaEval 2017 Multimedia Satellite Task [2]. We propose an ensemble learning and tuning method for Disaster Image Retrieval from Social Media in order to overcome the challenge of using restricted visual features and metadata as well as to increase the accuracy of classification by using data from external resources. Details about the methodology are given in the section 2, while results on this task are reported and discussed in section 3, and the conclusion and future works are stated in the last section.

2 METHODOLOGY

Following subsections discuss each run required by the organizer.

2.1 Visual-based Image Retrieval

We formalize the problem as an ensemble learning and tuning task, in which we use visual features provided by the organizer associated with each image. These visual features are used with supervised learners to create classifiers. Our visual-based method includes three components: late fusion, tuning, and ensemble learning, designed as follows:

- **Stage 1**: a set of classifiers associated with each type of features is created by using supervised learners (SLs) with outputs reported in a regression form (i.e. the output variable takes continuous values). The late fusion technique is applied to these outputs to form the multimodal features combination (MFC), as follows: $MFC = \sum_{i=1}^{N}(w_i * SL_i)$ where $\sum(w_i) = 1$
- **Stage 2**: the bagging is utilized on these features and learners to increase the accuracy of the system. A multiple data set (MDS) is created by dividing randomly the training data set into m non-intersecting folders (DS_m). At the end of this stage, the combined classifier is created, namely the combined learner (CL).
- **Stage 3**: after running a testing phase on these data sets, a new training data set for tuning is created and learned by using a specific supervised learner and a certain subset of feature types. The output of this stage is called the tuning leaner (TL). The subsection 2.1.1 describes how to establish this tuning data set. The idea behind this stage is to apply boosting and bagging techniques to create a tuning classifier for samples which fall into a wrong side of a hyperplane zone of a previous classifier.
- **Stage 4**: the ensemble learner (EL) is created as $EL = w1 * TL + w2 * CL$ where $w1 + w2 = 1$.

2.1.1 *Creating a tuning data set.*

(1) Let $PR_k = \{pred_i, gt_i\}_{i=1:N}$ be an output set when using CL from the stage 2 of DS_k, where $pred_i$ and gt_i denote the predicted value and the ground-truth label of the i^{th} image, respectively. The descending sort algorithm is applied to PR_k so that the image with the biggest predicted value will be on the top. Here, the predicted label of the i^{th} image is calculated by $label_i = f(pred_i, threshold)$.
(2) For each i^{th} image whose ground-truth label gt_i and predicted label $label_i$ do not match, collecting its k nearest neighbour images whose gt_j equals $label_j$ (i.e. collect k *true positive* and *true negative* neighbours) by getting $k/2$ samples from position $(i-1)$ to 1 and from position $(i+1)$ to N, respectively.

2.2 Meta-data-based Image Retrieval

We formalize the image retrieval as a text categorization task, in which we extract textual features from meta-data associated with each image. Textual features are used as basic features for training a feed forward neural network (FFNN) with just one hidden layer. Our meta-based classification method includes three components: preprocessing, feature extraction, and training neural network, described as follows:

(1) **Pre-processing**: we clean text data by removing hyperlinks, image path, image names. We also remove all URL

MediaEval'17, 13-15 September 2017, Dublin, Ireland

in texts. We perform those steps by using some regular expressions. After that, we do word tokenization and remove all punctuations in a text. For a user tag containing multiple words, we join words in the tag to form a phrase and treat that phrase as a single word. We use *nltk toolkit* [1] to perform word tokenization.

(2) **Textual Feature Extraction**: For text categorization task, bag-of-words are basic and straightforward features. We just use bag-of-words features and represent an image as a n-dimension sparse vector in which n is limit of the number of words in the vocabulary extracted from the training data set. A feature is activated if meta-data of the image contains the corresponding word in the vocabulary. In our experiments, we use $n = 10,000$. We extract features from three attributes in the meta-data including: title, description, tags of an image.

(3) **Neural Network Architecture**: We use a feed-forward neural network with one hidden layer containing 128 units. In training the network, we used batch size 20, and applied drop-out technique with drop-out coefficient 0.5. In the output layer, we use softmax layer so that the final network can output probability values that an image is related to flood or not. We adopted keras framework [4] for building the neural network.

2.3 Visual-metadata-based Image Retrieval

The method used for visual-based image retrieval, described in subsection 2.1 is utilized for this task. All features and supervised learners of visual-based and metadata-based methods are reused.

2.4 Visual-metadata-external-resource-based Image Retrieval

Based on the observation that a water texture and colour and a spatial relation between the water area and its surrounding area can lead to the misclassification when using the proposed method with limited visual features provided by the organizer. Besides, a metadata content and it's associated image do not always synchronize by an event meaning. Hence, we propose the domain-specific algorithm to overcome these obstacles.

We utilize the faster R-CNN [6] with the pre-computed object detection model running on the Tensorflow platform [3] to generate a bag of words containing objects that semantically related to flooded areas, especially in *industrial*, *residential*, *commercial*, and *agricultural* areas. Moreover, we use location and time information described in a metadata to confirm whether a flood really happened by checking with weather databases that can be freely accessed on the Internet (e.g. Europe area[1], America area[2], and Australia area[3]). This task can be done by reusing the method introduced in [5].

The former component, namely syn-content model (SC) is used to give more weighted values to the pair image-metadata that shares the similar content. The latter component is to strengthen the accuracy of the meta-data-based model, namely spatio-temporal model (ST). These components are used to re-label and create a tuning data set in the stage 3.

[1]https://www.eswd.eu/
[2] http://www.weather.gov/, https://water.usgs.gov/floods/reports/
[3]http://www.bom.gov.au/climate/data/

3 EXPERIMENTAL RESULTS

We use the data set and evaluation metrics provided by the organized [2]. All parameters used by our approach are set as follows:

- **RUN 1**: (1) *Stage 1*: we use features set $\{CED, EH, JC\}$ and SVM (*SVM-Type: eps-regression, SVM-Kernel: radial, cost: 1, gamma: 0.006944444, epsilon: 0.1*) to create SLs, the weighted w_is are set by 1/3, (2) *Stage 2*: we divide the development set in to m=10 non-intersect data set, (3) *Stage 3*: k is set by 10, and random forest (RF) is used to create TL (*type of random forest: regression, number of trees: 500, No. of variables tried at each split: 48*) with the JC feature as the input, and (4) *Stage 4*: we set $w_1 = 0.4, w_2 = 0.6$
- **RUN 2**: We perform 5-fold cross validation on the development set and report the average scores of 5 folds. In testing, we train the model using development set and use the trained model for the prediction on the test images.
- **RUN 3 and RUN 4**: We use pre-computed object detection model without changing any parameter [6][3]. We also reused methods in RUN 1 and RUN 2 with the same setup.

Table 1 shows the evaluation results on the development and test data.

Table 1: Evaluation Results on Development and Test Data

	Dev. Set		Test Set	
Run	AP@480	MAP	AP@480	MAP
Run 1	82.17	88.84	77.62	87.87
Run 2	83.4	87.8	57.07	57.12
Run 3	85.78	92.86	85.41	90.39
Run 4	92.53	98.38	90.69	97.36

For run 1 and 2, there is a big gap between results on the development set and on test set. Possible explanations are that (1) the data distribution of the development set is different than of the test set, and/or (2) there are too much non-synchronize between text and image contents. In the current work, we did not deal with inflections such as "flood" and "flooded", so the feature space is quite sparse. *For run 3*, the fusion of visual-based and metadata-based outputs can improve the accuracy of flood images retrieval, around 3% higher. It proves that these two different approaches can compensate their weaknesses. *For run 4*, there is a significant improvement of the accuracy when using SC model and ST information whilst the visual-metadata-based method (e.g. run 3) reaches it's limitation.

4 CONCLUSIONS AND FUTURE WORKS

We introduce the domain-based late-fusion method to retrieve flood images using social media. Although the achievement at this stage is acceptable, there are so many things can be improved to get better results, especially to overcome the non-synchronized content between image's and metadata's and the suitable features and/or learners to distinguish water/flooded areas by its color, texture, and its spatial relation with surrounding areas.

REFERENCES

[1] Steven Bird, Ewan Klein, and Edward Loper. 2009. Natural Language Processing with Python.

[2] Benjamin Bischke, Patrick Helber, Christian Schulze, Srinivasan Venkat, Andreas Dengel, and Damian Borth. The Multimedia Satellite Task at MediaEval 2017: Emergence Response for Flooding Events. In *Proc. of the MediaEval 2017 Workshop* (Sept. 13-15, 2017). Dublin, Ireland.

[3] Xinlei Chen and Abhinav Gupta. 2017. An Implementation of Faster RCNN with Study for Region Sampling. *arXiv preprint arXiv:1702.02138* (2017).

[4] François Chollet and others. 2015. Keras. https://github.com/fchollet/keras. (2015).

[5] Minh-Son Dao, Giulia Boato, Francesco G.B. De Natale, and Truc-Vien Nguyen. 2013. Jointly Exploiting Visual and Non-visual Information for Event-related Social Media Retrieval. In *Proceedings of the 3rd ACM Conference on International Conference on Multimedia Retrieval (ICMR '13)*. ACM, New York, NY, USA, 159–166. https://doi.org/10.1145/2461466.2461494

[6] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Advances in Neural Information Processing Systems (NIPS)*.