Multimedia Tools and Applications manuscript No.
(will be inserted by the editor)

Neural relational inference for disaster multimedia retrieval

Samuel G. Fadel* · Ricardo da S. Torres

Received: date / Accepted: date

Abstract Events around the world are increasingly documented on social media, specially by the people experiencing them, as these platforms become more popular over time. As a consequence, social media turns into a valuable source of data for understanding those events. Due to their destructive potential, natural disasters are among events of particular interest to response operations and environmental monitoring agencies. However, this amount of information also makes it challenging to identify relevant content pertaining to those events. In this paper, we use a relational neural network model for identifying this type of content. The model is particularly suitable for unstructured text, that is, text with no particular arrangement of words, such as tags, which is commonplace in social media data. In addition, our method can be combined with a CNN for handling multimodal data where text and visual data are available. We perform experiments in three different scenarios, where different modalities are evaluated: visual, textual, and both. Our method achieves competitive performance in both modalities by themselves, while significantly outperforms the baseline on the multimodal scenario. We also demonstrate the behavior of the proposed method in different applications by performing additional experiments in the CUB-200-2011 multimodal dataset.

Keywords Machine learning · Neural networks · Information Retrieval · Natural language processing · Multimodal

Mathematics Subject Classification (2010) 68T05 · 68T45 · 68T50

S. G. Fadel
Institute of Computing, University of Campinas, Av. Albert Einstein, 1251, Campinas 13083-852, Brazil.
E-mail: samuel.fadel@ic.unicamp.br

R. da S. Torres
Department of ICT and Natural Sciences, Faculty of Information Technology and Electrical Engineering, NTNU - Norwegian University of Science and Technology, Ålesund, Norway.
E-mail: ricardo.torres@ntnu.no

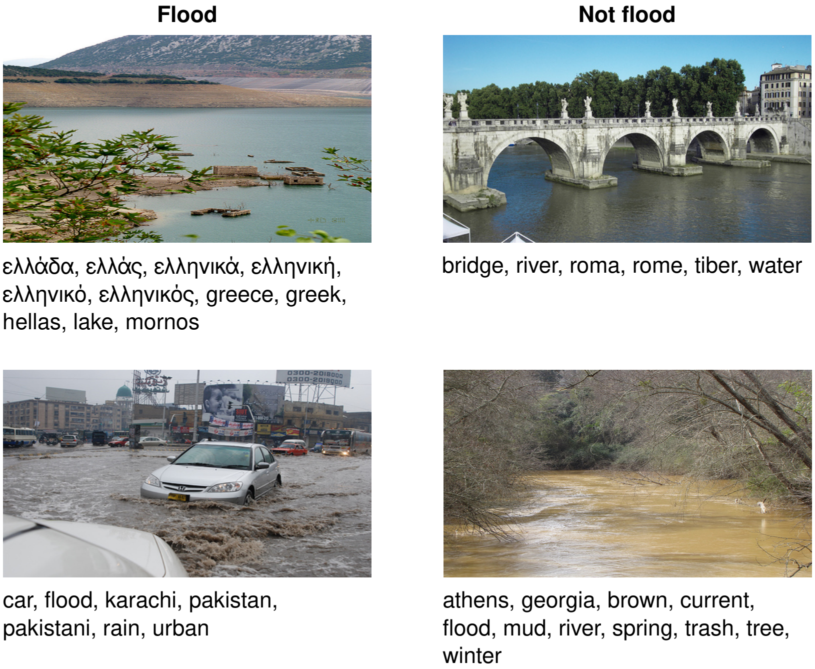
Flood
Not flood
ελλάδα, ελλάς, ελληνικά, ελληνική, ελληνικό, ελληνικός, greece, greek, hellas, lake, mornos
bridge, river, roma, rome, tiber, water
car, flood, karachi, pakistan, pakistani, rain, urban
athens, georgia, brown, current, flood, mud, river, spring, trash, tree, winter

Fig. 1 Some examples of multimedia items from the Disaster Image Retrieval from Social Media sub-task of the Multimedia Satellite Task. Items consist of an image either depicting a flooding event or not, alongside user-provided tags. Notice that tags might have words from multiple languages.

1 Introduction

The pervasiveness of social media has led many to use them as means of communication and a source of information and news. Using social media, events around the world are documented not only by news outlets, but also by the people experiencing them. As a consequence, social media streams are becoming a valuable source of data for understanding, detailing, and assessing such events. Among those events are natural disasters, which are of particular interest for emergency response operations and environmental monitoring, given their destructive potential. However, automatically identifying posts, pictures, and videos related to natural disasters has become both a necessity and a challenge due to the sheer amount of information on social media platforms.

The Multimedia Satellite Task [5], which was part of the Multimedia Evaluation Benchmark (MediaEval) 2017, proposed a retrieval task of flooding evidence from social media. The task consists of ranking a collection of multimedia items comprised of either images, their associated metadata, or both, such that those depicting flooding events should be ranked higher than others. Examples of such multimedia items are shown in Figure 1. Our particular interest in this task lies in handling text data where words are not arranged in any particular order, as it is commonplace with tags in social media posts.

In this paper, we introduce a solution to this problem that addresses learning from unstructured text data, that is, instead of regular sentences, we use words that are not arranged in any particular order. Specifically, it consists of a relational inference model based on neural networks that can infer relational structure from a set of objects. We consider scenarios using only text data and using both text and visual data.

In summary, our contributions are twofold: (i) proposal of a relational inference model to support multimodal representation of multimedia objects; (ii) demonstration of its effectiveness in the retrieval task related to natural disasters introduced by the Multimedia Satellite Task. We also demonstrate the behavior of the proposed method to different applications by performing additional experiments in the CUB-200-2011 multimodal dataset.

2 Related work

This section provides an overview of related work focusing on multimodal analysis in the context of the Multimedia Satellite Task.

Bischke et al. [5] proposed a solution to the Multimedia Satellite Task using a Support Vector Machine (SVM) with a radial basis function kernel. The SVM was trained using visual features extracted via a pre-trained Convolutional Neural Network (CNN) based on the X-ResNet architecture [13] and the metadata was represented using word2vec [21] embeddings, trained on the metadata itself. For handling both modalities at the same time, they used a concatenation of both visual and textual feature vectors.

More recently, Dourado et al. [7] proposed the use of multiple descriptors instead of relying on a single representation for objects, producing one similarity-based ranking per object. Then, those rankings are aggregated using graphs capturing the relationships to other objects based on the different rankings, resulting in one graph for each object. This naturally allows their approach to handle multiple modalities. The produced graphs are embedded into a vector representation, which is used as the object representation. Graph-based formulations were also exploited in [23]. In such formulations, graphs are used as an early-fusion approach, which exploits embedding procedures based on bags of graphs, to generate multimodal vector representations.

Those approaches then rank the objects by using the score produced by an SVM trained on the vector representations of the objects. While their performance is reasonable, the metadata is handled by simply averaging their vector representations into a single vector representing all words. Moreover, compared to using modalities by themselves, both approaches result in a small improvement over the scores obtained by using only a single data modality.

In contrast, Zaheer et al. [29] proposed a neural network architecture, called DeepSets, that can process sets of objects. Their formulation, however, does not explicitly model the relationship between items in the input set. Thus, items are only taken into account with respect to each other when aggregating all of them, making it more difficult to express pairwise relationships.

In summary, existing approaches can be roughly categorized into early-fusion methods – some of them based on the concatenation of visual and textual feature vectors [5,6,8,10,19,27] or graph formulations [23] – and late-fusion approaches [1,2].

3 Neural relational inference in social media data

Neural networks currently attain state-of-the-art performance in a number of applications, prompting us to consider them for this task. Since we use both with visual data (images depicting flooding events) and their associated textual metadata, we consider both data modalities in our approach to this task. For visual data, Convolutional Neural Networks (CNNs) are the essential building block of current state-of-the-art models. Moreover, ResNets [11] are competitive in transfer learning scenarios [17] while being straightforward to train due to a significantly smoother loss surface, encouraging their use.

For the textual metadata, we resort to a relational approach. Neural networks designed for natural language data exploit the sequential nature of sentences, relying on the order in which words are presented on input. As a consequence, most of them are based on either RNNs [14,20] or CNNs [9,15]. However, here we focus on applications where text data are available as tags, where a document is simply a collection of words related to its subject, in no particular order. Consequently, we refrained from looking at the metadata from a sentence understanding perspective. We instead use a model that can produce decisions from collections of objects given as input.

Relation Networks (RNs) [25] provide an elegant framework for our task. Put simply, a RN combines two neural networks, f_ϕ and g_θ, whose parameters ϕ and θ are learned together. Given a collection of objects $\mathcal{O} = \{x_1, x_2, \ldots, x_n\}$ as input, g_θ takes each pair of objects as input and encodes them. These encoded values are then added up to represent the whole collection and given as input to f_ϕ, which produces the output of the RN. These operations can be expressed as

$$\mathrm{RN}(\mathcal{O}) = f_\phi\left(\sum_{x_i, x_j \in \mathcal{O}} g_\theta(x_i, x_j)\right). \tag{1}$$

This way of handling the input is advantageous for this task, since RNs can take a variable number of objects as input. With this, we are not restricted to a fixed number of words in each document. Moreover, these models can infer which word relations are important without explicit supervision over them.

3.1 Architecture

The first step is to map each one of the input words into t-dimensional vectors, which the RN uses as object representations. We adopted two different strategies for this, namely a word lookup table and distributed word representation.

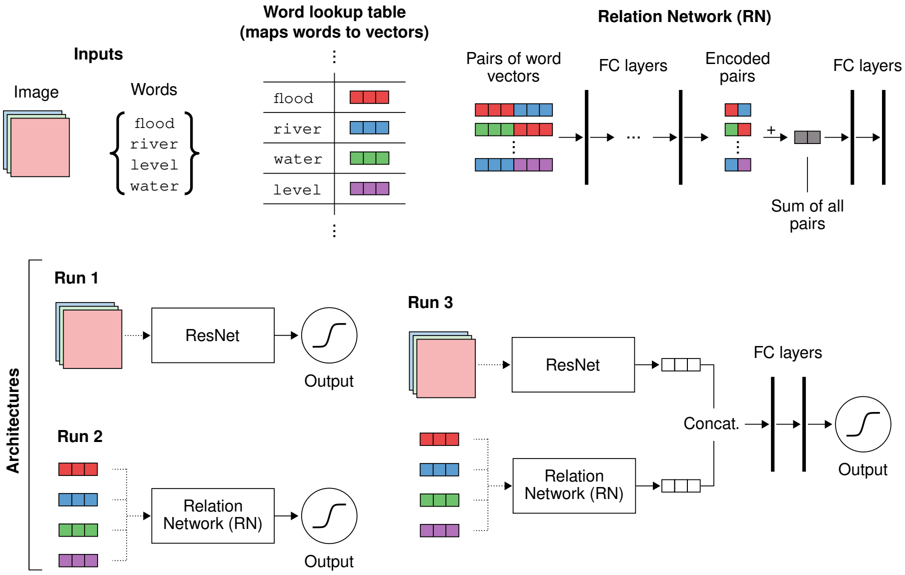

Fig. 2 The neural network modules and architectures used in our experiments. The RN causes the models to look into the word relationships, not only presence, then summarizing those relationships into a single vector used for downstream tasks.

Our intent with this is to evaluate the RN under different conditions of input representation and whether these can significantly affect its performance.

Let V be the (totally) ordered set of all words in the training set. Note that this order is arbitrary and does not affect the learning process. A word lookup table is a matrix $E \in \mathbb{R}^{|V| \times t}$, where each row of E represents some word in V. In this approach, we initially represent each word as a 1-hot $|V|$-dimensional vector, where the index of the nonzero value is the index of this word in V. The final word representation is computed by simply projecting the 1-hot vector using E, producing the t-dimensional vector. As the vector representations are learned from scratch, this approach has two main advantages. First, they are constantly improved while training the neural network. Second, we do not require a large number of parameters dedicated to learning word representations. This comes with the drawback that our vocabulary is limited to what is seen in the training data, thus any unseen words are mapped to vectors of zeros and the network must learn to ignore them. An overview of the architecture illustrating this strategy is shown in Figure 2.

In the case of distributed word representations, we replace our word lookup table with fastText [22] word embeddings, a general-purpose distributed continuous representation model, which was trained on a corpus of 16B tokens.

In both architectures, f_ϕ and g_θ are comprised of fully-connected layers of SELUs units, followed by an additional layer with a sigmoid unit for producing the classification output. We chose this architecture based on two main results from previous work. First, Klambauer et al. [16] have recently shown on a large number of datasets that SELUs, when used in architectures consisting of fully-connected layers, outperform other widely used activation functions, such as ReLUs, while not requiring techniques such as batch normalization [12]. Additionally, Santoro et al. [25] successfully used 4- and 3-layer fully-connected ReLU networks, for f_ϕ and g_ϕ respectively, in a scenario involving 25 objects per data sample, while in our case there are 10 objects (tags) on average.

For the multimodal scenario, where we have both images and their metadata, we use a larger architecture that incorporates a CNN for image data and the same RN-based architecture previously described for the metadata. The CNN we employ is a ResNet-18 [11] pre-trained on ImageNet, but with its classification layer replaced by a fully-connected layer of 512 units. Then, the CNN and RN outputs are fully-connected by a layer with one sigmoid unit for classification.

4 Experimental setup

4.1 Datasets

We evaluate our proposed architecture on two datasets. In both, samples from the validation set are not used in the test set.

DIRSM: The "Disaster Image Retrieval from Social Media" was subtask of the Multimedia Satellite Task from MediaEval 2017. Its dataset is comprised of 6,600 images from the YFCC100M dataset [26] alongside their metadata, such as description, tags, and title. Each set of words is the union of all individual words found in all tags, that is, we break tags into individual words if there are more than one per tag. Individual words are obtained by employing the NLTK [3] word tokenizer. Since the dataset is not large, this was done to ease the difficulty of learning better representations for tags, as their occurrence rates should be higher. We then remove multiple occurrences of the same words in the same document. The dataset has a predefined test set corresponding to 20% (or 1,320) samples. We set aside 1,054 samples from the training set for validation purposes.

CUB-200-2011: The Caltech-UCSD Birds-200 [28] dataset consists of 11,788 images of birds, labeled into 200 classes. Each image contains 312 binary attributes describing characteristics of the bird therein. As they include a certainty level between 1 and 4, we filter out all attributes associated with each bird whenever it has a certainty below 3. We transform those remaining into a set of tags, making this similar to the previous scenario. The dataset has predefined train and test sets with 5,994 and 5,794 samples, respectively. We used 25% (or 1,158) of test samples for validation purposes.

Table 1 Hyperparameters and search domains used by Hyperband ($R = 243$ and $\eta = 3$). In the DIRSM dataset, we select as optimal the hyperparameters from the model that achieved the lowest validation loss among 10 complete Hyperband runs.

Hyperparameter	Search domain
Batch size	32, 64, 128
Learning rate	2.5×10^{-3}, 2.5×10^{-4}, 2.5×10^{-5}
Weight decay	10^{-2}, 10^{-3}, 10^{-4}
# layers (f_ϕ)	2, 3, 4, 5
# layers (g_θ)	2, 3, 4, 5
# units/layer (f_ϕ)	128, 256, 384, 512
# units/layer (g_θ)	128, 256, 384, 512
Dropout	Uniform in $[0.1, 0.3]$
t	50, 100, 200, 300

4.2 Evaluation

In the DIRSM dataset, we adopt the same evaluation protocol as the Multimedia Satellite Task competition [5]. It consists of three different scenarios, each of them evaluating one modality: in *run 1*, only images are provided as input; in *run 2*, only the metadata are provided; finally, in *run 3*, both images and metadata are used.

In the test phase, we evaluate the test set items ranked by the confidence that they depict a flooding event, from most to least confident. They are evaluated using the Average Precision (AP) at cut-off 480 and the mean AP at cut-offs 50, 100, 250, and 480.

In the CUB-200-2011 dataset, we evaluate the F1 score of the models in classifying samples of birds into one of the 200 classes, averaged over all classes. We consider two scenarios: only tags, and both images and tags.

4.3 Baselines

We compare our method to three baselines. The first two, proposed by Bischke et al. [4] and Dourado et al. [7], described earlier, use a SVM with the RBF kernel as a classifier to produce scores for ranking. We also compare to a fully-connected neural network with multiple layers (MLP), with no relational step, using the mean vector of all words as input, to highlight the influence of the relational step in our approach. Furthermore, we use fastText [22] embeddings pre-trained on a large corpus as an alternative text representation to evaluate our method and the MLP and SVM baselines. Specifically for the CUB-200-2011 dataset, we do not perform comparisons using fastText embeddings as the tags are non-textual.

4.4 Training details

We train the proposed methods using AMSGrad [24]. Each training session lasts for at most 50 epochs, evaluating the model against the validation set at each epoch. If the validation loss does not improve for 10 epochs, we stop the training procedure earlier.

Hyperparameters were optimized using the Hyperband [18] algorithm on the validation set, with $R = 243$ and $\eta = 3$. The optimal hyperparameters are selected individually for each modality, that is, one for each task: *run 1* (images), *run 2* (text), and *run 3* (multimodal). In addition, we individually optimize hyperparameters for the RN and MLP variants that use fastText. These hyperparameters are summarized in Table 1. We also optimize hyperparameters for the SVM on the validation set, but using a grid search. The search is over $10^{-2} \leq C \leq 10^{10}$ and $10^{-9} \leq \gamma \leq 10^{3}$, both spaced evenly on a logarithmic scale.

5 Results and discussion

The results for each scenario are summarized in Table 2. In order to better highlight the differences in performance, we also compare the precision at each cut-off from 1 to 480 of all neural network approaches, both in *run 2* and *run 3*, shown in Figure 3 and Figure 4, respectively. While we are focused on the performance of the RN for learning from text data (*run 2*), we still consider the performance of the CNN on the image-only scenario (*run 1*), thus we can compare how the CNN performs by itself and when used in the multimodal scenario (*run 3*). Even though our CNN based on ResNet-18 has almost 4 times fewer parameters than one of the baselines [4], while the other [7] uses multiple deep neural networks as feature extractors, all approaches have similar performance, with the baselines being slightly better.

In *run 2*, we see that the neural network architectures achieved competitive results with the SVM baseline using fastText, all of those being significantly better than the baselines in AP@480, but close to the best approach in terms of MAP. In particular, the RN with embeddings learned from scratch had very similar performance to the SVM with fastText embeddings. While using fastText embeddings with the RN led to better AP@480, its MAP had a larger variance. The MLP baseline, however, was inferior to both of those, using either learned embeddings or fastText embeddings. Inspecting Figure 3, however, both the RN and the MLP display a large variance in the results at the first few positions. On average, the RN-based networks pull ahead at around $k = 50$, the RN with fastText approach being the best on average at around 250, and then they become similar for the rest of the elements. The MLP with fastText falls behind specially after the first 400 elements, where the precision drops faster than other approaches.

As already mentioned, the learned lookup table is restricted to the words seen during training. Since it cannot produce representations for unknown

Table 2 Average Precision at 480 (AP@480) and Mean Average Precision (MAP) at cut-offs 50, 100, 250 and 480 on the test set of the DIRSM dataset. Values presented for our models (detailed in Section 3.1) are the mean and standard deviation of 10 distinct training sessions with randomly initialized weights. Models that use fastText representations are indicated with $^{\mathrm{fT}}$. In all cases, ResNet-18 was initialized from weights of a pre-trained (ImageNet) model. We use the best result by Dourado et al. [7] of their proposed variants in each scenario.

		AP@480 (%)	MAP (%)
Run 1	ResNet-18	84.8 ± 1.3	95.1 ± 0.8
	Bischke et al. [4]	86.64	95.71
	Dourado et al. [7]	**88.41**	**96.74**
Run 2	RN	82.8 ± 1.0	85.6 ± 1.5
	MLP	82.0 ± 0.6	84.4 ± 0.8
	RN$^{\mathrm{fT}}$	83.2 ± 1.5	85.5 ± 2.3
	MLP$^{\mathrm{fT}}$	81.9 ± 1.1	83.5 ± 1.6
	SVM (rbf)$^{\mathrm{fT}}$	**84.00**	85.97
	Bischke et al. [4]	63.41	77.64
	Dourado et al. [7]	73.81	**88.09**
Run 3	ResNet-18 + RN	97.2 ± 0.3	99.1 ± 0.3
	ResNet-18 + MLP	95.1 ± 0.7	97.5 ± 1.3
	ResNet-18 + RN$^{\mathrm{fT}}$	**97.3** ± 0.3	**99.2** ± 0.1
	ResNet-18 + MLP$^{\mathrm{fT}}$	97.3 ± 0.2	99.1 ± 0.1
	Bischke et al. [4]	90.45	97.40
	Dourado et al. [7]	90.96	97.63

words, they are mapped into vectors of zeros and the model must learn to ignore them. While the performance attained with both approaches for representing text are close with the RN, the simpler learned representations are more sensitive to the hyperparameters described in Table 1. This suggests that the RN can perform reasonably well with good object representations, being more robust to the architecture, but hyperparameter search also plays a significant role.

Finally, in *run 3*, the models using only neural network solutions were the best performing, being above other baselines by a significant margin. Both previously published results [5, 7] improve slightly in the multimodal scenario compared to the ones where only images or text are available. The MLP showed the largest improvement in the use of fastText embeddings, while the RN had very similar performance both with and without fastText, as was the case in *run 2*. Upon inspection of precision scores, we see that, with the exception of the MLP with learned embeddings, the methods were indeed very close in performance and had relatively small variances. Moreover, the RN with fastText embeddings had the best precision, on average, at the first 200 positions in the produced rankings, dropping slightly below others after.

These results highlight an advantage of using neural networks when fusing different modalities: the CNN and RN were trained jointly in an end-to-end manner, requiring no significant changes to their individual architectures.

We summarize the results on the CUB-200-2011 dataset in Table 3. In this case, we see the MLP baseline clearly outperforms the RN. This can be at-

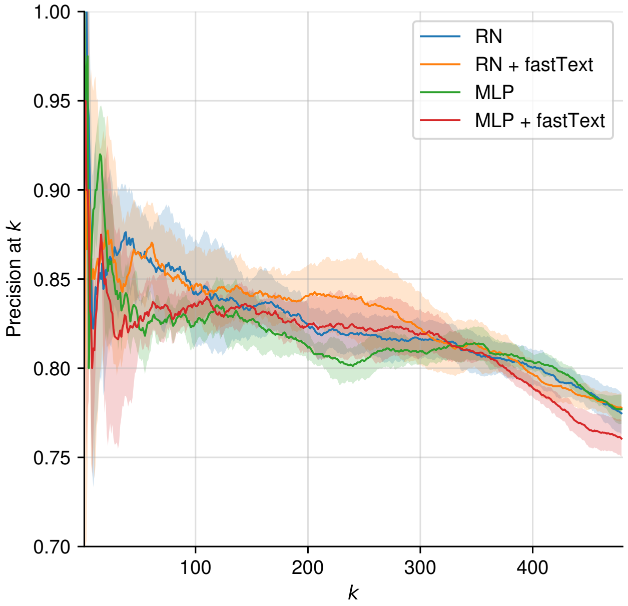

Fig. 3 Precision scores at each cut-off (k) from 1 to 480 attained by the neural network architectures in run 2 (detailed in Section 4.2). The scores shown are the average of 10 training procedures with randomly initialized weights; shaded areas represent standard deviation.

Table 3 F1 scores on the test set of the CUB-200-2011 dataset. Values presented are the mean and standard deviation of 5 distinct training sessions with randomly initialized weights. In all cases, ResNet-18 was trained from scratch.

		F1 score (%)
Run 2	RN	2.6 ± 0.6
	MLP	9.2 ± 1.8
Run 3	ResNet-18 + RN	1.7 ± 1.1
	ResNet-18 + MLP	8.0 ± 1.2

tributed to two main factors. First, the dataset contains a much higher number of objects (tags) per sample, resulting in a high number of pairs to be analyzed by the RN. Second, as opposed to the previous dataset, the tags represent binary attributes. Thus, a model that only accounts for their presence, rather than their relations, can perform the necessary task. This observation goes in line with the one observed earlier: when the relationships between tags show

6 Conclusions

In this paper, we proposed a relational approach using neural networks, based on Relation Networks (RN) [25], for scenarios where textual data is available not as proper sentences, but as a collection of words (tags). We compare its

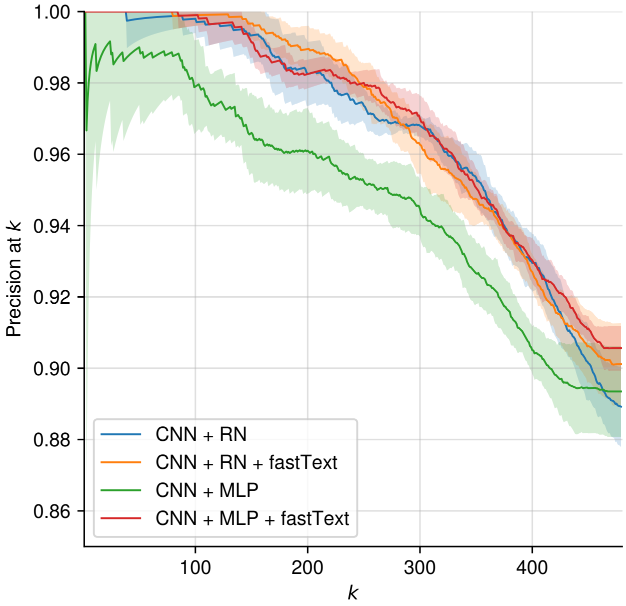

Fig. 4 Precision scores at each cut-off (k) from 1 to 480 attained by the neural network architectures in run 3 (detailed in Section 4.2). The scores shown are the average of 10 training procedures with randomly initialized weights; shaded areas represent standard deviation.

performance to more straightforward approaches, one using a feedforward neural network and one using SVM, in the case of social media data referring to natural disasters. Additionally, we also compare different types of vector representations for words and analyze how they affect the model performance in each case.

The experiments suggest that RNs are a competitive approach for this task, performing slightly better than the MLP and SVM baselines. They also indicate that distributed word representations such as fastText can lead to improvements, but those are not as significant. However, better object representations for the RN makes it more robust to hyperparameters, leading to a wider range of hyperparameters that can attain competitive results, while the simpler lookup table representation is more sensitive to them.

As they can be jointly trained with a CNN, we also performed experiments in a multimodal scenario. These show that neural network approaches, both RN and MLP, have a significant advantage over SVMs, even though the SVM uses visual features extracted using a CNN with almost four times more parameters. While MLPs are also competitive in this scenario, we observed that it is more sensitive than the RN with respect to the word representation employed.

Acknowledgements This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), Brazil, Finance Code 001, and grant

#88881.145912/2017-01. Authors are grateful to CNPq (grant #307560/2016-3), FAPESP (grants #2017/24005-2, #2014/12236-1, #2015/24494-8, #2016/50250-1, and #2017/20945-0), and the FAPESP-Microsoft Virtual Institute (grants #2013/50155-0, #2013/50169-1, and #2014/50715-9).

References

1. Ahmad, K., Pogorelov, K., Riegler, M., Conci, N., Pal, H.: Cnn and gan based satellite and social media data fusion for disaster detection.
   In: Proceedings of the MediaEval Workshop. Dublin, Ireland (2017)
2. Ahmad, S., Ahmad, K., Ahmad, N., Conci, N.: Convolutional neural networks for disaster images retrieval.
   In: MediaEval Workshop. Dublin, Ireland (2017)
3. Bird, S., Loper, E., Klein, E.: Natural Language Processing with Python.
   O'Reilly Media Inc. (2009)
4. Bischke, B., Bhardwaj, P., Gautam, A., Helber, P., Borth, D., Dengel, A.: Detection of flooding events in social multimedia and satellite imagery using deep neural networks.
   In: Working Notes Proceedings of the MediaEval 2017 Workshop, vol. 1984 (2017)
5. Bischke, B., Helber, P., Schulze, C., Venkat, S., Dengel, A., Borth, D.: The multimedia satellite task at MediaEval 2017.
   In: Working Notes Proceedings of the MediaEval 2017 Workshop, vol. 1984. Dublin, Ireland (2017)
6. Dao, M.S., Pham, Q.N.M., Nguyen, D., Tien, D.: A domain-based late-fusion for disaster image retrieval from social media.
   In: Proceedings of the MediaEval Workshop. Dublin, Ireland (2017)
7. Dourado, I., Tabbone, A.S., Torres, R.d.S.: Event Prediction based on Unsupervised Graph-Based Rank-Fusion Models.
   In: 12th International Workshop on Graph Based Representation in Pattern Recognition. Tours, France (2019)
8. Fu, X., Bin, Y., Peng, L., Zhou, J., Yang, Y., Shen, H.T.: Bmc@mediaeval 2017 multimedia satellite task via regression random forest.
   In: Proceedings of the MediaEval Workshop. Dublin, Ireland (2017)
9. Gan, Z., Pu, Y., Henao, R., Li, C., He, X., Carin, L.: Learning generic sentence representations using convolutional neural networks.
   In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 2390–2400. Association for Computational Linguistics, Copenhagen, Denmark (2017)
10. Hanif, M., Tahir, M.A., Khan, M., Rafi, M.: Flood detection using social media data and spectral regression based kernel discriminant analysis.
    In: Proceedings of the MediaEval Workshop. Dublin, Ireland (2017)
11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition.
    In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
12. Ioffe, S., Szegedy, C.: Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift.
    In: Proceedings of the 32nd International Conference on Machine Learning, pp. 448–456 (2015)
13. Jou, B., Chang, S.F.: Deep cross residual learning for multitask visual recognition.
    In: Proceedings of the 2016 ACM on Multimedia Conference - MM '16, pp. 998–1007. ACM Press, New York, New York, USA (2016).
    DOI 10.1145/2964284.2964309
14. Jozefowicz, R., Zaremba, W., Sutskever, I.: An empirical exploration of recurrent network architectures.
    In: Proceedings of the 32nd International Conference on Machine Learning, pp. 2342–2350. PLMR, Lille, France (2015)
15. Kim, Y.: Convolutional Neural Networks for Sentence Classification.

In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1746–1751. Association for Computational Linguistics, Stroudsburg, PA, USA (2014).
DOI 10.3115/v1/D14-1181

16. Klambauer, G., Unterthiner, T., Mayr, A., Hochreiter, S.: Self-normalizing neural networks.
In: Advances in Neural Information Processing Systems 30, pp. 971–980 (2017)
17. Kornblith, S., Shlens, J., Le, Q.V.: Do better imagenet models transfer better?
CoRR **abs/1805.08974** (2018)
18. Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A., Talwalkar, A.: Hyperband: A Novel Bandit-Based Approach to Hyperparameter Optimization.
Journal of Machine Learning Research **18**(185), 1–52 (2018)
19. Lopez-Fuentes, L., van de Weijer, J., Bolanos, M., Skinnemoen, H.: Multi-modal deep learning approach for flood detection.
In: Proceedings of the MediaEval Workshop. Dublin, Ireland (2017)
20. Melis, G., Dyer, C., Blunsom, P.: On the state of the art of evaluation in neural language models.
CoRR **abs/1707.05589** (2017)
21. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient Estimation of Word Representations in Vector Space.
CoRR **abs/1301.3781** (2013)
22. Mikolov, T., Grave, E., Bojanowski, P., Puhrsch, C., Joulin, A.: Advances in pre-training distributed word representations.
In: Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018) (2018)
23. de Oliveira Werneck, R., Dourado, I.C., Fadel, S., Tabbone, S., da S. Torres, R.: Graph-based early-fusion for flood detection.
In: 25th IEEE International Conference on Image Processing (ICIP), pp. 1048–1052 (2018).
DOI 10.1109/ICIP.2018.8451011
24. Reddi, S.J., Kale, S., Kumar, S.: On the Convergence of Adam and Beyond.
In: International Conference on Learning Representations (2018)
25. Santoro, A., Raposo, D., Barrett, D.G., Malinowski, M., Pascanu, R., Battaglia, P., Lillicrap, T.: A simple neural network module for relational reasoning.
In: Advances in Neural Information Processing Systems 30, pp. 4974–4983 (2017)
26. Thomee, B., Elizalde, B., Shamma, D.A., Ni, K., Friedland, G., Poland, D., Borth, D., Li, L.J.: YFCC100M: the new data in multimedia research.
Communications of the ACM **59**(2), 64–73 (2016).
DOI 10.1145/2812802
27. Tkachenko, N., Zubiaga, A., Procter, R.: Wisc at mmediaeval 2017: Multimedia satellite task.
In: Proceedings of the MediaEval Workshop. Dublin, Ireland (2017)
28. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The Caltech-UCSD Birds-200-2011 Dataset.
Tech. Rep. CNS-TR-2011-001, California Institute of Technology (2011)
29. Zaheer, M., Kottur, S., Ravanbakhsh, S., Poczos, B., Salakhutdinov, R.R., Smola, A.J.: Deep Sets.
In: Advances in Neural Information Processing Systems 30, pp. 3391–3401 (2017)