Collected data and classification of relevant one

- In this paper, we used social media messages posted on Twitter during catastrophic events. Although our system is able to use data from other social media (e.g., Facebook or Flickr), Twitter has been chosen because it is widely used in this application context as it allows to download large amounts of data through public APIs. Other social media, although more widespread and used than Twitter (Facebook and Instagram), do not allow researchers to download users' posts on a certain topic and therefore appear to be unusable.

- We used Twitter APIs for searching and collecting tweets matching keywords related to earthquakes, including those that occurred in Barletta (May 21, 2019) and Peru (May 26, 2019). From the analysis of the collected data, we noticed that some tweets report the earthquake and the problems/sub-events it generated (*relevant*), while others do not refer to the catastrophic event (*not relevant*).

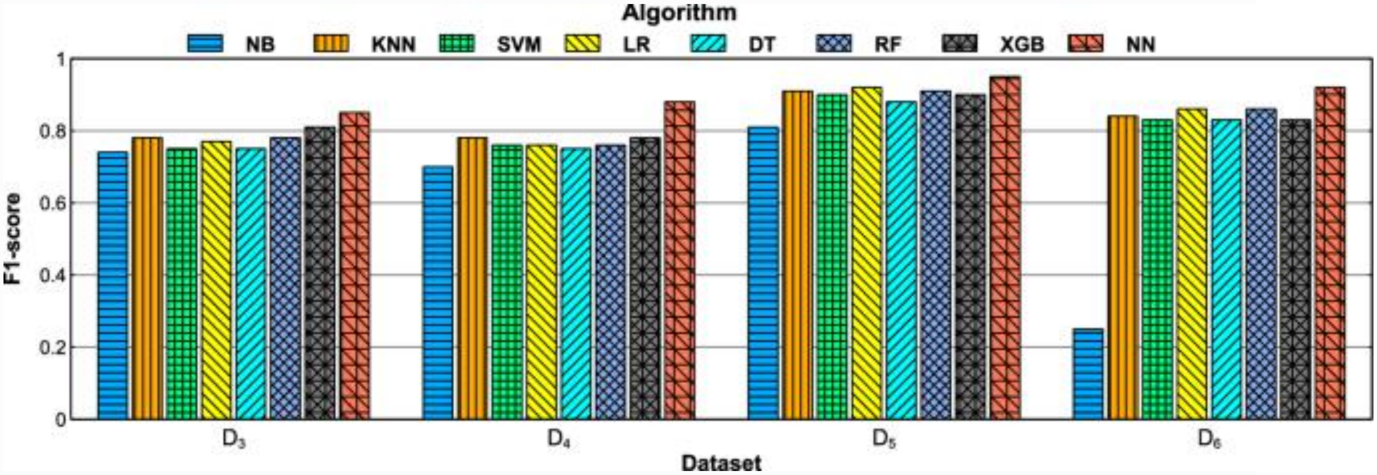
- Starting from the collected data, we created a manually classified dataset (D_1D1) composed of 5000 tweets, half *relevant* and half *not relevant*. Such data have been used to train different machine learning algorithms, which are Naïve Bayes (NB), K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Logistic Regression (LR), Decision Tree (DT), Random Forest (RF), XGBoost (XGB), and Neural Networks (NN). In particular, we used the implementations included in the scikit-learn library[Footnote4], together with Keras[Footnote5], TensorFlow[Footnote6] and Word2Vec [42] for creating neural networks.

- The obtained classification models take into account different features of tweets, such as length and presence of keywords, hashtags or bi-grams that are typically used to refer to disasters.
  Let $P=\{p_1,p_2,\ldots,p_n\}$P={p1,p2,…,pn} be a set of social media posts, where a generic post $p_i$pi is a social media content (e.g., a tweet) posted by a user after a catastrophic event *E*. Specifically, a generic post $p_i$pi includes:

- *user_id*, containing the identifier of the user who posted $p_i$pi;
- *timestamp*, indicating when (date and time) $p_i$pi was posted;
- *text*, containing a textual description of $p_i$pi;
- *tags*, containing the tags associated to $p_i$pi;
- *coordinates*, which consists of latitude and longitude of the place from where $p_i$pi was created (often this field is undefined);
- *profile_geo*, containing public location information provided by the user in its profile;
- *length*, indicating the length of the text of $p_i$pi;
- *numKeywords*, indicating the number of relevant keywords (e.g., earthquake, flooding, magnitude, lack of water, electrical problems) contained in the text of $p_i$pi;

- For the different algorithms, the classification models have been trained using dataset *D*1. Then, such models have been tested on five datasets [43], different from *D*1, which are related to different natural disasters (i.e., floods and earthquakes) that occurred in the period 2009–2019 (see Table 2). In such a way, the training and testing datasets are completely decoupled, which enables to evaluate how well the models are generalized to deal with new unseen data. It is worth noting that some datasets are unbalanced because the two classes, *relevant* and *not relevant*, are not equally represented. In order to correctly evaluate the classification models, the training datasets have been balanced before building For the different algorithms, the classification models have been trained using the models

- With all the datasets, the classification algorithms were able to separate relevant tweets from non-relevant ones with high accuracy. As an example, Table shows the results obtained by the different algorithms on the D_2D2 dataset (similar behaviors we obtain with the other datasets). The algorithm based on neural networks was the most accurate with an accuracy of 83%, followed by the algorithms XGBoost (81%) and Random Forest (80%). Figure reports the classification results obtained with the other four datasets (D_3D3, D_4D4, D_5D5, D_6D6), which assess the high accuracy obtained by neural networks in all four tests. For this reason. such a model has been used for classifying posts into *relevant* and *not relevant* with high accuracy.



Comparative analysis among several machine learning algorithms, evaluating the F1-score obtained by our approach for each dataset used in this work

Evaluation of the classification models made on the D2D2 testset

Algorithms	Acc	Prec	Rec	F1
Naïve Bayes	0.753	0.735	0.753	0.739
KNN	0.807	0.803	0.807	0.781
SVM	0.776	0.765	0.776	0.751
Logistic Regr.	0.790	0.773	0.790	0.766
Decision Tree	0.744	0.755	0.744	0.753
Random For.	0.795	0.794	0.790	0.783
XGBoost	0.815	0.812	0.815	0.809
Neural Net.	**0.830**	**0.826**	**0.864**	**0.845**