On the relation between solar flares and corona mass ejections, from a machine learning view

Ruizhu Chen
Department of Physics
Stanford University
rzchen@stanford.edu

Xin Zheng
Department of Electrical Engineering
Stanford University
xzheng3@stanford.edu

Abstract

We apply machine learning techniques to predict whether a solar flare is associated with any Corona Mass Ejections (CMEs) given the features of solar flares, including direct observables and derived quantities using satellite data from SDO and GOES. Naïve Bayes, logistic regression and support vector machine are explored, with positive cases upweight, feature selection by correlation and feature dimension degeneration by PCA. Promising results are achieved with generalization error < 0.2 and K-score > 0.6, on a small training set of 290 data points with class label confirmed. Efforts are also made on a 10 times larger train set with less confident class labels.

1 Introduction & Related work.

Solar flares and corona mass ejections (CMEs) are both energetic events with eruptions. Solar flares are explosions in solar atmosphere, while CMEs are eruptions that blow high-energy particles from the sun to the space, which have significant impact to human activities when faced towards earth. Some of the solar flares and CMEs are associated with each other, especially the strong ones. Although the mechanisms of both events and their relation have been heavily studied during the past decades (Kahler, 1992 review[1]), the relation between solar flares and CMEs is still unclear.

Most of the previous works are case-by-case studies on the physics process (Zhang et al., 2001,etc.[2]). It is indicated that flares and CMEs may be two type of phenomena due to the same process. Recently, two works draw large samples of flare-CME pairs and study the statistical relation on one or two characteristic properties (Yashiro, 2008[3]; Youssef, 2012[4]). It is found that flares with larger flux or longer duration tend to have associated CMEs.

In this project, we try to model the flare-CME association from machine learning view. We use Logistic Regression, Naïve Bayes, and Support Vector Machines (SVM), with input of properties of flares to predict whether the flare is CME associated.

There are not many studies using machine learning techniques in this field. On the same topic, Qahwaji(2007)[5] used Cascade Correlation Neural Networks (CCNN) and SVM with four features of flare: intensity, flare duration, and duration of decline and duration of growth to predict association with CME. The training class labels are determined automatically by event detection times. We think we can improve in two aspects: First, we use case-by-case confirmed association of flare-CME pairs as class label, at the cost of a smaller data size. The automatic determination of association by detection time turns out to be very noisy. In section 5 we build a better criteria using both the location and event starting time (also counting propagation time before detected), and even that is not robust enough compared to the case-by-case confirmed pairing. Second, we use 32 features including more physical quantities, from continues-time full-disk satellite data from SDO/HMI and SDO/AIA.

On similar topics, Al-Omari (2010)[6] used SVM to associate filaments and CMEs; Qahwaji(2008)[7] used AdaBoost to predict CMEs by filaments; Qu(2003)[8] used multi-layer perceptron (MLP), radial

basis function (RBF), SVM to predict solar flares; Bobra(2015)[9] used SVM to predict solar flares. Of all these work, Bobra(2015) is the first to use data from SDO/HMI, the first instrument to continuously map the full-disk photospheric vector magnetic field from space. We also include those features. SVM algorithm is used most in the existing literature, and many claim it to be working best. In this work we will also use SVM, together with Logistic Regression and Naïve Bayes.

2 Dataset and Features

2.1 Data Generation and Preprocess

We used the solar flare events catalog from GOES satellite and the corresponding CME associations from SOLARHAM catalog if available. An example is shown in Table 1. The class labels, 'YES'=1 and 'NO' =0, are confirmed by researchers by manually check in a case-by-case way. The features are generated in the following way:

- We used duration (end time – peak time) and peak flux as features directly.
- Info like time, AR Region (the index number of the active region where the flare originated), and location are not good features for the study, instead we used them as indexes to quote for measurements and properties of the same events in SDO/AIA and SDO/AIA databases.
- From SDO/AIA database we get filtered image (Figure 1) of the Sun at 9 different wavelengths at the peak time of a flare. For simplicity we use the mean value of each image as a feature.
- From SDO/HMI database we get 21 SHARP (Spaceweather HMI Active Region Patch) quantities or segments (Figure 2) and use the mean value within the active region as a feature.

Flare Class	Date	Start Time	End Time	Peak Time	AR Region	Location	CME
X1.2	1/7/14	18:04	18:58	18:32	1944	/	YES
M1.0	1/7/14	3:49	3:56	3:53	1946	N07E08	NO
M9.9	1/1/14	18:40	19:03	18:52	1936	S14W47	NO

Table 1: A example of solar flare catalog (GOES + SOLARHAM)

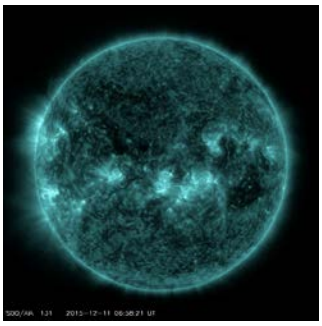

Figure 1 An image at 131Å from AIA database

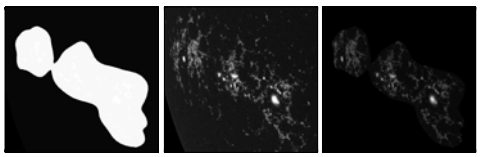

Figure 2 An example of SHARP segments. Left panel shows area of active region, used as a mask applying on middle panel, magnetic field. The average of masked field (right panel) is used as a feature.

2.2 Feature and data summary

Below is a summary of feature and class.

Column 1	Column 2	Column 3-11	Column 12-32	Class
Duration	Peak flux	Intensity of 9 wavelengths	22 Host active region field parameters from SHARP	Y={0,1} (as factor)

Table 2: summary of generated feature and class

We have a total of 290 data points, 25% of which are positive class. The relatively small data size is limited by the availability of flare-CME associations confirmed. 242 data points have all 32 features, 21% of which are positive, others only have the first 11 features.

We cleaned outliers and scale each feature so that the histogram is close to normal or uniform, or double-bell distribution. Additional preprocess methods are introduced in section 3 to improve machine learning performance under different circumstances.

3 Methods

3.1 Metrics

For examination, we use generalization error, recall, precession and k-score as the metrics. K-score is defined as

$$K_{score} = \frac{recall*precesion}{recall+precision} * 2. \quad (1)$$

3.2 Algorithm

3.2.1 Logistic Regression (LR)

In LR, decision boundary $\theta^T x$=0 is found by maximizing the likelihood on training set, $L(\theta) = \prod_{i=1}^{m} p(y = y_i \,|x = x_i;\, \theta)$, where we assume a model with probability

$$p(y = 1\,|x;\theta) = h_\theta = \frac{1}{1+\exp(-\theta^T x)}\ , \qquad p(y = 0\,|x;\theta) = 1 - h_\theta \quad (2)$$

We use leave-one-out cross validation to achieve metrics values.

3.2.2 Naïve Bayes (NB)

NB is to predict class through comparing $p(y = 1\,|x)$ and $p(y = 0|x)$, where we assume the features are conditionally independent so that

$$p(x|y) = \prod_{i-1}^{i=n} p(x_i|y), \quad \text{and} \quad p(y|x) = \frac{p(x|y)p(y)}{p(x)} = \frac{\prod_{i=1}^{i=n} p(x_i|y)\cdot p(y)}{\sum_y \prod_{i=1}^{i=n} p(x_i|y)\cdot p(y)} \quad (3)$$

The performance of NB depends on the difference between posterior distribution given y=1 and y=0. In our case, it's hard to model some of the feature distributions using continuous models. So for simplicity we then discretize each feature uniformly into the same number of bins. We choose the number of bins as tuning parameter from 5 to 100, and select the optimal number of bins by evaluating performance through leave-one-out cross validation.

3.2.3 SVM

Support vector machine finds a separating hyperplane that maximizes the margin between the separating hyperplane and training data. We apply both linear- and radial-kernel SVM with regularization, using a series of regularization parameter C and search for the optimal value (figure 2). The optimal C is decided by best generalization error and K-score, estimated by K-fold cross-validation (k=10).

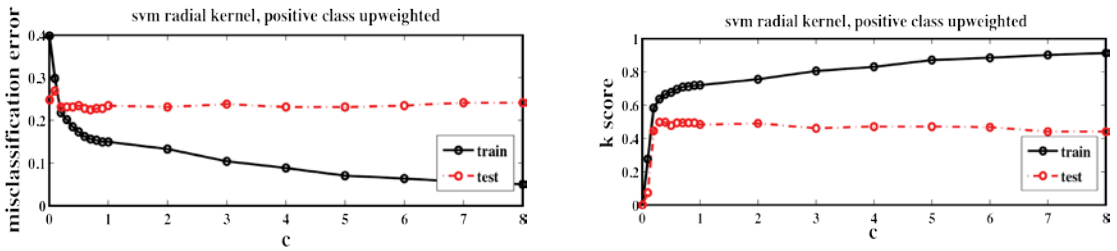

Figure 3 An example of finding optimal parameter through performance. Train set is significantly overfitting as C increases. The optimal C is chosen at 0.2 where error is smallest and K-score is largest.

3.3 Upweight

To solve the unbalanced class problem (the ratio of positive class ~25%), we upweight the positive class in training sets by a factor of 2.

3.4 Feature Process

- We first use the 11 direct-observable features for training. This is of interest because we could have real-time prediction without the hassles of inversion problems to get other features.
- The whole 32 features are then used for training.
- Feature selection (FS) methods are used to find the most correlated features: calculate the correlation between y and each feature and eliminate the least correlated feature one-by-one until the best performance.
- PCA is applied to decrease redundancy in feature dimension. The final feature size is around 10.

4. Results

	error	precision	recall	k-score	error	precision	recall	k-score
	Logistic Regression				Naïve Bayes			
11× 290	0.21	0.61	0.38	0.47	0.21	0.65	0.31	0.42
+ UW	0.21	0.58	0.50	0.54	0.26	0.49	0.86	0.62
32×242	0.18	0.58	0.51	0.54	0.29	0.38	0.61	0.47
+UW	0.16	0.63	0.59	0.61	0.22	0.48	0.80	0.60
+UW + FS	0.16	0.63	0.59	0.61	0.24	0.46	0.88	0.60
+UW +FS +PCA	0.18	0.55	0.57	0.56	0.19	0.54	0.84	0.66

Table 4: Logistic Regression and naïve Bayes main results (11×290 stand for data set with 11 features and 290 samples; UW stands for upweight; FS stand for feature selection)

	error	precision	recall	k-score	error	precision	recall	k-score
	SVM(linear)				SVM(radial)			
11×290	0.21	0.71	0.32	0.44	0.22	0.70	0.40	0.51
+ UW	0.23	0.57	0.59	0.58	0.26	0.52	0.68	0.59
32×242	0.19	0.60	0.51	0.55	0.17	0.73	0.49	0.59
+UW	0.19	0.58	0.61	0.60	0.18	0.67	0.46	0.55
+UW + FS	0.19	0.63	0.64	0.64	0.20	0.58	0.58	0.58
+UW +FS +PCA	0.17	0.64	0.58	0.61	0.13	0.88	0.42	0.57

Table 5: SVM main results

Table 4 and Table 5 summarize the machine learning performance.

Upweight improves performance significantly for all models. This is not surprising considering more balanced class will give more balanced description of both class during training and thus give better predictions.

Feature selection works best for SVM when removing 4 insignificant features, but it doesn't improve performance for LR and NB. There is a tradeoff between removing unrelated information and removing useful information. When we keep the features with large correlation to class labels, we assume the features and classes are linearly related. However, real system based on physics law may be non-linear, and thus our feature selection may also remove useful information.

PCA improves performance for NB significantly, while doesn't work for LR and NB. PCA removes redundancy in feature space. For Naïve Bayes, this works because PCA forms orthogonal basis for data and thus help to satisfy the conditional independence assumption. For the other methods, some useful information may be lost after PCA, thus cause slightly worse performance. However PCA will decrease the computation demand significantly.

Best performance is achieved by Naïve Bayes after upweight, feature selection and PCA. The highest k-score is 0.66.

5. Generate a larger data set and train.

The results on the small set are very promising. We then decide to try generating a larger set of a typical machine learning size. There are far more solar flare and CME events recorded by GOES and LASCO, but to determine the association of solar flare and CME pairs is hard. A robust but too expensive way is by manually check the processes of events, like how it's done for the small set. Instead, we try an empirical paring criterion that requires spatial and temporal coincidence, the same method used in Yashiro, 2008[3]; Youssef, 2012[4].

The method is not robust because it cannot correctly classify the given pairs in the 290 data points we have. The limitation comes from: 1. The start time of CME is derived by a simplified model using CME detection time, average CME velocity, flare center location and geometry, which doesn't count in flare size, shape, and CME velocity change. 2. The universal location and time window used do not work for all sizes of flare and CME. However this is the best we found in literature.

We still generate a set of 2900 data points. The sampling of this large set is more general, including weaker events, while the small sets are sampled unevenly favoring strong or interesting events. The overall positive rate is about 14%, and positive rate for flare M class and above (same as small set) is about 25%, comparable to the small set. The positive class is then upweighted by a factor of 4.

We used either 11 direct observables or 14 downsized features by feature selection and PCA. Using all 32 features are not affordable but it's proved in Sector 4 that the feature downsizing only cost a little bit performance to dramatically save the computation demand.

The optimal results shown below in Table 6. The results are very poor compared to the small set. Results are limited by the robustness of train set class labels. We found the probability distribution of features on both class (Figure 4) are so similar that no feature can significantly distinguish the two classes.

method	error	precision	recall	k-score
11× 290	0.18	0.21	0.19	0.20
PCA, UW	0.27	0.23	0.23	0.23

Table 6: Summary of SVM results using large data set.

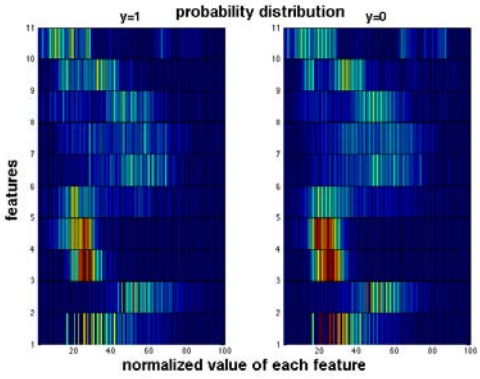

Figure 4: probability distributions of features on both class.

6 Conclusion and Future Work

We applied algorithms of LR, NB and SVM on a small data set of 290 data points and 32 features, with upweight and feature manipulations, and got results with generalization error <0.2 and K-score >0.6 for all algorithms. Our best result is achieved using NB, with generalization error =0.19 and K-score =0.66.

These results are promising given the small size of training data. While we also generate and try on a 10 times larger set, the empirical class labels are not robust enough for a good study. However, most existing work with large data sets used similar or simpler empirical criteria for pairing solar flares and CMEs, and here we learn that the true association should be more complicated than this apparent temporal and special coincidence. When more confirmed pairs of flare and CMEs are published and accumulate to a considerable amount, the model will be improved.

References

[1] Kahler, S. W. "Solar flares and coronal mass ejections." Annual Review of Astronomy and Astrophysics 30 (1992): 113-141.

[2] Zhang, J., et al. "On the temporal relationship between coronal mass ejections and flares." The Astrophysical Journal 559.1 (2001): 452.

[3] Yashiro, Seiji, and Nat Gopalswamy. "Statistical relationship between solar flares and coronal mass ejections." Proceedings of the International Astronomical Union 4.S257 (2008): 233-243.

[4] Youssef, M. "On the relation between the CMEs and the solar flares." NRIAG Journal of Astronomy and Geophysics 1.2 (2012): 172-178.

[5] Qahwaji, R., et al. "Automated Prediction of CMEs Using Machine Learning of CME–Flare Associations." Solar Physics 248.2 (2008): 471-483.

[6]Al-Omari, M., et al. "Machine leaning-based investigation of the associations between cmes and filaments." Solar Physics 262.2 (2010): 511-539.

[7] Qahwaji, R., et al. "Using the real, gentle and modest AdaBoost learning algorithms to investigate the computerised associations between coronal mass ejections and filaments." Communications, Computers and Applications, 2008. MIC-CCA 2008. Mosharaka International Conference on. IEEE, 2008.

[8]Qu, Ming, et al. "Automatic solar flare detection using MLP, RBF, and SVM."Solar Physics 217.1 (2003): 157-172.

[9]Bobra, Monica G., and Sebastien Couvidat. "Solar Flare Prediction Using SDO/HMI Vector Magnetic Field Data with a Machine-Learning Algorithm."The Astrophysical Journal 798.2 (2015): 135.