2018 IEEE 15th International Conference on e-Business Engineering (ICEBE)

Prediction of Crowdfunding Project Success with Deep Learning

Pi-Fen Yu, Fu-Ming Huang*, Chuan Yang, Yu-Hsin Liu, Zi-Yi Li, Cheng-Hung Tsai
School of Big Data Management
Soochow University
Taipei, Taiwan
watasilu@gmail.com, fmhuang@gm.scu.edu.tw, a1205ab@gmail.com, 04170229@gm.scu.edu.tw, caco2114@gmail.com, sssh10130338@gmail.com

***Abstract*— Over the past century there has been a dramatic increase in crowdfunding activity, which offers an alternative for both creators and backers to sell products and invest in creative businesses respectively. However, empirical analysis shows that only one-third of crowdfunding campaigns could meet their fundraising goal. The aim of this paper is to develop a model that predicts the success of crowdfunding project with deep learning. The datasets are retrospectively collected from Kaggle and contain historical records of Kickstarter campaigns. The model could provide insights in pre-lunching stage and in early stage of fundraising. The proposed MLP model can provide accountable results when applied to different crowdfunding platforms that have not been addressed before. Comprehensive experiments are conducted and a variety of classification algorithms have been tested to support this prediction engine and they concluded that the MLP model has the best outcome with the highest degree of confidence.**

Keywords*— Crowdfunding; Deep learning; Multi-layer perceptron; Prediction; Open data

I. Introduction

A. Background

Crowdfunding is flourishing in the Internet Age and crowdfunding platforms have become a booming market as a new fundraising channel in the digital world. Belleflamme et al. (2010) described crowdfunding as an open call, particularly through the internet, to get financial support in forms of donation or exchanging rewards (Belleflamme, P., Lambert, T., & Schwienbacher, A., 2010). Yang et al. (2008) also pointed out that crowdfunding focuses on fundraising from the public by using internet-based solution and, encourage contribution from backers by offering adequate reward options (Yang, J., Adamic, L. A., & Ackerman, M. S., 2008).

B. Brief History

The origin of crowdfunding dates back to several centuries ago. The activity aimed to collect a small amount of money from each individual of massive population for a variety purposes. One famous examples was fundraising for the Statue of Liberty. In 1885, the US government had no budget for pedestal construction of the statue. A renowned publisher, Joseph Pulitzer, launched a fundraising campaign in his newspaper. He published a series of articles and proposed to print the names of each contributor on the front page, no matter the amounts. The campaign was so popular that it raised over $100,000 dollars from more than 160,000 people within six months and contributors are made by children, businessmen, street cleaners and politicians. As the result, the campaign helped resolve the financial problem the new landmark faced (BBC NEWS, 2013).

The first successful on-line crowdfunding activity ever recorded occurred in 1997 where the society only had minimal access to the internet. The keyboardist of a British rock band, Marillion, raised £39,000 (or $60,000 dollars) for their US tour through internet as they were short of money. They sent emails to their fans asking for financial support to the tour. They received positive replies and, their tour was successfully funded. A few years later, ArtistShare setup the first crowdfunding platform which focused on “fan-funding” music projects on the website and made a great achievement in 2005 as the platform helped funded the first Grammy-Award-winning album that was not released by a mass retailer. Being inspired by ArtistShare’s success, more crowdfunding platforms were launched worldwide, including Indiegogo in 2008 and Kickstarter in 2009 (Freedman, D. M., & Nutting, M. R., 2015) .

With the rise of sharing economy, as more creators rely on getting fundraising via the internet, more crowdfunding platforms are born to cater to this niche market. In the past decade, the number of crowdfunding platforms increased significantly from double digits to four digits. The fundraising volume reached $34 billion in 2015 from Massolution Crowdfunding Industry Report. According to a report issued by CrowdFundBeat, the number of crowdfunding platforms grew from only 53 in 2009 and to over 1,000 in 2015 (crowdfundbeat.com; Report: Global Crowdfunding Market 2016-2020) .

C. Roles in Crowdfunding

Different from the traditional fundraising, crowdfunding platform adopts modern e-commerce practices and provides a deck where creators present their products and ideas to the general public on the internet. The general public then back the items or activities of their liking while surfing the internet. Tomczak (2013) defined crowdfunding as an action of acquiring third-party financing from the general public via a web-based platform (Tomczak, A., & Brem, A., 2013) . Since about a decade ago, crowdfunding platforms have been fostering and developing rapidly as a result of general public’ acceptance to make contribution to the funding activities (Aaker, J. L., & Akutsu, S., 2009).

There are three roles involved in crowdfunding projects. The first one is “intermediary”, also known as “platform”, acts as the matchmaker between creators and backers. The second role is “fundraisers” or “creators”, who raise money over the platforms. The last role, “investors” or “backers”, contributes money to projects of their choice (Tomczak, A., & Brem, A., 2013). Fundraisers have to describe the projects with detail information on the platform (Belleflamme, P., Lambert, T., & Schwienbacher, A., 2010). In terms of business model, platforms serve as service providers that charge service fee based on predefined rules. Tomczak (2013)

978-1-5386-7992-0/18/$31.00 ©2018 IEEE
DOI 10.1109/ICEBE.2018.00012

gave two examples: Kickstarter charges 5% of total money raised and Indiagogo charge 4% of funds raised from the successfully projects and 9% from the failed projects (Tomczak, A., & Brem, A., 2013).

D. Types of Crowdfunding

In general, crowdfunding can be categorized into four types: donation-based, reward-based, lending-based and equity-based. In donation-based crowdfunding, people donate money and expect nothing in return. In reward-based crowdfunding, creators offer non-financial rewards to backers for their contribution. Instead of receiving non-financial rewards, backers in lending-based category could claim their money back, sometimes with interest. For equity-based projects, unlike lending- and rewards-based, creators provide returns in forms of equity, share of revenue, or share of profits (Bannerman, 2013). Giudici (2012) explained the concepts in a similar fashion. Backers make monetary donations without receiving compensation in donation-based crowdfunding. In lending-based, backers are paid back with interests under certain conditions. For reward-based crowdfunding, creators provide non-monetary benefits in the forms of products or services. As for equity-based, creators reward backers with shares of residual income (Giudici, G., Nava, R., Rossi Lamastra, C., & Verecondo, C. , 2012).

There are two types of financing models underneath each crowdfunding platform: “All-or-Nothing” and “Keep-it-All”. In the “All-or-Nothing” model, creators receive money only if the project has reached the stated fundraising threshold before its deadline and receive nothing for failed projects. In contrast, the “Keep-it-All” model gives creators the entire amount raised even for failed projects (Cumming, D., Leboeuf, G., & Schwienbacher, A, 2014) . The second largest crowdfunding partner, Indiegogo, offers both models. The research suggests that the crowd is more willing to put money in “All-or-Nothing” model.

E. Motivation and Purpose

Crowdfunding offers an alternative way for both creators to sell products and backers to invest in creative businesses. By launching projects on platforms, small business could gather information on market sentiment before releasing their products to the market by assessing whether their consumers are willing to receive pre-orders (Belleflamme, P., Lambert, T., & Schwienbacher, A., 2010). On Kickstarter, one of the world largest crowdfunding site, the number of projects had grown 85x from 2009 to 2015 and the total amount raised had increased from 1.6 million to more than 600 million US dollars in the same period.

Leveraging the crowdfunding platforms, creators are able to provide information and communicate with backers constantly. However, only one-third of campaigns could reach the goal. Many projects could not meet their targets within the prescribed period while the number of proposals are constantly increasing. Based on Kickstarter stats-web, less than 40 percent of proposals succeed on average (icopartners.com; Kickstarter in 2017 – Year in review, 2018). Under the rule of “all-or-nothing”, many of projects failed regrettably by a small amount on the last day. It is therefore important to both creators and backers to know whether the project could be funded successfully well ahead of time.

This paper aims to adopt Deep Learning technique and develop a model that provides a faster, more accurate and more resource-efficient method. Creators could take timely action, for example, to increase their visibility in case of low possibility of meeting targets, while backers may look into projects with possible extensions to their fundraising period due to their high success rate (Etter, V., Grossglauser, M., & Thiran, P., 2013). Denton et al. (1990) evaluated the performance of a neural network as a classifier and found that its performance is comparable to the best among other methods under a wide variety of modeling assumptions (Denton, J. W., Hung, M. S., & Osyk, B. A., 1990) . Neural Network is suitable for classification and more appropriate for solving the nonlinear problem. A tool with the ability to evaluate the success rate of a given project at the early stage could help creators adjust their marketing strategies and help backers achieve their investment return.

II. LITERATURE REVIEW

There are two hot topics in regards to crowdfunding research, where one focuses on analyzing the factors that contribute to the success of a project while the other on predicting success of fundraising campaign. There has been a number of studies carried out on reward-based crowdfunding platform with the focus on its features and prediction of a successful project.

By analyzing projects and users on Kickstarter, Chung et al. (2015) made several experiments and proposed four feature sets: project features, user features, temporal features and twitter features. Furthermore, the analysis identified two peaks during a project’s lifecycle: in the beginning and the end of a project duration (Chung, J., & Lee, K., 2015). Greenberg et al. collected 13,000 projects from Kickstarter and extracted 13 features to develop prediction engines. After building several prediction models and compared with different machine learning algorithms (including J48 Trees, Logistic Model Trees, Random Forests, SVM and AdaBoost) , researchers found decision tree algorithms performed the best with roughly 68% accuracy while SVM only yielded an average accuracy as baseline (Greenberg, M. D., Pardo, B., Hariharan, K., & Gerber, E., 2013). The research focus on dominant crowdfunding platforms and develop a method to predict the success of Kickstarter campaigns with two predictors: money pledged in time-series and social attributes. Four hours after launch of a campaign, combining both predictors helps improve the prediction accuracy by 4%, even higher than the 76% accuracy in SVM model (Etter, V., Grossglauser, M., & Thiran, P., 2013). Kamath et al. (2016) concluded that Neural Network perform better (up to 94% accuracy) in predicting success of Kickstarter campaigns after building several supervised learning models which include Naïve Bayes, Neural Network, Random Forest and Decision Tree (Kamath, R. S., & Kamat, R. K., 2016). On the linguistic aspect, Sawhney (2016) demonstrated the 92% accuracy in predicting Kickstarter campaigns by linear-kernel SVM (Sawhney, K., Tran, C., & Tuason, R., 2016). Combining both classification and censored regression as of survival analysis based, a new approach helps build a robust prediction model which perform in the best rate of 0.8029. The approach has been shown to deliver a better outcome by employing both successful and failed projects while using both temporal and social network features. Furthermore, improvements can be made on prediction to achieve the best accuracy of 0.9030, if the model includes information on progress a project makes within first 3 days after launching (Li, Y., Rakesh, V., & Reddy, C. K., 2016). Chen et al. (2013) trained a SVM model on Kickstarter projects and obtained approximately 90% accuracy with Android application and Chrome extension employed (Chen, K., Jones, B., Kim, I., & Schlamp, B., 2013). Rakesh et al. (2016) provided in-depth analysis on the work of probabilistic recommendation model which recommends projects to Kickstarter users by incorporating the dynamic-status of ongoing projects. (Rakesh, V., Lee, W. C., & Reddy, C. K., 2016). After constructing prediction models using, GLM, Random Forest, Gradient Boosting and XgBoost, Lai et al.(2017) demonstrated an accuracy up to 96.8% by XgBoost (Lai, C. Y., Lo, P. C., & Hwang, S. Y. , 2017).

The literatures presented above shows recent trends of Machine Learning applying in predicting campaign success, mostly in reward-based platform. Given Kamath et al. (2016) has proven a good performance with Neural Network, it may be worth considering to include the approach for the proposed MLP model.

III. METHODOLOGY

A. Data Collection

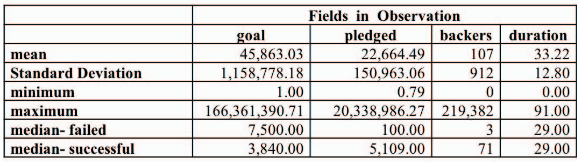
This research is conducted with historical records of Kickstarter since Kickstarter is not only the market-leading in crowdfunding platforms which make evaluation result reliable and trustworthy, but also contains sufficient amount of open data to utilize in deep learning computation. The dataset was retrospectively collected from Kaggle where sharing an extensive data for competition and data science as well. It consists of historical projects from Kickstarter (Kaggle Dataset, 2018) and contains 378,611 records in total between May 2009 to March 2018.

This dataset also included project duration and string length of project name. Successful campaigns were recorded as 133,851, around 36% of total campaigns while numbers of failed were 197,611, around 53% of total campaigns. The rest 12% of campaigns were counted as others which including live, suspended and canceled. The first glance at the highest amount of pledge goes to Pebble (Pebble Time - Awesome Smartwatch, No Compromises) which achieved significant success as stated in introduction. The lowest amount of successful pledged is only 1 Canadian dollar contributed by 1 backer. The average project duration were 33.2 days while the shortest was 0 which means project achieved the goal on the date it was launched. Average number of backers in successful projects were 264 while only 16 in failed projects. There are fifteen columns in this dataset to accommodate with basic information of projects and supplemental information of computation results. The general information such as project ID, project name, project category, subcategory, project goal, pledged amount, location of project taking off, launching date of projects, and status of projects. Kickstarter divide projects into 15 main categories and 52 subcategories. The 15 categories are: Art, Comics, Crafts, Dance, Design, Fashion, Film & Video, Food, Games, Journalism, Music, Photography, Publishing, Technology, and Theater (Kickstarter-Our Rules, 2018). TABLE I describes detail data field information, including field name, data type and short descriptions. Another observation in descriptive statistics are given in TABLE II.

TABLE I. DATA TYPE AND DESCRIPTION OF EACH FIELD

Field	Data Type	Description
project id	int64	Project ID
name	object	Project Name
main_category	object	Project Category; Kickstarter list out 15 categories for projects.
category	object	Project Subcategory; Kickstarter list out 52 sub-categories for projects.
currency	object	Project currency
deadline	object	Project deadline
goal	float64	Project goal in USD
launched	object	Project lunching date
pledged	float64	Pledged amount
state	object	Project Status; five status of historical projects: failed; successful; canceled; live; suspended
backers	int64	Number of Backers
country	object	Location of project taking off
usd_pledged_real	float64	Amount of pledge in USD
usd_goal_real	float64	Amount of Goal in USD

TABLE II. DESCRIPTIVE STATISTICS OF DATA FIELDS



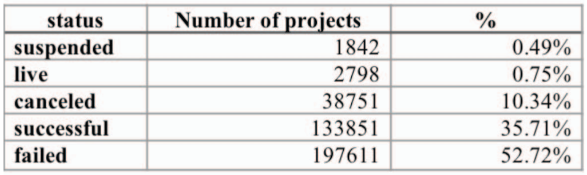
	Fields in Observation			
	goal	pledged	backers	duration
mean	45,863.03	22,664.49	107	33.22
Standard Deviation	1,158,778.18	150,963.06	912	12.80
minimum	1.00	0.79	0	0.00
maximum	166,361,390.71	20,338,986.27	219,382	91.00
median- failed	7,500.00	100.00	3	29.00
median- successful	3,840.00	5,109.00	71	29.00

B. Exploratory Data Analysis

Exploratory Data Analysis (EDA) is an approach to gain more insight of data by employing different techniques, mostly in creating graphical and numerical summaries.

The data pre-processing task aims to identity missing values and, noisy data (Kamath, R. S., & Kamat, R. K., 2016). On the first observation, there are 3,801 records which could be removed as data leakage. Another finding is that incorrect data type for "deadline" and "launched" fields were incorrectly set as "object" which is not considered as time series data. Therefore, converted data type to "datetime" for the two fields. In order to present the project duration and length of project name, two extra columns were added as workaround. Noisy data, such as extreme project durations outliers (>10,000 days) cause by incorrect launch date were omitted as well. TABLE III display number of projects in different statuses after data cleaning.

TABLE III. TOTAL PROJECS BY STATUS



status	Number of projects	%
suspended	1842	0.49%
live	2798	0.75%
canceled	38751	10.34%
successful	133851	35.71%
failed	197611	52.72%

Presenting EDA with visualization helps readers measure and compare variables at the same time while Python package ("plotly" and "iplot") provides effective graphing capability.

Below charts provides the distribution of different projects by category. The most popular is "Film & Video", followed by "Music" and "Publishing". Fig. 1 presents the details sorted by number of projects. Fig. 2 displays percentages of successful projects versus failed projects in each category.

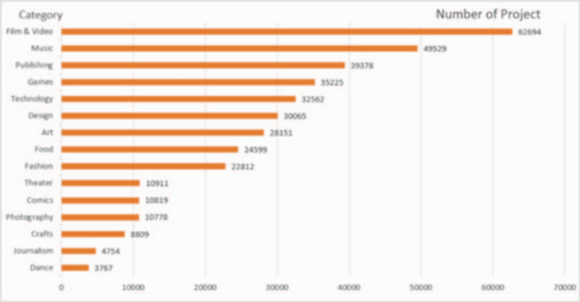


Fig. 1. All projects count by category

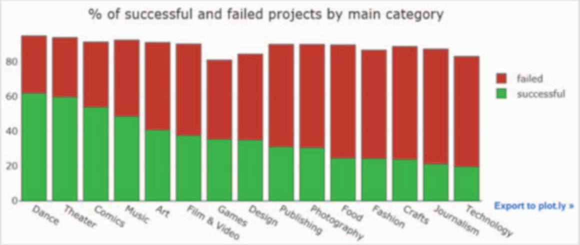


Fig. 2. Successful projects versus Failed projects by category

Comparing the number of projects and amount pledged in each category, Fig. 3 shows "Film & Video" and "Music" are top two of both and different ranking in rest of categories. Projects under Technology were funded considerably in spite of being low quantity. People appeared more inclined to invest in Games, then new products, and technology.

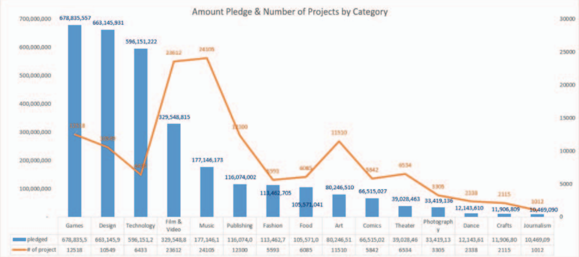


Fig. 3. Comparing number of projects and amount pledged by category

From Fig. 4, nearly 80% projects were raise in United States, 9% in United Kingdom.

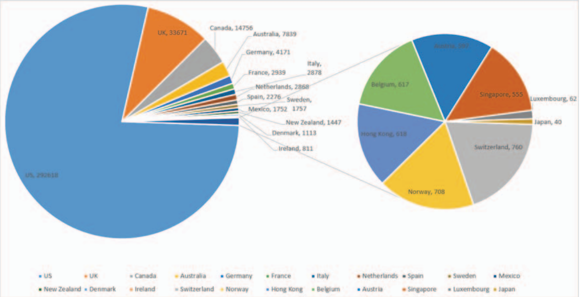


Fig. 4. Numbers of projects by Country

Most of creators set their target duration to be within one month, shown as Fig. 5.

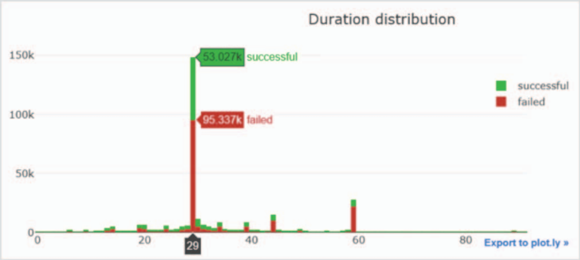


Fig. 5. Projects distribution by Duration

Pearson correlation is one way to show if a feature is correlated with the target variable. To do this, values of project status have to be converted from string to numeric: 1 represent for successful while 0 represent for failure. Fig. 6 presents correlations between variables. The results show that the amount pledged is moderately correlated with the number of backers, and goal amount is correlated with success status . The rest of features reflect no correlation in between.

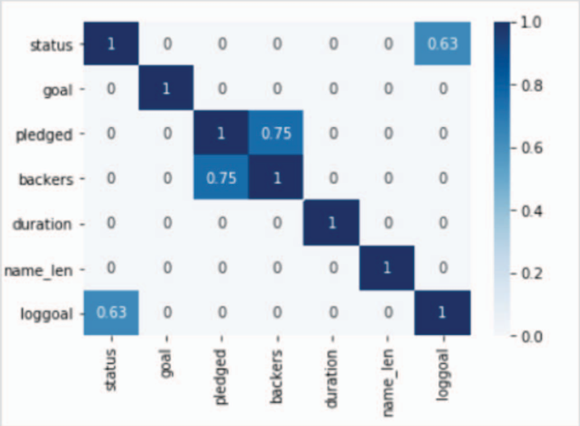


Fig. 6. Pearson correlation in features

Comparing the distribution of successful projects against failed projects with log-transform, Fig. 7, 8, 9 demonstrate the positive trend in different features. In contrast to failed projects, successful projects tend to be at an advantage of numbers in backers and pledged. Project goal amount and duration in Figure 9 shows the higher amount of goal is set at, the more likely a project fails.

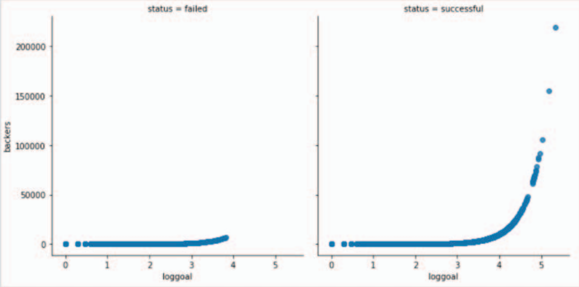


Fig. 7. Backers Distribution on a Logarithmic Scale

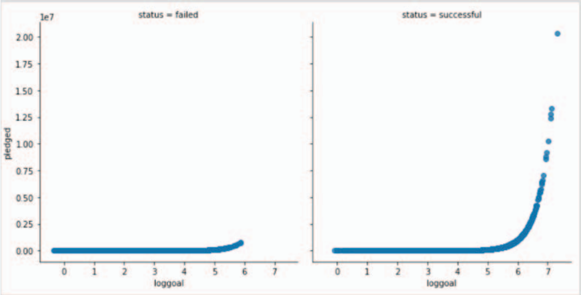


Fig. 8. Pledged Distribution on a Logarithmic Scale

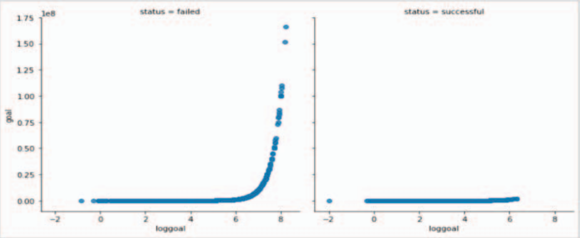


Fig. 9. Goal Distribution on a Logarithmic Scale

Regarding project duration and the length of project name, in Fig. 10 and 11, there is not much difference made in successful and failed projects.

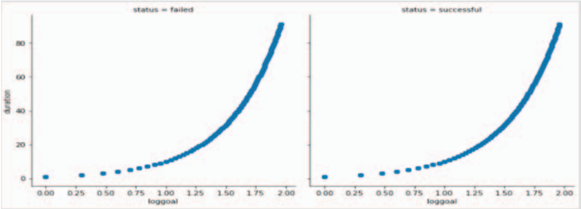


Fig. 10. Duration Distribution on a Logarithmic Scale

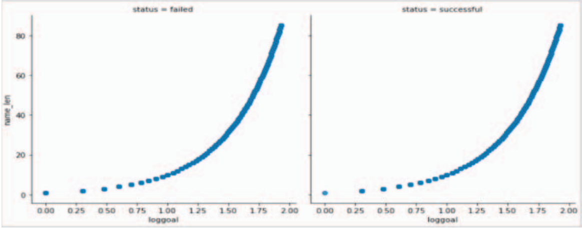


Fig. 11. Length of Project Name Distribution on a Logarithmic Scale

C. Model Construction

After examining and exploring most of the features, next step is to construct a prediction model. MLP is a mathematical model that includes batches of neurons connected in different layers, which work like neural networks human utilizes to compute certain information.

There are 331,462 records under statuses “successful” and “failed”. Based on data analysis already performed, all numeric columns will be kept except “project id” while dropping most string fields except “category” and “status”. To simulate the initial stage of projects, “amount pledged” is removed. In order to apply the Sequential model in Keras, categorical values need be mapped to integer values by using one hot encoders. A one hot encoding is the process of presenting categorical variables into binary vectors 0 and 1. The experimental dataset includes 1 class label and 32 features. The allocation of training set and testing is set at 80/20 ratio with a random seed.

Multi-Layer Perceptron, is one of the most commonly used Neural Network architecture to solve non-linear problems. MLP is a fully connect feedforward network consisting of a input layer, hidden layers and an output layer. The number of attributes in the dataset is set as the number of neurons in the input layer. Neurons in the output layer is the number of target classes. Embedded one or more non-liner layers as hidden layer between input layer and output layer, MLP can learn non-liner function (Hemalatha, K., & Rani, K. U., 2017) .

“The most significant elements of MLP are the connection weights and biases. The output of each node is calculated in two steps [6]. In first step, the weighted summation of the input is calculated using equation:

$$\mathrm{S}_j = \sum_{i-1}^{n} W_{ij}jij\text{-}I_j + \beta_j$$

where I_j is the input variable, W_{ij} is the connection weight between I_j and hidden neuron j, β_j is bias. ” (Hemalatha, K., & Rani, K. U., 2017)

Fig. 12 illustrates a multilayer perceptron with four layers and each layer is connected to the last one. The leftmost is the source dataset which is converted into two arrays, features and target value. The dataset used in this research, contains 32 features and one target (class label). These 32 features represent a set of neurons in Input Layer that are fully connected to neurons in the hidden layer. Each neuron in the hidden layer-1 transforms values with a weighted linear summation and feedforward to hidden layer-2, also fully connect. The output layer receives the values from the hidden layer-2 and transforms them into output value.

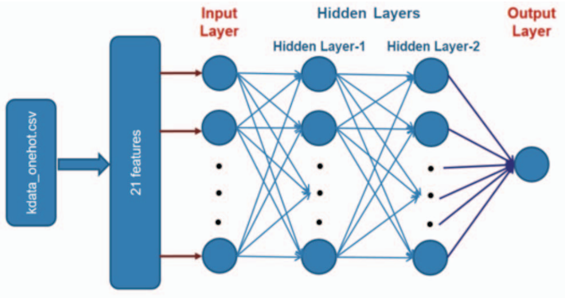


Fig. 12. Multilayer Perceptron Model

By utilizing a Python package Keras under TensorFlow environment, MLP model can be easily constructed without extensive programing effort. TensorFlow is an interface to express, implement and execute machine learning algorithms (Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., ... & Ghemawat, S., 2015). Keras provides a high-level neural networks API, written in Python and support multiple backend engines including: TensorFlow, CNTK, and Theano. Tensorflow and Theano are the most-used numerical platforms in Python to build Deep Learning algorithms. Keras uses TensorFlow by default as its tensor manipulation library (Keras Documentation).

Input layer is the first layer in a MLP model. It is fully connected to the first hidden layer, and defines the input features to 32 while 100 neurons in hidden layer apply activation function as “relu”. The second hidden layer consists of 60 neurons with the same activation function as “relu”. The last layer, output layer, receives values from the second hidden layer and transforms them into output values to represent the probability of success. Activation function is used to compute the predicted output of each neuron in each layer by using inputs, weights and bias. The activation function is set to “sigmoid” in the output layer since its range of 0 to 1 is fulfilled with the probability of success (Sharma, 2017).

After defining layers, the MLP model needs to be compiled with numbers of parameters: the optimization algorithm, the loss function and metrics. Optimization is the process of achieving the best outcome in a given operation (Hemalatha, K., & Rani, K. U., 2017), the most commonly used optimization algorithms are SGD and Adam. Keras supports a suite of optimization algorithms, including SGD, RMSprop, Adam, Adagrad, Adadelta and so forth (Keras Documentation). Adam is chosen in this model as it works well in practice and outperforms other Adaptive techniques (Walia, 2017). Binary-crossentropy is used as loss function because it deals with binary classification, and accuracy as the evaluation metric. The network is trained using the backpropagation algorithm which requires the network be trained for a specified number of epochs with the training dataset. To learn and make better predictions, a number of epochs (training cycles) are executed until an acceptable range of error is achieved. Each epoch can be partitioned into groups of input-output pattern pairs called batches (Hemalatha, K., & Rani, K. U., 2017).

The model is compiled and trained for 100 epochs with a batch size of 30, optimized using ADAM as optimization algorithm, binary-crossentropy as loss function, and accuracy as the evaluation metric. After training is completed, a list of evaluation metrics provide the training history of accuracy and loss. By composing a plotting function, the graph in Fig. 13 presents the vibration status during the training.

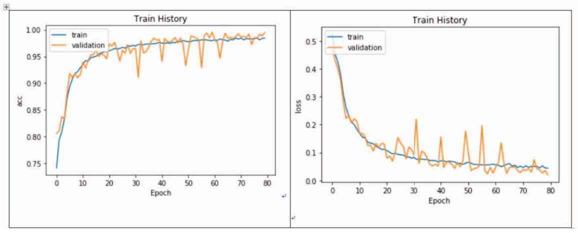


Fig. 13. Vibration status of training history

IV. EXPERIMENTS AND RESULTS

A. Parameters Tuning

The first experiment is to seek the best parameters in MLP by setting different numbers of layers, neurons, adopting available optimizers, and evaluating the corresponding performances. By examining the principle of algorithm, some parameters are already in good shape hence no change is required. Activation functions are used to determine the output of neural network in the formats of “yes” or “no” then resulting values are mapped depending on their respective function. For the hidden layer, activation function is set as ReLU (Rectified Linear Unit). ReLU has ranged from 0 to infinity and it decreases effectivity of negative values since all the negative values become zero.

The following parameters have been evaluated in the experiments: (1)Number of hidden layers, (2)Number of neurons in the hidden layers, (3)Number of batches, (4)Types of optimizer. Firstly, training and testing combinations of different number of hidden layers with different numbers of neurons in each. Given the input features are not too complex, comparison between two hidden layers and three hidden layers with various number of neurons is conducted. As the results, combination of consisting two hidden layers with 180 and 80 neurons separately performs the best. The model has a combination of parameters (hidden-1=180, hidden-2=80, activation in hidden-1=Relu, activation in hidden-2=Relu, activation in output layer=sigmoid) and ready for testing the loss function. Since the model gives categories of either success or fail, “binary_crossentropy” performs better than “mean_squared_error”.

The model requires an “optimizer” to compile. Keras provides several optimizer, such as, SGD (Stochastic gradient descent), RMSprop, Adagrad, Adadelta, Adam; Adamax, and Nadam. Adam performs the best. Kingma et al (2015) stated that Adam is an algorithm for first-order gradient-based optimization of stochastic objective function which is straightforward to implement, is computationally efficient (Kingma, D. P., Ba, J. , 2015).

Comprehensive experiments are conducted in order to explore the best combinations. TABLE IV shows the fine-tuned parameters and the testing history are presented in TABLE V.

TABLE IV. COMBINATION OF PARAMETERS

number of hidden layers	2
number of neurons in hidden-1	180
activation in hidden-1	Relu
number of neurons in hidden-2	80
activation in hidden-2	Relu
batch_size	180
loss_function	binary_crossentropy
optimizer	ADAM

TABLE V. HISTORY RECORDS OF FINE-TUNE PARAMETERS

	Training	**Testing**
Round-01	0.93137	0.93043
Round-02	0.93285	0.93151
Round-03	0.93330	0.93200
Round-04	0.93205	0.93239
Round-05	0.93245	0.93146
Round-06	0.93286	0.93020
Round-07	0.93240	0.93137
Round-08	0.93146	0.93087
Round-09	0.93096	0.93063
Round-10	0.93245	0.93152

B. Comparison of Different Algorithms

The previous literatures reveal that several machine learning techniques were developed as predict engine. To empirically evaluate the proposed MLP model, additional models have been built by utilizing "scikit-learn". Scikit-learn is a free software machine learning library for the Python programming. It is built on NumPy, SciPy, and matplotlib, and provides an easy implementation for various types of classification, regression and clustering algorithms including Support Vector Machine (SVM), Random Forest, Logistic Regression, Adaptive Boosting (AdaBoost), K Nearest Neighbors, Naïve Bayes, and Decision Tree (Pedregosa et al., 2011). Those classification methods are adopted to represent a range of experiments in comparison with MLP. The tested methods are: Adaptive Boosting (AdaBoost), Decision Tree, Logistic Regression, Naïve Bayes, Random Forest, and Support Vector Machine (SVM).

The major dataset was split by 80/20 ratio with a random seed. Training set contains 265,169 records and testing set contains 66,293 records in testing set. TABLE VI shows that data fields are divided as 1 class label and 33 features since applying supervised learning techniques. By doing comprehensive searches on the best parameters, the MLP model ranks the first with 93% accuracy and AUC with 0.9290. They are both, slightly higher than Random Forest and AdaBoost while Naïve Bayes produces the lowest outcome. The performance metrics are presented in TABLE VII.

TABLE VI. LABLE AND FEATURES OF KICKSTARTER DATASET

Data Field	*Usage*	*Description*
project status	label	successful=1; failed=0
project goal	features	project goal
backers	features	number of backers
duration	features	project duration
length of project name	features	number of characters in project name
category	features	one-hot encoding to 15 features
location	features	one-hot encoding to 14 features

TABLE VII. PERFORMANCE METRICS - KICKSTARTER

Model	***Accuracy***	***AUC***
Multilayer Perceptron (MLP)	0.9320	0.9323
Random Forest	0.9293	0.9272
Adaptive Boosting (AdaBoost)	0.9242	0.9233
Support Vector Machine (SVM)	0.9065	0.8928
Decision Tree	0.9029	0.8990
Logistic Regression	0.8907	0.8716
Naïve Bayes	0.7102	0.6500

C. Scalability Testing

Scalability testing is a kind of performance testing which focus on changing of volume. In this session, scalability test is conducted to have better understanding in the differentiation of volume changes. Base on the major dataset records, the data is randomly divided as one-fourth, two-fourth, three-forth, and full portion for the testing. TABLE VIII represents the volume of data and training/testing records accordingly.

TABLE VIII. VOLUME OF SAMPLING DIVIDED BY FOUR PORTION

portion	***volume***	***successful vs. failed***			***training set***	***testing set***
1/4	62,149	successful	25,123	40.42%	49,719	12,430
		failed	37,026	59.58%		
2/4	165,731	successful	66,795	40.30%	132,584	33,147
		failed	98,936	59.70%		
3/4	248,596	successful	100,262	40.33%	198,876	49,720
		failed	148,334	59.67%		
4/4	331,462	successful	133,851	40.38%	265,169	66,293
		failed	197,611	59.62%		

The performance is very stable in results of scalability testing and MLP takes the first position in comparing with other machine learning methods. The result is shown as TABLE IX.

TABLE IX. RESULTS OF SCALABILITY TESTING

	Sampling - 25%		Sampling - 50%		Sampling - 75%		Sampling - 100%	
Model	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC
Multilayer Perceptron (MLP)	93.02%	0.9318	93.09%	0.9318	93.07%	0.9315	93.10%	0.9330
Adaptive Boosting (AdaBoost)	92.42%	0.9231	92.64%	0.9246	92.42%	0.9231	92.54%	0.9234
Random Forest	92.48%	0.9228	92.36%	0.9220	92.48%	0.9228	92.28%	0.9214
Decision Tree	90.16%	0.8977	90.19%	0.8977	90.12%	0.8972	90.11%	0.8972
Support Vector Machine (SVM)	90.23%	0.8880	89.84%	0.8830	90.23%	0.8880	90.11%	0.8866
Logistic Regression	88.42%	0.8634	87.32%	0.8497	88.42%	0.8634	88.68%	0.8675
Naïve Bayes	70.16%	0.6361	70.20%	0.6396	70.16%	0.6361	70.17%	0.6377

D. Discussion

The proposed prediction model, Multi-Layer Perceptron, has been developed and showed experimental results comparing with several classifier algorithms. The results prove that the MLP model performs well in terms of predicting probability of project success using only basic projects information. The results can also be replicated in different crowdfunding platforms.

By utilizing historical datasets from Kickstarter, MLP model is proven to fulfill the demand of effective prediction in terms of data size and platforms. In searching of best parameters, MLP demonstrates the time cost in average 10 to 15 minutes each round whereas Random Forest and Decision Tree cost more time. Comparing to other machine learning methods, the results revels that MLP takes the first position while Random Forest and Adaptive Boosting (AdaBoost) follows closely behind. On the whole, MLP and Random Forest are preferable approaches to solve nonlinear multivariate problems. Boosting simple algorithms, such as AdaBoost, could further improve the results. However, neural network only gives the probability of successful projects as it works like a black box hence not able to know the weights of each features. Besides prediction model, there are some interesting findings to be addressed while conducting EDA. First, games attracts the most investment, followed by creative designs and technology products. This reflects how innovative industries adopting new technologies such as eSports, innovative are thriving over the recent years. Second, although the platform is web-based, 80% of Kickstarter projects are raised domestically and similar phenomenon is found in Indiegogo as well. Third, the amount pledged demonstrates the positive correlation between the goal amount and number of backers. The logarithmic graphs also shows successful projects tend to have higher numbers of backers and amount pledged. In contrast, a project with high goal amount was less to successfully raise the desired funding. This may imply that, from a backer/investor's perspective, projects with smaller goal amount are more likely to succeed. Lastly, duration and length of project name are less associated with project success. Regression analysis does not yield, an obviously trend line in amount pledged, but, it shows the mass density versus price clustering. It also exposes the price barrier in goal amount set - higher density of data is found under 100,000 US dollars and duration between 60 and 30 days.

V. CONCLUSION AND FUTURE WORK

The aim of this paper is to develop a model to predict success of crowdfunding projects utilizing deep learning. The model could be applied in pre-launching or early stage of project's lifecycle. The proposed MLP model is preferable and performs consistently on different platforms which was not addressed before. To support this prediction engine, a variety of classification algorithms have been tested, ranging from Decision Trees, Logistic Regression to SVM. The MLP is the best performing model with the best degree of confidence.

Only one-third of projects pass the minimum fundraising goal and execution thoroughly. Prospective creators, especially novice creators, need a tool to help predict if the campaigns will achieve the goal so they could make plans on promotion or increasing backers interaction accordingly. This could also help backers identify favorite products.

In future work, would suggest to power up this model to a user-facing tool which serves as a real-time predictor of project funding. Another suggestion is to not only predict the probability of success, but also a range of expected amount pledged by incorporating information on the progresses each project is making. Beyond the success, future research may investigate how promotion activities inspire backers and help goal achieving. More broadly, it's worthwhile to develop a crowdfunding risk assessment platform for project data collection, credit rating, risk assessment and warning system.

Acknowledgment

This work is supported by Ministry of Science and Technology, Taiwan under grants MOST 107-2218-E-031 -001 -MY2, MOST 107-2514-S-031-001.

References

Aaker, J. L., & Akutsu, S. (2009). Why do people give? The role of identity in giving. Journal of Consumer Psychology 19 (2009), pp. 267-270.

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., ... & Ghemawat, S. (2015). TensorFlow:Large-Scale Machine Learning on Heterogeneous Distributed Systems. arXiv preprint arXiv:1603.04467. Retrieved from scikit-learn.org: http://scikit-learn.org/stable/modules/neural_networks_supervised.html

Bannerman, S. (2013, 6). Crowdfunding Culture. Journal of Mobile Media, 7(01), pp. 1-30. Retrieved from Wi: Journal of Mobile Media: http://wi.mobilities.ca/crowdfunding-culture

BBC NEWS. (2013, 4 25). The Statue of Liberty and America's crowdfunding pioneer. Retrieved from BBC News: http://www.bbc.com/news/magazine-21932675

Belleflamme, P., Lambert, T., & Schwienbacher, A. (2010). Crowdfunding: An industrial organization perspective. In Prepared for the workshop Digital Business Models: Understanding Strategies. Paris.

Chen, K., Jones, B., Kim, I., & Schlamp, B. (2013). KickPredict: Predicting Kickstarter Success. Technical report, California Institute of Technology.

Chung, J., & Lee, K. (2015). A long-term study of a crowdfunding platform: Predicting project success and fundraising amount. 26th ACM Conference on Hypertext & Social Media, (pp. 211-220). Cyprus.

Crowdfundbeat.com; Report: Global Crowdfunding Market 2016-2020. (n.d.). Retrieved from crowdfundbeat.com: https://crowdfundbeat.com/mobile/2016/02/03/report-global-crowdfunding-market-2016-2020/

Crowdsourcing, L. L. C. (2012). Crowdfunding industry report: market trends, composition and crowdfunding platforms. Retrieved from crowdsourcing. org: http://www. crowdsourcing. org/document/crowdfunding-industry-report-abridged-version-market-trends-compositionand-crowdfundingplatforms/14277

Cumming, D., Leboeuf, G., & Schwienbacher, A. (2014). Crowdfunding models: Keep-it-all vs. all-or-nothing.

Denton, J. W., Hung, M. S., & Osyk, B. A. (1990). A neural network approach to the classification problem. Expert Systems with Applications, pp. 1(4) , 417-424.

Etter, V., Grossglauser, M., & Thiran, P. (2013). Launch hard or go home!: predicting the success of kickstarter campaigns. Switzerland.

Freedman, D. M., & Nutting, M. R. (2015). A brief history of crowdfunding. Retrieved from Recuperado de: http://www. freedmanchicago. com/ec4i/History-of-Crowdfunding. pdf

Giudici, G., Nava, R., Rossi Lamastra, C., & Verecondo, C. . (2012). Crowdfunding: The new frontier for financing entrepreneurship? Milano.

Greenberg, M. D., Pardo, B., Hariharan, K., & Gerber, E. (2013). Crowdfunding support tools: predicting success & failure. CHI'13 Extended Abstracts on Human Factors in Computing Systems, (pp. 1815-1820). Paris, France.

Hemalatha, K., & Rani, K. U. (2017). Advancements in Multi-Layer Perceptron Training to Improve Classification Accuracy. International Journal on Recent and Innovation Trends in Computing and Communication. vol.5, pp. 353-357.

icopartners.com; Kickstarter in 2015–Review in numbers. (2016). Retrieved from icopartners.com: http://icopartners.com/2016/02/2015-in-review/

icopartners.com; Kickstarter in 2017 – Year in review. (2018). Retrieved from icopartners.com: http://icopartners.com/2018/01/kickstarter-2017-year-review/

Kaggle Dataset. (2018). Retrieved from Kaggle: https://www.kaggle.com/geekycb/kickstarter-data/data

Kaggle; IndieGoGo Project Statistics. (2017, 4). Retrieved from Kaggle: https://www.kaggle.com/kingburrito666/indiegogo-project-statistics

Kamath, R. S., & Kamat, R. K. (2016). Supervised learning model for kickstarter campaigns with R mining. International Journal of Information Technology, Modeling and Computing (IJITMC) Vol. 4, No.1,.

Keras Documentation. (n.d.). Retrieved from Keras: The Python Deep Learning library: https://keras.io/

Kickstarter-Our Rules. (2018). Retrieved from Kickstarter: https://www.kickstarter.com/rules

Kickstarter-stats. (2018). Retrieved from Kickstarter: https://www.kickstarter.com/help/stats

Kingma, D. P., Ba, J. . (2015). ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION. International Conference on Learning Representations.

Lai, C. Y., Lo, P. C., & Hwang, S. Y. . (2017). Incorporating Comment Text into Success Prediction of Crowdfunding Campaigns. PACIS 2017 Proceedings. 156.

Li, Y., Rakesh, V., & Reddy, C. K. (2016). Project success prediction in crowdfunding environments. the Ninth ACM International Conference on Web Search and Data Mining, (pp. 247-256). San Francisco.

Number of total and repeat Kickstarter project backers as of January 2018. (2018). Retrieved from Statista - The Statistics Portal: https://www.statista.com/statistics/288345/number-of-total-and-repeat-kickstarter-project-backers/

Pedregosa et al. (2011). scikit-learn Machine Learning in Python. Journal of Machine Learning Research, 12, pp. 2825-2830. Retrieved from scikit-learn: http://scikit-learn.org/stable/index.html

Rakesh, V., Lee, W. C., & Reddy, C. K. (2016). Probabilistic group recommendation model for crowdfunding domains. the Ninth ACM International Conference on Web Search and Data Mining (pp. 257-266). ACM.

Sawhney, K., Tran, C., & Tuason, R. (2016). Using Language to Predict Kickstarter Success. Stanford University.

Sharma, S. (2017). ctivation Functions: Neural Networks. Retrieved from towardsdatascience.com: https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6

The Kickstarter Blog: The History of #1. (2012, 4). Retrieved from Kickstarter: https://www.kickstarter.com/blog/the-history-of-1-0

The Kickstarter Blog: when creators return. (2015, 3). Retrieved from Kickstarter: https://www.kickstarter.com/blog/by-the-numbers-when-creators-return-to-kickstarter

The Kicksterter Blog: The History of #1 UPDATED. (2014, 8). Retrieved from https://www.kickstarter.com/blog/the-history-of-1-updated

Tomczak, A., & Brem, A. (2013). A conceptualized investment model of crowdfunding. Venture Capital: An International Journal of Entrepreneurial Finance, 15:4, pp. 335-359.

Walia, A. S. (2017). Types of Optimization Algorithms used in Neural Networks and Ways to Optimize Gradient Descent. Retrieved from Towards Data Science: https://towardsdatascience.com/types-of-optimization-algorithms-used-in-neural-networks-and-ways-to-optimize-gradient-95ae5d39529f

Yang, J., Adamic, L. A., & Ackerman, M. S. (2008). Crowdsourcing and knowledge sharing: strategic user behavior on taskcn. the 9th ACM conference on Electronic commerce, (pp. 246-255).