ИНФОРМАТИКА. ИНФОРМАТИКАНЫ ОҚЫТУ ӘДІСТЕМЕСІ. БІЛІМ БЕРУДІ АҚПАРАТТАНДЫРУ
ИНФОРМАТИКА. МЕТОДИКА ПРЕПОДАВАНИЯ ИНФОРМАТИКИ. ИНФОРМАТИЗАЦИЯ ОБРАЗОВАНИЯ

МРНТИ 28.23.00
УДК 004.89

EARTHQUAKE TIME PREDICTION WITH MULTI-AGENT SYSTEMS

Abyurova A.A.

Al-Farabi Kazakh National University, Almaty, Kazakhstan

Abstract

This article is devoted to the development of multi-agent systems for predicting the time of earthquakes based on seismic signals. The work uses a dataset from laboratory signals, which used to calculate the time predicted before the next earthquake. The MadKIT platform and the Python programming language are used to build multi-agent systems. The "tsfresh" package is used to calculate a large number of time series characteristics, so-called features, from seismic signals for further use in regression. The article considers one of the regression models - LightGBM. Using it, a set of data was processed and the predicted time of the earthquake was obtained. This article shows the relevance and prospects of the research area, describes the functionality of the created agents.

Keywords: multi-agent system, earthquake prediction, models of prediction, regression models, agents, artificial intelligence.

Аңдатпа

А.А. Абюрова
әл-Фараби атындағы Қазақ Ұлттық университеті, Алматы қ., Қазақстан

МУЛЬТИАГЕНТТІК ЖҮЙЕЛЕРДІ ПАЙДАЛАНЫП ЖЕР СІЛКІНІСІ УАҚЫТЫН БОЛЖАУ

Берілген мақала сейсмикалық сигналдар негізінде жер сілкіну уақытын болжау үшін мультиагенттік жүйелерді әзірлеуге арналған. Бұл жұмыста зертханалық сигналдар арқылы келесі жер сілкінісі болжанатын уақыт жүзеге асырылатын деректер жинағы қолданылды. Мультиагенттік жүйелерді құру үшін MadKIT платформасы және Python бағдарламалау тілі қолданылды. Деректер жинағы тек екі сипаттамадан құралған соң "Tsfresh" пакеті сейсмикалық сигналдардан қосымша сипаттамаларды есептеп шығаруга септігін тигізді. Алынған нәтижелер ары қарай модельдің бірінде регрессияда қолданылды. Мақалада регрессиялық модельдердің бірі - LightGBM қарастырылған. Оның көмегімен деректер жиынтығы өңделіп, жер сілкінісінің болжамдық уақыты алынды. Бұл мақалада зерттелетін саланың өзектілігі мен келешектегі пайдасы көрсетілген және құрылатын агенттердің функционалы сипатталған.

Түйін сөздер: мультиагенттік жүйе, жер сілкінісін болжау, болжау моделі, регрессиялық модельдер, агенттер, жасанды интеллект.

Аннотация

А.А. Абюрова
Казахский национальный университет им. аль-Фараби, г.Алматы, Казахстан

ПРОГНОЗИРОВАНИЕ ВРЕМЕНИ ЗЕМЛЕТРЯСЕНИЙ С ИСПОЛЬЗОВАНИЕМ МУЛЬТИАГЕНТНЫХ СИСТЕМ

Статья посвящена разработке мультиагентных систем для прогнозирования времени землетрясений на основе сейсмических сигналов. В работе используется набор данных из лабораторных сигналов, с помощью которого осуществляется время, прогнозируемое до следующего землетрясения. Для построения мультиагентных систем использована платформа MadKIT и язык программирования Python. Пакет “tsfresh” применен для вычисления большого количества характеристик временных рядов, так называемых признаков, из сейсмических сигналов для дальнейшего использования в регрессии. Рассмотрена одна из регрессионных моделей - LightGBM. С помощью нее обработан набор данных и получено прогнозируемое время землетрясения. В данной статье показана актуальность и перспективность исследуемой области, описан функционал создаваемых агентов.

Ключевые слова: мультиагентная система, прогноз землетрясений, модели прогнозирования, регрессионные модели, агенты, искусственный интеллект.

Introduction

The forecast of an earthquake is one of the topical problems of our time, which has not only scientific, but also serious practical significance. The need for an early solution to the problem of earthquake forecasting is steadily increasing. Current earthquake forecasting scientific studies focus on three main points: when the occasion will happen, where it will happen, and how huge it will be.

In the twentieth century, intensive international attempts were made to solve such a dangerous problem for humanity, but they did not achieve any significant success.

Thus, according to V.I.Keylis-Borok, the existing earthquake prediction systems are able to provide the following highly probable estimates of the accuracy and its characteristics [1]:

a) the place of the upcoming earthquake-hundreds of kilometers;

b) the possible energy of the expected earthquake is six orders of magnitude;

c) implementation time – years.

It is obvious that such a forecast has no practical value. In addition to generating false anxiety, it can also generate false calm. This is what happened in 1989 in California, when a strong earthquake was expected in Parkfield (near the San Francisco in 300km), and it occurred in San Francisco. In recent decades, Geophysics has been dominated by two radically opposite approaches to assessing seismic risk.

The first approach is based on the direct detection method the epicentre of the impending earthquake – deformation-geodetic method, which allows, as suggested by his followers, make an accurate forecast of the hearth and to determine the maximum possible strength of future earthquakes.

The next approach is based on a method for solving inverse problems – from analyzing the behavior of anomalies in various geophysical fields, the so-called anomaly strategy.

Such a strategy for implementing an earthquake forecast is based on the ideas of detecting the hearth and tracking the processes occurring in it by scattered indirect signs – anomalies generated by the preparing hearth in various fields: seismic, deformation, hydrogeological, geochemical and electromagnetic.

Used technologies

The prediction of earthquake as a whole has proven to be a challenge which is essentially impossible. With modern computing resources, machine learning methods and a dramatically narrow focus, however, maybe some progress can be made in this important task. One of the technologies in order to achieve the task is multi agent systems. Multi-agent system – a set of interrelated agents that can interact with each other and the environment, have certain intellectual abilities and the ability to individual and joint actions [2].

MAS has the following characteristics:

• each individual agent does not have enough information or ability to solve the problem, and thus does not have a complete vision of the global task to be completed;

• there is no global control in the system, i.e. there are no agents managing the entire system;

• no centralized data storage;

• agents are at least partially independent;

• limited representation, i.e. none of the agents have an idea of the entire system or the system is too complex for knowledge about it to have practical application for the agent;

• calculations are performed asynchronously [3].

The aim of this work is to predict the timing of laboratory earthquakes using seismic signals. The dataset originates from a notable test set-up used to study physics of the earthquake. There are two parameters in dataset:

- acoustic data - input seismic signal;
- time to failure - remaining time before the next earthquake.

In this project, I consider three agents. First agent formats the data, as our dataset is too large in order to run it on a normal computer. It is going to convert to an appropriate way for use. As we have a dataset that consists of only two features - the seismic signal and the time until the next earthquake, we need to make a prediction when the next earthquake will occur with only one feature, thus we need to pull additional features from the data, which can be used in regression. This is a mission of the second agent. The last agent will predict when the next earthquake will occur according to the model (Figure 1).

The total acoustic data in the training set is shown below. Each of the big spikes represents an earthquake event, and the training set includes around 16 earthquake events. In addition, the training data comprises 630 million perceptions. Orange plotted line is a time to failure.

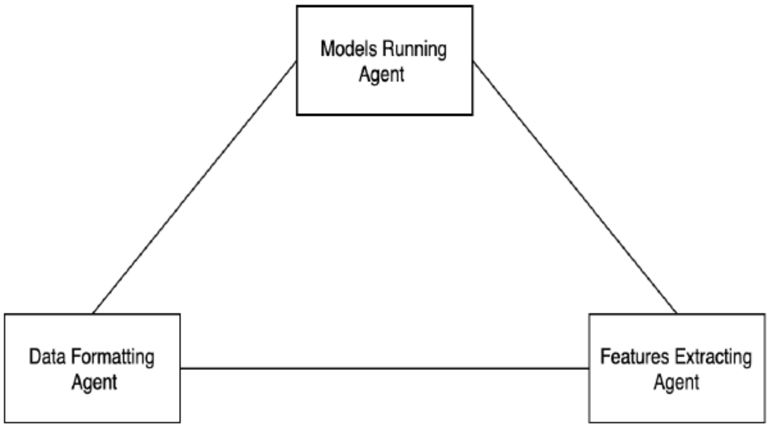

Figure 1. Interactions of agents

As you can see, the time to failure decreases linearly until an earthquake strikes, then resets. The mission of this project is to take a set of 140 000 observations and predict where the orange line could be (Figure 2).

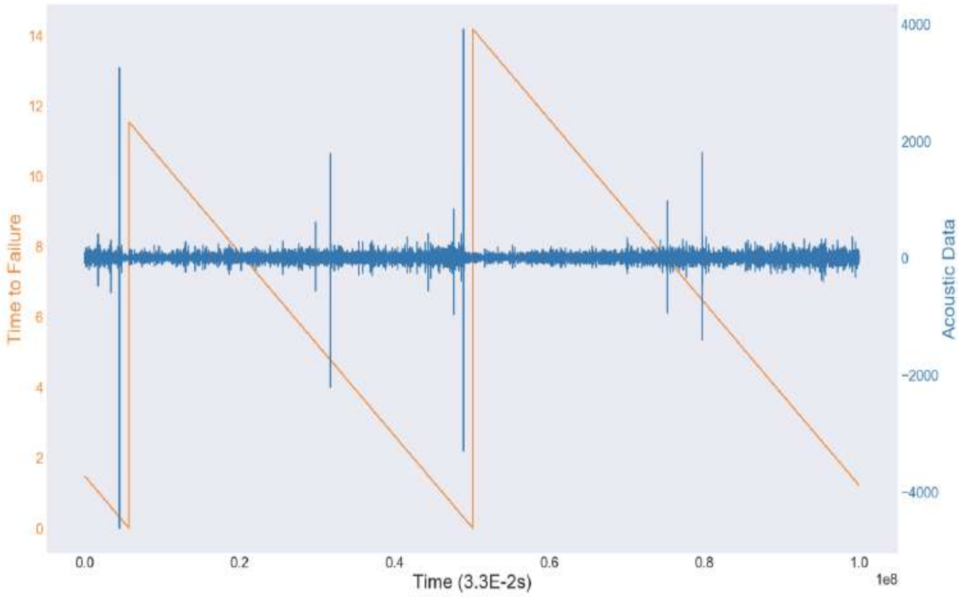

Figure 2. Acoustic data and time to failure.

Featuring extraction and modeling

Once there are only two features available in the dataset, then I have to get additional features. Some of the features are simple aggregate functions such as a segment mean, IQR, standard deviation and so on. However, to get better results, I use more complicated features that are prevalent in the analysis of time series.

For this reason, the "tsfresh" module of BlueYonder Tech will be helpful. This module extracts up to 1100 time series related features. In the table given below shown the formulas of the theory of probability used in method. All with one method only.

Table 1. Statistical features

Measure	*Formula*	*Description*
Mean	$\bar{x} = \frac{x_1 + x_2 + \cdots + x_n}{n}$ *or* $\bar{x} = \frac{\sum x}{n}$	*The average (mean) value is the sum of all values in the dataset divided by the number of values in the dataset.*

Median	$n + \frac{1}{2}$ *Position*	*The average score for a data set that was ordered by order of magnitude.*
Mode	*None*	*Most frequent*
Standard deviation	$N \sum \lvert x - \mu \rvert 2$	*A measure of the number of changes or variance in a set of values.*
Kurtosis	$\frac{1}{n-1} \sum_{i=1}^{n} \left(\frac{x_i - Mn}{Sd} \right)^4$	*It is an indicator that reflects the sharpness of the vertex and the thickness of the tails of a one-dimensional distribution*
Skewness	$\frac{1}{n-1} \sum_{i=1}^{n} \left(\frac{x_i - Mn}{Sd} \right)^3$	*An asymmetric distribution can be characterized by the terms*

The structure of the agent for featuring parameters consists from the model and agent classes. Both of them are child classes of generic Mesa classes: Model and Agent.

```
class FeatureParamAgent(Agent):

    def __init__(self, uni_id, mdl):
        super().__init__(uni_id, mdl)
        self.wealth = 1

class FeatureParamModel(Model):
    "A model with certain nmber of agnts."
    def __init__(self, N):
        self.num_agnts = N
        # Create agents
        for i in range(self.num_agnts):
            a = FeatureParamAgent(i, self)
```

The extracted training data demonstrated in the figure 3 below.

```
temp = statistical_features.join(rollingwindow_features)
data = temp.join(fourier_features)
print(data.shape)
data.head()
```

(4194, 132)

	mean	std	max	min	abs_mean	mean_first_10000	mean_first_50000	mean_last_10000	mean_last_50000	abs_std	...	img_FFT_5quantile	img
0	4.884113	5.101106	104.0	-98.0	5.576567	5.1820	4.96210	4.899007	5.01594	4.333325	...	-1605.470156	
1	4.725767	6.588824	181.0	-154.0	5.734167	4.7772	4.69840	4.712293	4.69448	5.732777	...	-1859.559966	
2	4.906393	6.967397	140.0	-106.0	6.152647	4.6814	4.70610	4.886771	4.81588	5.895945	...	-1944.064062	
3	4.902240	6.922305	197.0	-199.0	5.933960	5.0364	4.84364	4.882936	4.83663	6.061214	...	-1889.301741	
4	4.908720	7.301110	145.0	-126.0	6.110587	4.9405	4.89116	4.923021	4.94855	6.329485	...	-1965.867577	

5 rows × 132 columns

Figure 3. Training data

When all the features are extracted, it is time to start modeling. There are several of the popular regressors that are currently available for the modeling: SVM, OLS, Bagging Regressor, XGBoost, CatBoost, Random Forest, Light Gradient Boost Method. CatBoost and LGBM are more recent, and their tree computing method is slightly different, although I personally found LGBM to be generally the fastest of the three. Above is a list of the features that LGBM determined were most important.

The architecture of multiagent systems looks like as in the figure 4,5 below. The process starts from the input data illustrated in the left side. Seismic detectors receive signals, put through sensor processing and starts the preprocessing level. The preprocessing level load input parameters, then feature more of them. All obtained data stored and used in the next level – modeling. As shown in the figure, user interface is available from the beginning till the end.

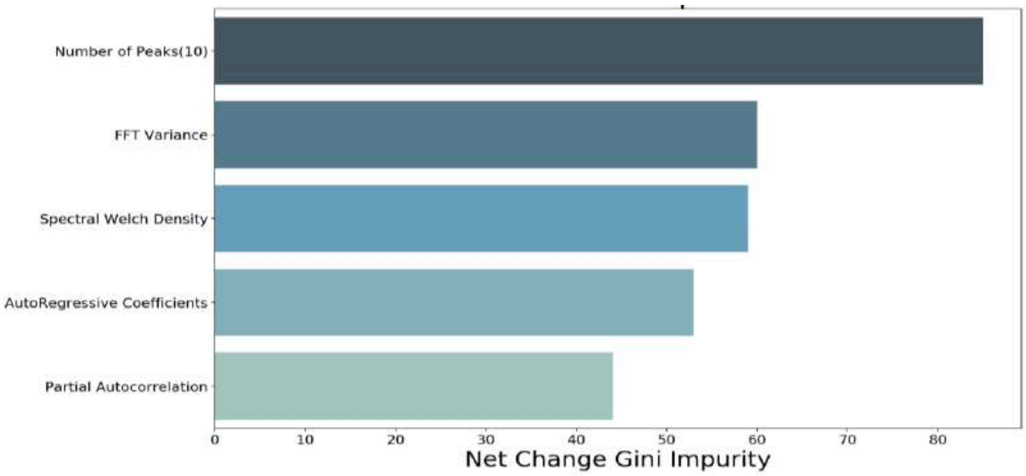

Figure 4. LGBM features

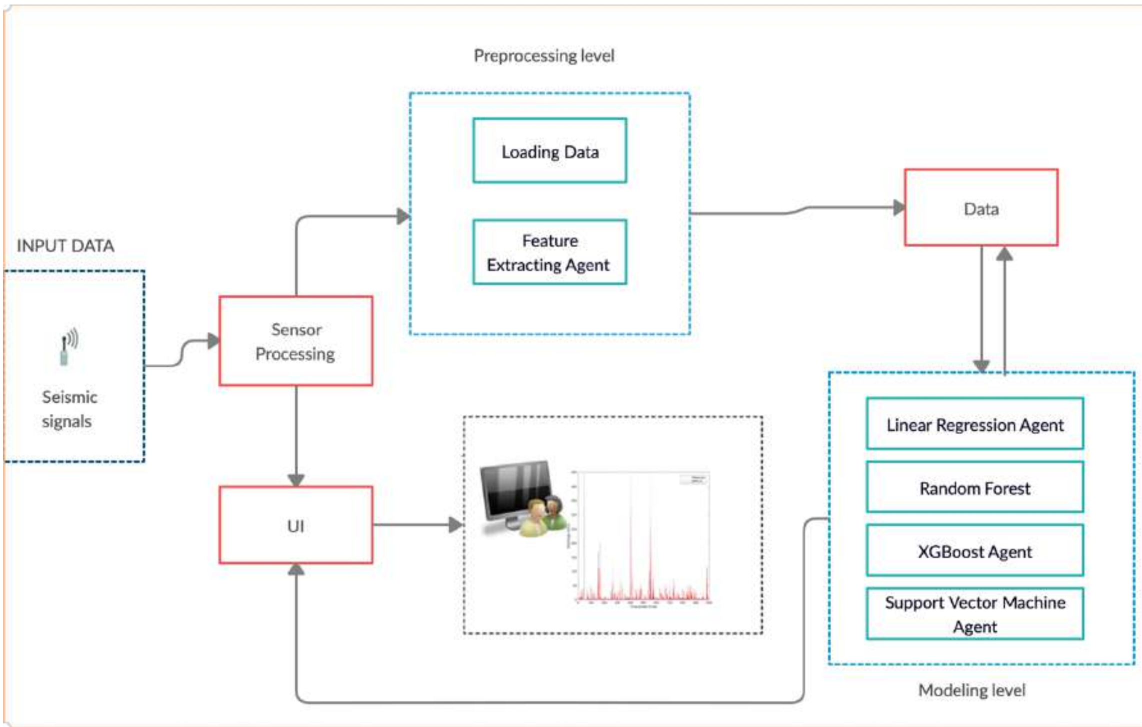

Figure 5. MAS architecture

Conclusion

After running the modelling in LGBM, the pivot score was 2.4175. This is a good result, where the ideal result is 2.26589. LGBM's one of the best advantages was its rapidity in standard computers. In this work, I learnt how to work with huge data – divide into segmentations and build the model. For the future, I plan to use DASK computers, which can easily run many millions of data rows in different regression models in parallel. It allows me to figure out which model suits my project best. Overall, one of the agents will select the model that will give the prediction with the least amount of error.

References:

1 Ferber J. Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence, Addison-Wesley Longman Publishing Co., Inc.Boston, MAUnited States, 1999 pp 88-90.

2 Wooldridge M. An Introduction to Multiagent Systems, John Wiley & Sons Ltd Baffins Lane Chichester, England, 2002 pp 36-40.

3 Minayev V.A., Fadeev A.O. and Abramova A.V. Mathematical modeling of seismic risks, Earth Sciences and related environmental Sciences, 2013 pp 63-68.