Использование методов поиска паттернов в последовательности событий для прогнозирования поломок сложных технических систем

Герман Новиков

Институт проблем передачи информации РАН,
german.novikov@phystech.edu

Аннотация Анализ редких событий является областью, включающей в себя методы для обнаружения и прогнозирования событий, например вторжений в сеть или отказов двигателя, которые происходят редко, но имеют существенное влияние на систему. Для этой задачи применяются различные методы из области статистики и анализа данных. Целью статьи является анализ методов и алгоритмов, которые используются для прогнозирования редких событий в различных системах и обозначение трудностей, которые возникают при решении задачи прогнозирования аномалий.

Ключевые слова: временные ряды, распознавание паттернов, бинарная классификация, несбалансированная классификация

1 Введение

Задача распознавания паттернов - это неотъемлемая часть анализа данных, для решения которой используется множество различных подходов. Эта задача возникает в различных приложениях, начиная построением ассоциативных правил в прогнозировании цепочек из покупок в супермаркетах и, впоследствии, соответствующего размещения товеров на полках [4], и заканчивая предсказанием поломок в сложных системах, таких как компьютерные жесткие диски [1] и двигатели самолётов, или прогнозированием резких скачков экономических показателей [11]. Среди приведённых примеров наиболее важной является задача детектированя поломок [17] в силу того, что от точности её решения напрямую зависит безопастность жизни людей.

Типичным и наиболее популярным решением данной задачи является построение модели, использующей стандартные методы классификации, а именно, строится классификатор для прогнозирования вероятности происхождения события $Y(t+h) = 1$ по наблюдениям $X(t-L), ...X(t)$, где Y - это и есть индикатор предсказываемой поломки (1 - поломка есть, 0 - нет), X -

многомерный временной ряд, элементы которого показывают зависимость показаний каждого из датчиков состояния системы от времени t.

В литературе по машинному обучению и статистике описывается обилие методов детектирования выбросов с учителем и без: статистические, методы кластеризации, основанные на нейронных сетях, SVM и другие [3]. Однако, эти методы не всегда годятся, так как:

1. Размер искомого класса - множества поломок, очень мал по сравнению с размером класса событий, когда поломок не происходит, а значит мы имеем дело с несбалансированной классификацией (imbalanced classification problem), для которой разработано множество особых подходов [8,15].
2. Метрика, которая оценивает качество прогноза в этой задаче, не может иметь ничего общего с обычной метрикой качества классификации. При прогнозировании поломок самолётов нам необходимо получить информацию о предстоящем выходе из строя какого-либо агрегата в точно определенный отрезок времени перед этим событием. Если мы получаем сигнал о поломке слишком рано, то это уменьшает эффективность действий по её устранению. Аналогично, если он получен слишком поздно. Также недопустимо, если поломка начнётся раньше спрогнозированного времени и допустимо, если немного позже.
3. Данные, содержащиеся в $X(t)$ зачастую представляют собой не какие-то физические/действительные признаки, а индикаторы событий, некоторые типы которых происходят редко. Соответственно, если мы научимся детектировать какие-то редкие последовательности типов событий в многомерной последовательности событий $X(t)$, которые будут являться аномалиями в нашем понимании, то это может служить предвестником поломки и помочь в ее прогнозировании.

Таким образом, цель данной работы состоит в том, чтобы:

1. Дать обзор методов несбалансированной классификации на основе ресемплинга
2. Описать в математических терминах метрику, наиболее подходящую для классификации в случае вышеописанных особенностей задачи
3. Дать краткий обзор методов выявления последовательностей событий, установив их применимость к прогнозированию поломок

2 Обзор методов ресемплинга для несбалансированной классификации

В реальных задачах случай несбалансированности возникает крайне часто, и среди этих задач очень важным оказывается точное детектирование объектов, которые принадлежат именно меньшим классам. Примерами для бинарной классификации могут служить такие задачи как распознавание

вторжения в сеть [12] или мошенничества с кредитными картами [5], предсказание поломок в сложных технических системах [17]. В таких случая стоит задача изменения метода оценки точности классификатора всвязи с несбалансированностью, так как точность детектирования объектов, принадлежащих именно меньшему классу здесь очень важна, а при использовании обычных методов оценки простое присваивание всех объектов большему классу даёт почти идеальный результат, однако, это никак не помогает решить поставленую задачу.

Поэтому для улучшения работы алгоритмов классификации для несбалансированной задачи используется несколько методов, которые позволяют увеличить важность меньшего класса:

- Адаптация порога для классификатора, определяющего вероятности принадлежности классам
- Модификация функции ошибки в сторону увеличения стоимостью ошибки за неправильную классификацию меньшего класса
- Ресемплинг:
  - Oversampling, то есть добавление дополнительных, искуственно сгенерированных объектов, принадлежащих меньшему классу
  - Undersamplig, то есть удаление случайно или определенным образом выбранных объектов из большего класса
  - Смешанный метод, то есть добавление редких и удаление частных объектов из обучающей выборки

2.1 Алгоритмы ресемплинга

Обозначения. Обозначения, приятые в данном разделе: набор данных S, показатель несбалансированности $IR = \frac{|C_0|}{|C_1|}$, где $C_0 = \{(x_i, 0)\}$ множество элементов первого класса в обучающей выборке, $C_0 = \{(x_i, 1)\}$ множество элементов второго класса, и пусть C_0 - преобладающий класс. Мультипликатором алгоритма $m > 1$ будем называть $m = \frac{IR(S)}{IR(r(S))}$, где $r(S)$ - соответственно, один из алгоритмов ресемплинга, примененный к начальному набору данных.

Random Oversampling. Также известен как bootstrap oversampling. Добавляет в начальное множество $(m-1)|C_1(S)|$ элементов первого класса, используя равномерное распределение вероятности выбора каждого из них.

Random Undersampling. Удаляет из $C_0(S)$ произвольное подмножество размера $\frac{m-1}{m}|C_0(S)|$, причём каждое из всех возможных подмножеств может быть удалено с равной вероятностью.

SMOTE. Synthetic Minority Oversampling Technique (SMOTE) [6], метод оверсемплинга, использующий метод k-ближайших соседей для генерации новых объектов, которые будут добавлены $C_1(r(S))$, где k - дополнительный параметр алгоритма:

1. Инициализируем пустое множество $S_{new} := \emptyset$
2. Повторяем $(m-1)|C_1(S)|$ раз:
   (a) Выбираем случайный элемент x_i из $C_1(S)$.
   (b) Из k его ближайших соседей выбираем один произвольный x_j
   (c) Добавляем в S_{new} произвольный элемент x из отрезка $[x_i, x_j] : S_{new} = S_{new} \cup (x, 1)$
3. $S = S \cup S_{new}$

Другие методы Менее популярными являются такие методы ресемплинга как Tomek Link Detection [13], Evolutionary Undersampling [9], borderline-SMOTE [10], Neighborhood Cleaning Rule [14] и многие другие.

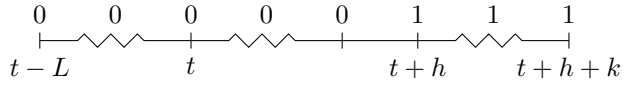
2.2 Выводы

Приведенные выше алгоритмы в большинстве случаев позволяют улучшить точность классификации [12], в следствии чего использование ресемплинга чаще всего является целесообразным в случае несбалансированной классификации.

3 Математическое описание метрики

Описанные в постановке задачи особенности накладывают на метрику некоторые условия. Введем обозначения, и покажем, каким образом каждое из свойств должно учитываться в конечной метрике, используемой для анализа точности прогнозирования поломок.

Пусть рассматривается промежуток времени $[t-L, t+h+k]$, где $[t-L, t]$ - допустимое окно детектирования поломки, $t+h$ - время начала поломки поломки.



F - множество поломок; при этом поломкой будем называть последовательность подряд идущих единиц (то есть если она возникла и продолжается, то мы считаем это одной поломкой)

– Время за которое должно быть известно о каждой из поломок не превосходит h,

$$E_1 = \sum_{i \in F} c(\Delta t_i, h)$$

и при этом коэффициент стоимости $c(\Delta t_i, h)$ зависит от времени (Δt_i), оставшегося до поломки($\Delta t_i \leq h$) это можно объяснить возрастанием экономической сложности устранения неполадки с её приближением

– Штраф

$$E_2 = \sum_{i \in F} Ind(Predicted\, fault_i\, time > t_i + h) \times ERROR_1$$

за предсказание, что каждая поломка произойдёт в промежутке $[t_i + h + 1, t_i + h + k]$ должен быть порядка штрафа за непойманную поломку, так как момент поломки определен неправильно и использование технического средства в момент $t_i + h$ является невозможным. Здесь $ERROR_1$ является коэффициентом стоимости для несвоевременного предсказания

– Штраф за неспрогнозированные поломки :

$$E_3 = \sum_{i \in F} Ind(fault_i\, unpredicted) \times ERROR_2$$

где $ERROR_2$ - коэффициент стоимости непредсказанной неполадки.

– Некоторый штраф за преждевременное сообщение о предстоящей поломке, например на промежутке $[t_i - L, t_i]$ линейно падающий от некоторой небольшой константы до 0, а при удалении от $t_i - L$ к началу отсчёта времени зависящей как степенная (или какая либо другая быстро растущая) функция $F(\delta t)$, где $\delta t = t_i + h - t$ - время до поломки, что можно объяснить необходимостью отсеивать лишние, то есть фактически ложные сообщения о поломках:

$$E_4 = \sum_{i \in F} (Ind(t \in [t_i - L, t_i]) \times \delta t \times K + Ind(t < t_i - L) \times F(\delta t)$$

Таким образом, перед нами стоит задача построения классификатора, минимизирующего функцию вида $E = E_1 + E_2 + E_3 + E_4$. Конкретный вид функции должен быть подобран в зависимости от конкретных параметров задачи.

4 Применяемые методы выделения последовательностей

Одним из наиболее популярных методов детектирования последовательностей, предшествующих редким событиям является применение метода построения ассоциативных правил [3], основанного на алгоритме *Apriori* [2]. Основная идея данного алгоритма в простом правиле - если некоторый набор событий является достаточно частым, то есть превосходит некоторый уровень поддержки s, то и любое подможество этого набора должно быть не менее частым. Применение его к детектированию редких событий сводится

к выделению перед всеми редкими событиями в обучающей выборке окна некоторого размера L и дальнейшего выделения из них правил с использованием *Apriori*. После этого построенные правила используются в режиме онлайн для сопоставления с рядом происходящих событий и предсказания целевого события.

Используются так же методы, применяющие скрытые Марковские модели (Hidden Markov Models) [18,16]. При этом для каждой поломки, используется Алгоритм Баума — Велша, строящий HMM. На стадии предсказания приходящие в режиме онлайн данные подаются на вход созданным моделям, которые в каждый момент времени показывают вероятности перехода в состояние поломки.

Методы, основанные на HMM, в терминах теории формальных языков схожи по вычислительной мощности с вероятностными регулярными грамматиками (PRG). Однако, существует более мощный класс грамматик - вероятностные контекстно-свободные грамматики ($PCFG$), которые также применимы при выделении последовательностей [7].

Таким образом, все наиболее популярные в приложениях методы детектирования последовательностей созданы для решения задачи в одномерных рядах. Однако, в случае рассматриваемых нами сложных технических систем, большая (например для некоторых подсистем самолёта $\sim$100) размерность признакового пространства не позволяет напрямую использовать эти методы. Все полученные во время исследования алгоритмы с их применением имеют высокую вычислительную сложность и в реальной задаче неприменимы.

5 Модель для задачи и эксперимент

Для начала сделаем простое и физически обоснованное предположение о том, что существенными для предсказания поломок в сложных технических системах являются показания датчиков, которые переходят некоторые критические значения. Тогда будем использовать метод бинаризации признаков с помощью квантилей. Для каждого вещественного признака определим некоторое количество квантилей K, и будем из каждого имеющегося признака генерировать K новых бинарных, которые показывают - превысили ли показания данного датчика соответствующий квантиль или нет.

Также будем предполагать, что существенным поводом для возникновения поломки является одновременное обращение в экстремальные значения показаний некоторых наборов датчиков. Что также является достаточно обоснованным с физической точки зрения, так как к поломке в системе часто ведёт именно одновременное возникновение критической нагрузки на разные её подсистемы.

Для моделирования в исследовании была рассмотренна задача, схожая с задачей детектирования поломок у двигателей самолётов. Были сгенерированы данные, отвечающие описанным в работе свойствам: количество измерений (9000) соответствует количеству полётов нескольких десятков само-

лётов за несколько лет, количество признаков (100) примерно соответствует количеству датчиков, отвечающих одному агрегату, частота возникновения поломок (120 на 9000 измерений) примерно отвечает типичной частоте поломок для двигателя самолёта. Поломки расставлялись в соответствии с предположениями, описанными выше, то есть были выбраны 4 разных пары признаков и поломка генерировалась тогда и только тогда, когда хотя бы одна пара одновременно обращалась в 1. Начальные признаки были сгенерированны случайным образом - каждое из измерений моделируемых датчиков с вероятностью 5% объявлялось экстремальным (то есть равным 1). Были сгенерированны дополнительные признаки, являющиеся логическими & для всех пар начальных признаков.

В результате первоначального отбора классификаторов, было выявлено, что обычные классификаторы (SVN, $RandomForest$, NN, kNN), как и ожидалось, очень плохо работают с такой размерностью и разбалансированностью, наилучшим оказался результат $RandomForest$. Также были применены статистические модели $ttest$ и логистическая регрессия, результаты которых оказались намного лучше. Для дальнейшего применения на реальных было решено сравнить результаты $ttest$ и логистической регрессии для разных уровней зашумленности данных как на обучающей выборке так и на тестовой.

В Таблице 1 показаны зависимости среднего количества нераспознанных поломок для каждого из алгоритмов при разном уровне шума. Уровень шума отложен по горизонтали и показывает отношение количества случайно добавленых поломок к реальному их числу (120).

	0	2	4	6	8	10	12	14	16	18
Logistic Regression	0.8	2.4	3.8	7.8	11.0	14.4	17.6	20.4	32.6	34.2
ttest	0.1	0.3	0.5	1.1	1.5	1.9	2.1	3.1	3.9	4.5
Random Forest	5.4	6.6	8.1	19.8	22.1	42.3	69.9	74.0	89.0	114.1

Таблица 1

Также необходимо оценить количество неправильно предсказанных поломок. В Таблице 2 аналогично показана зависимость среднего количества ложных тревог от уровня шума.

	0	2	4	6	8	10	12	14	16	18
Logistic Regression	0.0	0.1	0.1	0.5	1.1	1.7	3.5	7.1	14.5	21.8
ttest	0.0	0.0	0.0	0.0	0.1	0.1	0.3	0.5	0.7	1.1
Random Forest	0.0	4.1	5.8	3.3	2.1	1.0	0.1	0.0	0.0	0.1

Таблица 2

Полученные результаты показывают, что обычные методы классификации плохо подходят для задачи такого рода и здесь лучше справляются статистические методы. Таким образом, в дальнейшем предстоит исследовать возможность использования статистических методов совместно с методами выделения последовательностей, что позволит расширить класс детектируемых поломок.

6 Выводы

Итак, мы установили, что имеются постановки задач прогнозирования редких событий, а именно - поломок сложных технических средств, которые не поддаются решению стандартными методами. В работе рассмотрены несколько важных особенностей таких задач и приведено описание новых, возникающих в связи с этим подзадач. Таким образом, обозначены сложности и приведены возможные методы борьбы с ними, применимость которых будет в дальнейшем исследована более детально. А также на модельных данных проведено моделирование простого, но эффективного метода детектирования поломок в конкретном случае, который часто возникает в системах типа отдельных агрегатов летательных средств.

Исследование выполнено в ИППИ РАН за счет гранта Российского научного фонда (проект № 14-50-00150).

Список литературы

1. V. Agarwal, C. Bhattacharyya, T. Niranjan, and S. Susarla. Discovering rules from disk events for predicting hard drive failures. In *Machine Learning and Applications, 2009. ICMLA '09. International Conference on*, pages 782–786, Dec 2009.
2. Rakesh Agrawal and Ramakrishnan Srikant. Fast algorithms for mining association rules in large databases. In *Proceedings of the 20th International Conference on Very Large Data Bases*, VLDB '94, pages 487–499, San Francisco, CA, USA, 1994. Morgan Kaufmann Publishers Inc.
3. CHRISTOS BERBERIDIS and IOANNIS VLAHAVAS. Detection and prediction of rare events in transaction databases. *International Journal on Artificial Intelligence Tools*, 16(05):829–848, 2007.
4. Sergey Brin, Rajeev Motwani, Jeffrey D. Ullman, and Shalom Tsur. Dynamic itemset counting and implication rules for market basket data. pages 255–264. ACM Press, 1997.
5. Philip K. Chan and Salvatore J. Stolfo. Toward scalable learning with non-uniform class and cost distributions: A case study in credit card fraud detection. In *In Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining*, pages 164–168. AAAI Press, 1998.
6. Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.
7. Witold Dyrka and Jean-Christophe Nebel. A probabilistic context-free grammar for the detection of binding sites from a protein sequence. *BMC Systems Biology*, 1(Suppl 1), 2007.
8. Vaishali Ganganwar. An overview of classification algorithms for imbalanced datasets. *International Journal of Emerging Technology and Advanced Engineering*, 2(4):42–47, April 2012.
9. Salvador García and Francisco Herrera. Evolutionary undersampling for classification with imbalanced datasets: Proposals and taxonomy. *Evol. Comput.*, 17(3):275–306, September 2009.

10. Hui Han, Wen-Yuan Wang, and Bing-Huan Mao. Borderline-smote: A new over-sampling method in imbalanced data sets learning. In De-Shuang Huang, Xiao-Ping Zhang, and Guang-Bin Huang, editors, *Advances in Intelligent Computing*, volume 3644 of *Lecture Notes in Computer Science*, pages 878–887. Springer Berlin Heidelberg, 2005.
11. Jim Wang Ionut Florescu, Khaldoun Khashanah. Hidden markov models for failure diagnostic and prognostic. 2012.
12. Christopher Kruegel, Darren Mutz, William Robertson, and Fredrik Valeur. Bayesian event classification for intrusion detection. In *IN: PROCEEDINGS OF ACSAC 2003, LAS VEGAS, NV*, page 14, 2003.
13. Miroslav Kubat and Stan Matwin. Addressing the curse of imbalanced training sets: One-sided selection. In *In Proceedings of the Fourteenth International Conference on Machine Learning*, pages 179–186. Morgan Kaufmann, 1997.
14. Jorma Laurikkala. Improving identification of difficult small classes by balancing class distribution. In *Proceedings of the 8th Conference on AI in Medicine in Europe: Artificial Intelligence Medicine*, AIME '01, pages 63–66, London, UK, UK, 2001. Springer-Verlag.
15. Dr. Latesh Malik Mr.Rushi Longadge, Ms. Snehlata S. Dongre. Class imbalance problem in data mining: Review. *INTERNATIONAL JOURNAL OF COMPUTER SCIENCE AND NETWORK*, 2(1), February 2013.
16. Felix Salfner and Miroslaw Malek. Using hidden semi-markov models for effective online failure prediction. *Reliable Distributed Systems, IEEE Symposium on*, 0:161–174, 2007.
17. Cristophe Brand Evgeniy Burnaev Pavel Erofeev Artem Papanov Stephane Alestra, Cristophe Bordry. Rare event prediction techniques in application to predictive maintenance of aircraft. In *Proceedings of ITaS conference*, 2014.
18. D.A. Tobon-Mejia, K. Medjaher, N. Zerhouni, and G. Tripot. Hidden markov models for failure diagnostic and prognostic. In *Prognostics and System Health Management Conference (PHM-Shenzhen), 2011*, pages 1–8, May 2011.