УДК 004.048

І.В. БАКЛАН, О.А. МОРОЗОВА, Є.А. НЕДАШКІВСЬКИЙ
Національний технічний університет України "Київський політехнічний інститут"

ПОРІВНЯННЯ ЛІНГВІСТИЧНИХ ЛАНЦЮЖКІВ ПРИ ПРОГНОЗУВАННІ ЧАСОВИХ РЯДІВ

У даній статті розглядається поетапний процес побудови лінгвістичної моделі для системи автентифікації користувача за рухами мишкою. Приведена математична постановка окремих етапів.

Ключові слова: лінгвістичне моделювання, лінгвістична модель.

И.В. БАКЛАН, А.А. МОРОЗОВА, Е.А. НЕДАШКОВСКИЙ
Национальний технический университет Украины "Киевский политехнический институт"

СРАВНЕНИЕ ЛИНГВИСТИЧЕСКИХ ЦЕПОЧЕК ПРИ ПРОГНОЗИРОВАНИИ ВРЕМЕННЫХ РЯДОВ

В данной статье рассматривается поэтапный процесс построения лингвистической модели для системы аутентификации пользователя по движениям мышкой. Приводится математическая постановка отдельных этапов.

Ключевые слова: лингвистическое моделирование, лингвистическая модель.

I.V. BAKLAN, O.A. MOROZOVA, Ye.A. NEDASHKOSKYI
National technical university of Ukraine "Kiyv polytechnical institute"

MATHEMATICAL STATEMENTS OF STEPS TO CONSTRUCT A LINGUISTIC MODEL

This paper considers step-by-step to constract linguistic model for user authentication system by movement of computer manipulator. The mathematical statement of each step is given.

Keywords:Linguistic modelling,Linguistic model.

Постановка проблеми

У роботі досліджується новий підхід для представлення часових рядів, застосування лінгвістичного моделювання для задачі прогнозування часових рядів.

Завданням лінгвістичного моделювання є перетворення часових рядів, експериментальних даних до лінгвістичних послідовностей та відновлення за ними формальної граматики мови. В основі лінгвістичного моделювання лежить лема існування ізоморфізму відтворення чисельних даних до лінгвістичних послідовностей, на основі яких може бути побудована мова. Як висновок існування унікальної мови, яка фактично уособлюється наборами чисельних даних.

Побудова лінгвістичних моделей, в тому числі, включає застосування інтервального підходу для розбиття множини значень числового ряду та подальшої побудови відображення множини. Для переходу від множини чисельних даних до символьного вигляду потрібно розбити цю множину на чисельні інтервали. Розбиття відбувається таким чином, щоб кількість елементів різницевих рядів початкового ряду в кожен інтервал потрапляла у відповідності до певного розподілу, тобто частота попадання елементів до інтервалу дорівнювала теоретичній імовірності. Розглядаються такі типи інтервалізації, як рівнозначні інтервали, логарифмічні інтервали, рівноймовірнісні інтервали, інтервали за певним розподілом (Пуассона, нормальним, бета-розподілом).

Завданням даного дослідження є виявлення оптимальних параметрів побудови алгоритму лінгвістичного моделювання та обґрунтування доцільності використання методу для прогнозування часових рядів.

Аналіз останніх досліджень та публікацій

Завданням лінгвістичного моделювання є перетворення часових рядів, експериментальних даних до лінгвістичних послідовностей та відновлення за ними формальної граматики мови. Це дозволяє вирішувати такі проблеми, як: прогнозування часових рядів, розпізнання образів [1]. Лінгвістичне моделювання – комплекс методів, методик та алгоритмів, які використовують процес перетворення числових масивів інформації до лінгвістичних послідовностей, на основі яких відновлюється формальна граматика [2]. Цей тип моделювання треба розглядати як специфічний вид математичного моделювання для обробки даних у символьному (не чисельному вигляді) [3]. Лінгвістичне моделювання базується на трьох основних підходах: структурний підхід та математична лінгвістика, інтервальні обчислення та робастні методи, сучасні методи імовірнісного моделювання. [4, 5]. Одним з методів прогнозу є структурний (синтаксичний) підхід. Він базується на деяких принципах розпізнавання образів, який складається з трьох основних частин – блока

попередньої обробки, блока опису об'єкта, блока синтаксичного аналізу [6]. Кожний попередньо опрацьований об'єкт сегментується на власні складові частини на підставі наперед визначених операцій та операцій об'єднання. В свою чергу кожна складова частина об'єкта ідентифікується за допомогою заданого набору зразків. Тепер кожний об'єкт подається набором зразків із визначеними синтаксичними операціями, тобто правилами їх поєднання [7].

Формулювання мети дослідження

Дана стаття має на меті розв'язати наступні завдання: виконати огляд відомих результатів з прогнозування часових рядів; виконати формалізацію задачі побудови граматики мови як основи лінгвістичної моделі; виконати порівняльний аналіз результатів на різних параметрах алгоритму.

Викладення основного матеріалу дослідження

На базі структурного підходу розроблено наступний метод створення лінгвістичної послідовності на базі числового ряду. Слід виконати наступні перетворення: порахувати різниці $\Delta y(i)=y(i)-y(i+1),\ i=\overline{1;N}$, між сусідніми значеннями ряду; виключити дублікати з отриманого ряду; відсортувати ряд за зростанням (чи спаданням); виокремити від'ємну ($a(k)$) та додатну ($b(k)$) послідовність зі значень $\Delta y(i)$.

Відомі різні способи опису мов. Кінцеву мову можна описати простим перерахуванням її ланцюжків. Оскільки формальна мова може бути і нескінченною, потрібні механізми, що дозволяють кінцевим чином представляти нескінченні мови. Можна виділити два основних підходи для такого уявлення: механізм розпізнання та механізм породження (генерації). Механізм розпізнавання є процедурою спеціального виду, яка за заданим ланцюжком визначає, чи належить він мові. Якщо належить, то процедура зупиняється з відповіддю "так", тобто допускає ланцюжок; інакше – зупиняється з відповіддю "ні", або зациклюється. Мова, яка визначається таким чином, є множиною всіх правильно побудованих ланцюжків. Основний спосіб реалізації механізму породження – використання породжуючих граматик, які іноді називають граматиками Хомського.

Алгоритм "дистанція Левенштейна" (Levenshtein distance), так само відома як редакційна відстань або дистанція редагування. Ця "дистанція" – це мінімальна кількість правок одного рядка (під правками маються на увазі три можливі операції: стирання символу, заміна символу і вставка символу), щоб перетворити його в інший. Наприклад: levenshtein ('ABC', 'ABC') = 0 levenshtein ('ABC', 'ABCDEF') = 3 levenshtein ('ABC', 'BCDE') = 3 levenshtein ('BCDE', 'ABCDEF') = 2. Для того, щоб отримати дистанцію Левенштейна між рядками ***s*** і ***t*** (довжиною ***m*** і ***n*** відповідно, індексація починається з нуля) і редакційний припис (які саме правки потрібно вносити), розраховується матриця відстаней *D* (розмірністю (*m* + 1) × (*n* + 1)). Кожний елемент матриці *D* [*i*, *j*] містить дистанцію між першими ***i*** символами рядка ***s*** і першими ***j*** символами рядка ***t***. Наприклад, матриця дистанцій Левенштейна для рядків ***s*** = '*ABC*' і ***t*** = '*ABF*':

		A	*B*	*F*
	0	1	2	3
A	1	0	1	2
B	2	1	0	1
C	3	2	1	1

Рис. 1. Матриця дистанцій Левенштейна

Стовпці відповідають підрядкам рядів ***t***, а рядки матриці – підрядкам ***s***. Рядок і стовпець з нульовим індексом відповідають порожнім підрядкам ***s*** і ***t***. Кожений елемент цієї матриці містить відстань між підрядками відповідних йому індексів. Наприклад, *D* [3,2] = 1 – це відстань між *ABC* і *AB* (всього одна правка – видалити *C*). Таким чином, *D* [3,3] = 1 – це і є шукана дистанція між *ABC* і *ABF*. (Заміна *C* на *F*). Крім дистанції ця матриця містить в собі інформацію про ті поправки, які необхідно внести в рядок ***s***, щоб отримати рядок ***t*** – редакційний припис. Побудова матриці дистанцій схожа на прокладання маршруту через лабіринт: починаючи з лівого верхнього кута матриці-карти, ми повинні потрапити в правий нижній кут. Частину матриці можна заповнити без обчислень: стовпець і рядок з нульовими індексами заповнюються числами по порядку, починаючи з нуля. Це просто пояснити тим, що для того, щоб з порожнього рядка отримати якийсь рядок *T* (довжиною ***k***), потрібно рівно ***k*** вставок – по одній на кожний символ. Аналогічно і в зворотний бік: для того, щоб з рядка ***s*** довжиною l отримати порожній рядок, потрібно рівно l вилучення. Таким чином, числа в нульовому рядку і колонці не залежить від вмісту порівнюваних рядків [49].

Для розв'язання даної задачі використовується рекурентне співвідношення:

$$Di.j=\{Di-1,j-1+0(equal,nonchange)\ Di-1,j-1(replace)\ Di-1,j(incert)\ Di,j-1(delete)$$

Найпростішою технікою оптимізації даного алгоритму є кешування розрахованих значень ***Di,j***. Для цього можна скористатись двовимірним масивом. В такому випадку складність алгоритму та вимоги по пам'яті складатимуть O(N^2) [8].

Задача пошуку та порівняння підрядків використовується в різних галузях науки. Наприклад, для пошуку рядків, що містять задані підрядки, для порівняння двох та більше рядків на подібність, для пошуку підрядків, що найчастіше зустрічаються у заданій послідовності, для пошуку інформативних елементів у послідовностях (наприклад, у послідовностях протеїнів та амінокислот).

Опис алгоритму розрахунку помилки тестового прогнозу за відстанню Левенштейна. Вхідні дані алгоритму: Слово для тестового періоду; Прогноз для тестового періоду;

Вихідні дані алгоритму: Значення відстані Левенштейна для вхідних строк.

Послідовність виконання алгоритму:

Крок 1:Нехай M та N – довжини вхідних рядків $S1$ та $S2$, тоді для всіх i від 0 до M та для всіх j від 0 до N підраховуємо $D(i, j)$, та повертаємо $D(M,N)$, де

$$D(i,j)=\{0;\ i=0,\ j=0i;\ j=0,\ i>0j;\ i=0,\ j>0D(i-1,j-1);\ S1[i]=S2[j]min(D(i,\ j-1)+insertcountD(i-1,\ j)+ +deletecountD(i-1,\ j-1)+replacecount\};\ j>0,\ i>0,\ S1[i]\neq S2[j].$$

Min(a,b,c) повертає найменший з аргументів. Ціну вставки, видалення та заміни вважатимемо однаковою та рівною одиниці.

Результати чисельних експериментів

Вхідні дані для експериментів були взяті з world data center (WDC): http://wdc.org.ua/. WDC було створено для архівації та поширення даних, зібраних з наглядових програм. WDC був створений у 1957-1958 роках і він нині включений до 52 центрів в 12 країнах. Представлений алгоритм призначений для оперативного прогнозу зміни тренду ряду, який відстежується. Однак ряду необхідна "історія", за якою формуються правила зміни тренду. Трендом називають невипадкову функцію, яка формується під дією загальних або довгострокових тенденцій, що впливають на часовий ряд. У деяких випадках алгоритм може не мати готового правила виводу, в цьому випадку потрібна експертна оцінка. Такі випадки можуть виникати при відсутності "історії" ряду, за якою формуються правила, та при великій кількості випадкових флуктацій, викидів чи структурних зсувів.

Проведемо експерименти на виборці часових рядів, оцінимо результати експериментів при різних значеннях параметрів алгоритм: різних розмірах вхідного алфавіту, на різних значеннях дельта та при різних розмірностях вхідного ряду.

Для автоматичного послідовного завантаження вхідних даних та запису результатів експериментів було використано програму xstarter.

Експерименти, при яких використовувалися правила вигляду лише "два елементи до одного" і "один до одного" показали низьку якість прогнозу. Даний варіант алгоритму не несе практичного значення і далі не розглядається.

Було проведено ряд експериментів на таких типах розподілу для подальшої інтервалізації, як: нормальний, Пуассона, Лапласа, бета-, Стьюдента. Програмою, створеною в рамках дослідження, не передбачена оцінка параметрів розподілу, тому було використано програмне забезпечення MatLab.

Експерименти з різними розмірностями вхідного файлу

Перевіримо якість прогнозу на різний кількості вхідних даних. За максимальний розмір файлу візьмемо 4000 значень часового ряду. Будемо поступово зменшувати розмір кожного ряду до 200 з кроком 200. В результаті проведення експерименту з параметрами дельта рівне 2 та розміром алфавіту рівним 26, було отримано дані, зображені на рис. 2–3 для стаціонарного часового ряду.

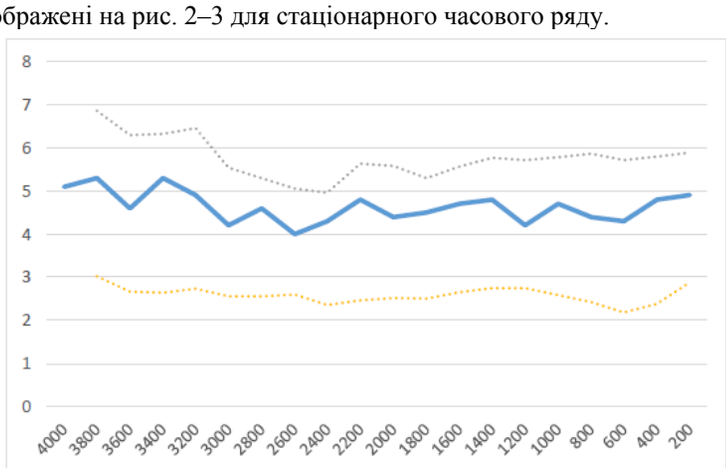

Рис. 2. Зміна кількості успішних прогнозів тренду для різної розмірності вхідного ряду.

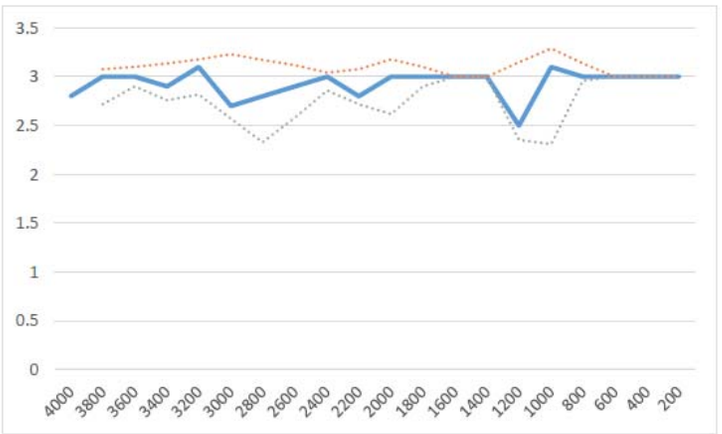

Рис. 3. Зміна кількості успішних прогнозів значень часового ряду для різної розмірності вхідного ряду.

Неперервною лінією позначимо усереднену кількість успішних прогнозів для тестової вибірки, штрихованими лініями – ковзне середнє відхилень від середнього значення прогнозу.

Ковзне середнє або рухоме середнє (процес ковзного (рухомого) середнього) — один із інструментів аналізу випадкових процесів та часових рядів, що полягає в обчисленні середнього підмножини значень. Ковзне середнє не є скаляром, а є випадковим процесом. Ковзне середнє може мати вагові коефіцієнти, наприклад, для посилення впливу новіших даних у порівнянні зі старішими. Найчастіше його використовують в аналізі часових рядів для згладжування раптових коливань та підкреслення довготермінових трендів або циклів. З математичної точки зору, ковзне середнє є різновидом згортки та схоже на фільтр низьких частот в обробці сигналів [9].

Як можна бачити з графіків, в середньому алгоритм прогнозує зміну тренду на 4-5 кроків вперед, а прогноз наступного символу в середньому є точним на три кроки вперед.

Перед виконанням алгоритму трансформуємо початковий ряд. Видалимо тренд за допомогою бібліотеки scikit-learn для Python. Щоб видалити тренд, побудуємо лінійну регресію від часу по кожному показнику та вирахуємо з початкових даних.

Лінійні тренди можуть бути просумовані лінійною моделлю, а нелінійні тренди краще за все, можуть бути просумованими з використанням поліноміального чи іншого методу апроксимації кривої. Прогнози цієї моделі становлять пряму лінію, яка може бути прийнята за лінію тренду для набору даних. Ці прогнози можуть бути вирахувані з початкового ряду, щоб надати версію набору даних без тренду. Відхилення від тренду являють собою детерміновану форму набору даних. Можливе також використання поліноміальної кривої та інших нелінійних моделей. Використаємо інструмент LinearRegression бібліотеки scikit-learn для обробки даних. Можемо бачити, що цей підхід ефективно відфільтровує набір даних.

Повторимо експеримент на різних розмірностях вхідного ряду для отриманих рядів без тренду. На рис. 4-5 показано отримані результати.

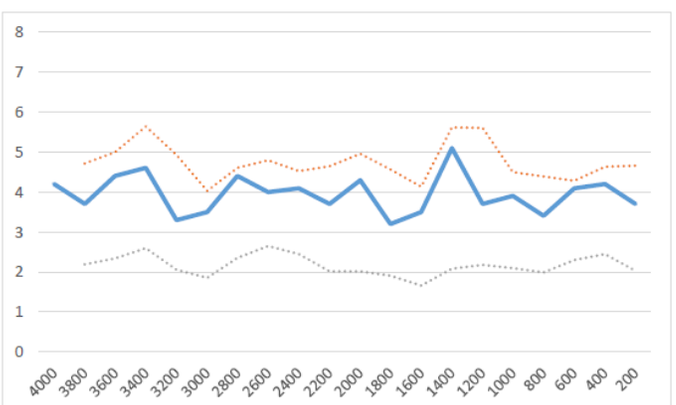

Рис. 4. Зміна кількості успішних прогнозів тренду для початкового ряду позбавленого тренду.

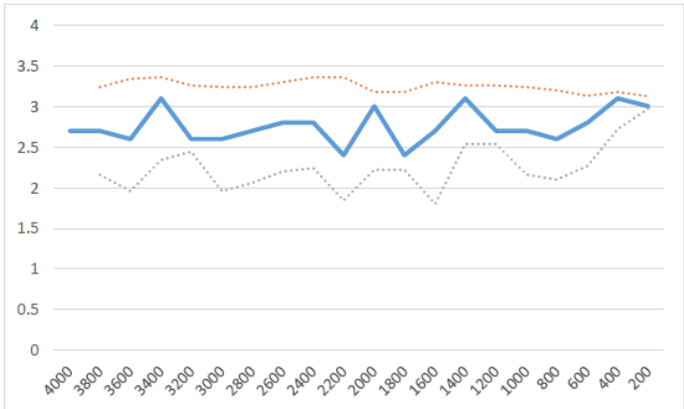

Рис. 5. Зміна кількості успішних прогнозів значень часового ряду для початкового ряду позбавленого тренду.

Як можна бачити, в середньому прогноз зміни тренду і прогноз наступного символу в послідовності в середньому майже не змінився.

Висновки

Авторами статті проведено ряд експериментів над часовими рядами з різними параметрами алгоритму. Було виявлено оптимальні параметри. Також експерименти показали, що алгоритм досить стабільно виконує оперативний прогноз значень ряду на 3-4 кроки вперед та прогноз зміни тренду на 4-5 кроків вперед. Було проаналізовано швидкодію алгоритму та виявлено, що алгоритм працює за лінійний час та має доволі низький час виконання – приблизно 60 мілісекунд для вхідного ряду у 200 пар символів, що у 7,5 разів швидше за алгоритм авторегресії другого порядку з ковзним середнім першого порядку від фірми Extreme Optimization. Ці результати доводять, що алгоритм є ефективним для виконання оперативних прогнозів.

Список використаної літератури

1. Vaseghi Saeed V. Advanced digital signal processing and noise reduction / Saeed V. Vaseghi. – 3ed. – Chichester: TLFeBook, 2006. – 454 p.
2. Fraser A.M. Hidden Markov models and dynamic systems / Andrew M. Fraser. – Philadelphia: Society for Industrial and Applied Mathematics. 2008. – 145 p.
3. Баклан І.В. Лингвистическое моделирование: основы, методы, некоторые прикладные аспекты //Системные технологии. –Днепропетровск, 2011. – Вып. 3 (74). – С. 10-19.
4. Fu K.S. Syntactic Methods in Pattern Recognition / K.S. Fu. – N.J.: Academic Press, 1974. – 306 c.
5. Fu K.S. Syntactic Pattern Recognition and Application / K.S. Fu. – N.J.: Prentice-Hall, Inc. Englewood Cliffs., 1982. – 596 p.
6. Елинек Ф. Распознавание непрерывной речи статическими методами / Ф. Елинек. – М.: ТИИЭР, 1976. – Т. 64. – №4. – С. 131-160.
7. Баклан І.В. Структурний підхід до розпізнання образів у системах безпеки / І.В. Баклан, Ю.М. Селін, О.О. Петренко // Міжнародна науково-практична конференція "Національна безпека України: стан, кризові явища та шляхи їх подолання" (Київ, 7-8 грудня 2005 р.). Збірка наукових праць. – К.: Національна академія управління – Центр перспективних соціальних досліджень, 2005. – С.375-380.
8. Відстань Левенштейна [Електронний ресурс] // Happy programmer. – 2012. – Режим доступу до ресурсу: http://happy-programmer.blogspot.com/2012/10/blog-post.html.
9. Рухоме середнє [Електронний ресурс] // msd.in.ua. – 2017. – Режим доступу до ресурсу: https://msd.in.ua/kilkisni-statistichni-metodi-prognozuvannya.