2. Для аналізу великих даних потрібні потужні сервери та ефективні алгоритми оптимізації.

3. Якщо платформа пропонує широкий асортимент товарів (тисячі чи навіть мільйони позицій), виникає складність у створенні релевантних рекомендацій для кожного клієнта.

Отже, системи рекомендацій продовжують розвиватися. Наприклад, сучасні дослідження зосереджуються на інтеграції нейронних мереж та моделей глибокого навчання. Алгоритми на основі рекурентних нейронних мереж (RNN) або трансформерів дають змогу прогнозувати вподобання клієнтів з урахуванням контексту та часу. До того ж активно розробляються пояснювані рекомендаційні системи (explainable AI), які пояснюють клієнту, чому той чи інший товар йому пропонується.

Можна зауважити, що системи рекомендацій є невід'ємною частиною сучасних онлайн-магазинів, які прагнуть персоналізувати взаємодію з клієнтами та збільшити продажі. Попри наявні виклики, розвиток технологій обробки даних і машинного навчання робить ці системи дедалі ефективнішими. Майбутнє цих технологій відкриває нові горизонти для бізнесу, покращуючи досвід споживачів і сприяючи росту компаній.

Список використаних джерел

1. Дослідження методів побудови рекомендаційних систем в мережі Інтернет / Є. В. Мелешко та ін. *Збірник наукових праць «Системи управління, навігації та зв'язку».* Полтавський національний технічний університет ім. Ю. Кондратюка. 2018. Т. 1, № 47. С. 131–136.

2. Лобур М. В., Шварц М. Є., Стех Ю. В. Моделі і методи прогнозування рекомендацій для колаборативних рекомендаційних систем. *Вісник НУ «Львівська політехніка». Серія: Інформаційні системи та мережі.* 2018. № 901. Львів: Видавництво Львівської політехніки. С. 68–75.

3. Моделі формування рекомендацій у інтелектуальних системах електронної комерції / О. Ю. Чередніченко, О. В. Янголенко, О. В. Іващенко, О. М. Матвєєв. *Системи обробки інформації*. 2020. № 1(160). С. 32–39.

4. Клюйко Г. Товарні рекомендації для ecommerce на сайті та в листах. 2021. URL: https://esputnik.com/uk/blog/tovarni-rekomendaciyi-dlya-ecommerce-na-sajti-ta-v-listah

УДК 004.422.5

Мигун Р. С., здобувач вищої освіти,
Потапова Н. А., канд. екон. наук, доцент,
доцент кафедри інформаційних технологій

ЗАСТОСУВАННЯ АЛГОРИТМУ ПОБУДОВАНОЇ ЦИФРОВОЇ ПІДПИСКИ З ВИКОРИСТАННЯМ ЕЛІПТИЧНИХ КРИВИХ (ECDSA) У БЛОКЧЕЙН-ТЕХНОЛОГІЯХ

Донецький національний університет імені Василя Стуса, м. Вінниця

ECDSA (абревіатура від *Elliptic Curves Digital Signature Algorithm*, алгоритм побудованої цифрової підписки з використанням еліптичних кривих) –

це схема криптографії на основі еліптичних кривих (Elliptic Curve Cryptography або ECC).

ECDSA використовується в багатьох системах безпеки, зокрема для захисту повідомлень і в основі безпеки Bitcoin. До того ж він застосовується для забезпечення безпеки транспортного рівня (TLS), що замінив SSL, шляхом шифрування з'єднань між веббраузерами та вебпрограмами. HTTPS-з'єднання, позначене іконкою замка у браузері, також захищається за допомогою підписаних сертифікатів через ECDSA.

Еліптична криптографія (ECC) має суттєву перевагу перед RSA завдяки меншому розміру ключів за умови однакового рівня безпеки. Наприклад, ECC з ключем 256 біт забезпечує таку ж криптографічну стійкість, як RSA з ключем 3 072 біт. Це робить ECC більш ефективною з погляду використання пам'яті, обчислювальних ресурсів і пропускної здатності, що є критично важливим для блокчейн-екосистем.

Ключовими компонентами ECDSA є:

1) Генерація пари ключів: це передбачає вибір випадкового закритого ключа, який є секретним числом, і використання його для отримання відкритого ключа, яким можна поділитися публічно. Відкритий ключ виходить шляхом множення фіксованої точки на еліптичній кривій на закритий ключ.

2) Хешування повідомлення: повідомлення, яке потрібно підписати, хешується за допомогою криптографічної хеш-функції, яка створює хеш-значення фіксованої довжини.

3) Підписання повідомлення. Цифровий підпис ECDSA створюється шляхом виконання ряду математичних операцій над хеш-значенням і закритим ключем. Результатом цих операцій є унікальний підпис, який може бути згенерований лише закритим ключем відправника.

4) Перевірка підпису: одержувач повідомлення використовує відкритий ключ для перевірки підпису. Якщо підпис дійсний, одержувач може бути впевнений, що повідомлення надіслано власником закритого ключа.

5) Параметри домену: еліптична крива E, визначена в дискретному просторі Fq з характеристикою p і параметрами домену базової точки G, може бути розподілена групою об'єктів або унікальною для одного користувача.

Безпека ECDSA залежить від складності розв'язання проблеми дискретного логарифмування еліптичної кривої, яка передбачає пошук закритого ключа за відкритим ключем і точкою генератора. Вважається, що проблема дискретного логарифмування еліптичної кривої обчислювально неможлива для достатньо великих розмірів ключів, що робить ECDSA надійним алгоритмом цифрового підпису.

Під час роботи автори вирішили реалізувати класи PrivateKey та PublicKey на мові програмування C#. У випадку з PrivateKey – ми маємо клас, який містить у собі криву та секретне число і може підписувати повідомлення:

```
public class PrivateKey
{
    private Curve Curve { get; }
    private BigInteger Secret { get; }
```

```
    public PrivateKey(Curve curve, BigInteger secret)
    {
        this.Curve = curve;
        this.Secret = secret;
    }

    public PublicKey PublicKey()
    {
        var publicPoint = MathHelper.Multiply(this.Curve.G, this.Secret, this.Curve.N, this.Curve.A, this.Curve.P);
        return new PublicKey(publicPoint, this.Curve);
    }

    public Signature Sign(string message)
    {
        var messageBytes = Encoding.UTF8.GetBytes(message);
        var hashBytes = SHA256.HashData(messageBytes);
        var numberMessage = new BigInteger(hashBytes.Reverse().ToArray());
        var curve = this.Curve;

        BigInteger r;
        BigInteger s;
        Point randSignPoint;
        do
        {
            var randNum = BigIntegerRandom.Between(1, curve.N - 1);
            randSignPoint = MathHelper.Multiply(curve.G, randNum, curve.N, curve.A, curve.P);
            r = randSignPoint.X % curve.N;
            s = ((numberMessage + r * this.Secret) * (MathHelper.Inv(randNum, curve.N))) % curve.N;
        } while (r == 0 || s == 0);

        int recoveryId = (int)(randSignPoint.Y & 1);
        if (randSignPoint.Y > curve.N)
        {
            recoveryId += 2;
        }

        return new Signature(r, s, recoveryId);
    }
}
```

Тепер перейдемо до класу PublicKey, який є парою до PrivateKey і може перевіряти, чи було підписано це повідомлення конкретно цим приватним ключем.

```
public class PublicKey
{
    public Point Point { get; }
    public Curve Curve { get; }

    public PublicKey(Point point, Curve curve)
    {
        this.Point = point;
        this.Curve = curve;
    }

   public bool Verify(string message, Signature signature)
    {
        var messageBytes = Encoding.UTF8.GetBytes(message);
        var hashBytes = SHA256.HashData(messageBytes);
        var numberMessage = new BigInteger(hashBytes.Reverse().ToArray());
        var curve = this.Curve;
```

```
        if (signature.R < 1 || signature.R > curve.N - 1)
        {
            return false;
        }

        if (signature.S < 1 || signature.S > curve.N - 1)
        {
            return false;
        }

        var w = MathHelper.Inv(signature.S, curve.N);
        var u1 = (numberMessage * w) % curve.N;
        var u2 = (signature.R * w) % curve.N;
        var point = MathHelper.Add(MathHelper.Multiply(curve.G, u1, curve.N, curve.A, curve.P),
MathHelper.Multiply(this.Point, u2, curve.N, curve.A, curve.P), curve.A, curve.P);

        if (point.X == 0 && point.Y == 0)
        {
            return false;
        }

        return point.X % curve.N == signature.R;
    }
}
```

Отже, ECDSA забезпечує високу криптографічну безпеку для блокчейн-транзакцій, використовуючи еліптичні криві. Реалізація класів PrivateKey і PublicKey на C# дає змогу ефективно підписувати та перевіряти повідомлення, що є основою для безпеки у блокчейн-мережах.

Список використаних джерел

1. Elliptic Curve Digital Signature Algorithm. *HYPR.* URL: https://surl.li/lxxejp (дата звернення: 20.12.2024).

2. Raza S. Ethereum’s Elliptic Curve Digital Signature Algorithm (ECDSA). *Medium.* 2023. URL: https://fitsaleem.medium.com/ethereums-elliptic-curve-digital-signature-algorithm-ecdsa-88e1659f4879

УДК 004.415.28

Перепелиця А. С., здобувач вищої освіти,
Бабаков Р. М., д-р техн. наук, професор
кафедри інформаційних технологій

ІНФОРМАЦІЙНА ПЛАТФОРМА ДЛЯ АНАЛІЗУ НАВЧАННЯ ТА КАР’ЄРНИХ МОЖЛИВОСТЕЙ

Донецький національний університет імені Василя Стуса, м. Вінниця

Проблема вибору майбутньої професійної траєкторії є актуальною для старшокласників, які часто стикаються з невизначеністю, зумовленою недостатньою поінформованістю про їхні здібності, інтереси та можливості на ринку праці [1, 2]. Для розв’язання цієї проблеми було розроблено інформаційну