ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «Алгоритм вивчення іноземних слів на основі антропоцентричних обчислень та методів обробки природної мови»

на здобуття освітнього ступеня магістра
зі спеціальності 121 Інженерія програмного забезпечення
освітньо-професійної програми «Інженерія програмного забезпечення»

Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

_____ Андрій СОКОЛОВСЬКИЙ
(підпис)

Виконав: здобувач вищої освіти групи ПДМ-61
Андрій СОКОЛОВСЬКИЙ

Керівник: Оксана ЗОЛОТУХІНА
канд. техн. наук., доц.

Рецензент: _____

Київ 2026

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

Навчально-науковий інститут інформаційних технологій

Кафедра Інженерії програмного забезпечення

Ступінь вищої освіти Магістр

Спеціальність 121 Інженерія програмного забезпечення

Освітньо-професійна програма «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ
Завідувач кафедри
Інженерії програмного забезпечення
_____ Ірина ЗАМРІЙ
«_____» _____ 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Соколовському Андрію Вадимовичу

1. Тема кваліфікаційної роботи: «Алгоритм вивчення іноземних слів на основі антропоцентричних обчислень та методів обробки природної мови»

керівник кваліфікаційної роботи Оксана ЗОЛОТУХІНА, канд. техн. наук, доцент, доцент кафедри ІПЗ

затверджені наказом Державного університету інформаційно-комунікаційних технологій від «30» жовтня 2025 р. №467.

2. Строк подання кваліфікаційної роботи «19» грудня 2025 р.

3. Вихідні дані до кваліфікаційної роботи: науково-технічна література, метод TF-IDF, методи моделювання когнітивної складності слів, вимоги до ефективності засвоєння лексики користувачем.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Дослідження методів обробки природної мови для відбору та аналізу лексичних одиниць.

2. Аналіз когнітивних та антропоцентричних моделей складності засвоєння іноземної лексики.
3. Розробка та валідація алгоритму відбору та повторення слів на основі методів обробки природної мови та когнітивного моделювання.

5. Перелік ілюстративного матеріалу: *презентація*
1. Використані методи алгоритму.
2. Схема алгоритму оптимізації вивчення слів.
3. Модель визначення пріорітетності слова.
4. Практичний результат.
5. Демонстрація роботи застосунку.
6. Порівняльний аналіз алгоритму з базовим TF-IDF.

6. Дата видачі завдання «31» жовтня 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз наявної науково-технічної літератури	31.10 - 03.11.2025	
2	Ознайомлення з науковою літературою та методами обробки природної мови для відбору лексики	03.11 - 08.11.2025	
3	Дослідження когнітивних та антропоцентричних моделей складності слів	08.11 - 13.11.2025	
4	Аналіз впливу когнітивних факторів на ефективність відбору слів	13.11 - 22.11.2025	
5	Вивчення технологій NLP та алгоритмів розрахунку пріоритету слів	22.11 - 25.11.2025	
6	Застосування алгоритму відбору та повторення слів у системі адаптивного навчання	25.11 - 27.11.2025	
7	Оформлення роботи: вступ, висновки, реферат	27.11 - 03.12.2025	
8	Розробка демонстраційних матеріалів	03.12 - 19.12.2025	

Здобувач вищої освіти _____ Андрій СОКОЛОВСЬКИЙ
(підпис)

Керівник
кваліфікаційної роботи _____ Оксана ЗОЛОТУХІНА
(підпис)

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня магістра: 60 стор., 2 табл., 11 рис., 25 джерел.

Мета роботи – оптимізація процесу вивчення іноземних слів за рахунок використання методів обробки природної мови та антропоцентричних обчислень.

Об'єкт дослідження – процес вивчення іноземних слів.

Предмет дослідження – технології оптимізації вивчення іноземних слів на основі методів обробки природної мови та антропоцентричних обчислень.

У роботі використано різноманітні методи, такі як обробка природної мови, статистичний аналіз текстів, когнітивне моделювання складності слів та алгоритми адаптивного повторення. Основна увага приділена інтеграції NLP-технологій із моделями когнітивної складності для автоматичного визначення значущих слів у текстах користувача та формування ефективних наборів для повторення.

Проведено аналіз сучасних методів навчання словникового запасу іноземних мов. Розглянуто традиційні системи повторення та підходи, що оцінюють інформативність слів у тексті, частотність та когнітивні характеристики. Особливу увагу приділено показникам довжини слова, кількості складів, морфологічній нерегулярності та семантичній конкретності, які використовуються для обчислення складності кожного слова. Досліджено способи комбінування статистичних метрик, таких як TF-IDF, із когнітивними факторами для визначення пріоритетності слів у навчальному процесі.

Розроблено та оцінено алгоритм статичного відбору слів та формування наборів для повторення. Кожне слово отримує оцінку пріоритету на основі поєднання інформативності, складності та частотності у корпусі користувача. Алгоритм дозволяє відсіювати менш значущі слова і формувати компактні набори для ефективного засвоєння. Для підготовки навчальних карток використано токенізацію, лематизацію та визначення частини мови, а також добір прикладів із

тексту користувача та перекладів.

Реалізовано програмну систему на основі інтеграції Python та Electron. Python забезпечує обробку текстів, розрахунок складності та формування наборів слів, а фронтенд на React надає зручний інтерфейс для введення текстів, перегляду результатів аналізу та проведення сесій повторення. Система дозволяє користувачу переглядати картки зі словником, прикладами речень та оцінками складності, а також контролювати включення чи виключення слів із навчального набору.

Проведено експерименти для перевірки точності визначення значущих слів та правильності підбору прикладів. Встановлено, що статичний підхід ефективно виділяє важливі слова та забезпечує користувача збалансованими навчальними наборами. Результати демонструють практичну доцільність використання комбінації NLP та когнітивного моделювання для оптимізації процесу вивчення іноземної лексики.

КЛЮЧОВІ СЛОВА: ОБРОБКА ПРИРОДНОЇ МОВИ, КОГНІТИВНА СКЛАДНІСТЬ, АДАПТИВНЕ ПОВТОРЕННЯ, АНАЛІЗ ТЕКСТУ, ЛЕКСИЧНИЙ ВІДБІР.

ABSTRACT

Text part of the master's qualification work: 60 pages, 2 tables, 11 pictures, 25 sources.

The purpose of the work is to optimize the process of learning foreign words by using methods of natural language processing and anthropocentric computing.

Object of research – the process of learning foreign words.

Subject of research – technologies for optimizing the learning of foreign words based on natural language processing methods and anthropocentric computing.

Summary of the work:

The work employs a variety of methods, including natural language processing, statistical text analysis, cognitive modeling of word complexity, and adaptive repetition algorithms. The primary focus is the integration of NLP technologies with cognitive complexity models to automatically identify significant words in user-provided texts and generate effective repetition sets.

An analysis of modern vocabulary acquisition methods for foreign languages has been conducted. Traditional repetition systems and approaches for evaluating word informativeness, frequency, and cognitive characteristics are examined. Particular attention is given to indicators such as word length, number of syllables, morphological irregularity, and semantic concreteness, which serve as the basis for calculating word complexity. Methods for combining statistical metrics such as TF-IDF with cognitive factors to determine word prioritization in the learning process are explored.

A static algorithm for word selection and formation of repetition sets has been developed and optimized. Each word receives a priority score based on a combination of informativeness, complexity, and frequency in the user's corpus. The algorithm filters out less important words and produces compact sets for efficient memorization. Tokenization, lemmatization, part-of-speech tagging, as well as extraction of example sentences from the user's text and translations, are used to prepare learning cards.

A software system has been implemented using Python and Electron. Python handles text processing, complexity calculation, and word-set generation, while a React-based frontend provides a user-friendly interface for inputting texts, viewing analysis results, and conducting repetition sessions. The system enables users to browse vocabulary cards containing example sentences and complexity metrics, as well as manage the inclusion or exclusion of words from the learning set.

Experiments were conducted to evaluate the accuracy of identifying significant words and selecting appropriate examples. Findings indicate that the static approach effectively highlights important vocabulary and offers users balanced study sets. The results demonstrate the practical value of combining NLP and cognitive modeling for optimizing foreign vocabulary learning.

KEYWORDS: NATURAL LANGUAGE PROCESSING, COGNITIVE COMPLEXITY, ADAPTIVE REPETITION, TEXT ANALYSIS, LEXICAL SELECTION.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

TF-IDF - частота терміну, обернена частота документу (term frequency, inverse document frequency)

NLP - обробка природної мови (natural language processing)

TOEIC - тест з англійської для міжнародної комунікації (Test of English for International Communication)

UI - інтерфейс користувача (user interface)

UX - досвід користувача (user experience)

SRS - система інтервальних повторень (spaced repetition system)

СУБД - система управління базою даних

JSON - запис об’єктів джаваскріпт (javascript object notation)

CEFR - Загальноєвропейські Рекомендації з мовної освіти (Common European Framework of Reference for Languages)

ВСТУП

Серед сучасних підходів до вивчення іноземних мов одним із ключових елементів є системи інтервального повторення, що значно підвищують ефективність запам’ятовування. Проте у практичному використанні виникає суттєва проблема: більшість користувачів механічно додають до SRS усі незнайомі слова, незалежно від їхньої складності, частотності або реальної навчальної цінності. Це призводить до надмірного накопичення карток, перевантаження пам’яті та падіння мотивації. Як наслідок, значна частина навчального часу витрачається не на важливі або складні лексеми, а на слова, які користувач може засвоїти природно, під час звичайного читання чи перегляду контенту (занурення в мову).

Тому актуальним постає завдання побудувати алгоритмічний механізм, що здатний автоматично визначати реальну вагу слова для вивчення: чи є воно рідкісним, чи складним, чи формально важким, чи важливим у контексті поточного корпусу. Використання інструментів аналізу природної мови дає змогу враховувати морфологічні особливості, частотні характеристики, контекстуальну значущість та структурні властивості тексту. Антропоцентричні підходи забезпечують визначення конгітивної складності слів. У поєднанні із SRS такі методи дозволяють розробити систему, яка не перевантажує користувача, а навпаки — відфільтровує лексичний шум і фокусує увагу на словах, що дійсно потребують окремого опрацювання.

Мета роботи – оптимізація процесу вивчення іноземних слів за рахунок використання методів обробки природної мови та антропоцентричних обчислень.

Об’єкт дослідження – процес вивчення іноземних слів.

Предмет дослідження – технології оптимізації вивчення іноземних слів на основі методів обробки природної мови та антропоцентричних обчислень.

У межах цієї дипломної роботи було виконано низку ключових завдань, що забезпечили побудову та перевірку розробленого алгоритму:

1. Аналіз предметної галузі та літератури — опрацьовано наукові праці щодо статистичних моделей ваги лексики, когнітивних аспектів засвоєння лексики, NLP-методів та сучасних підходів до лексичного ранжування.
2. Аналіз засобів реалізації алгоритму — проведено огляд технологій для лінгвістичної обробки тексту, інструментів побудови корпусів, методів обчислення TF-IDF, а також способів інтеграції додаткових метрик, включно з розробленою позиційною метрикою LPS.
3. Розробка алгоритму — створено модель, що поєднує модифікований TF-IDF, LPS та об’єднану метрику важливості, здатну ранжувати слова за сумарною пріорітетністю для навчання.
4. Оцінка роботи алгоритму — виконано порівняння з базовими підходами (частотний аналіз і класичний TF-IDF), проведено тест стабільності результатів та перевірено відповідність оцінок рівням складності CEFR.

Завдяки такій архітектурі розроблений підхід має практичну цінність для освітніх платформ, систем персоналізованого навчання, рекомендаційних моделей, адаптивних лексичних тренажерів та інструментів фільтрації текстів за рівнем складності. Він може значно зменшити когнітивне навантаження на користувача, підвищити ефективність опрацювання словника та зробити процес вивчення мови більш природним, керованим і результативним.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

1.1. Аналіз сучасних підходів до вивчення іноземних слів

Вивчення лексики іноземної мови - одна з ключових складових компетенції володіння мовою. Незалежно від стеку технологій чи методики, успішне засвоєння нових лексичних одиниць означає не лише їх запам’ятовування, але й здатність активного використання у мовленні чи читанні. У сучасній педагогіці і лінгводидактиці виокремлюють кілька підходів, які варто проаналізувати перед розробкою власної системи.

Традиційні методи.

До класичних методів належать навчання за списками слів, перекладна практика, мнемонічні методики (наприклад, метод ключового слова та активне використання карток). Наприклад, метод ключового слова передбачає створення асоціації між звучанням слова та образом або значенням, що значно покращує запам’ятовування.

Також у дистанційних чи мультимовних контекстах дослідження показують, що учителі активно застосовують такі стратегії як пов’язування з особистим досвідом учня, використання синонімів/антонімів та читання вголос.

Недоліком цих методів є: переважна орієнтація на ізольовані слова без контексту, малий рівень адаптації під індивідуальні когнітивні особливості учня, відсутність аналітики щодо ефективності запам’ятовування та закріплення.

Метод інтервального повторення.

Уже декілька десятиліть методика інтервального повторення (SRS) є наріжним каменем ефективного вивчення лексики. Суть полягає в тому, що словникові одиниці повторюються через певні інтервали, які з часом збільшуються — таким чином використовують криву забування та практику відновлення. Наприклад, дослідження показують, що при розподілених інтервалах

забезпечується краща довготривала пам’ять, ніж при масованому (без пауз) вивченні.

Більш конкретно, експеримент із саудівськими студентами показав, що група зі інструкціями по розподілу демонструвала статистично значуще кращі результати як на тесті, так і на відстроченому тесті, порівняно з групою масованого навчання. [1]

Інші дослідження, наприклад серед університетських ESL-учнів в Україні, підтверджують, що повторення словникових одиниць через інцидентний аудіо-контент значно підвищує утримання нових слів. [2]

Водночас, хоча SRS і має сильні докази, він також має обмеження: багато систем не враховують контексту використання слова, семантичні зв’язки або індивідуальну когнітивну складність слова.

Сучасні технологічні підходи: цифрові інструменти та контекстне навчання

У цифрову епоху дедалі більше ресурсів пропонують використання мобільних застосунків, веб-платформ, адаптивних систем, які автоматизують повторення словникових одиниць. Наприклад, дослідження серед китайських студентів EFL виявили, що використання мобільного застосунку для словникової роботи значно підвищує мотивацію та утримання, коли враховано фактори сприйняття користувачем (усвідомлена корисність, легкість використання).

Також активно досліджується поєднання методу SRS з інтерливінговими (перемішаними) завданнями, коли слова вивчаються у зміненому порядку чи контексті, що покращує глибше засвоєння.

Ці підходи підкреслюють значущість контексту, цифрової доступності та інтерактивності у сучасному вивченні лексики.

Перехід до більш персоналізованих підходів.

Останнім часом увага дослідників зосереджена на тому, як зробити навчання лексики більш персоналізованим: тобто враховувати когнітивні, поведінкові, мотиваційні характеристики конкретного учня. Цей підхід тісно збігається з поняттям «антропоцентричного» навчання — навчання, орієнтоване на людину як суб’єкта, а не лише на технологію чи загальну методику.

У цьому ключі важливо враховувати такі чинники: частоту вживання слова, морфологічну складність, абстрактність чи конкретність поняття, попередню успішність учня, індивідуальні інтервали повторення. При цьому, наукові роботи підкреслюють: навіть потужні SRS-системи мають обмеження, якщо вони не враховують такі фактори.

Оскільки лексика іноземної мови часто включає сотні чи тисячі одиниць, персоналізована адаптація має стати пріоритетом: не всі слова однаково складні для кожного учня, і алгоритм, який враховує цю відмінність, може значно підвищити ефективність навчання.

Обмеження сучасних підходів.

Незважаючи на те, що сучасні методи значно просунулися, вони мають низку обмежень, які слід враховувати при розробці нових систем:

- Часто слова вивчаються поза контекстом, що знижує їх активне використання.
- Алгоритми часто не враховують семантичні зв’язки та рідкість слова, що може вплинути на запам’ятовування.
- Системи не завжди адаптуються до індивідуальних когнітивних профілів учнів (наприклад, користувач із високим рівнем навичок має інші потреби, ніж новачок).
- Багато платформ орієнтовані лише на повторення, але не на відбір слова — тобто не на те, чи слово має велику ймовірність бути корисним чи часто вживаним в їхньому реальному житті.
- Дослідження зосереджені переважно на кількісному проникненні (скільки слів вивчено), але менше – на їхній активній, продуктивній використанні.

Ці обмеження створюють поле для подальших досліджень і технологічних рішень — зокрема, саме тому виникає потреба в алгоритмах, які поєднують відбір лексики, ранжування за складністю та адаптивне повторення.

1.2. Огляд систем інтервального повторення (SRS) та їх обмеження

У сучасному мовному навчанні системи інтервального повторення (англ. *Spaced Repetition Systems*, SRS) стали однією з ключових технологій для засвоєння великої кількості лексичних одиниць. Цей підрозділ має на меті надати ґрунтовний огляд принципів SRS, їхньої теоретичної бази, практичного застосування у навчанні іноземних слів, а також — виявити їхні суттєві обмеження. Це дозволить сформулювати політики та критерії, які будуть враховані при розробці нашої системи відбору та ранжування лексики.

Теоретичні основи SRS.

Ефективність SRS спирається на класичні дослідження в галузі пам’яті й навчання. Герман Еббінгауз ще в 1885 році довів, що без повторення більшість нової інформації втрачається впродовж короткого часу — це так звана «крива забування».

Метод інтервального повторення передбачає, що навчальні одиниці (наприклад, нові слова) повторюються через поступово зростаючі інтервали часу — таким чином, коли пам’ять про слово стає слабшою, система викликає його повторення для зміцнення.

Згідно популярному опису, «SRS – це коли слово повторюється лише тоді, коли шанс забути його вже високий», що дозволяє не витрачати час на речі, які учень вже добре запам’ятав.

У психолінгвістичному контексті це означає: замість масового «зубріння» словників, ефективнішою є стратегія, коли відбір і повторення слів координуються із кривою пам’яті користувача.

Практичне застосування SRS у вивченні іноземної лексики.

Багато освітніх платформ і застосунків ввели SRS як базову частину функціональності (наприклад, Anki). У контексті навчання лексики показано, що систематичне використання SRS-інструментів веде до статистично значущого підвищення утримання слів. Так, дослідження серед японських студентів, які вивчали слова за списком TOEIC із SRS-підтримкою, виявило кореляцію між

кількістю повторень у SRS і приростом результатів тесту TOEIC: загальні балли зросли значущо ($p < .05$). [3]

Ще одне дослідження серед п'ятикласників Кувейту показало, що поєднання методів інтервального повторення та практики відтворення (retrieval practice) дає суттєво кращі результати за звичайне навчання.

Впровадження SRS системи в класах покращує збереження словникового запасу та дає додаткові відомості щодо мотивації та використання.

Таким чином, SRS виглядає як ефективний інструмент для підтримки лексичного навчання, особливо коли мова йде про великі обсяги слів і мету — довготривале утримання знання.

Основні алгоритмічні моделі та системи.

Найбільш відомими системами інтервального повторення є ті, що базуються на алгоритмі SuperMemo (SM-2) та його подальших розробках [4]. Наприклад, дослідження «Unbounded Human Learning: Optimal Scheduling for Spaced Repetition» пропонує стохастичну модель для SRS і оптимізацію введення нових карток у чергу.

У практичних застосунках флеш-карток система визначає показник успіху (наприклад, чи нове слово було правильно відтворене), на основі цього здійснюється переміщення картки у «довшу чергу» повторення, або повернення до коротшого інтервалу. Наприклад, алгоритм SM-2 використовує оцінку (рейтинг) для кожної карти та коригує інтервал повторення. Більш сучасні системи включають моделі машинного навчання, які аналізують історію повторень великої кількості користувачів і на їх основі прогнозують оптимальний час повторення [4].

Існують також прості правила, які часто застосовують: якщо користувач правильно відтворив слово тричі поспіль — картка переходить у довгу чергу, і наступний перегляд відбувається через тижні чи місяці; якщо допущено помилку — інтервал скорочується. Ці системи дозволяють автоматизувати процес повторення та зменшити навантаження на користувача, водночас збільшуючи продуктивність запам’ятовування.

Обмеження систем інтервального повторення.

Хоча SRS має значні переваги, слід виділити ряд важливих обмежень і викликів, які варто мати на увазі при впровадженні таких систем.

1. Обмежена глибина засвоєння та контекстуалізація

Одним з найчастіших зауважень є те, що SRS добре працює для розпізнавання або впізнавання слів, але менш ефективний для активного використання лексики у контексті. Студенти, які навчались із SRS, мали хороше пізнання нових слів, але слабшу здатність використовувати їх у реальному мовленні.

Тобто SRS має бути лише частиною комплексного підходу до навчання, а не єдиним засобом.

2. Вхідний стан потребує первинного засвоєння

Ще одна важлива проблема — якщо слово не було належним чином засвоєно (наприклад, не було пов’язано з контекстом, не було засвоєно в пам’яті), то інтервальний повторення може не мати відчутного ефекту. У досвіді учні часто згадують, що «SRS працює, коли ви вже знаєте слово трохи; він не замінює первинне навчання». Цей момент підкреслює, що SRS – не “чарівна таблетка”; потрібен якісний початковий етап навчання (до прикладу, контекстне засвоєння, читання).

3. Одномірність даних і алгоритмічна гнучкість

Більшість комерційних SRS-систем використовують заздалегідь визначені інтервали й алгоритми, які не враховують повною мірою індивідуальних когнітивних особливостей, типу слова, частоти його вживання, рідкості чи морфологічної складності. Наприклад, хоча моделі, як MEMORIZE, намагаються вирішити це як оптимізаційну задачу, алгоритми SRS загалом ще не достатньо адаптовані до складніших мовних сценаріїв.

Це обмежує можливості системи підбирати оптимальний час і частоту повторень для кожного слова та користувача, особливо в умовах індивідуалізації.

4. Проблема мотивації та регулярності користувача

Навіть найкращі SRS-алгоритми не працюють без дисциплінованого використання системи. Якщо користувач пропускає сесії, інтервали порушуються,

і ефективність падає. Системи самі зазвичай не компенсують великі проміжки відсутності, і це призводить до втрати переваг SRS.

Крім того, для багатьох учнів повторення карток стає рутинним і одноманітним, що знижує мотивацію та призводить до перебору карток без реального залучення.

5. Відбір лексики — «що варто повторювати»

Більшість SRS-систем орієнтовані на повторення наявних карток і не пропонують автоматизованого методу відбору слів, які мають найвищий пріоритет для навчання. Якщо користувач імпортує великий набір слів без сортування за складністю чи релевантністю, система може витрачати час на слова, які вже відомі, або які малоймовірно будуть використані. Як свідчать дослідження, потрібен додатковий компонент «розумного» відбору слів на основі частоти, контексту чи когнітивної складності.

Це особливо актуально для систем, що орієнтовані на автоматичну генерацію колод.

Наслідки для розробки системи.

Для реалізації успішної системи навчання іноземної лексики з використанням SRS потрібно враховувати наведені переваги й обмеження.

Зокрема:

- Перш за все, необхідний добре продуманий механізм відбору слів, який врахує релевантність, частоту, рівень складності і когнітивне навантаження — це дозволить уникнути витраченого часу на повторення «легких» чи малокорисних слів.
- Алгоритм SRS має бути адаптованим під користувача: враховувати його успіхи/помилки, час, коли він був активний чи не активний, дозволяти гнучке коригування інтервалів.
- Важливо поєднувати SRS з контекстним навчанням, наприклад, вправами на використання слова в реченні, читанням/аудіюванням, підвищенням залученості — щоб система не лиш мала функцію «зубріння», але формувала реальну продуктивну компетенцію.

- Необхідно дбати про мотивування та зручність використання системи: UI/UX має бути зручним, сесії — короткими і регулярними, щоб учень не втрачав регулярності.
- Враховувати, що система має бути лише частиною загального процесу навчання: занурення, читання, прослуховування, активне використання слів ісплатні. SRS — це підсилюючий компонент, а не заміна.

Отже, системи інтервального повторення без сумніву мають міцну теоретичну основу та доведену ефективність у підвищенні утримання словникового запасу. Однак їхня ефективність значною мірою залежить від якості початкового засвоєння матеріалу, відбору релевантних слів, регулярності використання та інтеграції з контекстним навчанням. У контексті розробки інтелектуальної системи вивчення іноземних слів це означає, що SRS-модуль має бути доповнений алгоритмом відбору та ранжування слів, і адаптивним механізмом, який враховує когнітивні характеристики слова і користувача.

1.3. Аналіз застосування методів обробки природної мови (NLP) у мовному навчанні

У сучасному мовному навчанні дедалі більше уваги приділяється тому, як технології обробки природної мови можуть покращувати ефективність засвоєння лексики, граматики, читання та письма. Цей підрозділ має на меті проаналізувати характерні напрями використання NLP-технологій у контексті вивчення іноземних слів, оцінити їх сильні сторони й обмеження, а також визначити ті аспекти, які релевантні для розробки нашої системи.

Використання NLP-методів у підтримці лексичного навчання.

Одним з найшвидших напрямів впровадження NLP у мовну освіту є автоматизоване формування навчальних матеріалів — наприклад, генерація вправ на слова, адаптація текстів під рівень учня, виділення ключової лексики. У дослідженні Peng (2024) проведено широкий огляд застосувань NLP у навчанні

другої мови, зокрема у частинах “підтримка читання”, “зворотний зв’язок при письмі”, “усна взаємодія”, “персоналізоване навчання”. [5]

Наприклад, автоматизація створення карток зі словами може спиратися на аналіз наявного текстового корпусу: NLP-інструменти виконують токенізацію, лематизацію, частотний аналіз, виділення нових чи важких слів, що учень ще не засвоїв. Це дозволяє створити більш релевантну колоду, ніж просте застосування «стандартного списку» слів.

Адаптація текстів та виділення лексики.

NLP-методи також застосовуються для адаптації вхідного навчального матеріалу за рівнем учня. Наприклад, використання NLP-технологій дозволяє створювати вправи на лексику, змінювати контекст, додавати пояснювальні підказки, генерувати варіанти речень з новими словами.

Це означає, що система може аналізувати великий корпус текстів, виділяти слова, які є менш відомими, або мають вищу когнітивну складність, і на їх основі створювати навчальні одиниці. Таким чином, відбір не базується лише на списках, а на аналізі реальних текстів користувача або інтересів.

Семантичні моделі та розуміння значення слова.

Підвищення ефективності лексичного навчання також пов’язане з переходом від простої частотної ваги слова до семантичних показників. У NLP-дослідженнях широко використовуються embeddings (наприклад, Word2Vec, GloVe), які перетворюють слова у вектори, що відображають їхні семантичні зв’язки.

У навчальному контексті це може дозволити системі визначити, наскільки слово «рідке», наскільки воно близьке до вже відомих користувачу слів, і, відповідно, вирішити, чи воно потребує активного повторення.

Приклад: якщо слово має багато аналогів (синонімів) у відомих учню одиницях, його когнітивна складність може бути нижчою, тому алгоритм може знижувати пріоритет. А слово, що не має явних семантичних зв’язків з вже відомою лексикою, може отримати вищий пріоритет для включення у навчальну колоду.

Персоналізація навчання за допомогою NLP.

Персоналізація — це один з ключових принципів ефективного навчання. NLP-інструменти можуть аналізувати індивідуальні дані користувача: тексти, які він читає, частоту використання слів, помилки у сесіях, і на основі цього створювати персоналізовані списки слів чи адаптувати інтервали повторення. У роботі Phomprasert (2021) показано, що застосування NLP-інструментів для учнів початкових класів у сільській школі дозволило покращити вимову та мотивацію в навчанні англійської мови.

Таке рішення дозволяє перейти від «одного розміру для всіх» до системи, що враховує когнітивний профіль, історію помилок, рівень володіння та інтереси учня — тобто, саме антропоцентричний підхід.

Інтеграція NLP з моделями SRS.

У поєднанні з методикою інтервального повторення, NLP-методи забезпечують можливість автоматичного відбору слів, оцінки їх складності (частотні та семантичні показники), а також адаптації порядку і інтервалів повторення. Наприклад, система може аналізувати текст користувача, виділяти нові слова за допомогою TF-IDF або embeddings, визначати когнітивну складність і включати ці слова у SRS-сесію з відповідним пріоритетом.

Цей підхід дає змогу уникнути ситуацій, коли користувач витрачає час на слова, які він вже добре знає, або слова, що занадто легкі чи нерелевантні. Тим самим підвищується ефективність навчання і знижуються витрати часу.

Обмеження застосування NLP у лексичному навчанні

Незважаючи на широкі можливості, застосування NLP у навчанні лексики має ряд обмежень. По-перше, багато моделей побудовані на великих корпусах даних і можуть бути менш ефективними для мов із обмеженими ресурсами (low-resource-languages).[6]

По-друге, автоматичне виділення слів або побудова речень може призводити до втрати лінгвістичної чи культурної релевантності — наприклад, безконтекстні приклади або погано адаптовані тексти можуть зменшувати залученість користувача.

По-третє, інтеграція NLP-елементів у навчальні платформи потребує значних технічних ресурсів та налаштувань — токенізація, лематизація, побудова embeddings, бази даних користувача, а також коректне зберігання й обробка даних — все це має бути реалізовано якісно.

По-четверте, технології NLP можуть створювати ілюзію автоматичного навчання, але без інтеграції з активним контекстним використанням лексики (читання, говоріння, писання) їх ефективність обмежена.

Для системи, створеної на основі антропоцентричних обчислень та NLP-методів, важливо врахувати такі принципи:

- Використовувати NLP-аналіз текстового корпусу користувача для виділення релевантної лексики.
- Створити математичну модель, що поєднує показники семантичної рідкості із когнітивною складністю слова (за довжиною, морфологією, частотою) для ранжування слів.
- Інтегрувати вибір слів із системою SRS так, щоб інтервали повторення враховували характеристики слова та користувача.
- Забезпечити адаптацію рішення під мову і користувача — особливо важливо, якщо система орієнтована на менш ресурсні мови або специфічні корпуси.
- Усвідомлювати, що NLP-елементи — це інструменти підтримки, але система має поєднувати адаптивний відбір, інтервальне повторення і контекстне використання лексики для досягнення реальних результатів.

1.4. Антропоцентричні обчислення як основа персоналізованого навчання

Сучасні підходи до автоматизованого навчання дедалі частіше акцентують увагу на персоналізації процесу. Однією з концептуальних основ такого підходу є антропоцентричні обчислення, які передбачають проектування систем, що враховують індивідуальні когнітивні, поведінкові та контекстуальні особливості

користувача. У контексті вивчення іноземних слів антропоцентричний підхід дозволяє підвищити ефективність засвоєння лексики, мінімізувати «шум» та оптимізувати витрати часу учня. [7, 8]

Основні принципи антропоцентричних обчислень.

Антропоцентричні обчислення базуються на трьох ключових принципах:

1. Фокус на користувача – система повинна орієнтуватися на потреби, здібності та поведінку конкретного користувача. У навчанні лексики це означає адаптацію контенту та методів повторення під когнітивні особливості учня, його рівень знань і мотивацію.
2. Інтерактивність та зворотний зв'язок – важливо забезпечити постійний обмін інформацією між користувачем та системою, щоб остання могла динамічно коригувати навчальний процес. Це включає оцінку успішності повторень, швидкості засвоєння нових слів, частоти помилок та часу реакції.
3. Контекстуалізація – навчання не повинно відриватися від реального контексту користувача. Для мовних систем це означає врахування типів текстів, які користувач читає, тематики контенту та частоти вживання слів у реальному мовленні.

Наукові дослідження показують, що інтеграція антропоцентричних принципів у навчальні системи значно підвищує ефективність засвоєння нових знань і мотивацію користувача. [9]

Антропоцентричний підхід у мовному навчанні.

У практичному аспекті антропоцентричні обчислення дозволяють розробляти системи персоналізованого відбору та ранжування слів. Наприклад, замість стандартного списку 5000 найпоширеніших слів, система може враховувати індивідуальні знання користувача:

- слова, які він вже засвоїв, виключаються або отримують низький пріоритет;
- слова, що зустрічаються рідко або є складними для конкретного учня, отримують високий пріоритет;

- інтервали повторення адаптуються залежно від швидкості засвоєння конкретної лексичної одиниці.

Таким чином, антропоцентричні обчислення дозволяють підвищити ефективність навчання через адаптацію алгоритмів під когнітивний профіль користувача. Це особливо важливо при використанні систем інтервального повторення, де ключовим фактором є правильне визначення пріоритету слів і моменту повторення.

Використання даних користувача

Антропоцентричні системи збирають і аналізують різні типи даних:

- Статистичні показники повторень: скільки разів користувач правильно відтворив слово, скільки разів помилився, середній час реакції.
- Когнітивні індикатори: складність слова за морфологією, довжиною, частотою вживання, наявність семантичних зв’язків із вже відомими словами.
- Контекст використання: тематика текстів, з яких виділяється лексика, специфічність мови, індивідуальні інтереси користувача.

На основі цих даних алгоритм формує індивідуальний профіль навчання і визначає, які слова слід включити в поточну колоду SRS, з яким пріоритетом і в якій послідовності повторювати. Це забезпечує баланс між автоматизацією процесу та збереженням контролю користувача над навчанням.

Інтеграція з NLP-методами

Антропоцентричні обчислення і NLP-методи взаємодоповнюють один одного. NLP дозволяє аналізувати текстовий контент, визначати частоту та складність слів, виділяти нові лексичні одиниці. А антропоцентричний підхід адаптує ці дані під конкретного користувача. Наприклад:

- TF-IDF або embeddings використовуються для оцінки «важливості» слова в тексті [10, 11];
- когнітивна складність слова оцінюється на основі довжини, морфології, наявності зв’язків із відомими словами;

- алгоритм визначає пріоритет повторення у SRS-колоді для конкретного користувача, враховуючи історію успіхів і помилок.

Таким чином, система стає гнучкою та індивідуалізованою, що є ключовою вимогою антропоцентричного підходу.

Переваги антропоцентричних обчислень у навчанні лексики

1. Підвищення ефективності засвоєння – слова з високою когнітивною складністю повторюються частіше, легкі слова відсуваються, що оптимізує час учня.
2. Мотивація користувача – персоналізовані сесії SRS менш монотонні, що підвищує зацікавленість і регулярність навчання.
3. Адаптивність – система динамічно реагує на прогрес користувача, коригує пріоритети та інтервали повторення.
4. Реалістичне засвоєння лексики – врахування контексту, тематики та семантичних зв’язків дозволяє ефективніше закріплювати слова у довготривалій пам’яті.

Обмеження та виклики.

Хоча антропоцентричні обчислення підвищують ефективність, вони потребують:

- точного збору та обробки даних користувача;
- правильної інтеграції з алгоритмами NLP і SRS;
- забезпечення приватності та безпеки персональних даних;
- продуманого UI/UX для підтримки мотивації та зручності.

Без цього ефект персоналізації може бути знижений або зовсім не проявитися.

Антропоцентричні обчислення створюють основу для персоналізованого навчання, де алгоритм підлаштовується під когнітивний та контекстний профіль користувача, забезпечуючи оптимальне формування колод для SRS. У поєднанні з методами NLP вони дозволяють не лише автоматизувати процес навчання, а й значно підвищити його ефективність, релевантність та мотиваційний потенціал.

2 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ТА АЛГОРИТМІВ ДЛЯ ВИЗНАЧЕННЯ СКЛАДНОСТІ ТА ВІДБОРУ ЛЕКСИКИ

2.1 Аналіз сучасних підходів до визначення складності текстів і лексичних одиниць

Визначення складності текстів та окремих лексичних одиниць є ключовим елементом у побудові систем персоналізованого мовного навчання. Рівень складності безпосередньо впливає на ефективність засвоєння інформації, швидкість прогресу та рівень когнітивного навантаження на користувача. Тому дослідження доступних методів оцінювання складності є необхідним для побудови надійної та валідної моделі відбору та структурування іншомовної лексики. У сучасній лінгвістиці та комп'ютерній лінгвістиці застосовуються численні підходи, які можна умовно поділити на три основні групи: традиційні лінгвістичні методи, статистичні та частотні підходи, а також методи на основі обробки природної мови, включно з моделями глибинного навчання. Кожен із них має свою сферу застосування, переваги та обмеження.

Традиційні лінгвістичні підходи до визначення складності.

Одними з найдавніших і водночас найпоширеніших підходів є формальні лінгвістичні показники. У рамках цих методів складність розглядається через призму мовних характеристик, що впливають на легкість сприйняття мовного матеріалу. До основних параметрів належать:

- довжина слова (у буквах або складах),
- морфемна структура,
- кількість значень (полісемія),
- регулярність словотворчих моделей,
- приналежність до конкретної частини мови.

Довжина слова часто використовується як базовий індикатор складності, оскільки коротші слова статистично частіше зустрічаються в розмовній мові й

підлягають швидшому засвоєнню. Разом із тим у багатьох мовах довгі слова мають регулярну морфемну структуру, що значно спрощує їх обробку. Це створює середовище, у якому самостійне застосування метрики «довжина слова» є недостатнім.

Морфологічна складність розглядає структурну організацію слів. Наприклад, слова аналітичних мов із малою кількістю флексій можуть бути короткими, проте їх семантичне значення часто залежить від контексту, що збільшує когнітивне навантаження.

Для формального опису морфологічної складності інколи застосовується така узагальнена формула (2.1):

$$C_{morph} = \alpha \cdot M + \beta \cdot Ir, \quad (2.1)$$

де M — кількість морфем у слові;

Ir — показник морфологічної нерегулярності;

α, β — вагові коефіцієнти, що визначають значущість параметрів.

Подібні моделі дозволяють кількісно оцінювати структурні властивості слова, проте вони часто не відображають когнітивні характеристики сприйняття, зокрема вплив частоти вживання чи семантичної зв’язаності.

Методи оцінювання складності тексту.

У контексті аналізу цілих текстів широко використовуються формули читабельності. Вони з’явилися ще у XX столітті та використовуються в освітніх стандартах у багатьох країнах. Найпоширенішими прикладами є формула Флеша-Кінкейда, індекс Ганнінга, Coleman–Liau та інші. Основною ідеєю цих формул є визначення складності тексту на основі таких параметрів:

- середньої довжини речення,
- середньої кількості складів на слово,
- співвідношення коротких і довгих слів.

Ці підходи дозволяють оцінити загальну складність текстового корпусу, однак вони не придатні для задачі ранжування лексичних одиниць, оскільки працюють лише з агрегованими значеннями.

Для прикладу розглянемо одну з найпоширеніших формул — індекс зручночитабельності Флеша (2.2):

$$RE = 206.835 - 1.015 \cdot ASL - 84.6 \cdot ASW\ , \qquad (2.2)$$

де ASL — середня довжина речення;

ASW — середня кількість складів на слово.

Хоча ця формула широко застосовується, вона була створена для англійської мови й не враховує структурних особливостей інших мов, а також не здатна відрізняти когнітивну складність конкретних слів.

Статистично-частотні методи визначення складності.

У сучасній лексикографії одними з найдієвіших показників складності є частотні характеристики. Частотність слова в корпусах природної мови корелює з легкістю запам’ятовування та швидкістю розпізнавання. Чим частіше слово зустрічається, тим легше воно засвоюється.

Проте частотою не вичерпується повний набір статистичних характеристик. При визначенні складності інколи враховують:

- ентропію контекстів,
- ступінь розподіленості значень,
- співвідношення базових та контекстуально зумовлених значень,
- частотний ранг у різних джерелах.

Ентропія контекстів дозволяє оцінити непередбачуваність появи слова. Якщо слово може з’являтися в ширшому спектрі контекстів, це може ускладнювати його засвоєння.

Ентропія визначається формулою (2.3):

$$H = -\sum_{i=1}^{n} p_i \log p_i\ , \qquad (2.3)$$

де p_i — ймовірність появи слова в конкретному контекстному класі.

Частотні методи є добре формалізованими, легко інтерпретуються та масштабуються на великі корпуси, однак мають певні обмеження: вони не враховують семантичну складність, полісемію та міжлексичну подібність.

Підходи на основі NLP та семантичного моделювання.

Останніми роками особливого поширення набули методи, що ґрунтуються на обробці природної мови та семантичному моделюванні. Їхня головна перевага — здатність враховувати не лише поверхневі параметри лексики, а й глибинні взаємозв'язки між словами, значеннями та контекстами.

Ключові напрями NLP-методів, що застосовуються для оцінки складності:

- векторні моделі слів (Word2Vec, GloVe);
- контекстуальні мовні моделі (ELMo, BERT-подібні архітектури);
- семантичні відстані та подібність;
- класифікаційні моделі читабельності;
- предиктивні моделі труднощів засвоєння.

На відміну від традиційних методів, моделі NLP дозволяють розглядати складність слова як функцію його семантичного розташування в латентному просторі значень (2.4):

$$C_{sem} = f\big(d(w_i, S)\big)\,, \tag{2.4}$$

де $d(w_i, S)$ - середня семантична відстань між словом w_i та іншими словами тематичного поля S;

$f(\cdot)$ — функція нормалізації.

Такі підходи формують багатовимірний погляд на складність, однак їхнє застосування потребує великих обчислювальних ресурсів, а інтерпретація результатів може бути складною.

Огляд сучасних підходів показує, що жоден окремий метод не забезпечує повного опису складності лексики. Лінгвістичні підходи забезпечують структурний аналіз, частотні — статистичний, а NLP — семантичний, проте лише їхнє комбінування дозволяє сформувати повноцінну модель оцінювання лексичної складності, необхідну для оптимізації процесу вивчення іноземних слів у персоналізованих навчальних системах.

2.2 Методи статистичного та частотного аналізу в задачах мовного навчання

Статистичні та частотні методи посідають центральне місце у визначенні складності лексичних одиниць у сучасних системах мовного навчання. Ці підходи ґрунтуються на припущенні про те, що розподіл слів у природному мовленні відображає реальну когнітивну доступність лексики для носіїв мови та для осіб, які її опановують. Статистичні закономірності дозволяють формалізувати процес оцінки складності, забезпечуючи об’єктивні метрики, які можуть слугувати основою для автоматизованих алгоритмів відбору лексики. У цьому підрозділі розглядаються ключові принципи статистичного аналізу, найбільш поширені частотні показники, а також обмеження, пов’язані з їх використанням у контексті індивідуалізованого мовного навчання.

Загальні принципи частотного аналізу.

Частотний аналіз передбачає визначення кількості появ певного слова в текстовому корпусі або множині корпусів. На основі частотності сформульовано кілька важливих статистичних закономірностей, серед яких найвідомішою є закон Ципфа. Відповідно до цього закону, частота слова обернено пропорційна його рангу (2.5):

$$f(r) = \frac{C}{r^k}, \tag{2.5}$$

де f(r) — частота слова з рангом r;

C — нормувальна константа;

k — параметр, що описує крутість спадання частот.

У більшості мов слово з рангом 1 (найчастіше вживане слово) зустрічається у десятки разів частіше, ніж слово з рангом 10, і в сотні разів частіше, ніж слово з рангом 100. Це означає, що певна відносно невелика група слів забезпечує значну частку мовного потоку. Саме тому частотність слугує одним із ключових індикаторів, що визначають доступність слова для вивчення.

Застосування частотного аналізу в мовному навчанні часто передбачає використання частотних словників, які відображають типові патерни вживання слів

у різних жанрах і стилях мовлення. Такі словники будуються на основі національних корпусів, що репрезентують різноманітні типи текстів — від художньої літератури до медіа та усного мовлення.

Абсолютна та відносна частота як індикатори складності

У межах частотного аналізу виділяються два основні показники:

- абсолютна частота — кількість поява слова у корпусі;
- відносна частота — число появ на певну кількість слів (як правило, на мільйон).

Абсолютна частота залежить від загального розміру текстового масиву й часто використовується при порівнянні слів всередині одного корпусу. Відносна частота є універсальнішою, оскільки дозволяє нормалізувати дані, забезпечуючи їхню порівнюваність між різними корпусами.

Відносну частоту часто визначають як (2.6):

$$f_{rel} = \frac{f}{N} \cdot M, \tag{2.6}$$

де f — абсолютна частота слова;

N — загальна кількість слів у корпусі;

M — масштабний множник (частіше за все 1 000 000).

Частота є важливим індикатором складності, адже численні психолінгвістичні дослідження демонструють прямий зв’язок між частотністю та швидкістю лексичного доступу. Слова високої частотності швидше розпізнаються, легше відтворюються та швидше переходять у довготривалу пам’ять.

2.3 Використання моделей обробки природної мови для лексичного відбору

Застосування моделей обробки природної мови у задачах відбору лексики сформувало окремий напрям у дослідженнях, спрямованих на підвищення точності оцінки складності та релевантності лексичних одиниць. У порівнянні зі статистичними методами, підходи на основі NLP дозволяють врахувати ширший спектр лінгвістичних та семантичних характеристик, що суттєво наближає

результати моделювання до реальних когнітивних процесів засвоєння інформації. У сучасних системах такі моделі застосовуються для кількісної оцінки семантичної схожості, класифікації складності, визначення корисності лексем та адаптації навчальних матеріалів відповідно до мовного рівня користувача.

Лексичні вектори та семантичні простори.

Одним із фундаментальних інструментів NLP у контексті лексичного відбору є моделі векторного подання слів. Такі моделі трансформують слова у точки багатовимірного простору, де геометричні відношення відображають семантичну близькість. Найпоширенішими є вбудовування типу Word2Vec, GloVe та їх подальші модифікації.

Основним принципом побудови таких моделей виступає дистрибутивна гіпотеза, згідно з якою значення слова визначається контекстами його вживання. У практичному застосуванні це дозволяє оцінювати:

- семантичну відстань між словами;
- ступінь складності через віддаленість слова від базової лексики;
- потенційну корисність конкретної лексеми у межах тематики тексту.

Формально відстань між словами у семантичному просторі найчастіше обчислюється через косинусну подібність (2.7):

$$sim(w_1, w_2) = \frac{(v_{w_1} \cdot v_{w_2})}{||v_{w_1}|| \cdot ||v_{w_2}||}. \quad (2.7)$$

Цей показник активно застосовується у процесах відбору найближчих за значенням слів, групування лексики та визначення «центральних» понять тексту.

Контекстуальні моделі та трансформерні архітектури.

У сучасних дослідницьких і промислових системах провідну роль відіграють моделі на основі трансформерів — зокрема BERT, RoBERTa, XLM-R, DistilBERT. На відміну від статичних векторів, контекстуальні моделі здатні формувати представлення слова залежно від його оточення, що дозволяє досягти значно точнішої оцінки складності та ролі конкретної лексичної одиниці в тексті.

Для визначення складності використовуються такі характеристики:

- контекстуальна варіативність: висока зміна значення слова в різних контекстах корелює з підвищеною складністю;
- ентропія розподілу значень: більша полісемія ускладнює засвоєння;
- позиційні залежності в реченні: складні синтаксичні конструкції підвищують когнітивні витрати.

Контекстуальні моделі дозволяють обчислювати оцінки складності на рівні тексту або окремих лексичних одиниць. Наприклад, одним із підходів є використання лог-правдоподібності токенів у моделі (2.8):

$$C(w_i) = -\log P\,(w_i \mid\mid context). \qquad (2.8)$$

Вища негативна лог-правдоподібність вказує на більшу непередбачуваність токена, що корелює зі складністю для користувача.

Семантична класифікація та тематичне моделювання.

У задачах відбору лексики вагоме значення має узгодженість слів із темою тексту. Тематичні моделі, такі як LDA (Latent Dirichlet Allocation) або сучасні нейронні тематичні моделі, дозволяють:

- визначити тематичні кластери;
- оцінити релевантність лексики до головних тем;
- фільтрувати слова, що не несуть ключової інформації.

Хоча LDA має обмеження у роботі з короткими текстами та складними синтаксичними структурами, вона залишається типовим інструментом у системах, де необхідна інтерпретованість.

Моделі визначення складності лексики.

У зарубіжних дослідженнях широко застосовуються системи lexical complexity prediction, де обчислюється кількісна оцінка труднощі слова для носія іноземної мови. Такі моделі можуть поєднувати:

- морфологічні характеристики (довжина слова, кількість морфем);
- частотні показники;
- синтаксичні ознаки;
- контекстуальні представлення слова з трансформерних моделей.

У деяких випадках комбінуються кілька предикторів за допомогою лінійних або регресійних моделей (2.9):

$$Score = a_1 \cdot f_1 + a_2 \cdot f_2 + \cdots + a_n \cdot f_n, \quad (2.9)$$

де f_i — окрема ознака, а a_i — ваговий коефіцієнт.

Аналіз колокацій та багатослівних виразів.

Колокаційний аналіз відіграє суттєву роль у визначенні корисності лексики. Багато систем вивчення мов роблять наголос на словосполученнях, оскільки вони частіше запам’ятовуються та краще відображають природне мовлення.

NLP-моделі дозволяють оцінювати силу сполучуваності за такими метриками:

- PMI (Pointwise Mutual Information);
- t-score;
- log-likelihood.

Наприклад:

$$PMI(w_1, w_2) = \log \frac{P(w_1, w_2)}{P(w_1) \cdot P(w_2)}. \quad (2.10)$$

Високий PMI сигналізує про стійке сполучення, яке варто включати до навчальних матеріалів навіть за умови, що окремі слова не належать до частотної або простої лексики.

Синтаксичні моделі та аналіз залежностей.

Синтаксичний парсинг дозволяє враховувати структурні аспекти тексту. У системах лексичного відбору синтаксичні дерева використовуються для визначення:

- типу функції слова в реченні;
- глибини вкладеності синтаксичних конструкцій;
- ролі слова у ключових відношеннях.

Комплексність синтаксичних структур часто виступає індикатором загальної складності тексту, а отже — необхідності адаптації лексики для початкових рівнів.

Додаткові методи обробки мови

Серед останніх досліжень також можна визначити новітні підходи застосування та розширення базового TF-IDF.

TF-IGM (частота терміну і обернений момент тяжіння) ефективно покращує точність класифікації текстів. TF-IGM враховує нову статистичну модель, яка точно визначає здатність терміна розрізняти класи. Важливою особливістю є використання розподілу термінів серед різних класів тексту, що дає більш точну класифікацію. [12]

Пропонується також новий метод STF-IDF, який покращує точність оцінки важливості слів у неформальних текстах. Цей метод адаптує традиційний TF- IDF, використовуючи семантичні моделі та векторні представлення слів для підвищення результатів класифікації. Дослідження показує, що STF-IDF зменшує помилку середнього TF-IDF на 50%, покращуючи точність на неформальних документах. [13]

Поєднання класичного TF-IDF з моделлю word2vec дає кращий аналіз семантики слів. Цей метод класифікації враховує векторне представлення слів, що дозволяє зберегти значення синонімів, підвищуючи точність видобутку ознак у порівнянні з традиційним підходом. [14]

Поєднання таких параметрів, як довжина документів та розподіл частоти термінів для категоризації текстів значно вдосконалюють стандартний метод TF-IDF і допомагає підвищити точність класифікації текстів. Експериментальні результати демонструють перевагу нових стратегій у порівнянні з традиційним TF-IDF, що підтверджує їхню ефективність на різних наборах даних. [15]

Моделі NLP забезпечують багаторівневий аналіз лексики, що значно розширює можливості систем мовного навчання у порівнянні з традиційними статистичними методами. Їх використання дозволяє визначати складність, релевантність, контекстуальність та когнітивну цінність лексичних одиниць, що є основою для побудови адаптивних високоточних алгоритмів відбору навчального матеріалу.

У подальших підрозділах виконано порівняння описаних методів, визначено їх обмеження та наведено аргументацію щодо вибору конкретного підходу для реалізації в інформаційній системі.

2.4 Порівняльний аналіз існуючих підходів: переваги та недоліки

Порівняння сучасних підходів до визначення складності та відбору лексики є необхідним етапом для формування узгодженої методології, яка забезпечує точність, стійкість і практичну придатність результатів. Попередні підрозділи описували три головні групи методів: індекси читабельності та класичні лінгвістичні формули, статистичні та частотні підходи, а також методи, засновані на обробці природної мови. Кожна з них пропонує цінний, але різний за природою інструментарій для оцінки лексичної складності та релевантності.

Порівняння за критеріями точності та інформативності.

Методи читабельності орієнтуються переважно на поверхневі характеристики тексту: середню довжину слів, довжину речень, кількість складів. Формули, подібні до Flesch Reading Ease чи Gunning Fog Index, забезпечують швидку й інтерпретовану оцінку складності, однак вони слабо відображають реальну когнітивну складність лексики. Такі формули не враховують семантичних зв'язків, тематичної релевантності та контекстної варіативності слів. Їх точність зменшується для текстів із нетиповою синтаксичною структурою або високою часткою термінології.

Статистичні методи частотності, навпаки, забезпечують більш деталізоване бачення реальності мовного вживання. Частотні словники, ранжування слів, TF-IDF та інші метрики дозволяють визначити, наскільки слово є інформативним у рамках конкретного тексту або корпусу. Однак частотність не є прямим показником складності: деякі високочастотні слова можуть бути концептуально складними, тоді як низькочастотні — легкими для розуміння в конкретному контексті. Частотні підходи також мають обмеження щодо тем, де домінують рідкі або спеціалізовані терміни.

Моделі NLP демонструють значно вищу гнучкість і здатність відображати складні лінгвістичні взаємозв’язки. Контекстуальні моделі, трансформери й тематичний аналіз дозволяють враховувати не лише статистичні параметри, а й семантику, прагматику та структуру тексту. Проте такі підходи вимагають значних обчислювальних ресурсів, можуть бути складними для інтерпретації та іноді демонструють надмірну чутливість до шуму або низькоякісних даних.

Порівняння за масштабованістю та ресурсними вимогами.

Методи читабельності є найбільш економічними: їх обчислення вимагає мінімальних ресурсів, що дозволяє використовувати їх у мобільних або офлайн-системах без втрати продуктивності. Статистичні методи, хоча й трохи дорожчі у виконанні, залишаються доступними для більшості освітніх платформ завдяки високій швидкості обробки даних і простоті реалізації.

NLP-моделі, особливо трансформерні, вимагають великих обчислювальних потужностей та оптимізованих інфраструктур. Для статичних систем обмеження можуть бути суттєвими, особливо якщо модель не дозволяє адаптивного донавчання. Незважаючи на це, отримана якість аналізу значно перевищує можливості традиційних методів, що частково компенсує їхню ресурсоємність.

Порівняння за здатністю адаптуватися до різних жанрів і рівнів мови.

Поверхневі формули складності мало пристосовані до жанрових відмінностей: короткі речення або високий рівень фрагментації можуть призводити до некоректних оцінок. Частотні методи працюють добре для загальновживаної лексики, але часто втрачають точність при аналізі спеціалізованих текстів, де ключові терміни є низькочастотними, хоча й надзвичайно важливими.

Контекстуальні моделі демонструють найвищу здатність адаптуватися, тому що враховують зв’язки між словами у межах конкретного тексту. Вони здатні визначати навіть приховані семантичні патерни, що особливо важливо у складних або міждисциплінарних текстах. Проте їх ефективність залежить від якості й масштабу корпусу, на якому вони були навчені.

Порівняння за можливістю інтеграції у системи мовного навчання.

Методи читабельності забезпечують легку інтеграцію, але їхня користь у системах, орієнтованих на індивідуалізацію навчання, є обмеженою. Вони можуть виступати базовим заходом контролю складності текстів, але не здатні підтримувати персоналізований лексичний відбір.

Статистичні методи є оптимальними для систем, що потребують стабільних і передбачуваних механізмів ранжування. Вони забезпечують швидкі результати і добре масштабуються, що робить їх підходящими для великих платформ із високою кількістю користувачів.

Підходи NLP забезпечують найглибше розуміння текстів та лексики, тому підходять для систем, орієнтованих на точну персоналізацію та когнітивну адаптацію. Проте їх впровадження потребує ретельного налаштування та оптимізації, а також заздалегідь визначених обмежень, якщо модель є статичною.

Таблиця 2.1

Порівняльна таблиця підходів до визначення складності та відбору лексики

Підхід	Точність	Ресурси	Адаптивість	Інтеграція	Недоліки
Формули читабельності	Низька–середня	Дуже низькі	Низька	Висока	Поверхневий аналіз, ігнорування семантики
Частотні методи	Середня	Низькі	Середня	Висока	Не враховують контекст, неповні оцінки складності
NLP-моделі	Висока	Високі	Висока	Середня	Складність, ресурсоємність

2.5 Визначення обмежень сучасних систем для вирішення задачі оптимізації вивчення іноземних слів

Попри значну кількість доступних підходів до оцінки складності та відбору лексики, сучасні системи, орієнтовані на підтримку процесу вивчення іноземних слів, демонструють низку обмежень, що впливають на якість навчання, адаптивність та стійкість отриманих результатів. Аналіз існуючих методів дозволяє виділити ключові недоліки, пов’язані з використанням поверхневих формул складності, частотних характеристик та контекстуальних NLP-моделей, а також обмеження, що виникають на рівні архітектурних рішень сучасних навчальних платформ.

Обмеження класичних індексів читабельності та лінгвістичних формул.

Індекси читабельності орієнтовані на поверхневі характеристики тексту — довжину слів, кількість складів та структуру речень. У контексті вивчення іноземної лексики такі формули демонструють кілька суттєвих обмежень.

1. Відсутність урахування семантики. Значення та когнітивна складність слова не пов’язані з його довжиною чи структурними характеристиками. Слова з ідентичною довжиною можуть мати різний рівень абстракції та семантичної насиченості.
2. Слабка узгодженість із мовною специфікою. Формули, розроблені для англійської мови, погано відтворюють складність текстів іншими мовами, особливо тими, що мають аглютинативні чи флективні властивості.
3. Ігнорування контекстного використання. Слово може бути простим у одному контексті й складним у іншому, що такі підходи не здатні адекватно врахувати.

Унаслідок цього індекси читабельності не можуть бути основою для точного визначення індивідуального лексичного навантаження користувача.

Обмеження частотних та статистичних методів.

Методи, що базуються на частотності, є потужними засобами аналізу корпусних даних, проте в задачі оптимізації лексичного навчання вони також демонструють низку системних недоліків.

1. Неоднозначний зв’язок між частотністю та складністю. Високочастотні слова не завжди є легкими, а низькочастотні — складними. Багато абстрактних понять можуть бути частими у текстах високого рівня складності.
2. Залежність від доменного корпусу. Частотні характеристики сильно змінюються залежно від жанру та тематичного контексту; тому системи, що працюють із обмеженим корпусом, можуть генерувати хибні оцінки.
3. Відсутність когнітивних параметрів. Частотні методи не враховують навчальних факторів: робочу пам’ять, ментальні моделі, швидкість забування тощо.
4. Одновимірність оцінок. Статистичні підходи не здатні працювати з багатовимірними характеристиками слова — морфологією, полісемією, синонімією, прагматичними особливостями.

Отже, хоча статистичні методи забезпечують стабільність і швидкість, вони не здатні самостійно формувати багатокомпонентну оцінку лексичної складності.

Обмеження підходів, заснованих на NLP-моделях.

Методи NLP здатні враховувати контекст, семантичну схожість і синтаксичні зв’язки, проте їхнє практичне застосування в освітніх системах також має низку проблем.

1. Високі обчислювальні витрати. Навіть статичні моделі вимагають значних ресурсів для обробки тексту, що ускладнює їх застосування в реальному часі.
2. Складність інтерпретації. Контекстуальні вектори мають низьку прозорість; визначити, чому модель оцінила слово певним чином, часто неможливо.
3. Чутливість до шуму. Помилки розмітки або нетиповий синтаксис можуть суттєво впливати на результати аналізу.

4. Відсутність адаптивності у статичних моделей. У разі неможливості донавчання модель не враховує індивідуальний стиль користувача або його навчальний прогрес, що зменшує ефективність рекомендацій.

Таким чином, хоча NLP відкриває значно ширші можливості для аналізу лексики, обмеження таких моделей створюють серйозні виклики для інтеграції в освітні системи.

Архітектурні обмеження сучасних систем вивчення лексики.

Більшість платформ, що реалізують інтервальне повторення або контекстне навчання, використовують обмежений набір параметрів для оцінки складності слова користувача. Типові обмеження включають універсальні моделі, що не враховують індивідуальний досвід; надмірну прив’язку до тестових завдань замість контекстного аналізу; низький рівень пояснюваності прийнятих рішень; відсутність механізмів динамічного формування словникової пріоритетності.

Такі недоліки знижують персоналізованість навчального процесу та не дозволяють створити надійну модель лексичного навантаження.

Таблиця 2.2

Зведена таблиця недоліків

Метод	Основні обмеження
Інструменти читабельності	Поверхневий аналіз; залежність від мови; ігнорування контексту
Частотні підходи	Одновимірність оцінок; залежність від корпусу; відсутність когнітивних параметрів
NLP-моделі	Ресурсоємність; низька інтерпретованість; чутливість до шуму; обмеження статичності

Огляд обмежень доводить, що сучасні методи не забезпечують комплексного врахування лінгвістичних, когнітивних та контекстуальних параметрів, необхідних для точного визначення складності лексики та її ефективного відбору. Жоден окремий підхід не пропонує самодостатнього рішення; натомість виявляється

необхідність у комбінованій системі, що поєднує точність статистичних моделей, гнучкість NLP та лінгвістичну інтерпретованість класичних підходів. Саме така інтеграція здатна подолати окреслені обмеження й сформувати основу для ефективного алгоритму оптимізації процесу вивчення іноземних слів.

2.6 Обґрунтування вибору методів для реалізації запропонованого підходу

Для реалізації системи визначення складності та відбору лексичних одиниць обрана інтеграція методів обробки природної мови (NLP) із антропоцентричними обчисленнями. Такий підхід дозволяє усунути обмеження сучасних систем, проаналізовані у попередніх підпунктах, поєднуючи статистичні характеристики тексту, лінгвістичні параметри слів та когнітивно-орієнтовану оцінку складності, що забезпечує точніше і персоналізоване ранжування лексики.

Використані компоненти системи.

Мовна обробка на основі NLP. Використання бібліотеки spaCy забезпечує морфологічний та частотний аналіз тексту на рівні токенів та лем. Модель en_core_web_sm дозволяє визначати частини мови, виконувати лемматизацію, сегментацію речень та обробку морфологічних особливостей слів. Це усуває обмеження класичних статистичних методів, які зазвичай враховують лише поверхневі характеристики слова та ігнорують контекст.

Статистичний підхід через частотний аналіз та TF-IDF. Для оцінки інформативності слів застосовується TF-IDF в поєднанні з нормалізованою частотністю, що дозволяє виділяти слова, характерні для конкретного тексту. Це забезпечує багатовимірний підхід до оцінки слова, що враховує не тільки його поширеність, але і значущість у конкретному контексті.

Антропоцентричні обчислення складності слова. Система оцінює складність слова, враховуючи його довжину, кількість складів, морфологічну нерегулярність, частину мови та конкретність значення. Такий підхід враховує когнітивні аспекти сприйняття лексики користувачем, що дозволяє формувати персоналізовану оцінку

складності та подолати обмеження традиційних методів, які не враховують індивідуальні особливості сприйняття.

Вибір слів для включення у навчальний набір. На основі інтегрованої оцінки визначаються слова, що рекомендуються для навчання. Кожне слово аналізується з урахуванням його значущості, когнітивної складності та частоти в тексті, що дозволяє формувати релевантний і оптимальний для навчання список лексики.

Контекст та приклади використання. Для кожного слова система надає приклад речення з корпусу та переклад. Це забезпечує контекстне сприйняття лексики, підвищує зрозумілість і дозволяє уникнути абстрактного ранжування слів, властивого традиційним частотним або статистичним методам.

Система інтервальних повторень. Для оптимізації запам’ятовування виділених пріорітетних слів система додає їх у SRS, що довела свою доцільність для даної задачі в ряді наукових досліджень [21, 22, 23]. SRS засновується на алгоримті SM-2, що є підтвердженим, стандартним алгоритмом для створення оптимального графіку повторень.

Переваги запропонованого підходу.

Подолання обмежень класичних методів: інтеграція когнітивних параметрів і контекстуального аналізу усуває проблеми поверхневого оцінювання та відсутності семантичної інформації.

Багатовимірне ранжування: поєднання статистичних, лінгвістичних та когнітивних характеристик забезпечує більш точне і релевантне формування списку слів для навчання.

Статична, але точна модель: незважаючи на відсутність адаптації під конкретного користувача, модель забезпечує стабільну оцінку складності і зменшує ризик спотворення результатів.

Контекстуалізація та пояснюваність: приклади речень і метрики складності роблять процес навчання прозорим і зрозумілим для користувача.

Ефективність обчислень: використання легких NLP-моделей дозволяє обробляти великі корпуси текстів швидко, без необхідності складного машинного навчання чи додаткового навчання моделей.

Як висновок: використання інтегрованого підходу, що поєднує NLP та антропоцентричні обчислення, дозволяє реалізувати ефективну систему оцінки складності та відбору лексики. Така система усуває недоліки традиційних підходів: поверхневість оцінок, ігнорування когнітивних аспектів, відсутність контексту та низьку персоналізацію. В результаті створюється адаптований список лексики, що підвищує ефективність вивчення іноземних слів і забезпечує більш точне та контекстно обґрунтоване навчання.

3 РОЗРОБКА ТА ОЦІНКА АЛГОРИТМУ ВИВЧЕННЯ ІНОЗЕМНИХ СЛІВ НА ОСНОВІ АНТРОПОЦЕНТРИЧНИХ ОБЧИСЛЕНЬ ТА МЕТОДІВ ОБРОБКИ ПРИРОДНОЇ МОВИ

3.1. Постановка задачі та вимоги до системи

Завдання розробки полягає у створенні алгоритму та програмного прототипу, що забезпечують автоматизоване формування індивідуальних лексичних колод для вивчення іноземних слів на основі методів обробки природної мови (NLP) та принципів антропоцентричних обчислень. Система повинна оптимізувати процес засвоєння лексики шляхом інтелектуального відбору, оцінки складності та організації інтервального повторення.

Метою є побудова навчального середовища, у якому користувач отримує набір лексичних одиниць, сформований не за частотою або випадковим принципом, а з урахуванням когнітивної складності, рідкості та значущості кожного слова в контексті.

Основна концепція.

Система поєднує обчислювальні підходи з когнітивною моделлю навчання. Її робота базується на трьох послідовних етапах:

- Автоматичний аналіз тексту. Введений користувачем текст обробляється засобами NLP для виокремлення лексичних одиниць, лематизації, очищення від стоп-слів і підрахунку частотності.
- Оцінка ваги слів. Кожне слово оцінюється за статистичними та когнітивними характеристиками, що дозволяє визначити рівень складності його запам’ятовування.
- Побудова колоди та управління повторенням. На основі отриманих ваг формується індивідуальна навчальна колода, що використовується у сесіях інтервального повторення (SRS).

Ключова відмінність системи полягає у її антропоцентричному підході: відбір слів відбувається з урахуванням природних когнітивних закономірностей людини, а не лише статистичних показників тексту.

Постановка задачі.

Необхідно розробити програмно-алгоритмічний комплекс, що виконує такі дії: Аналіз текстових даних із використанням NLP; Оцінку когнітивної та статистичної складності слів; Формування навчальних колод за визначеними критеріями; Реалізацію інтервального повторення для підкріплення засвоєного матеріалу; Збереження результатів аналізу, колод і статистики користувача в локальній базі даних.

Функціональні вимоги.

1. Обробка тексту.

   Система повинна виконувати токенізацію, лематизацію, визначення частоти вживання та виключення нерелевантних одиниць (числа, власні назви, знаки пунктуації). Обробка має бути повністю автоматизованою, забезпечуючи користувачеві швидкий результат після введення тексту.

2. Визначення ваги слова.

   Для кожного слова обчислюється індекс важливості за принципом TF-IDF або його аналогами. Вага коригується з урахуванням когнітивних факторів: довжини слова, морфологічної складності, частини мови та новизни (чи є слово вже відомим користувачеві).

3. Побудова навчальної колоди.

   Система формує колоду зі слів, що мають найвищі вагові показники. Визначається порогове значення, нижче якого слова не включаються, оскільки вважаються такими, що засвоюються природним шляхом під час читання.

4. Інтервальне повторення.

   Застосовується модифікований алгоритм SM-2 (SuperMemo) або його спрощена версія для управління повтореннями. Інтервали між

повтореннями визначаються успішністю користувача під час навчальних сесій.

5. Збереження даних.

   Уся інформація про тексти, слова, колоди та результати навчання зберігається в локальній базі даних SQLite. Це забезпечує автономність роботи системи без підключення до інтернету.

6. Користувацький інтерфейс.

   Інтерфейс включає три головні розділи:

   - сторінку аналізу тексту;
   - сторінку управління колодами;
   - сторінку SRS-сесії для повторення слів.

   Інтерфейс реалізований на основі Electron, React та Tailwind, що забезпечує швидку реакцію та просту навігацію.

Нефункціональні вимоги.

1. Швидкість. Обробка тексту середнього обсягу (1 тис. слів) повинна виконуватися менш ніж за 10 секунд.
2. Стабільність. Збереження даних має бути гарантованим навіть при аварійному завершенні роботи програми.
3. Модульність. Архітектура системи має дозволяти подальше розширення функціоналу (наприклад, додавання перекладу, статистики або адаптації рівня складності).
4. Гарний користувацьких досвід. Інтерфейс має бути зрозумілим без необхідності додаткових інструкцій.
5. Адаптивність. Система повинна враховувати попередні результати користувача для динамічного регулювання складності повторення.

Архітектурна модель.

Система побудована на основі таких технологій:

- Python — реалізація NLP-процесів, обчислення вагових коефіцієнтів та управління даними;
- Electron — середовище для створення десктопного застосунку;

- React + Tailwind — реалізація користувацького інтерфейсу;
- SQLite — база даних для зберігання колод, слів і результатів повторення.

Компоненти системи взаємодіють через локальний API, який забезпечує обмін даними між фронтендом та бекендом. Така структура дозволяє швидко обробляти інформацію, мінімізуючи затримки.

Очікувані результати.

Реалізація описаної системи забезпечує автоматизацію процесу підготовки навчального матеріалу; інтелектуальний відбір лексики на основі статистичних і когнітивних показників; персоналізацію процесу навчання; збереження стабільної продуктивності навіть при роботі з великими текстами; формування бази для подальшого розширення функціоналу.

Розроблений алгоритм сприяє оптимізації вивчення іноземної лексики, оскільки він фільтрує слова, які природно засвоюються через контекст, і концентрується на тих, що мають високу когнітивну складність. Це відповідає принципам антропоцентричного підходу, у якому технологічні рішення підпорядковуються закономірностям людського сприйняття та пам’яті.

3.2. Математична модель оцінки складності засвоєння лексики

Математична модель оцінки складності засвоєння лексики є основним елементом алгоритму формування навчальних колод у системі. Вона забезпечує кількісне представлення складності кожної лексеми на основі сукупності лінгвістичних, статистичних та морфологічних параметрів, що обчислюються автоматично з текстового корпусу користувача.

Модель складається з двох основних частин:

Оцінка когнітивної складності слова — обчислення показника $D_c(w)$ для кожного токена на основі його внутрішніх характеристик (довжина, кількість складів, частина мови, частота вживання, морфологічна форма).

Обчислення показника навчального пріоритету — визначення загальної ваги слова у процесі вивчення за допомогою формули комбінування TF-IDF [11], когнітивної складності та частотності.

1. Формалізація когнітивної складності слова.

Для кожного токена w із корпусу визначається частковий показник когнітивної складності $D_c(w)$, який приймає значення в інтервалі $[0;\ 1]$.

Обчислення цього показника відбувається за такою формулою (3.1):

$$D_c(w) = a_1 \cdot L(w) + a_2 \cdot S(w) + a_3 \cdot P(w) + a_4 \cdot \big(1 - F_{lex}(w)\big) + a_5 \cdot M(w) + a_6 \cdot C(w), \quad (3.1)$$

де $L(w)$ — нормалізована довжина слова;

$S(w)$ — нормалізована кількість складів;

$P(w)$ — складність частини мови;

$F_{lex}(w)$ — частотність слова у словнику spaCy (індекс рангу);

$M(w)$ — показник морфологічної нерегулярності;

$C(w)$ — показник конкретності частини мови;

$a_1 \dots a_6$ — нормовані коефіцієнти ваги, що визначають вплив кожного фактора на загальну складність.

Кожен із параметрів має значення в межах [0; 1] і обчислюється автоматично за емпіричними правилами.

1.1. Довжина слова

Довжина відображає кількість символів у слові відносно середнього максимального значення. Нормалізована метрика (3.2):

$$L(w) = \frac{len(w)}{12}, \quad (3.2)$$

де 12 — умовна константа, що відповідає середній довжині складних англійських слів.

1.2. Кількість складів

Кількість складів визначається регулярним виразом, який підраховує послідовності голосних літер. Результат нормалізується до діапазону [0; 1] (3.3):

$$S(w) = \frac{\text{склади}(w)}{4}, \quad (3.3)$$

де 4 — максимальна очікувана кількість складів для більшості звичайних лексем.

1.3. Частина мови

Для частин мови призначаються вагові коефіцієнти залежно від складності запам’ятовування. Наприклад, дієслова мають вищу складність через варіативність форм, а прикметники — середню (3.4).

$$P(w) = \begin{cases} 0.7, & \text{якщо POS} = \text{VERB} \\ 0.5, & \text{якщо POS} = \text{ADJ або ADV} \\ 0.3, & \text{якщо POS} = \text{NOUN} \\ 0.1, & \text{якщо POS} = \text{DET, ADP, CCONJ, тощо} \\ 0.4, & \text{для інших випадків} \end{cases} \quad (3.4)$$

1.4. Частотний компонент

Частота береться з внутрішнього словника моделі spaCy (lex.rank). Цей параметр показує поширеність слова в мовному корпусі.

Частотна складова визначається як (3.5):

$$F_{lex}(w) = 1 - \min\left(\frac{rank(w)}{50000}, 1\right). \quad (3.5)$$

Таким чином, рідковживані слова (з високим рангом) мають вищий внесок у складність.

1.5. Морфологічна нерегулярність

Якщо лема слова відрізняється від його поверхневої форми, вважається, що воно має морфологічну варіативність, що підвищує складність (3.6).

$$M(w) = \begin{cases} 0.1, & \text{якщо } \text{lemma}(w) \neq \text{text}(w) \\ 0.0, & \text{інакше} \end{cases} \quad (3.6)$$

1.6. Конкретність частини мови

Оцінка конкретності визначається емпірично: іменники вважаються найбільш конкретними, дієслова — менш, а прислівники — абстрактними (3.7).

$$C(w) = \begin{cases} 0.2, & \text{якщо POS} = \text{NOUN} \\ 0.6, & \text{якщо POS} = \text{VERB} \\ 0.4, & \text{якщо POS} = \text{ADJ} \\ 0.5, & \text{якщо POS} = \text{ADV} \\ 0.5, & \text{для інших випадків} \end{cases} \quad (3.7)$$

2. Ключова модель навчального пріоритету.

Після обчислення когнітивної складності для всіх лексем формується узагальнений показник навчального пріоритету $LPS(w)$, який визначає доцільність включення слова до навчальної колоди.

Формула розрахунку (3.8):

$$LPS(w) = \alpha \cdot IDF(w) + \beta \cdot D_c(w) + \gamma \cdot \big(1 - F_{text}(w)\big), \quad (3.8)$$

де $IDF(w)$ — обернена частота документів, обчислена з TF-IDF [11];

$D_c(w)$ — когнітивна складність, описана вище;

$F_{text}(w)$ — нормалізована частота появи слова в корпусі користувача,

α, β, γ — вагові коефіцієнти (у поточній реалізації α =0.4, β =0.4, γ=0.2).

Таким чином, модель поєднує три ключові аспекти:

- інформативність слова (TF-IDF) — визначає, наскільки слово унікальне для корпусу користувача;
- когнітивну складність (D_c) — відображає внутрішню складність сприйняття слова людиною;
- реальну частоту вживання (F_{text}) — дозволяє знизити вагу надто поширених слів, які користувач засвоїть природним шляхом.

3. Нормалізація та пороговий відбір

Після обчислення $LPS(w)$ для всіх лексем формується розподіл значень, на основі якого визначається порогове значення T_{LPS} (3.9):

$$T_{LPS} = P_{1-p}(LPS), \quad (3.9)$$

де $P_{1-p}(LPS)$ — $(1-p)$-й процентиль, а p=0.4 — емпірично встановлений рівень вибірки.

Слова, для яких $LPS(w) \geq T_{LPS}(w)$, включаються до навчальної колоди, оскільки вони поєднують високу когнітивну складність і низьку частотність, що робить їх менш доступними для природного засвоєння.

4. Приклад обчислення

Нехай для слова “architecture” маємо:

довжина 12 символів → L=1.0,

4 склади → S=1.0,

POS = NOUN → P=0.3,

частота у словнику низька → F_{lex}= 0.2,

морфологічна варіація наявна → M=0.1,

конкретність середня → C=0.4.

Тоді:

$$D_c = 0.2(1) + 0.2(1) + 0.2(0.3) + 0.2(0.8) + 0.1(0.1) + 0.1(0.4) = 0.67$$

Нехай IDF=3.5, $F_{text} = 0.001$.

Тоді:

$$LPS = 0.4(3.5) + 0.4(0.67) + 0.2(1 - 0.001) = 1.4 + 0.268 + 0.1998 = 1.8678$$

Якщо поріг $T_{LPS} = 1.5$, в цьому випадку це слово потрапляє до колоди.

5. Висновок

Побудована модель оцінки складності засвоєння лексики інтегрує кілька рівнів лінгвістичного аналізу — статистичний (TF-IDF), морфологічний та когнітивний. Завдяки цьому система автоматично виявляє ті лексеми, які є найскладнішими для запам’ятовування, і відбирає їх для подальшого навчання в межах SRS-механізму.

Модель відзначається автономністю, оскільки не вимагає зовнішніх словників чи машинного навчання. Її структура є визначеною, а параметри мають чітке математичне тлумачення, що забезпечує прозорість, відтворюваність і можливість подальшої адаптації для інших мовних пар або корпусів.

3.3. Архітектура та функціональні модулі системи

Архітектура розробленої системи побудована за гібридною клієнт–серверною моделлю з інтеграцією вебтехнологій та локальних обчислювальних сервісів. Вона забезпечує високий рівень інтерактивності інтерфейсу користувача при збереженні можливостей для інтенсивної обробки природної мови на стороні Python. Система реалізована у вигляді кросплатформового застосунку на базі

Electron, що поєднує JavaScript-інтерфейс (React + Tailwind) із серверною логікою, написаною на Python.

Архітектура системи передбачає модульний підхід, який дає змогу незалежно розвивати компоненти, відповідальні за лінгвістичний аналіз, управління навчальними колодами, планування повторень та візуальне представлення результатів користувачу.

1. Загальна структура.

Система складається з трьох основних рівнів:

- Інтерфейс користувача (Electron/React Layer) – забезпечує взаємодію користувача із системою, введення тексту, перегляд результатів аналізу, формування навчальних колод і роботу з картками повторення.
- Серверна логіка (Python Backend Layer) – реалізує алгоритми обробки природної мови, обчислення складності лексики, формування SRS-плану та управління базою даних.
- Система збереження даних (SQLite Layer) – відповідає за зберігання користувацьких колод, статистики повторень та історії навчання.

Між рівнями здійснюється обмін даними через внутрішній API, який використовує протокол міжпроцесної взаємодії (IPC) Electron для передачі структур JSON між JavaScript та Python. Така архітектура дозволяє поєднувати зручність браузерного інтерфейсу з обчислювальною потужністю серверних алгоритмів NLP.

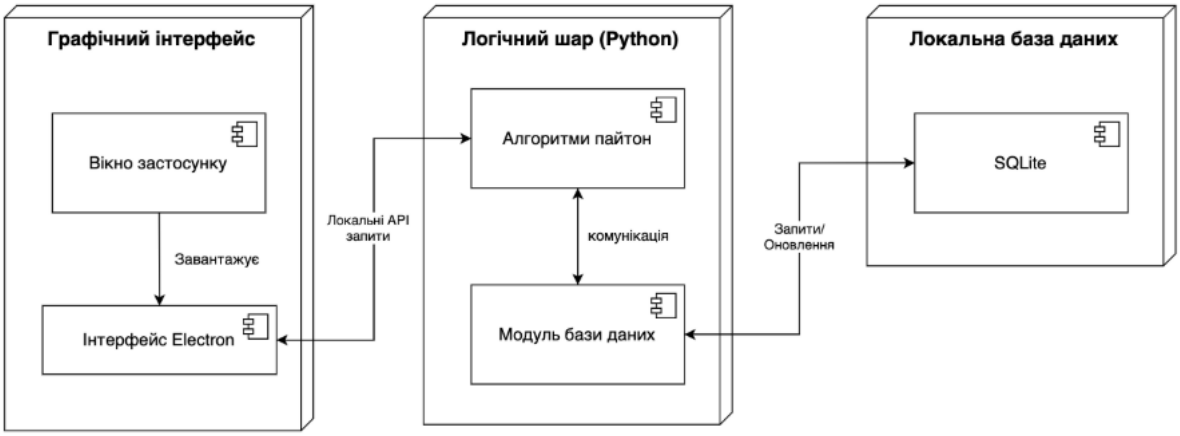


Рис. 3.1 Діаграма розгортання застосунку

2. Користувацький інтерфейс

Інтерфейс системи побудований з використанням бібліотеки React і фреймворку TailwindCSS, що забезпечує динамічну взаємодію з користувачем та адаптивність до різних екранів.

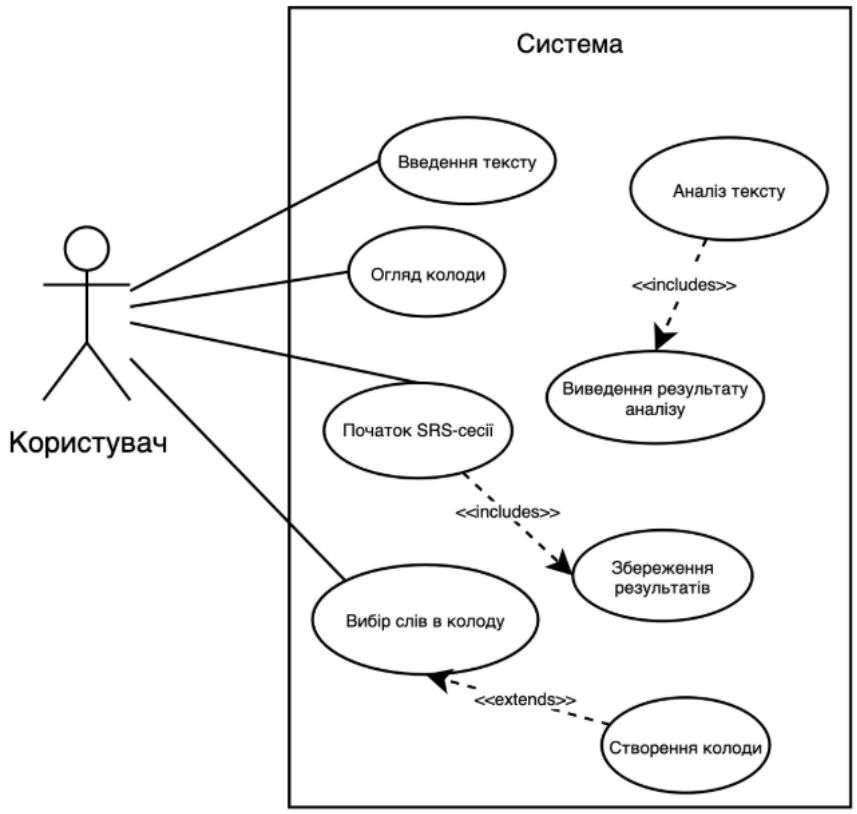


Рис. 3.2 Діаграма варіантів використання системи

Ключові компоненти інтерфейсу:

2.1. Сторінка WordMining

Цей модуль реалізує механізм попереднього аналізу тексту, який користувач вводить для обробки.

Функції включають: поле введення або вставки тексту; кнопка Analyze Sentence, що ініціює виклик до сервісу nlp_service; кнопка Build Deck, яка формує колоду на основі відібраних лексем; бічна панель попереднього перегляду карток.

Бічна панель відображає список лексем із вирахуваними параметрами складності, частоти та важливості. Користувач може вручну включати або виключати слова перед формуванням навчальної колоди.

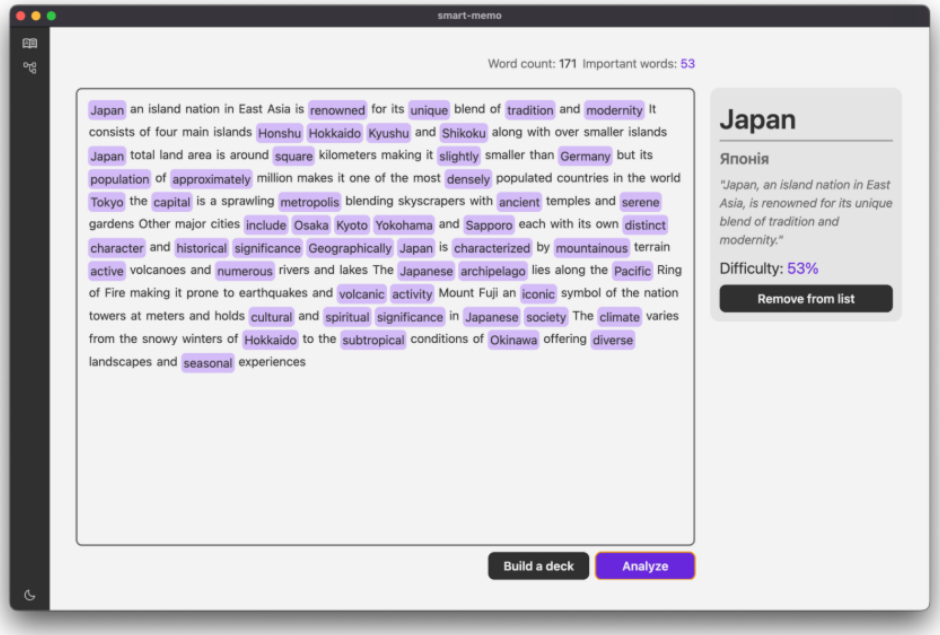


Рис. 3.3 Екран сторінки аналізу тексту

2.2. Сторінка Decks

Цей модуль реалізує огляд створених користувачем колод. Виводиться інформація про: кількість нових, запланованих до повторення та прострочених карток; загальний прогрес у межах кожної колоди; можливість запуску сесії повторення.

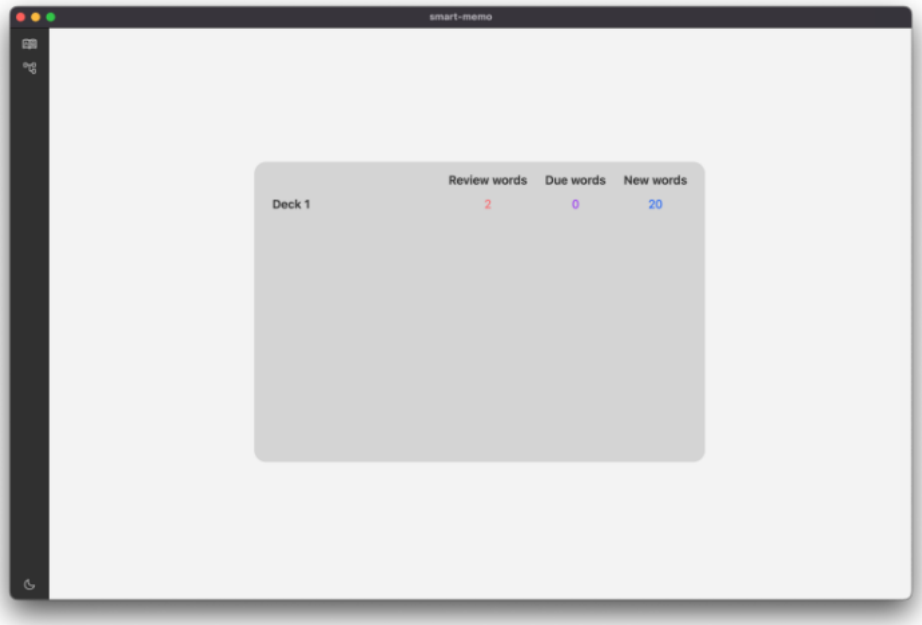


Рис. 3.4 Екран сторінки колоди слів користувача

2.3. Сторінка SpacedRepetition

Функціонально модуль відповідає реалізації механізму інтервального повторення.

Кожна картка має атрибути стану: new, learning, review, lapsed. Інтервали повторення коригуються на основі оцінки користувача після відповіді (“Again”, “Hard”, “Good”, “Easy”), що передається в Python-модуль srs_engine для оновлення графіка повторів.

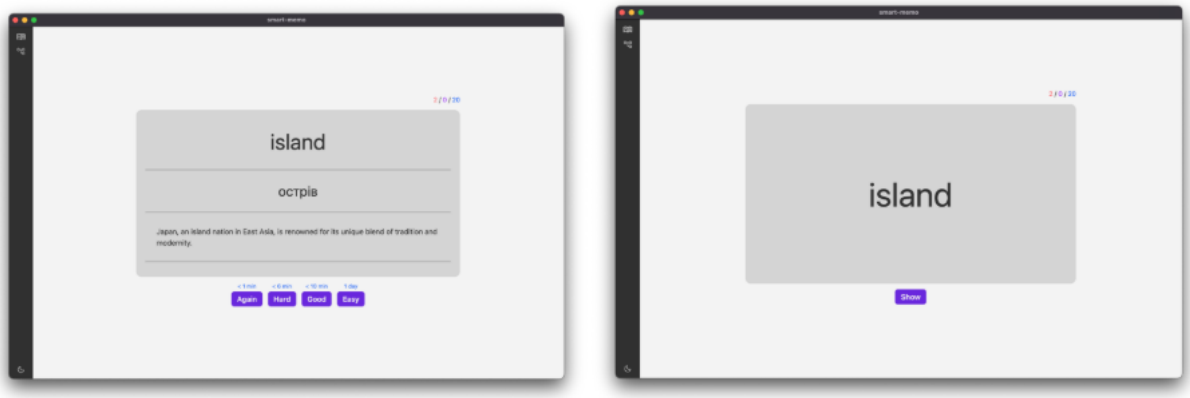


Рис. 3.5 Екрани сторінки інтервальних повторень

3. Серверна частина

Серверна частина системи реалізує три основних сервіси: nlp_service, srs_engine та user_deck. Кожен з них виконує окремі функції в межах єдиної архітектури.

3.1. Модуль nlp_service

Модуль відповідає за попередню обробку та аналіз тексту. Основні завдання:

- токенізація, лематизація, визначення частин мови (на базі spaCy);
- обчислення частотності лексем у межах користувацького корпусу;
- розрахунок індексу складності (див. модель у п. 3.2);
- комбінування показників TF-IDF, частотності та когнітивної складності для визначення Learning Priority Score (LPS);
- формування структури результату, який включає переклад, приклад речення та позначку included.

Таким чином, nlp_service виступає центральним аналітичним компонентом системи, який визначає релевантність кожної лексеми для включення до колоди.

3.2. Модуль srs_engine

Цей компонент відповідає за планування інтервалів повторення. Він реалізує алгоритм інтервального повторення, заснований на принципах SM-2, що враховує якість відповіді користувача. Для кожної картки підтримуються такі параметри, як repetitions (кількість успішних повторів); interval (поточний інтервал у днях); ease_factor (коефіцієнт складності); due_date (дата наступного повторення).

Розрахунок нових інтервалів здійснюється за класичною формулою (3.10):

$$I_{n+1} = I_n \cdot E, \tag{3.10}$$

де I_n — поточний інтервал;

E — коефіцієнт легкості, який коригується залежно від відповіді користувача.

3.3. Модуль user_deck

Забезпечує з’єднання між інтерфейсом користувача та серверною логікою повторення.

Основні функції включають створення нових колод після обробки тексту, оновлення стану карток після сесії повторення, надання даних для сторінки Decks (статистика та кількість карток за статусами), синхронізацію із базою даних SQLite.

Модуль виступає API-шаром для управління колодами на рівні користувача.

4. Система збереження даних

Для зберігання даних застосовується легка реляційна СУБД SQLite. Вона забезпечує зберігання таких сутностей:

- Decks — інформація про колоди користувача (ідентифікатор, назва, дата створення);
- Cards — окремі картки з атрибутами складності, леми, перекладу, прикладу речення, інтервалів повторення;
- Statistics — збереження історії повторень і результатів навчання.

Вибір SQLite зумовлений компактністю, відсутністю потреби в окремому сервері та повною інтегрованістю в локальні програми.

5. Взаємодія між компонентами

Процес обробки даних відбувається послідовно:

1. Користувач вводить текст на сторінці WordMining і натискає кнопку Analyze Sentence.
2. Electron через IPC передає текст до Python-модуля nlp_service.
3. nlp_service аналізує корпус, обчислює складність і повертає JSON із результатами (леми, складність, LPS, переклад, приклади).
4. Інтерфейс відображає результати у вигляді інтерактивного списку з можливістю ручного вибору лексем.
5. Після натискання Build Deck вибрані слова передаються до модуля user_deck, який створює відповідну колоду в базі SQLite.
6. Під час повторення модуль srs_engine оновлює інтервали та статистику карток, передаючи зміни через user_deck у базу даних.

Таким чином, забезпечується замкнутий цикл взаємодії між користувачем, аналітичним механізмом і системою повторення.

6. Принципи архітектурного проєктування

Архітектура системи дотримується таких принципів:

- Модульність — кожен функціональний компонент може розвиватися незалежно;
- Розширюваність — можливість додавання нових алгоритмів або типів контенту (аудіо, відео, діалоги);
- Прозорість обчислень — усі етапи аналізу виконуються детерміновано, без використання “чорних скриньок”;
- Інтерактивність — користувач має контроль над процесом формування колод і вибору слів;
- Кросплатформеність — завдяки Electron застосунок працює на Windows, macOS і Linux.

7. Висновок

Архітектура системи забезпечує глибоку інтеграцію аналітичних обчислень на Python із сучасним графічним інтерфейсом користувача, що дозволяє поєднати когнітивну модель навчання з практичною зручністю використання. Завдяки модульній структурі система залишається гнучкою, розширюваною та придатною до подальшого розвитку — зокрема, інтеграції нових мов, моделей NLP або аналітики користувацького прогресу.

3.4. Практичний внесок системи

Розроблена система — це інструмент для оптимізації вивчення лексики, який поєднує сучасні методи NLP, оцінку когнітивної складності й адаптивне інтервальне повторення. Вона автоматично відбирає й ранжує слова з корпусу користувача, оцінює їхню складність та формує персоналізовані колоди для повторення. Основна практична цінність полягає у фокусуванні навчання на словах, які важко запам’ятати природним шляхом, на відміну від традиційних SRS, що однаково трактують усю лексику.

Система аналізує будь-який користувацький контент, виділяючи слова, що мають найбільшу навчальну цінність. Використання TF-IDF дозволяє відфільтровувати тривіальну лексику, а когнітивна модель — враховувати довжину, морфологію, частину мови та частотність. Об’єднання цих факторів формує Learning Priority Score — ключовий критерій відбору слів.

Така інтеграція статистичних і когнітивних характеристик дає змогу будувати ефективні колоди та уникати зайвого повторення легких слів. Користувач може вручну коригувати підбір, що підвищує контроль і мотивацію. Модуль SRS будує індивідуальні графіки повторень, динамічно змінюючи інтервали залежно від труднощі слова й успішності користувача.

Система реалізована як кросплатформовий застосунок (Electron, React), а Python забезпечує аналіз корпусів та розрахунок складності. SQLite гарантує локальне збереження даних без потреби у сервері. Архітектура гнучка й дозволяє

розширення: додавання семантичних моделей, нових когнітивних метрик чи підтримку різних типів контенту — від книг до субтитрів.

Рішення демонструє, що поєднання NLP і антропоцентричних методів підвищує ефективність, зменшує когнітивне навантаження та підтримує автономне навчання. Воно придатне як для індивідуального використання, так і для інтеграції у мовні платформи, які хочуть автоматично добирати лексику на основі контенту користувача. Система забезпечує персоналізований адаптивний процес, що може стати основою для більш складних рішень із семантичними та мультимодальними можливостями.

3.5 Оцінка запровадженого алгоритму

Оцінка ефективності розробленого алгоритму є ключовим етапом у перевірці його здатності автоматично визначати пріоритетні слова для вивчення та формувати оптимальні навчальні набори. Для забезпечення комплексної та об’єктивної оцінки алгоритм тестується за трьома взаємопов’язаними напрямами:

1. Порівняння з базовими підходами – тестування алгоритму проти традиційних методів, таких як TF-IDF та частотний аналіз, дозволяє оцінити його здатність виділяти значущі лексичні одиниці та формувати більш ефективні набори для повторення.
2. Тест стабільності – перевірка стабільності та надійності алгоритму при різних входових даних або варіаціях текстового корпусу. Мета цього тесту – визначити, наскільки послідовно алгоритм ранжує слова і чи не змінюється пріоритетність ключових слів при невеликих змінах у корпусі.
3. Валідація складності через CEFR – оцінка того, наскільки обчислена складність слів та їх пріоритетність корелює з відомими лексичними стандартами, таким як CEFR, що дозволяє перевірити когнітивну та навчальну релевантність алгоритму.

Для кожного напряму обираються відповідні кількісні та візуальні метрики, що дозволяють порівняти результати алгоритму із базовими підходами, оцінити

його стабільність та відповідність лексичним стандартам. Використання таблиць, гістограм, діаграм розсіювання та коефіцієнтів подібності дозволяє наочно демонструвати ефективність алгоритму та робити науково обґрунтовані висновки щодо його застосовності у процесі навчання іноземної лексики.

Далі детально описані результати кожного з трьох тестів, їх порівняння з базовими методами та аналіз сильних і слабких сторін алгоритму з точки зору точності, надійності та навчальної ефективності.

1. Порівняння з базовими підходами.

Для оцінки здатності алгоритму автоматично визначати значущі слова та формувати оптимальні навчальні набори ми порівняли його результати з двома традиційними методами: TF-IDF та частотним аналізом. Розгляленмо метрики для аналізу.

Коефіцієнт Жаккара – вимірює схожість множин виділених слів: чим більше перетин з базовими методами, тим ближчий результат алгоритму до традиційних підходів.

Коефіцієнт кореляції рангу Спірмена – оцінює схожість ранжування слів між алгоритмом та TF-IDF; відображає, наскільки подібний порядок пріоритетності слів.

Візуалізація розподілу LPS та порівняння з TF-IDF – дозволяє наочно бачити скупчення слів, їх оцінку пріоритетності та відповідність частотним та інформативним характеристикам.

Як тестову вибірку було обрано три тексти, кожні по приблизно 100 слів.

Результати:

- Коефіцієнт Жаккара в порівнянні з базовим TF-IDF: 0.304
- Коефіцієнт Жаккара в порівнянні з частотністю слова: 0.200
- Коєфіцієнт Спірмена з базовим TF-IDF: 0.404

На основі цих даних проведемо аналіз.

Коефіцієнт Жаккара:

Алгоритм виділяє близько 30% слів, що збігаються з TF-IDF, та 20% – з частотним аналізом. Це свідчить, що наша модель не просто відбирає найчастіші

чи найінформативніші слова за TF-IDF, а комбінує кілька факторів (частота, складність, IDF), що формує оригінальний набір пріоритетних слів, зберігаючи при цьому часткову сумісність з класичними методами.

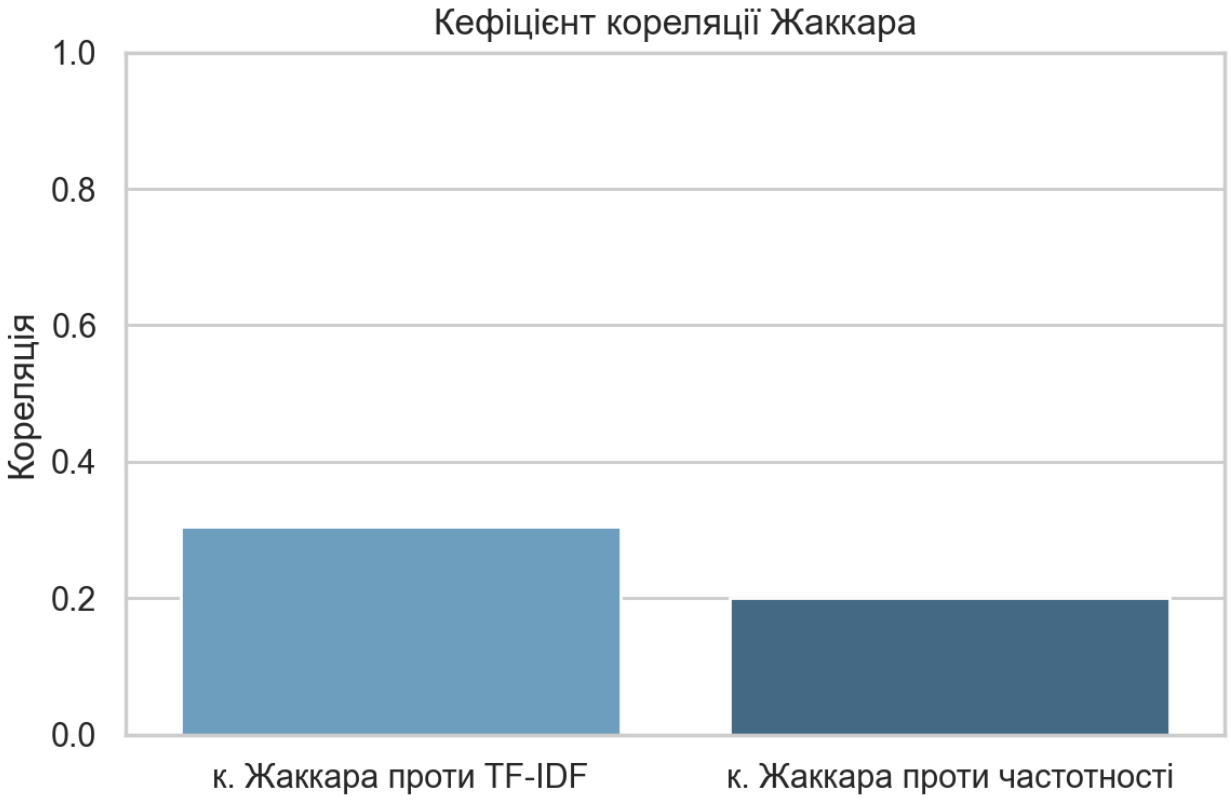


Рис. 3.6 Діаграма порівняння LPS проти TF-IDF та частотності

Коєфіцієнт Спірмена при порівнянні впровадженного LPS значення проти TF-IDF:

Розкид точок демонструє три основні кластери.

Одинокий крайній випадок (1; 0.8) може відповідати слову з низьким TF-IDF, але високим LPS через складність або рідкість у корпусі.

Малий кластер близько x=1.28 показує слова з помірною LPS, що частково узгоджуються з TF-IDF.

Великий кластер на x≈1.69 показує слова, для яких TF-IDF майже однаковий, але LPS різниться. Це демонструє, що алгоритм враховує не лише інформативність слова в корпусі (TF-IDF), а й когнітивні характеристики (довжина слова, складність, частота, морфологія), дозволяючи диференціювати пріоритети між словами з однаковим TF-IDF.

Спірменова кореляція 0.40 показує помірну узгодженість ранжування, що є очікуваним: алгоритм враховує складність та інші параметри, тому не повністю повторює TF-IDF.

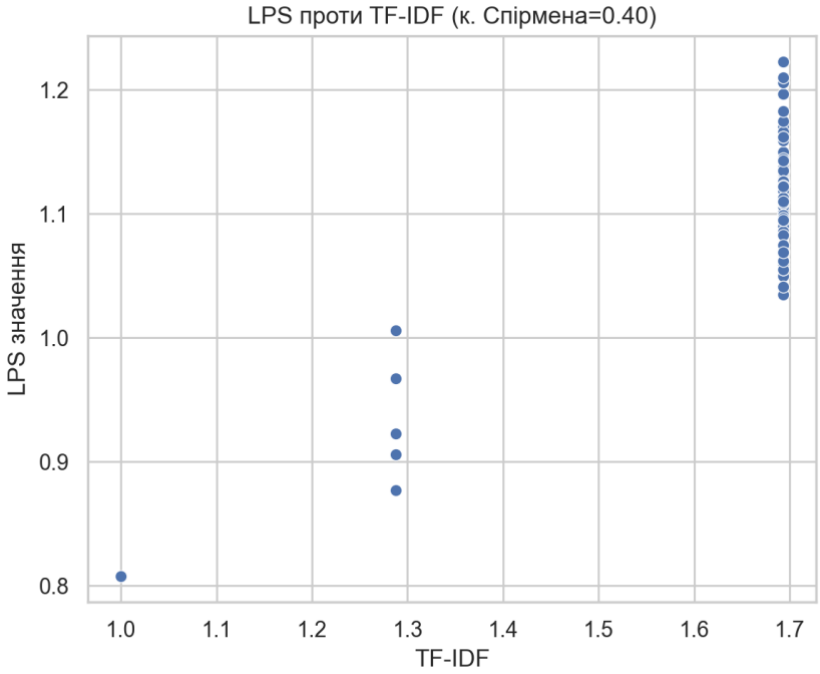


Рис. 3.7 Діаграма порівняння LPS проти TF-IDF

Розподіл LPS значень:

Основна маса слів зосереджена між 0.9–1.0 LPS, що формує чіткий піковий сегмент для пріоритетного навчання.

Діапазон LPS 0.81–1.22 демонструє, що алгоритм може диференціювати легкі та складні слова, створюючи адаптивні навчальні набори.

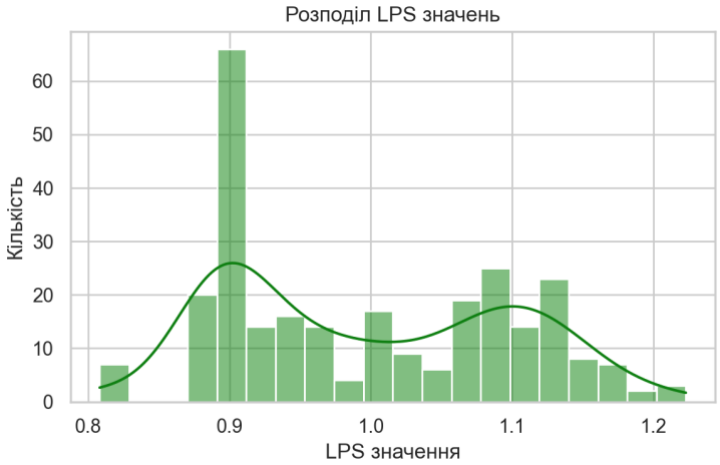


Рис. 3.8 Діаграма розподілу значень LPS

Розроблений алгоритм показує об’єктивну відмінність від базових методів, виділяючи слова з високим навчальним пріоритетом, що враховують як частотність, так і когнітивну складність.

Водночас часткова схожість з TF-IDF та частотним аналізом підтверджує, що алгоритм не втрачає інформаційно значущих слів і відтворює базові закономірності.

Візуалізація та метрики дозволяють наочно коментувати силу алгоритму, показуючи його здатність формувати навчальні набори, які одночасно інформативні та диференційовані за складністю.

2. Тест стабільності.

Для оцінки стабільності алгоритму було проведено експеримент, у якому порядок речень у корпусі випадково змінювався, після чого повторно обчислювалися ранги слів за LPS. Метою тесту було перевірити, наскільки алгоритм чутливий до перестановки речень і чи зберігається послідовність вибраних слів та їхні оцінки складності.

Як метрики використовувалися:

- Середній коефіцієнт Спірмена між ранжуваннями оригінального та переставленого корпусу (Spearman correlation), що показує узгодженість порядку слів;
- Середній показник перетину множин вибраних слів (selection overlap), який відображає стабільність у включенні слів до фінального списку;
- Середнє абсолютне відхилення LPS (LPS variation), що дозволяє оцінити, наскільки змінюються самі оцінки складності.

Результати для корпусу з двох текстів по 100 слів показали надзвичайно високу стабільність:

Середній коефіцієнт Спірмена склав 0.997, стандартне відхилення — 0.004, що вказує на майже повну збереженість порядку слів навіть після перестановки речень.

Показник перетину множин вибраних слів досяг 1.0, тобто всі ключові слова були збережені у фінальному списку незалежно від порядку речень.

Середнє відхилення LPS склало лише 0.00027, що свідчить про практично нульові зміни у значеннях оцінки складності.

Наступний графік ілюструє розподіл кореляцій у 20 випадкових перестановках: спостерігаються два основні стовпці — один біля 0.991–0.992 з частотою 12 і більший стовпець біля 0.999–1.0 з частотою 8. Це наочно демонструє, що більшість перестановок не впливають на ранжування, а лише невелика частина випадків демонструє незначне зниження кореляції.

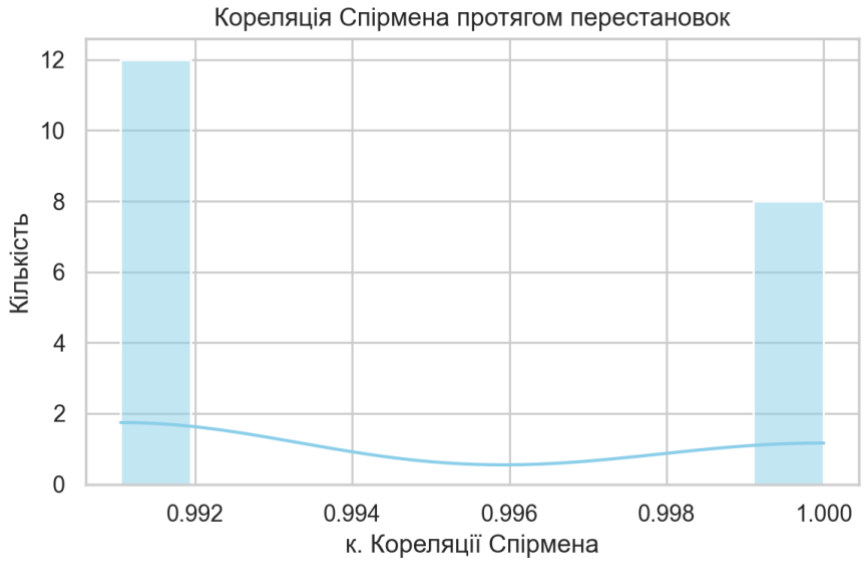


Рис. 3.9 Діаграма кореляції Спірмена протягом перестановок речень в корпусі

Отримані результати підтверджують високу стабільність алгоритму, що є важливим для практичного застосування в навчальному процесі: користувачі отримують послідовне ранжування ключових слів незалежно від локальних змін структури тексту.

3. Валідація складності через CEFR.

Наведені тести демонструють кількісну та якісну поведінку моделі LPS у співвідношенні з рівневою шкалою CEFR - загальноєвропейські рекомендації з мовної освіти. Вони включають збірник слів та їх відповідний орієнтовний рівень за стантардом рівня володіння мовою: A1, A2, B1, B2, C1, C2.

Перший експеримент порівнює кількість включених слів за LPS із значенням TF-IDF. На отриманому графіку видно, що TF-IDF генерує найбільшу кількість лем на рівнях A2 (≈13 слів) та B1 (≈7 слів), тоді як рівні C1 і C2 представлені значно слабше (≈5 і ≈9 слів відповідно). Такий розподіл відповідає типовій поведінці частотних моделей, що віддають перевагу простішим і більш уживаним словам.

У протилежність цьому, LPS демонструє виражений зсув у бік складнішої лексики. На графіку видно, що рівень B2 охоплює найбільшу кількість включених слів — приблизно 22. Це суттєво перевищує як відповідний показник baseline (≈11), так і значення для інших рівнів. Також спостерігається значна представленість рівнів C1 (≈11 слів) та C2 (≈12 слів). Рівні A1 та A2 майже не фігурують у підсумковому наборі — для A1 LPS взагалі не включив жодного слова, а для A2 лише близько 11. Такий розподіл уже на першому етапі свідчить про те, що модель не покладається виключно на частотність і системно виокремлює лексику середньо-високої складності, яку baseline не здатний виділити.

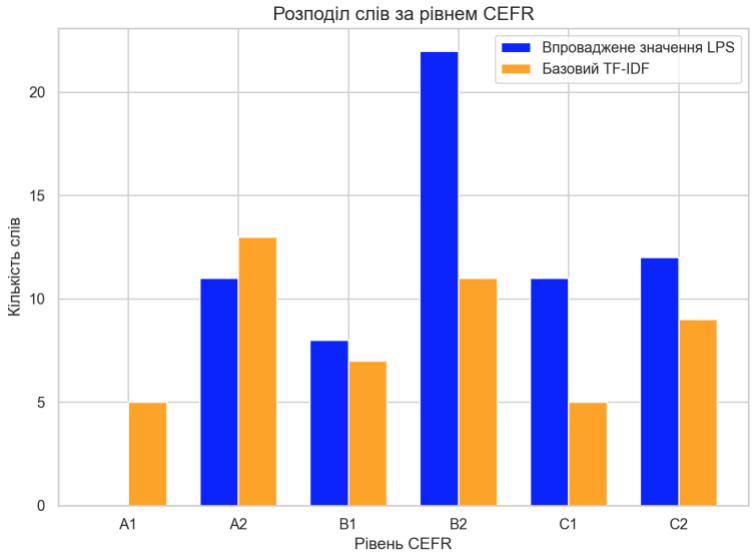


Рис. 3.10 Діаграма розподілу слів за рівнем CEFR (LPS проти TF-IDF)

Другий тест підтверджує ці тенденції в нормалізованому вигляді. Розподіл відсоткової частки включених LPS-слів серед CEFR-рівнів демонструє, що найбільшу частку становить рівень B2 — 34.4% усіх включених одиниць. Рівні C1 і C2 дають відповідно 17.2% та 18.8%, тобто разом майже 36% вибірки. Натомість рівні A1–A2 представлені мінімально: для A1 зафіксовано 0%, для A2 — 17.2%.

Рівень B1 займає 12.5%, що узгоджується з тим, що цей рівень зазвичай містить лексичний матеріал середньої складності, але менш інформативний порівняно з B2.

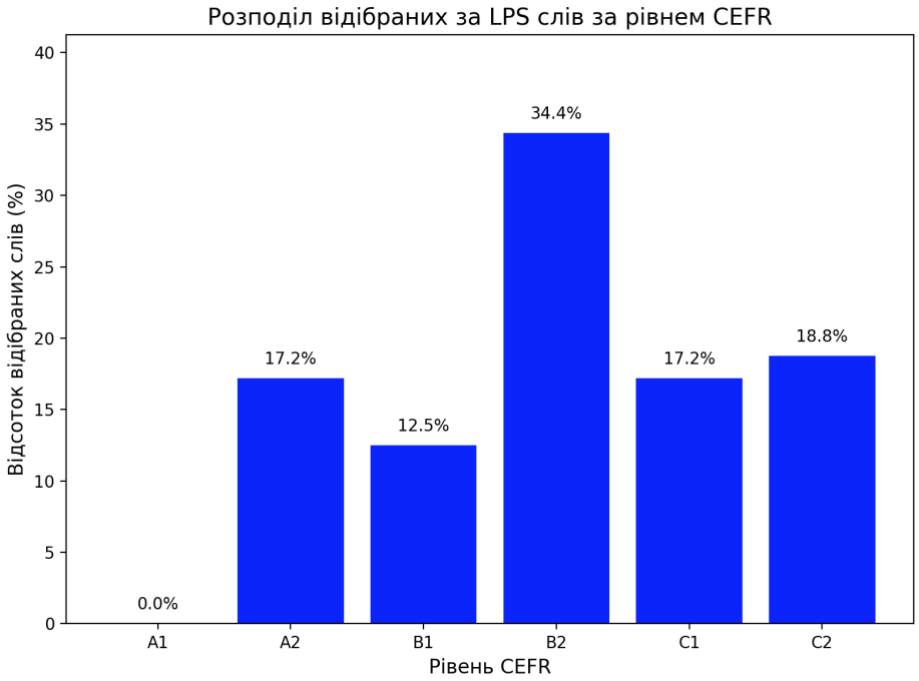


Рис. 3.11 Діаграма розподілу відібраних за LPS слів за рівнем CEFR

Сукупність цих даних підтверджує, що LPS систематично надає пріоритет словам, які марковані рівнями B2–C2. Порівняно з частотним baseline, який відтворює очікуваний пік у нижчих рівнях (A2–B1), модель LPS демонструє спрямовану поведінку: відбір переважно лексики, що виходить за межі елементарного та базового рівня і потенційно сприяє подальшому мовному просуванню. Це свідчить про те, що модель коректно узгоджується з рівневою структурою CEFR у частині виділення складнішої та навчально релевантної лексики, а результати двох тестів — як у вигляді абсолютних значень, так і у відсотковому форматі — демонструють стабільність цього ефекту.

ВИСНОВКИ

У результаті проведеного дослідження виконано комплексний аналіз сучасних підходів до визначення складності лексичних одиниць та формування навчальних наборів для засвоєння іноземної лексики. Теоретичний огляд продемонстрував, що частотні моделі, TF-IDF, традиційні системи повторення та рівневі метрики складності забезпечують корисні орієнтири, однак не враховують індивідуальних властивостей користувацького тексту та не інтегрують когнітивних характеристик, що впливають на навчальне навантаження. Розбіжність між принципами інформативності, частотності та когнітивної складності підтвердила необхідність комбінованого підходу до відбору лексики.

На основі цього було сформульовано та реалізовано алгоритм статистичного лексичного відбору, що використовує інтегровану систему показників: статистичні параметри корпусу, інформативність терміна та когнітивні фактори, зокрема довжину, морфологічну структуру та орієнтовну складність слова. Алгоритм призначений для автоматичного ранжування лексем відповідно до їх навчальної цінності та для формування компактних, релевантних наборів слів.

Практична реалізація системи в середовищі Python та Electron/React забезпечила можливість повного циклу обробки: токенізацію, лематизацію, обчислення складності та формування карток із прикладами.

Порівняння з базовими методами показало, що відбір алгоритму частково співпадає з традиційними підходами, але формує змістовно відмінні навчальні множини. Коефіцієнт Жаккара становив 0.304 у порівнянні з TF-IDF та 0.200 у порівнянні з частотним аналізом, що підтверджує: модель не повторює класичні стратегії, а створює власний тип наборів, які враховують додаткові параметри складності. Коефіцієнт рангової кореляції Спірмена 0.404 засвідчив помірну узгодженість із TF-IDF при збереженні відмінної структури пріоритетів.

Тест стабільності засвідчив високу надійність алгоритму: середній коефіцієнт Спірмена між оригінальними та переставленими корпусами становив

0.997, а середнє відхилення LPS — лише 0.00027. Водночас 100% ключових слів залишалися у фінальних наборах незалежно від змін порядку речень, що підтверджує відсутність чутливості до поверхневих структурних варіацій тексту.

Валідація через CEFR продемонструвала спрямованість моделі на відбір лексики середньої та підвищеної складності. Рівень B2 охопив 34.4% вибраних слів, тоді як рівні C1 і C2 сукупно — 36%. Натомість рівні A1–A2 були представлені мінімально (0% і 17.2%), що принципово відрізняється від частотного показника, орієнтованого на простішу лексику. Така поведінка підтверджує відповідність алгоритму навчальним потребам користувача на рівнях від B1 і вище.

Таким чином, розроблена система продемонструвала здатність формувати стабільні, когнітивно релевантні та статистично обґрунтовані набори лексики. Отримані кількісні показники свідчать про ефективність інтегрованого підходу, що поєднує NLP-методи та когнітивні моделі складності, та підтверджують практичну доцільність застосування алгоритму у задачах індивідуалізованого мовного навчання.

Робота пройшла апробацію. За її результатами було опубліковано наступні тези доповідей

1. Соколовський А.В., Герцюк М.М. Аналіз технологій для створення застосунку оптимізації вивчення іноземних слів. II міжнародна науково-практична конференція «Сучасні аспекти діджиталізації та інформатизації в програмній та комп'ютерній інженерії», 19-21 грудня 2024 р., Київ, Державний університет інформаційно-комунікаційних технологій. Збірник тез. К.: ДУІКТ, 2024. С.238-239.
2. Соколовський А.В., Золотухіна О.А. Підвищення ефективності вивчення іноземних слів використовуючи методи обробки природної мови та систему розподілених повторень. Всеукраїнська науково-технічна конференція «Застосування програмного забезпечення в ІКТ», 24 квітня 2025 р., Київ, Державний університет інформаційно-комунікаційних технологій. Збірник тез. К.: ДУІКТ, 2025. С.243-245.

ПЕРЕЛІК ПОСИЛАНЬ

1. Al-Khasawneh F. M. The acquisition of foreign language vocabulary: Does spacing effect matter?. *The Education and science journal*. 2023. Vol. 25, no. 3. P. 174–193. URL: https://doi.org/10.17853/1994-5639-2023-3-174-193 (date of access: 11.11.2025).
2. SPACED VOCABULARY ACQUISITION WHILE INCIDENTAL LISTENING BY ESL UNIVERSITY STUDENTS / T. Zubenko et al. *Advanced Education*. 2022. P. 79–87. URL: https://doi.org/10.20535/2410-8286.250501 (date of access: 11.11.2025).
3. Bower J. V., Rutson-Griffiths A. The relationship between the use of spaced repetition software with a TOEIC word list and TOEIC score gains. *Computer Assisted Language Learning*. 2016. Vol. 29, no. 7. P. 1238–1248. URL: https://doi.org/10.1080/09588221.2016.1222444 (date of access: 11.11.2025).
4. Ye J., Su J., Cao Y. A stochastic shortest path algorithm for optimizing spaced repetition scheduling. KDD '22: the 28th ACM SIGKDD conference on knowledge discovery and data mining, Washington DC USA. New York, NY, USA, 2022. URL: https://doi.org/10.1145/3534678.3539081 (date of access: 02.03.2025).
5. Peng J. A comprehensive review of the application of NLP technology in language learning. *Applied and Computational Engineering*. 2024. Vol. 92, no. 1. P. 163–168. URL: https://doi.org/10.54254/2755-2721/92/20241735 (date of access: 11.11.2025).
6. Pakray P., Gelbukh A., Bandyopadhyay S. Natural language processing applications for low-resource languages. *Natural Language Processing*. 2025. Vol. 31, no. 2. P. 183–197. URL: https://doi.org/10.1017/nlp.2024.33 (date of access: 11.11.2025).
7. Sebe N. Human-centered computing. Handbook of ambient intelligence and smart environments. Boston, MA, 2010. P. 349–370. URL: https://doi.org/10.1007/978-0-387-93808-0_13 (date of access: 01.03.2025).

8. Bringing humans at the epicenter of artificial intelligence: A confluence of AI, HCI and human centered computing / V. Vishwarupe et al. Procedia computer science. 2022. Vol. 204. P. 914–921. URL: https://doi.org/10.1016/j.procs.2022.08.111 (date of access: 01.03.2025).
9. Prospects of computational intelligence in society: human-centric solutions, challenges, and research areas / K. K. Reddy Chinthala et al. Journal of computational and cognitive engineering. 2024. URL: https://doi.org/10.47852/bonviewjcce42023330 (date of access: 01.03.2025).
10. Qaiser S., Ali R. Text mining: use of TF-IDF to examine the relevance of words to documents. International journal of computer applications. 2018. Vol. 181, no. 1. P. 25–29. URL: https://doi.org/10.5120/ijca2018917395 (date of access: 02.03.2025).
11. Research of Text Classification Based on Improved TF-IDF Algorithm / C.- z. Liu et al. 2018 IEEE International Conference of Intelligent Robotic and Control Engineering (IRCE), Lanzhou, 24–27 August 2018. 2018. URL: https://doi.org/10.1109/irce.2018.8492945 (date of access: 02.03.2025).
12. Turning from TF-IDF to TF-IGM for term weighting in text classification / K. Chen et al. Expert systems with applications. 2016. Vol. 66. P. 245–260. URL: https://doi.org/10.1016/j.eswa.2016.09.009 (date of access: 02.03.2025).
13. Semantic sensitive TF-IDF to determine word relevance in documents / A. Jalilifard et al. Lecture notes in electrical engineering. Singapore, 2021. P. 327–337. URL: https://doi.org/10.1007/978-981-33-6987-0_27 (date of access: 02.03.2025).
14. Text features extraction based on TF-IDF associating semantic / Q. Liu et al. 2018 IEEE 4th international conference on computer and communications (ICCC), Chengdu, China, 7–10 December 2018. 2018. URL: https://doi.org/10.1109/compcomm.2018.8780663 (date of access: 02.03.2025).
15. Roul R. K., Sahoo J. K., Arora K. Modified TF-IDF term weighting strategies for text categorization. 2017 14th IEEE india council international conference

(INDICON), Roorkee, 15–17 December 2017. 2017. URL: https://doi.org/10.1109/indicon.2017.8487593 (date of access: 02.03.2025).

16. Yaş E. The role of age in second language acquisition. Cankaya university journal of humanities and social sciences. 2024. Vol. 18, no. 2. P. 265–276. URL: https://doi.org/10.47777/cankujhss.1570045 (date of access: 02.03.2025).
17. Firdaus T. A. Accelerating second language acquisition for effective strategies for student. Jurnal pendidikan indonesia. 2024. Vol. 5, no. 10. P. 1002–1013. URL: https://doi.org/10.59141/japendi.v5i10.5692 (date of access: 02.03.2025).
18. Huang W. The influence of mother tongue transfer on second language acquisition and its countermeasures. Transactions on social science, education and humanities research. 2024. Vol. 11. P. 599–603. URL: https://doi.org/10.62051/4h68mq56 (date of access: 02.03.2025).
19. Flavia P. D., Padmanabha C. H. The role of implicit learning in second language acquisition. I-manager's journal on english language teaching. 2024. Vol. 14, no. 2. P. 66. URL: https://doi.org/10.26634/jelt.14.2.20689 (date of access: 02.03.2025).
20. Shang J., Cui S. Universal grammar and universal grammar's influence and related theories concerning second language acquisition. Scholars international journal of linguistics and literature. 2024. Vol. 7, no. 07. P. 182–186. URL: https://doi.org/10.36348/sijll.2024.v07i07.002 (date of access: 02.03.2025).
21. Rogers J. The spacing effect and its relevance to second language acquisition. Applied linguistics. 2017. P. amw052. URL: https://doi.org/10.1093/applin/amw052 (date of access: 02.03.2025).
22. Enhancing human learning via spaced repetition optimization / B. Tabibian et al. Proceedings of the national academy of sciences. 2019. Vol. 116, no. 10. P. 3988–3993. URL: https://doi.org/10.1073/pnas.1815156116 (date of access: 02.03.2025).
23. Kakitani J., Kormos J. The effects of distributed practice on second language fluency development. Studies in second language acquisition. 2024. P. 1–25. URL: https://doi.org/10.1017/s0272263124000251 (date of access: 02.03.2025).

24. Optimizing spaced repetition schedule by capturing the dynamics of memory / J. Su et al. IEEE transactions on knowledge and data engineering. 2023. P. 1–13. URL: https://doi.org/10.1109/tkde.2023.3251721 (date of access: 02.03.2025).
25. Spaced learning enhances episodic memory by increasing neural pattern similarity across repetitions / K. Feng et al. The journal of neuroscience. 2019. Vol. 39, no. 27. P. 5351–5360. URL: https://doi.org/10.1523/jneurosci.2741-18.2019 (date of access: 02.03.2025).

ДОДАТОК А. ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

НАВЧАЛЬНО–НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Магістерська робота

«Алгоритм вивчення іноземних слів на основі антропоцентричних обчислень та методів обробки природної мови»

Виконав: студент групи ПДМ-61 Андрій СОКОЛОВСЬКИЙ

Керівник: канд. техн. наук, доцент, доцент кафедри ІПЗ Оксана ЗОЛОТУХІНА

Київ - 2025

МЕТА, ОБ'ЄКТА ТА ПРЕДМЕТ ДОСЛІДЖЕННЯ

Мета роботи: оптимізація процесу вивчення іноземних слів за рахунок використання методів обробки природної мови та антропоцентричних обчислень.

Об'єкт дослідження: процес вивчення іноземних слів.

Предмет дослідження: технології оптимізації вивчення іноземних слів на основі методів обробки природної мови та антропоцентричних обчислень.

АКТУАЛЬНІСТЬ РОБОТИ

Модель / Метод	Суть	Ключові недоліки	Переваги
TF-IDF	Оцінює інформаційну важливість слова на основі частотності	• Не враховує когнітивну складність слова (довжину, морфологію, частину мови) • Не оцінює засвоюваність слова людиною • Дає перевагу рідкісним словам без зв'язку з навчальною доцільністю	• Ефективно визначає інформативність слів у межах конкретного текстового корпусу. • Має просту, формально обґрунтовану модель та широко застосовується в аналізі текстів.
Частотні словники	Розподіляють слова за загальною вживаністю	• Частота ≠ складність засвоєння • Не враховують індивідуальні тексти користувача • Не формують адаптивної навчальної черги	• Дають базове уявлення про поширеність лексики та її орієнтовний рівень. • Зручні для формування початкового словникового мінімуму.
Класичні системи інтервального повторення (SRS)	Інтервальне повторення без аналізу складності слів	• Не розподіляють лексику перед додаванням • Однаковий алгоритм для всіх слів • Не враховують когнітивних параметрів чи контексту	• Підвищують довготривале запам'ятовування завдяки оптимальному плануванню повторень. • Ефективні для закріплення вже відібраної лексики.

3

ЗАДАЧІ ДИПЛОМНОЇ РОБОТИ

1. Провести аналіз сучасних методів оцінювання складності та значущості лексичних одиниць, зокрема частотних моделей, статистичних підходів типу TF-IDF, CEFR-орієнтованих методів та класичних систем інтервального повторення, з метою виявлення їхніх можливостей і обмежень у контексті роботи з індивідуальними текстовими корпусами.
2. Розробити та теоретично обґрунтувати комбінований алгоритм відбору лексики, який поєднує інформативність слова в тексті, статистичні характеристики та показники когнітивної складності, і забезпечує обчислення пріоритетності лексем для формування оптимізованих навчальних наборів.
3. Реалізувати програмну систему для автоматизованого аналізу текстів і формування навчальних словникових наборів на основі запропонованого алгоритму з використанням технологій Python, Electron та React.
4. Провести експериментальну перевірку ефективності розробленого алгоритму шляхом порівняння його результатів з існуючими базовими підходами, зокрема TF-IDF, та подальшим аналізом отриманих результатів.

4

ВИКОРИСТАНІ МЕТОДИ ОБРОБКИ ПРИРОДНОЇ МОВИ ТА АНТРОПОЦЕНТРИЧНИХ ОБЧИСЛЕНЬ

Методи обробки природної мови:

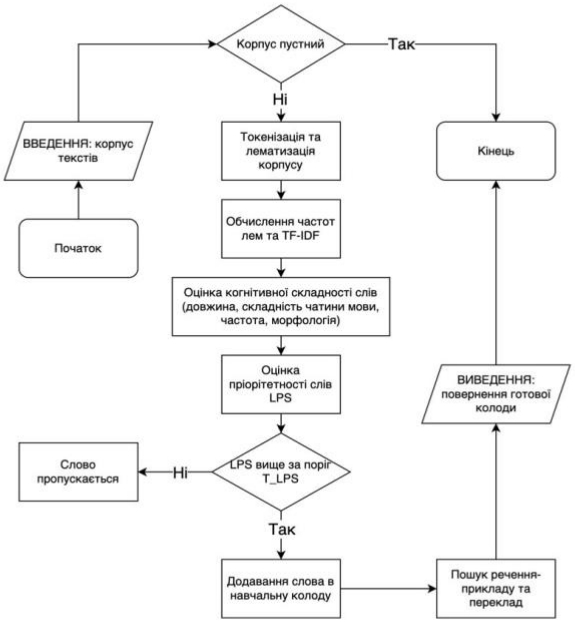
1. Лематизація та морфологічний аналіз.
2. Токенізація та сегментація на речення.
3. Маркування частини мови.
4. Частотний аналіз і нормалізація.
5. Обчислення TF-IDF для оцінки інформативності.

Антропоцентричні та когнітивні методи:

1. Модель когнітивної складності слова (довжина, складність, рідкість, морфологічна регулярність, частина мови та конкретність).
2. Крива забування Еббінгауза.
3. Контекстуалізація матеріалу через підбір прикладів з тексту.

5

СХЕМА АЛГОРИТМУ ОПТИМІЗАЦІЇ ВИВЧЕННЯ ІНОЗЕМНИХ СЛІВ НА ОСНОВІ АНТРОПОЦЕНТРИЧНИХ ОБЧИСЛЕНЬ ТА МЕТОДІВ ОБРОБКИ ПРИРОДНОЇ МОВИ



6

МОДЕЛЬ ВИЗНАЧЕННЯ ПРІОРІТЕТНОСТІ СЛОВА

1. Загальна формула

Показник пріорітетності слова для вивчення (Learning Priority Score – LPS) визначається за наступною формулою:

$$LPS(w) = \alpha \cdot IDF(w) + \beta \cdot D_c(w) + \gamma \cdot \big(1 - F_{text}(w)\big), \quad (1)$$

де $IDF(w)$ — обернена частота документів, обчислена з TF-IDF;

$D_c(w)$ — когнітивна складність;

$F_{text}(w)$ — нормалізована частота появи слова в корпусі користувача,

α, β, γ — вагові коефіцієнти (у поточній реалізації α =0.4, β =0.4, γ=0.2).

2. Формула компонента IDF

$$\mathrm{IDF}(w) = \log\left(\frac{N}{1 + df(w)}\right) \quad (2)$$

де *w* — слово або лема;
N — загальна кількість документів в корпусі;
df(w) — кількість документів, що включають це слово.

МОДЕЛЬ ВИЗНАЧЕННЯ ПРІОРІТЕТНОСТІ СЛОВА (ПРОДОВЖЕННЯ)

3. Показник когнітивної складності

$$D_c(w) = a_1 \cdot L(w) + a_2 \cdot S(w) + a_3 \cdot P(w) + a_4 \cdot \big(1 - F_{lex}(w)\big) + a_5 \cdot M(w) + a_6 \cdot C(w), \quad (3)$$

де $L(w)$ — нормалізована довжина слова;

$S(w)$ — нормалізована кількість складів;

$P(w)$ — складність частини мови;

$F_{lex}(w)$ — частотність слова у словнику spaCy (індекс рангу);

$M(w)$ — показник морфологічної нерегулярності;

$C(w)$ — показник конкретності частини мови;

$a_1 \dots a_6$ — нормовані коефіцієнти ваги

деталі обчислень та значень наведених вище показників - дивитися пункт 3.2 третього розділу диплому.

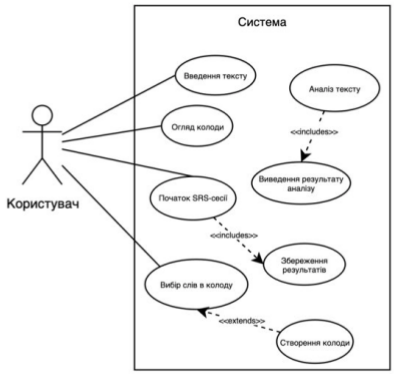
4. Нормалізована частота появи слова в корпусі користувача

$$F_{\text{text}}(w) = \frac{\text{кількість лем } w \text{ у корпусі}}{\text{загальна кількість лем у корпусі}}. \quad (4)$$

ПРАКТИЧНИЙ РЕЗУЛЬТАТ

1. Структура застосунку

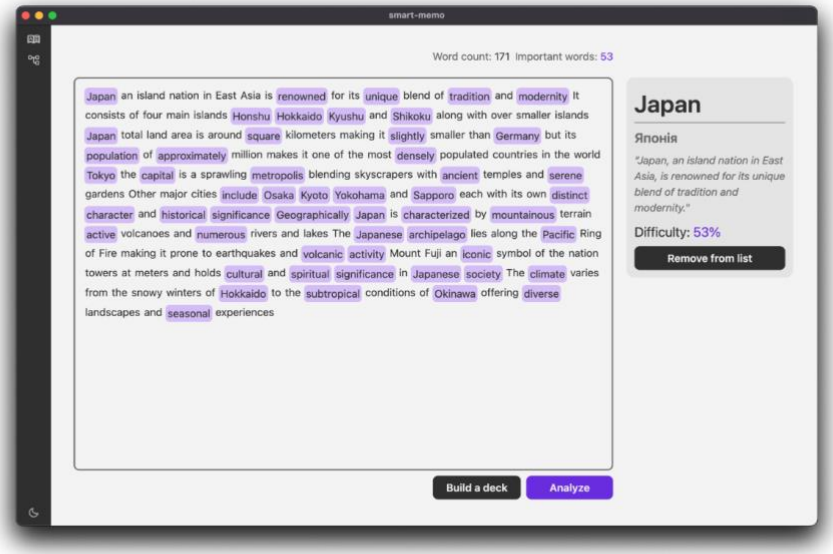
Діаграма розгортання застосунку



Діаграма варіантів використання системи

ПРАКТИЧНИЙ РЕЗУЛЬТАТ

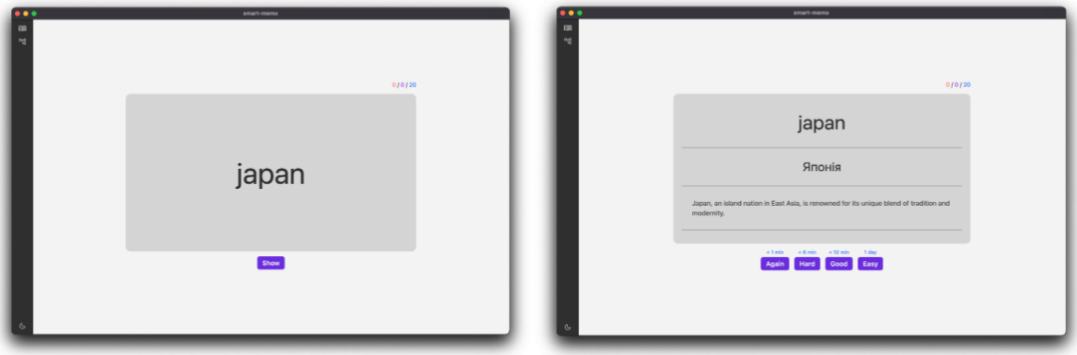
2. Екранні форми



Вікно вводу тексту

ПРАКТИЧНИЙ РЕЗУЛЬТАТ

2. Екранні форми



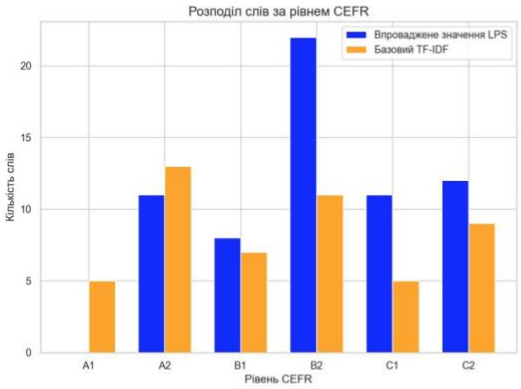
Вікно сесії інтервальних повторень

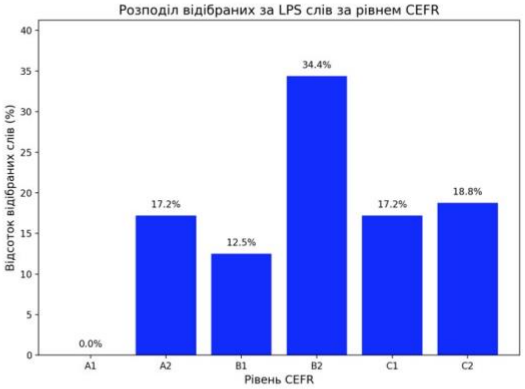
11

ДЕМОНСТРАЦІЯ РОБОТИ ЗАСТОСУНКУ

12

ПОРІВНЯЛЬНИЙ АНАЛІЗ ВПРОВАДЖЕНОГО АЛГОРИТМУ З БАЗОВИМ TF-IDF





13

ВИСНОВКИ

1. Проведено аналіз існуючих методів оцінювання складності та значущості лексики й встановлено, що частотні моделі, TF-IDF, CEFR-орієнтовані підходи та класичні SRS не забезпечують адаптації до індивідуального текстового корпусу користувача та не враховують когнітивних характеристик слів.
2. Запропоновано комбінований алгоритм відбору лексики, що поєднує інформативність слова (TF-IDF), статистичні ознаки та когнітивну складність. Алгоритм обчислює пріоритет кожної лексеми та формує оптимізовані набори слів, актуальні для конкретного матеріалу.
3. Розроблена програмна система на Python та Electron/React підтвердила практичну ефективність моделі: забезпечено автоматичне виділення ключової лексики, відсікання простої та неактуальної, формування навчальних наборів і зручну взаємодію.
4. Отримані результати аналізу демонструють, що інтеграція NLP-метрик і когнітивних моделей значно підвищує якість формування словникових наборів. Впроваджений алгоритм надає значну перевагу рідкісним та когнітивно складнішим словам рівня B2-C2, в той час як базові методи, такі як TF-IDF, надають пріорітет більш частотним та простішим словам. Відсотковий розподіл показує, що загальний відсоток виділених алгоритмом слів, що припадає на рівні B2-C2, складає 70,4%, і лише 29,6% - на рівні A1-B1.

14

ПУБЛІКАЦІЇ ТА АПРОБАЦІЯ РОБОТИ

Тези доповідей:

1. Соколовський А.В., Герцюк М.М. Аналіз технологій для створення застосунку оптимізації вивчення іноземних слів. II міжнародна науково-практична конференція «Сучасні аспекти діджиталізації та інформатизації в програмній та комп'ютерній інженерії», 19-21 грудня 2024 р., Київ, Державний університет інформаційно-комунікаційних технологій. Збірник тез. К.: ДУІКТ, 2024. С.238-239.
2. Соколовський А.В., Золотухіна О.А. Підвищення ефективності вивчення іноземних слів використовуючи методи обробки природної мови та систему розподілених повторень. Всеукраїнська науково-технічна конференція «Застосування програмного забезпечення в ІКТ», 24 квітня 2025 р., Київ, Державний університет інформаційно-комунікаційних технологій. Збірник тез. К.: ДУІКТ, 2025. С.243-245.

ДОДАТОК Б. ЛІСТИНГИ ОСНОВНИХ МОДУЛІВ

```
import spacy
import re
import math
from collections import Counter
from sklearn.feature_extraction.text import TfidfVectorizer
from translation import translation_table
import json
import numpy as np

nlp = spacy.load("en_core_web_sm")

def fast_syllables(word: str) -> int:
return max(1, len(re.findall(r"[aeiouy]+", word.lower())))

def compute_difficulty(token):
if token.is_punct or token.is_space:
return 0.0

length = len(token.text) / 12.0
syllables = fast_syllables(token.text) / 4.0

pos_difficulty = {
"NOUN": 0.3, "VERB": 0.7, "ADJ": 0.5, "ADV": 0.5,
"PRON": 0.2, "ADP": 0.1, "DET": 0.1, "AUX": 0.3,
"CCONJ": 0.1, "SCONJ": 0.2, "NUM": 0.3, "PROPN": 0.5
}.get(token.pos_, 0.4)

concreteness_pos = {"NOUN": 0.2, "VERB": 0.6, "ADJ": 0.4, "ADV": 0.5}
conc_score = concreteness_pos.get(token.pos_, 0.5)

morph_irreg = 0.1 if token.lemma_.lower() != token.text.lower() else 0.0

lex = nlp.vocab[token.lemma_]
freq_component = (1.0 - min(lex.rank / 50000, 1.0)) if lex.is_alpha else 0.0

dc = (
0.2 * length +
0.2 * syllables +
0.2 * pos_difficulty +
0.2 * (1 - freq_component) +
0.1 * morph_irreg +
0.1 * conc_score
)

return round(min(dc, 1.0), 3)


def _to_py_scalar(x):
"""Convert numpy scalars/arrays to native Python types where possible."""
if isinstance(x, (np.generic,)):
try:
return x.item()
except Exception:
return x.tolist() if hasattr(x, "tolist") else str(x)
return x


def compute_learning_priority(payload, threshold_percentile=0.4):
corpus = payload.get("corpus", [])
if not corpus:
return {"error": "No corpus provided"}

# TF-IDF vectorization
vectorizer = TfidfVectorizer(stop_words="english", max_features=5000)
X = vectorizer.fit_transform(corpus)
feature_names = vectorizer.get_feature_names_out()
idf_values = vectorizer.idf_
idf_dict = {fn: float(idf_values[i]) for i, fn in enumerate(feature_names)}

# spaCy batch tokenization
```

```
docs = list(nlp.pipe(corpus, disable=["ner", "parser"]))
all_tokens = [t for doc in docs for t in doc if t.is_alpha]

# Frequency normalization
lemmas = [t.lemma_.lower() for t in all_tokens]
freq = Counter(lemmas)
total = sum(freq.values()) or 1
freq_norm = {w: float(freq[w]) / float(total) for w in freq}

# Compute per-lemma difficulty and LPS
a, b, y = 0.4, 0.4, 0.2
lemma_stats = {}
for lemma in set(lemmas):
dummy_token = next(t for t in all_tokens if t.lemma_.lower() == lemma)
dc = float(compute_difficulty(dummy_token))
idf = float(idf_dict.get(lemma, math.log((len(corpus) + 1) / 1)))
f = float(freq_norm[lemma])
lps = float(a * idf + b * dc + y * (1 - f))
lemma_stats[lemma] = {
"dc": dc,
"lps": lps,
"idf": round(float(idf), 3),
"freq": round(float(f), 4),
}

# Determine threshold
lps_sorted = sorted([s["lps"] for s in lemma_stats.values()])
T_LPS = float(lps_sorted[int(len(lps_sorted) * (1 - threshold_percentile))]) if lps_sorted else 0.0

# Build final result with translation and example sentence
results = []
for token in all_tokens:
lemma_lower = token.lemma_.lower()
s = lemma_stats[lemma_lower]

# Translation: try exact token first, then lowercase lemma
translation = translation_table.get(token.text) or translation_table.get(lemma_lower, "")
# Example sentence from corpus
example_sentence = find_example_sentence(token.text, corpus)

token_obj = {
"token": str(token.text),
"lemma": lemma_lower,
"pos": str(token.pos_),
"difficulty": float(s["dc"]),
"idf": float(s["idf"]),
"freq": float(s["freq"]),
"lps": float(round(s["lps"], 3)),
"included": bool(s["lps"] >= T_LPS),
"translation": translation,
"example_sentence": example_sentence
}
results.append(token_obj)

return {"ranked_words": results}

def find_example_sentence(word: str, corpus: list[str]) -> str:
"""
Return the first sentence from the corpus that contains the target word or lemma.
Uses spaCy sentence segmentation for accurate boundary detection.
"""
if not word or not corpus:
return ""

target = word.lower()

for text in corpus:
doc = nlp(text)
for sent in doc.sents:
sent_lower = sent.text.lower()
if target in sent_lower:
return sent.text.strip()

return ""
```

```
import datetime, math
LEARNING_STEPS_MIN = [1, 10]
def update_srs_card(card, result: int):
"""
result: 0=Again, 1=Hard, 2=Good, 3=Easy
Returns dict with new scheduling fields, including 'status' and 'learning_step'.
"""
now = datetime.datetime.now()
ef = card.get("ease_factor", 2.5)
interval = card.get("interval_days", 0)
review_count = card.get("review_count", 0)
status = card.get("status", "new")
learning_step = card.get("learning_step", 0)

next_interval_days = interval
new_status = status
new_learning_step = learning_step

if status in ("new", "learning"):
if result == 0:
new_learning_step = 0
minutes = LEARNING_STEPS_MIN[0]
delta = datetime.timedelta(minutes=minutes)
new_status = "learning"
next_interval_days = 0

elif result == 1:
if new_learning_step + 1 < len(LEARNING_STEPS_MIN):
new_learning_step = new_learning_step + 1
minutes = LEARNING_STEPS_MIN[new_learning_step]
else:
minutes = LEARNING_STEPS_MIN[-1]
delta = datetime.timedelta(minutes=minutes)
new_status = "learning"
next_interval_days = 0

elif result == 2:
new_learning_step = new_learning_step + 1
if new_learning_step < len(LEARNING_STEPS_MIN):
minutes = LEARNING_STEPS_MIN[new_learning_step]
delta = datetime.timedelta(minutes=minutes)
new_status = "learning"
next_interval_days = 0
else:
new_status = "review"
next_interval_days = 1
delta = datetime.timedelta(days=next_interval_days)
new_learning_step = len(LEARNING_STEPS_MIN)

elif result == 3:
new_status = "review"
next_interval_days = 4
delta = datetime.timedelta(days=next_interval_days)
new_learning_step = len(LEARNING_STEPS_MIN)

else:
if result == 0:
ef = max(1.3, ef - 0.3)
next_interval_days = 1
elif result == 1:
ef = max(1.3, ef - 0.15)
next_interval_days = max(1, math.ceil(interval * 1.2))
elif result == 2:
ef = min(3.0, ef + 0.05)
next_interval_days = max(1, math.ceil(interval * ef))
elif result == 3:
ef = min(3.5, ef + 0.15)
next_interval_days = math.ceil(interval * ef * 1.3)

delta = datetime.timedelta(days=next_interval_days)
new_status = "review"
new_learning_step = new_learning_step
```

```
next_review = (now + delta).strftime("%Y-%m-%d %H:%M:%S")
review_count += 1

return {
"ease_factor": round(ef, 3),
"interval_days": next_interval_days,
"next_review": next_review,
"review_count": review_count,
"last_result": result,
"status": new_status,
"learning_step": new_learning_step,
}
```