

Шифр: NLP ПОШУК

Назва роботи: **NLP ДЛЯ ЕФЕКТИВНОГО ПОШУКУ ОСВІТНІХ
СТАНДАРТІВ З МАТЕМАТИКИ В СИСТЕМІ CCSS**

ЗМІСТ

Вступ.....	3
1. Теоретична частина.....	4
1.1 Аналіз системи CCSS з математики в освіті США.....	4
1.2 Огляд принципів і методів NLP.....	6
2. Методика.....	7
2.1 Вибір Джерела Даних.....	7
2.2 Методики аналізу даних.....	8
2.3 Вибір платформи та інструментів для розробки моделі.....	8
3. Реалізація проєкту.....	9
3.1 Аналіз та підготовка даних.....	9
3.2 Проєктування архітектури нейронної мережі.....	12
4. Реалізація моделі.....	14
4.1 Налаштування Embedding шару та обґрунтування його використання.....	14
4.2 Тренування моделі.....	15
4.3 Оцінка ефективності та точності моделі.....	16
5. Результати.....	18
6. Майбутнє використання та перспективи.....	20
6.1 Адаптація моделі для української освітньої системи.....	20
6.2 Подальша оптимізація та розширення функціонала моделі.....	21
Висновки.....	22
Список використаних джерел.....	23

Вступ

Науково-технічний прогрес у галузі обробки природної мови (NLP) та машинного навчання відкриває нові можливості для реформування підходів до навчального процесу, особливо у створенні та роботі з дидактичними матеріалами.

Попри стрімкий прогрес у галузі технологій, освітня система в багатьох країнах, включаючи Україну, часто зберігає консервативний характер, сповільнюючи інтеграцію інноваційних технологічних рішень. В цьому контексті, створення інструменту, який оптимізує процес пошуку і рекомендації математичних стандартів CCSS для педагогів, є важливим кроком у правильному напрямку. Даний інструмент забезпечить не лише полегшення та економію часу педагогічних працівників, які розробляють нові дидактичні матеріали для учнів, але і підвищить точність відповідності вибраних тем до відповідних освітніх стандартів.

Об'єктом дослідження є процес ідентифікації та відбору навчальних матеріалів, що відповідають специфічним освітнім стандартам, зокрема, з орієнтацією на математичні стандарти Common Core State Standards (CCSS). (1)

Предмет дослідження – створення інструменту на основі NLP для ефективного пошуку математичних стандартів CCSS, за допомогою лише короткого опису теми, пропонуючи потенціал для можливої адаптації до майбутніх освітніх стандартів України.

Мета дослідження полягає у розробці програмного інструменту, який здатний допомогти педагогам у пошуку потрібних їм стандартів CCSS з математики, з використанням алгоритмів NLP для аналізу та розуміння текстових запитів викладача.

Таким чином, розгорнуте дослідження покликане допомогти у формуванні технологічних підходів для спрощення пошуку стандартів у сфері шкільної освіти, і як результат, більш зручній та якісній роботі українських педагогів у

створенні нових дидактичних матеріалів, коли Україна оновить свої стандарти освіти з математики.

1. Теоретична частина

1.1 Аналіз системи CCSS з математики в освіті США.

Common Core State Standards (CCSS) з математики є ініціативою, розробленою з метою підвищення якості математичної освіти в США, концентруючи увагу на ключових навичках і поняттях, які мають опанувати учні. Основна проблема, яку мали на увазі розробники CCSS, це те, що математична програма була “з шириною у милю та глибиною у дюйм”, тобто надто обширною і не зосередженою. (1)

Основні Принципи та Структура CCSS з Математики:

- 1. Від Загального до Специфічного:** Стандарти математики CCSS відрізняються конкретністю і чіткістю, відмовляючись від широких загальних тверджень на користь більш точних і конкретних вимог.
- 2. Міжнародні Моделі та Досвід:** CCSS базуються на найкращих стандартах математики з різних штатів США та інтегруючи наукові дослідження та внески з широкого спектра джерел, включаючи освітні департаменти штатів, науковців, розробників методик оцінювання та дидактичних матеріалів.
- 3. Орієнтованість на Розуміння:** Стандарти вимагають, щоб учні не просто знали, але й розуміли математичні концепції, і водночас надають вчителям інструменти для перевірки цього розуміння, такі як вимога пояснити, чому конкретне математичне твердження є істинним.

Деталізація Математичних Стандартів:

- **Кількість та Розподіл Стандартів:** CCSS охоплює велику кількість стандартів, які детально описують, що і як повинен вчити учень на

кожному етапі своєї освітньої доріжки, надаючи вчителям чіткий план та послідовність для кожного класу.

- **Послідовність та Залежності:** Особливу увагу приділено структуризації тем та визначенню, як конкретні навички й знання будуються з року в рік, допомагаючи вчителям бачити, як навички та знання учнів розвиваються і побудовані у часі.
- **Покриття Різних Рівнів Освіти:** Стандарти охоплюють математичну освіту від дитячого садка до старшої школи, надаючи всебічний і послідовний шлях для розвитку математичних здібностей учнів на всіх етапах освіти.

Common Core State Standards – Mathematics						Standards Progressions			
Kindergarten	1	2	3	4	5	6	7	8	HS
Counting and Cardinality									Number and Quantity
Number and Operations in Base Ten					Ratios and Proportional Relationships				
			Number and Operations - Fractions			The Number System			
Operations and Algebraic Thinking						Expressions and Equations			Algebra
									Functions
Geometry						Geometry			Geometry
Measurement and Data						Statistics and Probability			Statistics and Probability

Рис. 1.1.1. Розподіл математичних галузей з садочка до старшої школи. (2)

Реалізація та Використання в Навчальному Процесі

Організація та Застосування: CCSS пропонує організований спосіб навчання концепцій протягом навчального року та між різними класами. Такий підхід допомагає учням розвивати концепції та навички у послідовний і логічний спосіб.

Проблемно Орієнтований Підхід: CCSS заохочує учнів застосовувати математичні знання для розв'язання проблем реального світу, спонукаючи їх розглядати математику як практичний і застосовний інструмент.

Grade 3 » Number & Operations in Base Ten

 PRINT THIS PAGE

Standards in this domain:

CCSS.MATH.CONTENT.3.NBT.A.1

CCSS.MATH.CONTENT.3.NBT.A.2

CCSS.MATH.CONTENT.3.NBT.A.3

Use place value understanding and properties of operations to perform multi-digit arithmetic.¹

CCSS.MATH.CONTENT.3.NBT.A.1

Use place value understanding to round whole numbers to the nearest 10 or 100.

CCSS.MATH.CONTENT.3.NBT.A.2

Fluently add and subtract within 1000 using strategies and algorithms based on place value, properties of operations, and/or the relationship between addition and subtraction.

CCSS.MATH.CONTENT.3.NBT.A.3

Multiply one-digit whole numbers by multiples of 10 in the range 10-90 (e.g., 9×80 , 5×60) using strategies based on place value and properties of operations.

Рис. 1.1.2. Опис стандартів у групі 3.NBT. (3)

1.2 Огляд принципів і методів NLP.

Обробка природної мови (NLP) – це міст між людською мовою та комп'ютерним розумінням, намагаючись вивчити та імітувати здатність людини засвоювати та використовувати мову в широкому діапазоні завдань. Суть NLP зводиться до перетворення текстової інформації в числовий формат, який можна обробити та аналізувати за допомогою моделей машинного навчання. (4)

Ембедінги слова: Ядром роботи більшості моделей NLP є процес ембедінгу, який конвертує слова у числові вектори. Ці вектори не просто кодують слова, але й захоплюють семантичні та контекстуальні зв'язки між ними. Для цього слова проєктуються у багатовимірний простір, де слова, що вживаються в схожих контекстах, розміщуються близько одне до одного. Це також дозволяє проводити семантичні операції за допомогою векторної арифметики: наприклад,

вектор("король") - вектор("чоловік") + вектор("жінка") може приблизно дорівнювати вектору("королева"). (5)

Архітектура Нейронної Мережі: Рекурентні нейронні мережі (RNN) і, зокрема, мережі з довгою короткочасною пам'яттю (LSTM), відіграють ключову роль у використанні контекстуальної інформації в тексті. Вони здатні "передавати" інформацію вздовж послідовності, забезпечуючи плавне збереження та оновлення контексту, що дозволяє моделі розуміти значення слова у контексті цілого речення чи тексту. Наприклад, LSTM зможе коректно інтерпретувати речення, враховуючи порядок слів та їхню взаємодію. (6)

Застосування цих принципів і методів у поєднанні з ретельно підбраною архітектурою та натренованою моделлю може забезпечити підтримку розширеного аналізу тексту і високу точність у виконанні задач обробки природної мови. Ці аспекти формують основу для розробки застосунків, що здатні взаємодіяти з текстовою інформацією в більш продуктивний та інтелектуальний спосіб, адаптуючи та оптимізуючи процеси обробки мовних даних.

2. Методика

2.1 Вибір Джерела Даних.

У контексті даного дослідження основним джерелом даних для аналізу було обрано платформу TeachersPayTeachers.com (ТРТ). Причини такого вибору базуються на популярності платформи серед вчителів США, адже сервісом активно користується понад 80% педагогів країни. (7) Також там велика база ресурсів створена вчителями, що містить майже 8 мільйонів навчальних матеріалів. Тож навіть мала частина цих даних це вже значний обсяг для аналізу. Також ТРТ дозволяє здійснювати деталізований відбір матеріалів за різноманітними критеріями (ціна, галузь, стандарт, тип та інші). (8)

Враховуючи результати попереднього аналізу та тестування різних стратегій пошуку на платформі, було прийнято рішення збирати по 300 перших ресурсів

для кожного освітнього стандарту. Дане рішення базується на спробах оптимізації об'єму даних та забезпеченні їхньої релевантності і якості для подальшого аналізу. Пошуковий запит був налаштований таким чином, щоб сортування ресурсів відбувалося за релевантністю, а також було встановлено фільтр, щоб включати лише ті продукти, ціна яких становить менше ніж 5 доларів. Такий підхід дозволяє зменшити кількість матеріалів, які можуть мати велику кількість тем, та сконцентруватися на більш специфічних та цільових ресурсах.

2.2 Методики аналізу даних.

Після збору даних, впроваджується подальший етап — визначення методик аналізу отриманих даних. Особливий акцент у цьому контексті робиться на критеріях, що будуть використовуватися для подальшого навчання нейронної мережі.

Для забезпечення якості даних та підвищення адекватності навчання моделі, обрано таку стратегію: включення тільки тих ресурсів, які отримали хоча б один відгук і мають середній бал не менше 4,0 з 5,0. Такі параметри сприяють концентрації на релевантних і якісних матеріалах, відсіюючи ті, що можуть бути менш корисними, чи не відповідати певному стандарту якості.

Додатково, було прийнято рішення виключити з аналізу ті продукти, що мають більше ніж 20 стандартів. Цей крок дозволяє сконцентрувати увагу на більш конкретних, специфічних матеріалах та уникнути переобтяження моделі даними, що можуть погіршити навчання мережі.

Таким чином, впроваджені методики аналізу даних орієнтовані на максимізацію якості вхідних даних для моделі, та спрямовані на забезпечення її здатності робити точні та релевантні прогнози, щодо взаємозв'язку між освітніми ресурсами та стандартами.

2.3 Вибір платформи та інструментів для розробки моделі.

При плануванні розробки моделі основний акцент було зроблено на ефективність, доступність та масштабованість. У зв'язку з цим, було вирішено використовувати наступний набір інструментів:

1. **Python**: Вибрано за велике розмаїття спеціалізованих бібліотек і фреймворків для машинного навчання й обробки даних.
2. **TensorFlow**: Обрано завдяки численним готовим рішенням, активній підтримці спільноти та зручності інтеграції з іншими сервісами Google, зокрема Google Cloud для створення вебдодатків. (9)
3. **Google Colab**: Ідеально підходить для експериментів з моделями завдяки безкоштовному доступу до GPU та зручності обміну результатами. (10)

Цей набір інструментів забезпечує оптимальне поєднання продуктивності, доступності ресурсів та масштабованості для досягнення цілей даного дослідження.

3. Реалізація проєкту

3.1 Аналіз та підготовка даних.

Робота з даними зазвичай починається з їхнього аналізу та підготовки. Маючи дві ключові змінні, titles (заголовки робіт) та csss (масиви стандартів), пріоритет було віддано ретельній попередній обробці цих даних для подальшого ефективного навчання моделі.

1. Обробка titles:

- Заміна символів: Усі символи "&" були замінені на " and " для забезпечення читабельності та уникнення проблем зі спеціальними символами в процесі обробки тексту.
- Фільтрація символів: Була застосована regex-функція, щоб утримати тільки символи [A-Za-z0-9], вилучаючи всі інші зайві символи з назв.
- Нормалізація тексту: Текст було приведено до нижнього регістру і подвійні пробіли були видалені для консистентності та оптимізації.

2. Обробка ccss:

- Бінаризація Категорій: ccss було трансформовано з допомогою MultiLabelBinarizer, конвертуючи масиви стандартів у бінарні вектори, де “1” позначає наявність конкретного стандарту, а “0” – його відсутність.

Також після видалення усіх записів, де більше ніж 20 стандартів залишилось 42 997 записів.

Після попереднього аналізу даних, було отримано наступні діаграми:

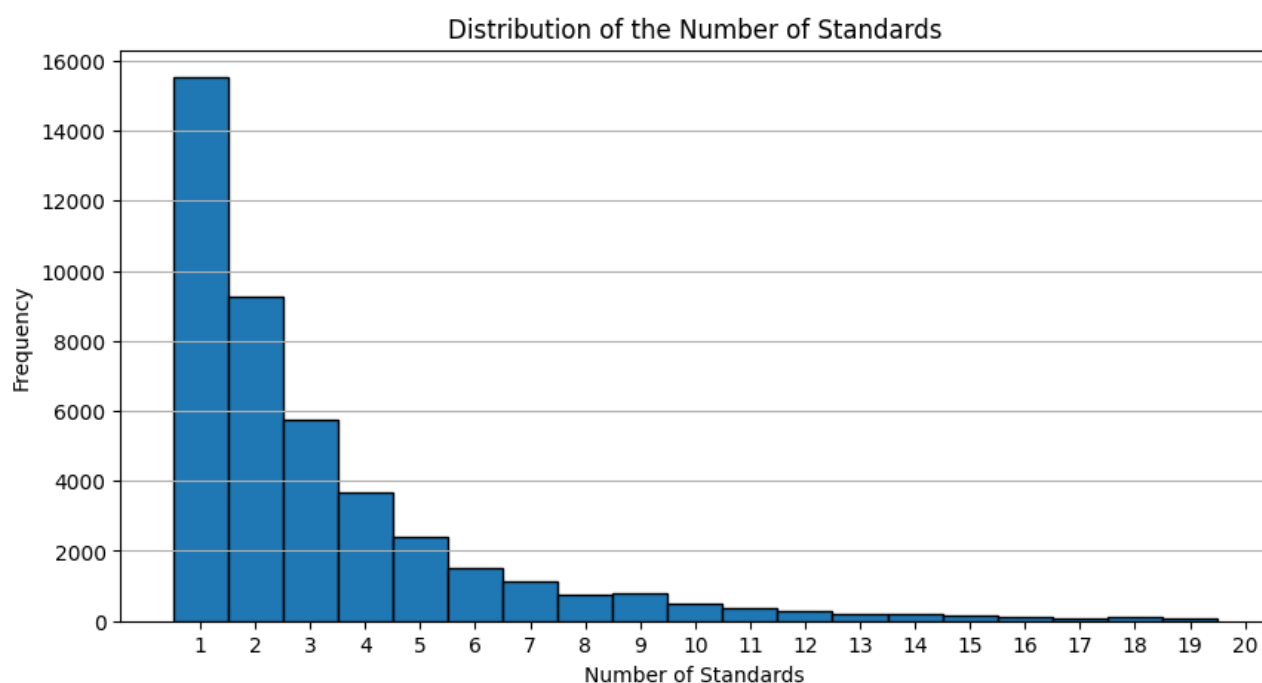


Рис. 3.1.1 Розподіл кількості стандартів у ресурсів.

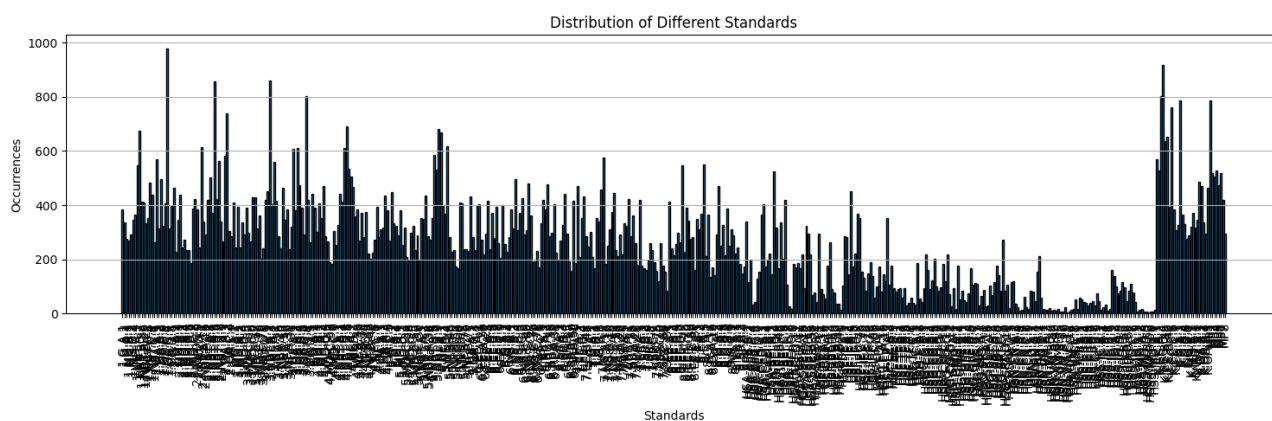


Рис. 3.1.2 Розподіл за кількістю стандартів по всім ресурсам.

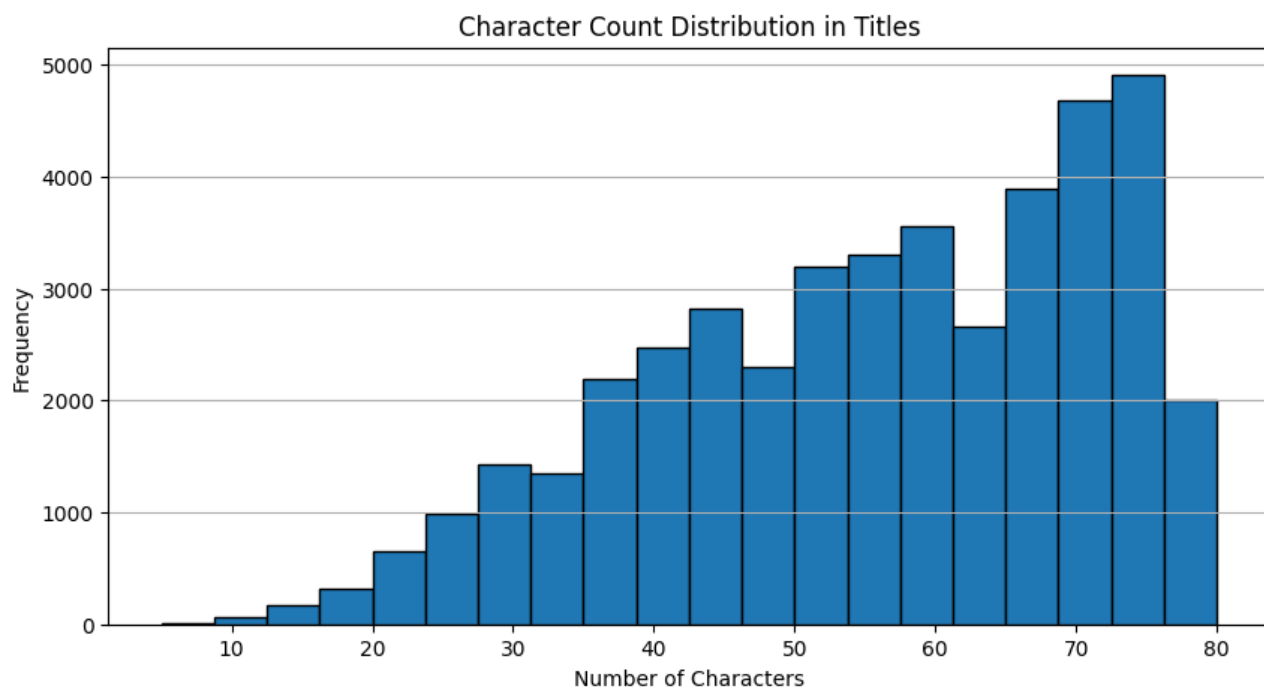


Рис. 3.1.3 Розподіл за кількістю символів у назвах.

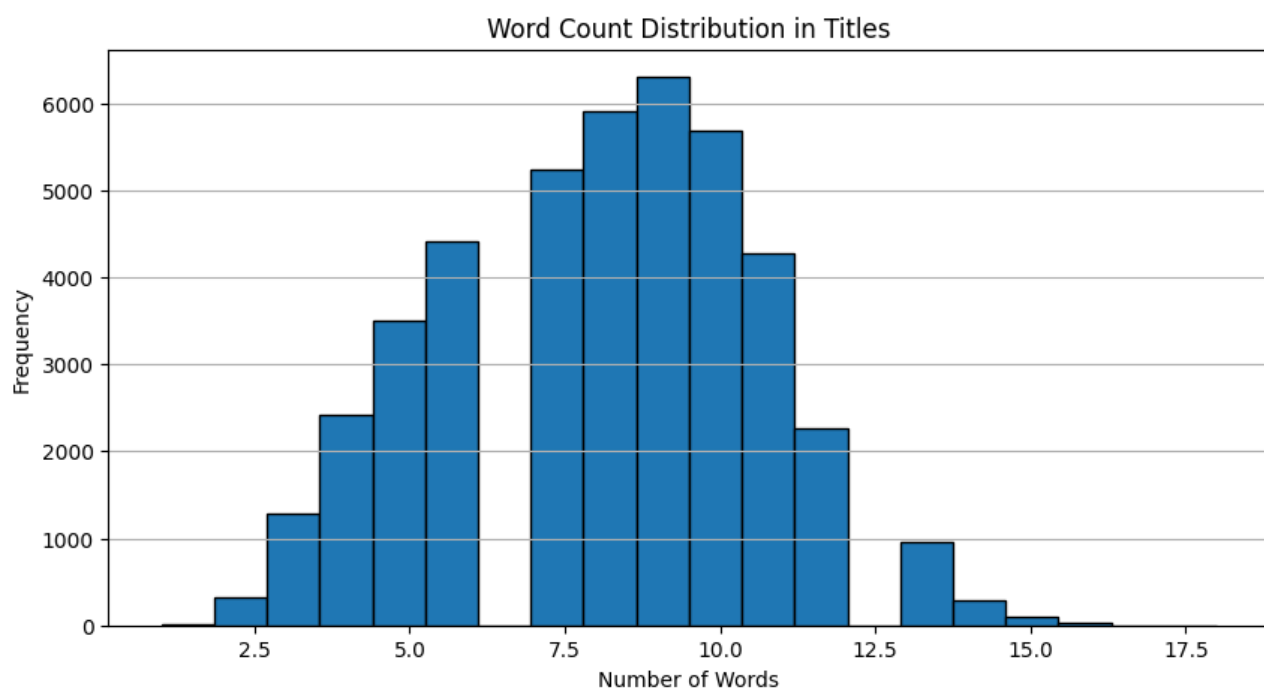


Рис. 3.1.4 Розподіл за кількістю слів у назвах

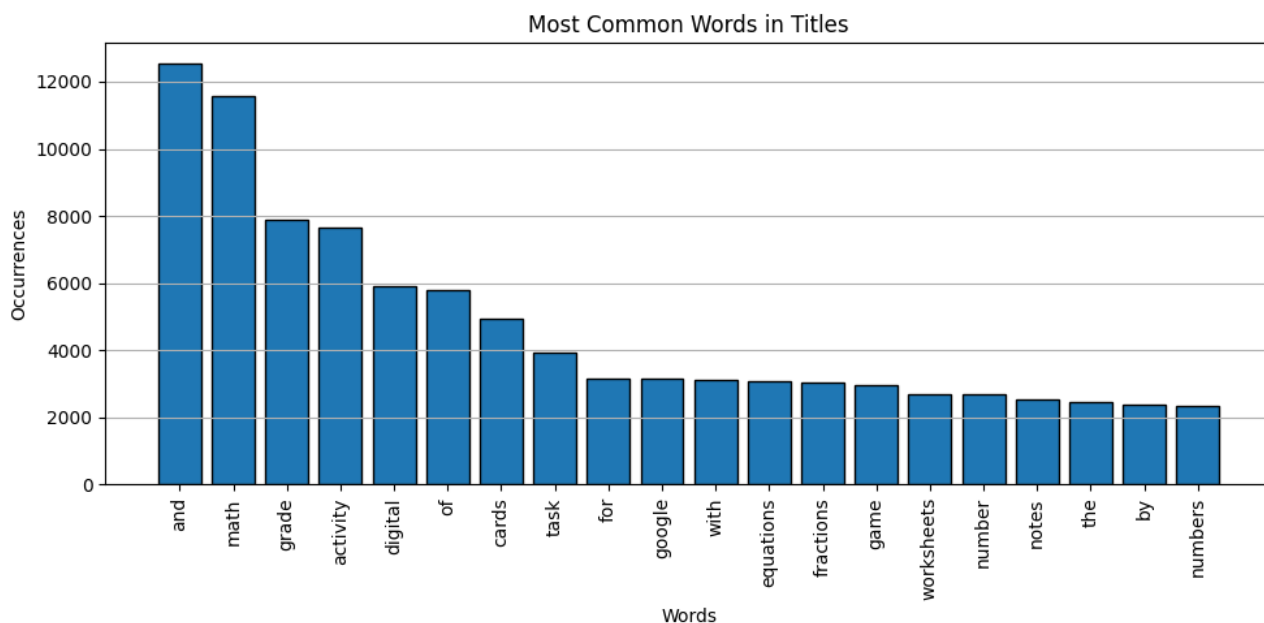


Рис. 3.1.5 Найчастіші 20 слів у назвах продуктів, та їх кількість.

Також, випадковим чином було поділено набір даних для навчання та валідації. 70% на навчання, 30% на валідацію.

3.2 Проектування архітектури нейронної мережі.

Архітектура нейронної мережі є критичним аспектом у розробці моделі, що впливає на її здатність ефективно навчитись на даних. У цьому дослідженні пропонуються такі шари нейронної мережі:

1. **Векторизація:** Вхідні текстові дані перетворюються у вектори, щоб надати нейронній мережі оптимальний формат для обробки.
2. **Bidirectional LSTM** (64 нейрони, `return_sequences=True`): LSTM (Long Short-Term Memory) шар, який має здатність "запам'ятовувати" інформацію протягом тривалих періодів часу, є критично важливим для обробки текстових послідовностей. Функціональність "Bidirectional" дозволяє LSTM передавати контекст в обидва напрямки (вперед і назад) через послідовність, значно підвищуючи здатність моделі вивчати залежності між словами у вхідній послідовності. (6)
3. **Bidirectional LSTM** (32 нейрони): Ще один шар LSTM із меншою кількістю нейронів для додаткового моделювання часових залежностей у

тексті, підвищуючи точність і відтворюваність при вивченні послідовностей даних.

4. **Dense Layer** (256 нейрони, активація 'relu'): Впровадження необхідної нелінійності для підтримки вивчення складних шаблонів.
5. **Batch Normalization**: Нормалізація активацій з метою прискорення навчання та збільшення стабільності моделі.
6. **Dropout** (0.2): Випадкове вимикання 20% нейронів з запобігати перенавчанню та покращення узагальнення моделі.
7. **Dense Layer** (256 нейрони, активація 'relu'): Ще один щільний шар для вивчення складних взаємозв'язків у даних.
8. **Batch Normalization**: Додатковий етап нормалізації активацій для поліпшення стабільності та прискорення навчання.
9. **Dropout** (0.2): Додаткова регуляризація з метою зменшення перенавчання і підвищення робастності моделі.
10. **Output Dense Layer** (кількість нейронів відповідає кількості класів, активація 'sigmoid'): Генерування ймовірностей приналежності вхідного тексту до кожного з класів, які потім інтерпретуються як прогнози моделі.

Загальна кількість параметрів для навчання: 356 357.

```
model.summary()
```

Model: "sequential"

Layer (type)	Output Shape	Param #
keras_layer (KerasLayer)	(None, 128)	124642688
reshape (Reshape)	(None, 1, 128)	0
bidirectional (Bidirectional)	(None, 1, 128)	98816
bidirectional_1 (Bidirectional)	(None, 64)	41216
dense (Dense)	(None, 256)	16640
batch_normalization (Batch Normalization)	(None, 256)	1024
dropout (Dropout)	(None, 256)	0
dense_1 (Dense)	(None, 256)	65792
batch_normalization_1 (Batch Normalization)	(None, 256)	1024
dropout_1 (Dropout)	(None, 256)	0
dense_2 (Dense)	(None, 517)	132869

=====
Total params: 125000069 (476.84 MB)
Trainable params: 356357 (1.36 MB)
Non-trainable params: 124643712 (475.48 MB)

Рис. 3.2.1 Структура моделі.

4. Реалізація моделі

4.1 Налаштування Embedding шару та обґрунтування його використання.

В ефективних моделях обробки природної мови (NLP) користування якісними векторними представленнями слів є ключовим. Вже натренований Embedding шар з TensorFlow Hub (11), зокрема NNLM з розмірністю вектора 128, вибрано через декілька переваг:

- **Економія Часу і Ресурсів:** Немає потреби тренувати векторне представлення слів від початку, що економить значний час та обчислювальні ресурси.
- **Експертиза:** NNLM уже "бачив" великі датасети та навчився враховувати контекст і семантичні відносини між словами, підвищуючи якість векторизації.
- **Стабільність:** Модель пройшла перевірку часом і використовується спільнотою, що гарантує стабільність і надійність.

Вбудований шар легко інтегрується в модель завдяки TensorFlow Hub, перетворюючи вхідні текстові дані в числові вектори, які можуть бути ефективно оброблені мережею. Цей процес не тільки полегшує розробку, але і вдосконалює результативність моделі, надаючи їй доступ до вже навчених шаблонів взаємозв'язку слів. Тож модель зможе працювати навіть зі словами, яких не «бачила» у даних для тренування.

4.2 Тренування моделі.

Під час розробки нейронної мережі важливо оптимізувати її конфігурацію та контролювати процес навчання.

Для компіляції моделі було вибрано оптимізатор Adam завдяки його добрим результатам у вирішенні великої кількості задач глибокого навчання. (12) Як функцію втрат використовувалася бінарна перехресна ентропія. Додатково, для моніторингу якості моделі було обрано метрики AUC (13), точність (Precision) (14) та повноту (Recall). (15)

Для розв'язання проблеми перенавчання, було використано Early Stopping з терпимістю в 5 епох та збереженням найкращих ваг, а також Model Checkpoint для збереження найкращої моделі.

Навчання планувалось на 1000 епох із `batch_size` 64. Але завдяки використанню `Early Stopping`, тренування зупинилося на 50-й епосі, щоб запобігти перенавчанню.

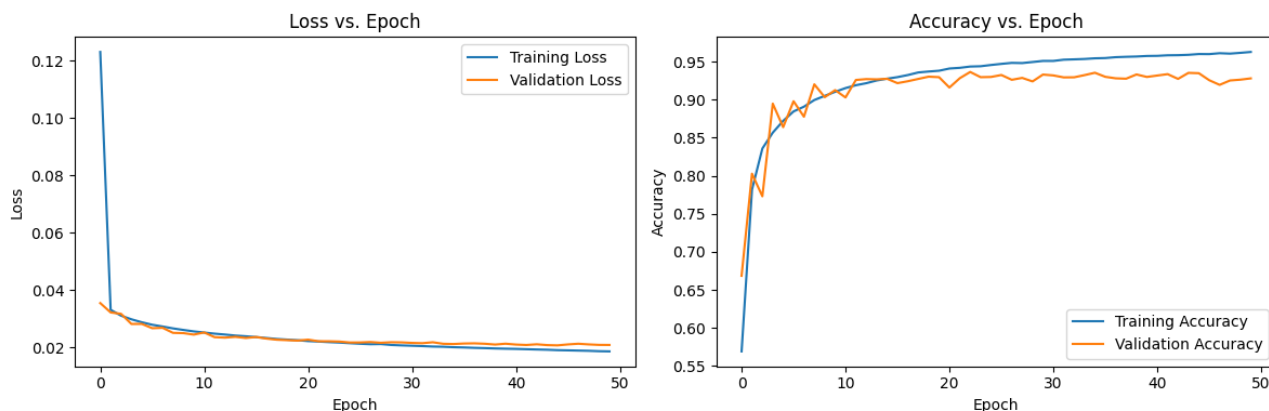


Рис. 4.2.1 Графік втрат та точності для тренувального та валідаційного наборів даних.

Аналіз графіків показує, що тренування було ефективним. Графік втрат демонструє, що втрати з тренувального та валідаційного наборів зменшилися протягом усього процесу навчання. Графік точності свідчить про стабільне покращення якості моделі на тренувальних та валідаційних даних.

На графіках також можна спостерігати, що лінії тренувального та валідаційного наборів даних рухаються синхронно, без різких стрибків або відхилень. Така узгодженість в динаміці обох графіків свідчить про добре збалансований набір даних. Крім того, відсутність сильних "коливань" та розривів між графіками вказує на те, що архітектура моделі була вдало обрана. Це не тільки свідчить про зменшення ризику перенавчання, але й підтверджує, що модель є стійкою до різних варіацій в даних та здатна узагальнювати знання ефективно.

4.3 Оцінка ефективності та точності моделі.

Після завершення тренування моделі, важливим кроком є аналіз її результатів на валідаційних даних. Розглянемо ключові метрики:

- **Loss** (Втрати): 0.0181 - Ця метрика відображає ступінь відхилення передбачених значень моделі від дійсних. Низьке значення втрат свідчить про високу здатність моделі до прогнозування.
- **AUC**: 0.9634 - Значення близьке до 1 свідчить про відмінну здатність моделі відокремлювати позитивні та негативні класи.
- **Precision** (Точність): 0.6242 - Модель з правильністю 62.42% визначала позитивний клас. Важливо зазначити, що "позитивним" за замовчуванням вважається те, на що модель дала оцінку більше ніж в 50%. Враховуючи «розмитість» даних, це непоганий результат, оскільки к у більшості випадків модель з >50% впевненістю порекомендує вірний стандарт.

Розглядаючи ці результати, можна сказати, що модель показала здатність до високоточного прогнозування. Проте, необхідно враховувати певні нюанси даних, які були використані для навчання. В цілому, результати свідчать про ефективність використаної архітектури та підходу до обробки даних, але й простір для майбутніх покращень.

5. Результати

Розбір декількох запитів до моделі:

Запит: «addition and subtraction to 20»

Результат:

1.OA.C.6: 81.13%

2.OA.B.2: 52.79%

1.OA.B.3: 43.28%

1.OA.C.5: 35.95%

1.OA.A.1: 22.83%

Початок опису стандарту 1.OA.C.6: «Add and subtract within 20 ...». (16)

Початок опису стандарту 2.OA.B.2: «Fluently add and subtract within 20 using mental strategies ...». (16)

Запит: «pythagorean theorem»

Результат:

8.G.B.7: 99.27%

8.G.B.6: 78.21%

8.G.B.8: 38.54%

HSG-SRT.C.8: 24.77%

HSG-SRT.B.4: 2.95%

Опис стандарту 8.G.B.7: «Apply the Pythagorean Theorem to determine unknown side lengths in right triangles in real-world and mathematical problems in two and three dimensions». (16)

Опису стандарту 8.G.B.6: «Apply the Pythagorean Theorem to find the distance between two points in a coordinate system.». (16)

Запит: «multiplying fractions by whole numbers»

Результат:

4.NF.B.4: 75.11%

4.NF.B.4a: 52.97%

5.NF.B.4: 44.01%

4.NF.B.4c: 40.99%

4.NF.B.4b: 33.53%

Опис стандарту 4.NF.B.4: «Apply and extend previous understandings of multiplication to multiply a fraction by a whole number». (16)

Хоч ці запити й були значно простіші, ніж назви продуктів, на яких вчилась нейронна мережа, модель все одно дає дуже точні результати, які можна використовувати на практиці. Але крім запитів на тему, можна ще й шукати найімовірніші стандарти для різних пір року або свят.

«3rd grade Halloween»

«3rd grade Christmas»

«3rd grade Easter»

3.NBT.A.1: 43.65%

3.NBT.A.1: 39.34%

3.NBT.A.1: 40.82%

3.NBT.A.2: 33.89%

3.NBT.A.2: 34.59%

3.MD.A.1: 37.92%

3.OA.C.7: 27.71%

3.OA.C.7: 31.30%

3.NBT.A.2: 34.91%

3.OA.A.4: 27.31%

3.OA.A.1: 27.38%

3.NF.A.2: 34.78%

3.OA.A.1: 26.51%

3.OA.A.4: 25.46%

3.OA.A.1: 26.87%

Такий спосіб не дає точних результатів, але може підказати тенденції щодо найімовірніших стандартів.

6. Майбутнє використання та перспективи

6.1 Адаптація моделі для української освітньої системи.

Вже сьогодні ведуться розмови про те, що *«Розвиток освіти в Україні має ґрунтуватися як на національних стандартах, так і на міжнародних нормах. У зв'язку з цим аналіз і узагальнення окремих освітніх інновацій у США становлять для нас не лише теоретичний інтерес, а й практичну цінність»* (17). Тож до цього слід готуватись вже сьогодні, а саме вчасно надати педагогам усі можливі інструменти для створення нових дидактичних матеріалів, які будуть сумісні з майбутньою системою освіти в Україні.

Але навіщо вчителям власноруч створювати дидактичні матеріали, якщо є підручники? Тому що кожен клас унікальний, та для кожного учня потрібен свій підхід, на жаль підручники вже не можуть забезпечити потрібного рівня зацікавленості учнів. Також, можливо, колись в Україні з'явиться власна онлайн платформа така як TeachersPayTeachers.com, щоб вчителі могли обмінюватись власними дидактичними матеріалами, та підбирати такі ресурси, які будуть цікаві саме їх учням.

Хоч одразу, таку модель як представлену у цій роботі навчити по новій на українському наборі даних не вийде, тому що не буде достатньої кількості даних. Але можна буде швидко адаптувати модель, наприклад на вході перекладати за допомогою Cloud Translation API запит вчителя з української мови, на англійську (18), а далі перенавчити лише останній шар нейронної мережі. Виходячи з того, що модель вже вміє знаходити закономірності в освітніх пошукових запитах, потрібно лише замінити ті 517 стандартів CCSS на свої, і це буде працювати з довільною кількістю стандартів, які будуть в українській системі. Такий підхід називається «Transfer Learning», та дозволяє навчити модель на значно меншому наборі даних. (19)

6.2 Подальша оптимізація та розширення функціонала моделі.

Ось кілька напрямків для подальшого удосконалення:

- **Параметр "надійності" записів:** При тренуванні моделі можливо встановлення різного рівня "довіри" для різних вхідних даних. Цей параметр може бути визначений в залежності від оцінки роботи, з якої було взято назву та стандарти, а також рівень надійності може бути зменшеним, якщо в роботі дуже багато стандартів. Це дозволить моделі приймати більш обґрунтовані рішення виходячи з надійності джерела.
- **Додавання нових вхідних параметрів:** З метою покращення точності моделі можна розглянути можливість додавання додаткових вхідних параметрів. Наприклад, "детальний опис" теми чи можливість вибору класу. Це дозволить моделі краще розуміти контекст запиту. Тому що зараз, якщо користувач введе "addition and subtraction", вказавши конкретний клас, модель матиме змогу точніше визначити, для якого класу відноситься запит – чи це додавання та віднімання в межах 20 для першого класу, або додавання та віднімання мішаних чисел для п'ятого класу.
- **Використання аугментації даних:** Щоб збільшити набір даних для тренування моделі, можна застосовувати аугментацію. Це допоможе створити нові варіанти запитів, використовуючи список синонімів до вже наявних слів. Такий підхід може збільшити розмаїття даних і покращити здатність моделі коректно реагувати на різноманітні запити.

Висновки

У даній роботі була розглянута актуальна проблема інтеграції новітніх технологічних рішень в освітній процес, особливо у контексті використання систем обробки природної мови для оптимізації пошуку освітніх стандартів. Звернуто увагу на необхідність створення інструментів, які спрощують та оптимізують процес вибору та розробки дидактичних матеріалів відповідно до конкретних освітніх стандартів. Це особливо актуально в контексті потреб педагогічних працівників, які постійно стикаються з необхідністю швидкого та точного доступу до відповідних математичних стандартів.

Основний акцент дослідження зосереджувався на створенні програмного інструменту для ефективного пошуку математичних стандартів Common Core State Standards (CCSS) за допомогою алгоритмів обробки природної мови.

Результатом роботи є програмний інструмент, який у майбутньому може спростити роботу педагогів при підготовці дидактичних матеріалів, надаючи можливість швидко та точно знаходити потрібні математичні стандарти CCSS, і, як наслідок, поліпшити якість навчального процесу в школах країни.

Список використаних джерел

1. Mathematics Standards | Common Core State Standards Initiative. URL: <https://www.thecorestandards.org/Math/>. (дата звернення: 16.10.2023).
2. Common Core Math Progressions - Biting Into Common Core Math. URL: <http://www.bitingintothecore.com/progressions-and-learning-trajectories.html>. (дата звернення: 16.10.2023).
3. Common Core State Standards Initiative. Grade 3 » Number & Operations in Base Ten | Common Core State Standards Initiative. URL: 2021 г. <https://www.thecorestandards.org/Math/Content/3/NBT/#CCSS.Math.Content.3.NBT.A.2>. (дата звернення: 16.10.2023).
4. Amazon. What is NLP? - Natural Language Processing Explained - AWS. URL: <https://aws.amazon.com/ru/what-is/nlp>. (дата звернення: 16.10.2023).
5. King – Man + Woman = Queen: The Marvelous Mathematics of Computational Linguistics | MIT Technology Review. URL: 17 September 2015 г. <https://www.technologyreview.com/2015/09/17/166211/king-man-woman-queen-the-marvelous-mathematics-of-computational-linguistics/>. (дата звернення: 16.10.2023).
6. aditya_taparia. Bidirectional LSTM in NLP - GeeksforGeeks. URL: <https://www.geeksforgeeks.org/bidirectional-lstm-in-nlp/>. (дата звернення: 16.10.2023).
7. Teacher Synergy LLC. About Us | TpT. URL: <https://www.teacherspayteachers.com/About-Us>. (дата звернення: 16.10.2023).
8. All Resources | TPT. URL: <https://www.teacherspayteachers.com/browse>. (дата звернення: 16.10.2023).
9. TensorFlow. URL: <https://www.tensorflow.org/>. (дата звернення: 16.10.2023).
10. Google Colab. URL: <https://research.google.com/colaboratory/faq.html>. (дата звернення: 16.10.2023).

11. TensorFlow Hub. URL: TensorFlow Hub. (дата звернення: 16.10.2023).
12. Banner, Pierre-Antoine. Understanding Adam : how loss functions are minimized ? | by Pierre-Antoine Banner | Towards Data Science. URL: <https://towardsdatascience.com/understanding-adam-how-loss-functions-are-minimized-3a75d36ebdfc>. (дата звернення: 16.10.2023).
13. Classification: ROC Curve and AUC | Machine Learning | Google for Developers. URL: <https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc>. (дата звернення: 16.10.2023).
14. tf.keras.metrics.Precision | TensorFlow v2.14.0. URL: https://www.tensorflow.org/api_docs/python/tf/keras/metrics/Precision. (дата звернення: 16.10.2023).
15. tf.keras.metrics.Recall | TensorFlow v2.14.0. URL: https://www.tensorflow.org/api_docs/python/tf/keras/metrics/Recall. (дата звернення: 16.10.2023).
16. Common Core State Standards Initiative. Common Core State Standards for Mathematics. URL: <https://learning.ccsso.org/wp-content/uploads/2022/11/ADA-Compliant-Math-Standards.pdf>. (дата звернення: 16.10.2023).
17. ІННОВАЦІЯ ЯК ТРЕНД РОЗВИТКУ ЗАГАЛЬНОЇ СЕРЕДНЬОЇ ОСВІТИ: ДОСВІД США. М., Кравченко С. 1, 2022 р., Освітня аналітика України, ст. 114.
18. Cloud Translation API | Google Cloud. URL: <https://cloud.google.com/translate/docs/reference/rest>. (дата звернення: 16.10.2023).
19. Donges, Niklas. What Is Transfer Learning? A Guide for Deep Learning | Built In. URL: <https://builtin.com/data-science/transfer-learning>. (дата звернення: 16.10.2023).