DOI 10.31891/2307-5732-2024-333-2-72
УДК 004.82

КЛІЧУК БОГДАН
Національний університет «Львівська політехніка»
https://orcid.org/0009-0009-0326-2165
e-mail: klichukb@gmail.com
ХАВАЛКО ВІКТОР
Національний університет "Львівська політехніка"
https://orcid.org/0000-0002-9585-3078
e-mail: viktor.m.khavalko@lpnu.ua

МЕТОД ВИЯВЛЕННЯ ДРЕЙФУ ТЕКСТОВИХ ДАНИХ В МОВНИХ МОДЕЛЯХ

У сучасному світі активного застосування штучного інтелекту (ШІ) у різних областях життєдіяльності виникає потреба в адаптації та постійному оновленні інтелектуальних систем, зокрема мовних моделей, до змінних умов та вимог реального світу. Важливу роль у цьому процесі відіграють системи MLOps, які забезпечують ефективне впровадження, моніторинг, та неперервне оновлення ШІ-систем з метою підтримки їх високої точності та адекватності до поточного контексту використання. Одним з ключових аспектів ефективності мовних моделей та назагал технологій обробки природної мови є здатність своєчасно виявляти зміни в середовищі, в якому оперують створені моделі, тобто зміна характеру даних, що поступають на вхід, та як це середовище відрізніться від того, на якому модель була натренована. Це задача виявлення дрейфу даних – змін у розподілах даних, якими оперують моделі, що може суттєво вплинути на їх продуктивність.

В статті проведено аналіз найбільших типових методів та алгоритмів виявлення дрейфу даних в текстових словах для оцінки поведінки мовних моделей. Проведено ряд експериментів з виявлення дрейфу даних у словах, обчислено дрейф, запропоновано новий метод та проведено порівняння результатів за допомогою додаткових метрик та здійснено візуальну їх оцінку.

Результати застосування методів демонструють доцільність їх використання для виявлення дрейфу, чутливість та реакцію на відсутність змін. Найбільш базові метрики визначення відстані між усередненими вкладеннями та слів, а також деякі методи з застосування кластеризації продемонстрували найточніші результати. Виявлення дрейфу в текстових даних може бути ефективно реалізовано за допомогою вкладень слів, що відображають внутрішнє представлення слів в мовних моделях, а також дозволяють якісно оцінити зміни їх семантики та контексту.

Ключові слова: мовні моделі, вкладення слів, дрейф даних, MLOps, моніторинг моделей, NLP.

KLICHUK BOHDAN, KHAVALKO VIKTOR
Lviv Polytechnic National University

METHOD FOR DETECTING TEXT DRIFT IN LANGUAGE MODELS

In the modern world of rapid artificial intelligence (AI) adaption across various aspects of life, a need for the adaptation and continual updating of intelligent systems to the changing conditions and demands of the real world emerges, including language models. MLOps systems play a crucial role by ensuring efficient implementation, monitoring, and continuous updating of AI systems to maintain their accuracy and relevance to the current usage context. A key aspect of the effectiveness of language models and technologies of natural language processing, in general, is their ability to timely detect changes in the environment in which the created models operate, i.e., changes like incoming data and how this environment differs from that with which the model was trained. This is the task of detecting data drift - changes in the data distributions that the models operate with, which can significantly affect their performance.

The most common methods and algorithms for detecting data drift in textual corpora to assess the behavior of language models are reviewed. Semantic changes are identified using word embeddings. A series of experiments are conducted to detect data drift in word embedding values using a set of text corpora, where new context and semantics of the target words is gradually injected. Then drift values are calculated using the selected methods, comparison of results using additional metrics and visual evaluation are proposed.

The results of applying these methods assess the appropriateness of their use for detecting drift, sensitivity, and response to the absence of changes. The most basic metrics for determining the distance between average embeddings and words and some clustering methods demonstrate the most accurate results. In language model application drift detection in text data can be efficiently performed using word embeddings and a variety of metrics for tracking change, since word embeddings specifically represent semantics and context of words in context-aware language models.

Keywords: language models, word embeddings, data drift, MLOps, model monitoring, NLP.

Постановка проблеми

Більшість сучасних систем штучного інтелекту використовують у своїй основі моделі штучного інтелекту, навчені на певній та обмеженій вибірці даних, що не передбачає їх самостійного до навчання, через що вони можуть працювати лише з наявним обсягом знань. З часом, іноді навіть безпосередньо після впровадження, такі моделі можуть ставати застарілими внаслідок змін у середовищі застосування, що проявляється як дрейф даних - зміни в розподілі вхідних або вихідних даних та зміни залежностей між вхідними і вихідними даними. Це породжує необхідність розробки інфраструктури та методик моніторингу змін в поведінці моделей штучного інтелекту (MLOps) з метою їх подальшого удосконалення та мінімізації негативного впливу на кінцеве середовище, що може включати бізнес, охорону здоров'я та життя, соціальні настрої, достовірність інформації та знань про світ та інше. У сфері обробки природної мови, моніторинг змін вхідних та вихідних даних мовних моделей становить один із ключових викликів у галузі NLP та MLOps. Це відкриває можливості для оперативного реагування на зміни у середовищі, перенавчання моделей або обмеження їх негативного впливу.

Деталізуючи зміни даних як загальні характеристики стану системи в даному середовищі, конкретними важливими факторами моніторингу є: більш детальне розуміння цих змін, наприклад розуміння

зміни семантики чи контексту конкретних слів, що можуть бути ключовими та описувати цілі тренди у тому, з якими даними має справу модель штучного інтелекту, зміна розподілу контексту слів, наприклад вже відомі слова, що все частіше зустрічаються в конкретному контексті, або ж поява нових слів або термінології, які невідомі моделі. Розвиток та побудова методів виявлення дрейфу даних в найменших цілісних одиницях текстової інформації, таких як слова, допоможе краще розуміти поведінку цих моделей, їх слабкі сторони, та вчасно реагувати їх оновленням, зменшуючи негативний вплив на середовище застосування та прийняті рішення.

В сфері моніторингу мовних моделей в середовищі застосування та виявлення дрейфу вхідних неструктурованих даних, одним з потужних інструментів є доступ до внутрішніх репрезентацій слів в мовних моделях, які називають вкладеннями (embeddings). Це багатовимірні вектори, кожен вимір яких характеризує слово або його частину в певний спосіб, особливість, виявлену нейронною мережею, що має числове відображення, що може або змінюватись від контексту або бути постійною, залежно від типу моделі. Розуміння як змінюються вкладення слів для виявлення дрейфу, вибір метрик та методів для порівняння є актуальною проблемою технологій обробки природної мови та рушієм побудови систем MLOps для мовних моделей.

Аналіз останніх досліджень

MLOps (Machine Learning Operations) передбачає цілий комплекс рішень для безперервного впровадження моделей штучного інтелекту в кінцеве середовище, та досліджується більше активно, як цілісна методологія [1], так і як окремі підзадачі та оцінку впливу застосування таких платформ для бізнесу [2].

Задача виявлення дрейфу є однією з найбільш типових для систем відстеження поведінки моделей штучного інтелекту, і багато методів вже відомо на сьогоднішній день для обробки структурованих даних.

Для більш деталізованого розгляду, розділимо різні поняття, що стосуються термінології дрейфу:

- *Дрейф вхідних даних* (віртуальний, коваріатний дрейф) - це зміна розподілу вхідних даних тобто p(X), і відповідно p(X|Y), що свідчить про потенційну зміну середовища, і появи даних, в випадку яких точність моделі може падати або ж модель може починати галюцинувати.
- *Дрейф вихідних даних* (дрейф цільової змінної) - це зміна розподілу вихідних даних тобто p(Y|X), тобто зміщення в розподілі вихідних даних, що може свідчити про незбалансованість вхідних даних.
- *Дрейф концепту* (дрейф семантики цільової змінної) - це зміна, яка суті цільової змінної, може не залежати від двох інших але свідчити про потенційну зміну співвідношення між вхідними і вихідними даними.

В даній роботі досліджується виявлення коваріатного дрейфу конкретних слів.

Робота [3] демонструє огляд великої кількості метрик та методів для виявлення коваріатного дрейфу вкладень слів, але упускає множину інших метрик, розглянутих в інших роботах. Пропрацьовано ряд типів агрегації вкладень, проведено експерименти по порівнянню метрик та підходів з різних точок зору.

В [4] проводиться порівняння різних методів виявлення дрейфу вхідних даних, використовуючи векторні репрезентації слів, обчислених різними способами: TF-IDF, doc2vec, та вкладень моделі BERT, і визначивши найбільш чутливі до змін метрики. Використані метрики для порівняння розподілу вкладень слова: критерій узгодженості Колмогорова-Смирнова, та максимальне середнє відхилення (MMD). Розробка та дослідження DetAIL [5] пропонує інструментарій для відслідковування коваріатного дрейфу, що працює над потоком даних, виявляючи зміни в узагальнених вкладеннях слів вхідного тексту, пропонуючи широкий спектр різних метрик для обрахунку дрейфу та працює з відомими фреймворками для машинного навчання.

В [6] досліджено проблему виявлення зміни значення слів, використовуючи відому модель BERT, та застосовуючи підхід з кластеризацією вкладень слова і подальшого визначення зміни розподілу входжень слова в знайдені кластери. Були використані метрики JSD (Дивергенція Єнсена-Шеннона) та Wasserstein Distance (Відстань Васерштайна). В [7] для виявлення змін контексту слів застосовано як кластеризацію, так і побудову усередненого прототипу вкладень слова, що досліджується. Використані метрики: JSD, евклідова відстань, відстань Гаусдорфа, тощо. В [8, 9] піднімається проблема дослідження дрейфу вхідних даних на основі вкладень слів, що натреновані на окремих часових відрізках нових даних, співставлених ти вирівняних, оскільки фактично через стохастичну природу деяких моделей виміри кінцевих векторів можуть між собою не співпадати.

В [10, 11] використано репрезентацію всієї тренувальної вибірки як усереднений вектор вкладень з вимірами, ідентичними до вимірів конкретного слова. Використані метрики: Відстань з косинусом, евклідова відстань, максимальне середнє відхилення. В [12] розроблено підхід до формування послідовностей векторів, що зв'язують значення векторів конкретного слова в різні проміжки часу, а також запроваджено поняття "компаса" для орієнтації та вирівнювання порівнюваних між собою вкладень слів.

Формулювання цілей статті

Метою роботи є аналіз та дослідження найбільш типових методів та метрик виявлення дрейфу вхідних даних у семантиці та контексті обраних ключових слів для прикладної контекстнозалежної мовної моделі, за допомогою вкладень слів. Використовуючи дослідження обраних методів та метрик, запропоновано метод на основі додаткових метрик, який дозволив виявити: зміни між тренувальними та новими текстовими даними моделі, залежність амплітуди виявлених значень дрейфу від коефіцієнту зміненої семантики досліджуваних слів та здійснити порівняння результатів.

Виклад основного матеріалу

Для проведення дослідження було обрано одну з відомих моделей мови, що оперує вкладеннями - BERT [13], та фреймворк PyTorch. Зауважимо, що ціллю роботи є побудувати метод, який буде працювати і для інших

моделей, що є контекстно-залежними. Опис загальної картини дрейфу вираховується за допомогою дослідження усіх входжень конкретного слова в вибірках, та накладання метрики та методів порівняння вкладень цих входжень. Деякі методи порівняння вимагають узагальнення вкладень слова у вибірці даних або іншого опису їх розподілу, для чого застосовується агрегація. Типи агрегації, використаних для проведення обчислень:

- Без агрегації. В цьому випадку використовується всі екземпляри вибірки для застосування подальшої метрики. Не вимагає додаткового кроку для обчислень, але менш практично з точки зору компактності інформації, що не обхідна для здійснення порівнянь.
- Агрегація за допомогою середнього (mean). Для всіх входжень слів у вибірці вибрати вкладення та знайти прототипне вкладення, кожен вимір якого містить середнє значення серед усіх входжень слова.
- Агрегація за допомогою кластеризації. В експериментах використаний метод k-середніх (де k = 7). В даних експериментах, при порівнянні двох вибірок даних D_1 та D_2 та вкладень цільового слова, кластеризація здійснюється на екземплярах входжень, з обох вибірок разом $D_1 \cup D_2$.

Перелік розглянутих методів та метрик:

- Евклідова відстань (Euclidean Distance) [7, 10, 11], тип агрегації - середнє значення. Евклідова відстань обчислює пряму лінійну відстань між двома точками у просторі будь-якої розмірності. Це найпростіша та найочевидніша міра відстані, яка використовує теорему Піфагора для визначення довжини відрізка.
- Відстань з косинусом (Cosine Distance) [10, 11], тип агрегації - середнє значення. Обчислює різницю між двома векторами у просторі за допомогою косинуса кута між ними. Вона вимірюється як *1-cosine_similarity(a, b),* де *cosine_similarity(a, b)* – косинус подібності двох векторів a, b.
- Відстань Гаусдорфа (Hausdorff Distance) [7], тип агрегації - середнє значення. Обчислює максимальну відстань між двома наборами точок. Ця метрика вимірює, наскільки далеко кожна точка набору A знаходиться від найближчої точки набору B, і навпаки, а потім вибирає найбільше з цих відстаней як кінцевий результат.
- Критерій узгодженості Колмогорова–Смирнова (Kolmogorov-Smirnov Test) [4]. Без агрегації. Обчислює схожість двох розподілів даних.
- Максимальне Середнє Відхилення (Maximum Mean Discrepancy, MMD) [4, 11]. MMD обчислюється як відстань між середніми значеннями двох розподілів у деякому просторі ознак. Якщо MMD мала, це означає, що розподіли є подібними; якщо велика — розподіли значно відрізняються.
- Дивергенція Єнсена-Шеннона (Jensen-Shannon Divergence, JSD) [6, 7], тип агрегації - кластеризація. Міра статистичної відстані між двома або кількома розподілами (або розподілом входжень слова в різні кластери). Симетрична та обмежена міра, що базується на ентропії, і э більш стабільною та надійною порівняно з дивергенцією Кульбака-Лейблера. JSD обчислює середню відмінність між двома розподілами та їх середнім розподілом.
- Відстань за косинусом (Cosine Distance), тип агрегації – кластеризація. Перетворивши розподіл входжень слова у кластери в двох вибірках на числові вектори, обчислюємо між ними відстань за косинусом.
- Евклідова відстань (Euclidean Distance), тип агрегації – кластеризація. Перетворивши розподіл входжень слова у кластери в двох вибірках, що порівнюються, на числові вектори, обчислюємо евклідову відстань.

Вибір прикладної проблеми. Прикладною проблемою обробки природної мови для дослідження дрейфу даних обрано ідентифікацію та класифікацію токсичного тексту. Використано датасет [14] з коментарями на Wikipedia, які розмічені 5 різними класами токсичності, а отже кожен коментар може належати до 0-5 таких класів. Для дослідження натреновано простий багатомітковий класифікатор на основі BERT моделі ('bert-base-uncased'), доналаштованої цим датасетом, за кроками поданими в [15]. Тренувальний датасет містить 157 тисяч промаркованих коментарів різної довжини англійською мовою - D_{train}, але не більше ніж 512 токенів. Кожен вектор вкладень для 1 токену в випадку моделі BERT є 768-вимірним.

Деякі з існуючих досліджень використовували сторонні моделі і алгоритми для обчислення вкладень, але оскільки ми досліджуємо працездатність та виявляємо потенційну неактуальність конкретної піддослідної моделі, доцільно використовувати саме її репрезентації слів та контексту, на противагу залученню сторонніх NLP алгоритмів та моделей. Зауважимо, що підхід сторонніх моделей також має місце, якщо піддослідна модель не надає досліднику доступу до вкладень слів або інакше представляє слова.

Доступні 5 класів токсичності: токсичність, сильна токсичність, образа, індивідуальна ненависть, погроза, непристойність. Усі ці класи мають характерний контекст, або ж прямі образи, нецензурні слова, приниження гідності, посилання на особисті якості.

Моделювання дрейфу виглядає наступним чином:

- В якості дрейфу запропонували як приклад новий тип токсичності - зневага до етнічності (ethnicity hate). Звужуємо розгляд до конкретних K слів, що є назвами етнічностей/національностей:

$K = \{american, austrian, british, canadian, french, german, hungarian, irish, italian, mexican, polish, scottish, spanish\}$

- Формуємо датасет D_{norm}: З додаткового промаркованого датасету в 60 тис коментарів вибираємо 2500 завідомо нетоксичних коментарів, які не бачив класифікатор, і які містять мінімум 1 з слів K.
- Формуємо датасет D_{tox}: За допомогою комбінацій з різних шаблонів конструюємо 2500 коментарів, де частина коментаря містить або непряму образу з посиланням на етнічність, або змішану з іншим типом токсичності, наприклад нецензурними словами або погрозами.
- Формуємо вибірки нових даних: множину датасетів $DD_{new,}$ (тобто усіх проведених експериментів), де кожен елемент є тестовим датасетом з 2500 коментарів, частина яких є токсичним текстом, тобто містить змінений контекст слів (D_{tox}), а решта - нормальним (D_{norm}).

Нехай I_{tox} - частки токсичного тексту усіх експериментів, *D(i)* - функція генерації нового датасету, що залежить від частки токсичного тексту, *RND(D, i)* - довільна вибірка рядків розміру *i* з датасету *D*.

$$I_{tox} = \{0, \dots, 0, 0.01,\ 0.02,\ 0.03,\ 0.04,\ 0.05,\ 0.1,\ 0.15,\ 0.2,\ \dots, 0.55,\ 0.6\},$$

$$D(i) = RND(D_{tox},\ i) \cup RND(D_{norm},\ 1 - i).$$

Тоді DD_{new} - множина, елементи якої є результатом $D(i)$, $\forall i \in I_{tox}$. I_{tox} містить ряд значень 0 (N = 10), що імітують відсутність дрейфу, а також послідовність зростаючих коефіцієнтів [0, 1], що імітують зростаючий коефіцієнт D_{tox}.

Хід проведення експериментів:

1. Визначити дрейф базис D_{base} - усі коментарі з D_{train}, які містять хоча б одне таке слово з *K*, (6000 рядків)
2. Знайти вкладення всіх входжень слів K для дрейф базису D_{base}.
3. Ітеративно знайти вкладення всіх входжень слів *K* для всіх вибірок дрейфу DD_{new}.
4. Застосувати до кожної групи вкладень передбачені типи агрегації.
5. Обрати метрики обчислення коваріатного дрейфу:
6. Імітуючи проміжки часу роботи моделі в середовищі з дрейфом вхідних даних, обчислити коваріатний дрейф для всіх вибірок нових даних $DD_{new.}$
7. Для кожного цільового слова $k \in K$ та випадку $D_{new} \in DD_{new}$ обчислити частку токсичних випадків цього слова в вибірці D_{new} та відобразити залежність значення дрейфу від частки токсичних випадків використання слова *k* в D_{new} до усіх.
8. Оцінити результати. В якості результату оцінимо монотонність зростання коваріатного дрейфу для всіх вибірок $DD_{new.}$ Для визначення монотонності зростання та узгодженості значення обчисленого дрейфу з часткою зміненого контексту обчислимо коефіцієнти кореляції рангу Кендалла та Спірмена для кожного слова К, та знайдемо усереднене їх значення для всіх слів.

Результати проведених експериментів

В результаті проведених експериментів за допомогою вкладень слів було застосовано кілька методів для обчислення дрейфу контексту слів у нових вхідних даних мовної моделі у порівнянні з тренувальними даними, здійснене чисельне порівняння цих методів та візуальна оцінка. Експерименти було зосереджено над визначеною множиною слів К, контекст яких був намірено, контрольовано та поступово змінений зі зростаючою силою у вибірках нових вхідних даних. Отримавши для кожного слова з *K* послідовність точок *(x, y)* залежності значення дрейфу від частки нового зміненого контексту цього слова, було обчислено оцінку монотонності та узгодженості цих послідовностей, а також візуальну оцінку. Згідно з результатами, всі методи за виключенням відстані Гаусдорфа виявились придатними для виявлення дрейфу. Найбільш узгоджені з коефіцієнтом зміненого контексту – усереднена відстань з косинусом та дивергенція Єнсена-Шеннона з кластеризацією k-середніх (k =7). Критерій узгодженості Колмогорова -Смирнова демонструє високий результат, але в експериментах з відсутністю дрейфу спостерігається високий рівень шуму. Використання класичних метрик (відстань з косинусом, евклідова відстань) між векторами, сформованими розподілом вкладень в кластерах, не продемонструвало доцільності для чисельного виявлення дрейфу.

Кінцеві результати чисельних порівнянь для всіх досліджуваних слів приведені в таб. 1. На рис. 1, 2 зображено результати обчислень на прикладі 2 слів: *hungarian, american* з відповідно найменшою та найбільшою частотою використання цих слів з-поміж усіх досліджуваних слів. Для візуального відображення значення були нормалізовані до проміжку [0, 1].

Таблиця 1

Результати порівняння роботи методів та застосованих метрик виявлення дрейфу, агрегованих для всіх досліджуваних слів з використанням RMSE

Позначення	Тип агрегації	Метрика	Коеф. Кендала	Коеф. Спірмана	P-val
Euc	Mean	Euclidean Distance	0.857	0.934	< 0.001
Cos	Mean	Cosine Distance	0.863	0.938	< 0.001
Haus	Mean	Hausdorff Distance	0.59	0.78	0.2
KS	N/A	Kolmogorov-Smirnov Test	0.817	0.91	< 0.001
MMD	N/A	Maximum Mean Discrepancy	0.834	0.922	< 0.001
KM + JSD	K-means(7)	Jensen-Shannon Divergence	0.864	0.937	< 0.001
KM + Cos	K-means(7)	Cosine Distance	0.806	0.909	< 0.01
KM + Euc	K-means(7)	Euclidean Distance	0.817	0.919	< 0.01

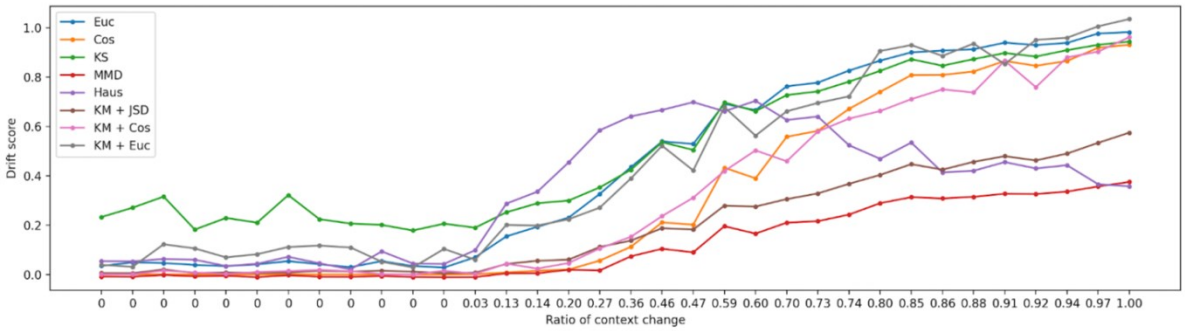

Рис 1. Залежність значення коваріатного дрейфу від частки зміненого контексту для слова hungarian (найменше даних)

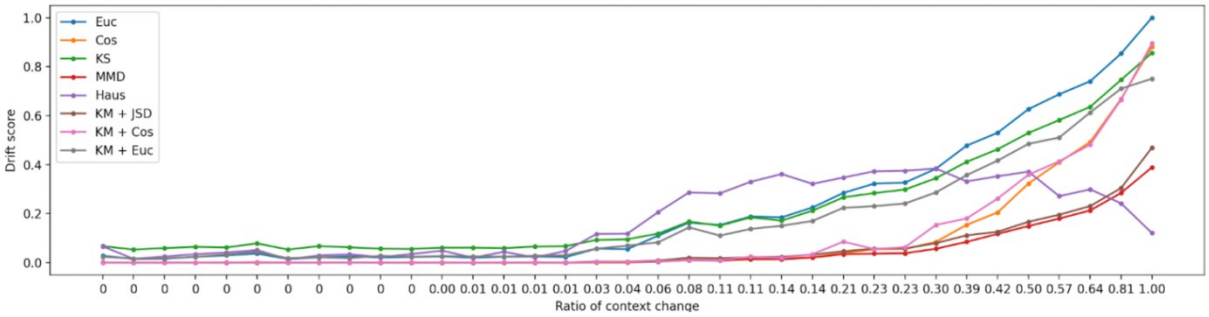

Рис 2. Залежність значення коваріатного дрейфу від частки зміненого контексту для слова american (найбільше даних)

Висновки

Продемонстровано та застосовано методи виявлення змін контексту та семантики слів у вхідних текстових даних прикладної мовної моделі на прикладі реальної системи для класифікації тексту, за допомогою вкладень слів. Проведено експерименти з використанням відомих методів агрегації вкладень слів та їх порівняння для виявлення коваріатного дрейфу. Використання базових класичних метрик виявлення відстані між усередненими векторами вкладень є простим, розширюваним підходом, який демонструє високі результати, утримуючись від високого рівня шуму. Варто зауважити, що, хоч, методи кластеризації, згідно з проведеними експериментами, не показують явної переваги у чисельному виявленні дрейфу, вони мають суттєву перевагу в поясненні змін та групуванні схожих сенсів та семантики слів, хоч і є обчислювально складнішими.

Подальший розвиток методів виявлення дрейфу даних в мовних моделях, окрім методів та метрик їх обчислення, повинен рухатись в напрямку пояснення таких змін, класифікації а також конкретизації. Для систем MLOps важливим є інтерпретація сигналу виявлення дрейфу – для розуміння необхідності дотренування системи обробки тексту та висновків щодо потенційного падіння точності цих систем, деталізація змін та пояснення їх походження з конкретними прикладами.

References

1. Zhengxin F.Yi,Y., Jingyu Z., Yue L., Yuechen M., Qinghua L., Xiwei X., Jeff W., Chen W., Shuai Z., Shiping C. MLOps Spanning Whole Machine Learning Life Cycle: A Survey. *arXiv*, 2023. https://doi.org/10.48550/arXiv.2309.10000

2. Berkmanas R. Why MLOps Is Important for Your Business? *Business / MLOps*. URL: https://easyflow.tech/why-mlops-is-important-for-your-business.

3. Montanelli S., Periti F. A Survey on Contextualised Semantic Shift Detection. *arXiv*, 2023. https://doi.org/10.48550/arXiv.2304.01666.

4. Sodar V., Sekseria A. Detecting covariate drift in text data using document embeddings and dimensionality reduction. *arXiv,* 2023, https://doi.org/10.48550/arXiv.2309.10000

5. Madaan N., Manjunatha A., Nambiar H., Goel A.K., Kumar H., Saha D., Bedathur S. DetAIL: A Tool to Automatically Detect and Analyze Drift In Language. *Proc. of the 35th Conference on Innovative Applications of Artificial Intelligence (AAAI/IAAI)*, 2023. https://doi.org/10.48550/arXiv.2211.04250

6. Montariol S., Martinc M., Pivovarova L. Scalable and Interpretable Semantic Change Detection. *In Proc. of the 2021 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* 2021.- *4642–4652p.* https://doi.org/10.18653/v1/2021.naacl-main.369

7. Wang B., Wang A., Chen F., Wang Y., Kuo C.-C. J. Evaluating Word Embedding Models: Methods and Experimental Results. *APSIPA Transactions on Signal and Information Processing*, 2019, vol 8, e19. DOI: https://doi.org/10.1017/ATSIP.2019.12

8. Jun, Y. Measuring Semantic Changes Using Temporal Word Embeddings. *Towards Data Science,* 2021 https://towardsdatascience.com/measuring-semantic-changes-using-temporal-word-embedding-6fc3f16cfdb4

9. Birunda S., Devi R.K. A Review on Word Embedding Techniques for Text Classification. *Innovative Data Communication Technologies and Applicatio*n, 2021, pp. 267-281. http://dx.doi.org/10.1007/978-981-15-9651-3_23

10. Lopatecki J. Monitoring Embedding/Vector Drift Using Euclidean Distance, 2023. URL: https://arize.com/blog-course/embedding-drift-euclidean-distance/

11. Filippova O., Samuylova E. Shift happens: we compared 5 methods to detect drift in ML embeddings. *Evidently AI blog,* 2023. URL: https://www.evidentlyai.com/blog/embedding-drift-detection

12. Di Carlo V., Bianchi F., Palmonari M. Training Temporal Word Embeddings with a Compass. *Proc. of the AAAI Conference on Artificial Intelligence*, 33(01), 2019.- pp.6326-6334. https://doi.org/10.1609/aaai.v33i01.33016326

13. Devlin J., Chang M.-W., Lee K., Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proc. of the 2019 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2019.- vol. 1, pp. 4171–4186. https://doi.org/10.18653/v1%2FN19-1423

14. Cjadams, Sorensen, J., Elliott, J., Dixon, L., McDonald, M., nithum, Cukierski, W. Toxic Comment Classification Challenge. *Kaggle*, 2017. https://kaggle.com/competitions/jigsaw-toxic-comment-classification-challenge

15. HowSun C., Qiu X., Xu.Y., Huang X. How to Fine-Tune BERT for Text Classification? *Chinese Computational Linguistics. CCL 2019. Lecture Notes in Computer Science*, vol 11856. Springer, Cham. DOI: https://doi.org/10.1007/978-3-030-32381-3_16