**Issue 14, 2023**

https://doi.org/10.23939/sisn2023.14.210

**УДК 004.89:004.738.5:004.415.3:004.82(045)**

# МЕТОД І МОДЕЛЬ ОПРАЦЮВАННЯ ТЕКСТОВОЇ ІНФОРМАЦІЇ НА НАВЧЕНОМУ ТРАНСФОРМЕРІ ДЛЯ БАЗИ ЗНАНЬ

**Василь Литвин[1], Володимир Тимчук[2]**

[1] Національний університет "Львівська політехніка", кафедри інформаційних систем та мереж, вул. С. Бандери, 12, Львів, Україна

[1–2] Національна академія сухопутних військ імені гетьмана Петра Сагайдачного, вул. Героїв Майдану, 32, Львів, Україна

[1] vasyl.v.lytvyn@lpnu.ua, ORCID: 0000-0002-9676-0180,

[2] v_tymchuk@yahoo.co.uk, ORCID: 0000-0002-3549-2813



**Невпорядкована база знань формується із різних множин нестандартизованих документів. У системі підтримки прийняття рішень ключовим є своєчасний доступ до інформації із бази знань. У статті описано модель інформаційно-пошукової системи щодо роботи з множиною знань, поданих у форматі PDF, одному із основних у військово-спеціалізованих базах знань. Модель розроблено на навченому трансформері із забезпеченням міжмовного перекладу, що загалом формує метод обробки текстової інформації.**

**Ключові слова: система обробки інформації; система підтримки прийняття рішень; метод обробки мови та тексту на навченому трансформері; машинне навчання; онтологія баз даних; множини знань.**

## Вступ. Загальна постановка проблеми

Потреба та здійснення захисту України в умовах воєнної агресії російської федерації стали, як і передбачено Конституцією України та іншими законодавчими актами, загальними. Задля забезпечення цього воєнна організація держави здійснює на постійній основі заходи комплектування наявних військових структур, а також створення нових військових формувань. Типових організаційно-штатних структур військових частин не існує, що пояснюється дуже широким номенклатурним рядом зразків озброєння та військової техніки (ОВТ), з якими доводиться мати справу військовослужбовцям та працівникам. Система накопичення та передавання знань тільки формується, особливо стосовно зразків ОВТ, які до повномасштабного вторгнення російської федерації в Україну не експлуатувалися, досвіду, знань і, найголовніше, часу помітно бракує. В цих умовах поєднання інструкторського навчання і самонавчання є вагомим підходом до підготовки персоналу щодо правильної експлуатації зразків ОВТ та виконання інших посадових обов'язків. Як інструкторське навчання, так і самонавчання повинно ґрунтуватися на достовірних джерелах знань. У загальній системі воєнної організації держави ці джерела знань є розрізненими, невпорядкованими, неописаними і часто неапробованими. Тож будь-які стратегії щодо систематизації та класифікації бази знань вкрай актуальні та значущі.

### Зв'язок висвітленої проблеми із важливими науковими та практичними завданнями

Невпорядкована база знань може формуватися із нормативних документів, які регламентують послідовність дій, підручників, навчальних посібників, методичних посібників і рекомендацій, пам'яток, інструкцій і настанов щодо експлуатації, мануалів тощо різними мовами (передусім українською, російською, англійською), різних років видання, різної якості та доступності. З метою поширення складових цієї бази знань її у той або інший спосіб оцифровують. Кінцевий результат – наявність матеріалів бази знань у певних форматах електронних документів. Вагому частку становлять документи у форматі PDF.

Використання формату PDF у базах знань має історичну та практичну основу, зокрема відомі його переваги щодо простоти збереження та наочності відтворення збережених даних, недопущення несанкціонованих змін у матеріалах, певної стійкості стосовно кіберзагроз, зручності тиражування для зберігання у надрукованому вигляді та можливості копіювання інформації в інші формати електронних документів.

Водночас за наявності багатьох документів у форматі PDF (як і в разі їх відсутності) дуже важко знайти необхідну інформацію. Тож створення бази даних для пошуку військово-спеціалізованої інформації вкрай потрібне.

У статті описано метод створення пошуково-інформаційної бази даних для інтелектуальної системи підтримки прийняття рішень (СППР) на основі розроблення моделі пошуку відповіді за інформаційним запитом у базі даних електронних документів у форматі PDF-файлів.

### Аналіз останніх досліджень та публікацій

Над цим завданням працювало чимало дослідників. У [1] для розпізнавання змісту текстового документа запропоновано стратегію діяльності програми-посередника, якою є інформаційна модель суб'єкта розпізнавання або уточнення стратегії на підставі виділених у цільовому текстовому документі даних. Цільовим вважають документ, щодо якого виконують цільові дії, наприклад розпізнавання тексту. В цитованій праці як цільові документи вибрані специфічні природномовні тексти – анотації наукових статей, які структуровані щодо правил написання у вигляді логічно зв'язаної послідовності стверджувальних речень. Окрім того, вони написані також англійською мовою, не містять графічного матеріалу та модальних зворотів. Задачу розв'язано за допомогою програмного пакета *CROCUS* [2], на вході підсистеми інформаційного пошуку якої була визначена множина ключових слів, а на виході – множина англомовних анотацій, розміщених у базі даних СУБД *MySQL*.

В [3] за допомогою методу резолюцій Робінсона для порівняння двох простих речень показано, що методи добування знань, які використовують окремі статистичні закономірності та пошук ключових слів, не здатні вилучати знання з інформації, оскільки не застосовують алгоритми лінгвістичної обробки текстів, тож авторка розвиває логіко-лінгвістичні моделі текстової інформації за змістом.

В [4] та в інших подібних працях за допомогою окремих програмних кодів реалізовано часткові завдання з пошуку інформації у базах даних у рядковому поданні, тобто, по суті, це завершальна стадія вирішення проблеми видобування знань. Самі моделі до тих або інших завдань теж розробляються, як-от абстрактна модель мовно-онтологічної інформаційної системи [5].

Проаналізовані джерела свідчать, що традиційним є комплексний підхід до проблеми – розроблення і моделі, і коду, і системи загалом. Власне за таким підходом виконано і ці дослідження.

**Основні завдання дослідження та їх значення**

Метою статті є розроблення методу опрацювання текстової інформації на навченому трансформері та моделі інформаційно-пошукової системи у військово-спеціалізованій базі знань.

**Постановка проблеми в загальному вигляді**

Прийняття рішень у військовій справі потребує, поміж іншим, великої сукупності даних, доступних за запитом. Забезпеченням цього доступу, пошуком інформації загалом та її опрацюванням традиційно займаються органи військового управління (ОВУ), делегуючи відповідні функції посадовим особам (ПО) та використовуючи передбачені засоби опрацювання інформації та програмне забезпечення.

У результаті типової роботи ОВУ та ПО на засобах опрацювання інформації зберігається великий обсяг довідкової, методичної, звітної та іншої інформації, яка постійно поповнюється. Іншою ознакою є те, що така інформація може зберігатися у вигляді документів у різних форматах представлення даних. Загалом, ця сукупність документів (і інформації) формує базу знань, до якої ПО звертаються у процесах підготовки прийняття рішень.

Подамо модель інформаційної СППР у військовій справі (рис. 1), зважаючи на циркуляцію такої інформації.

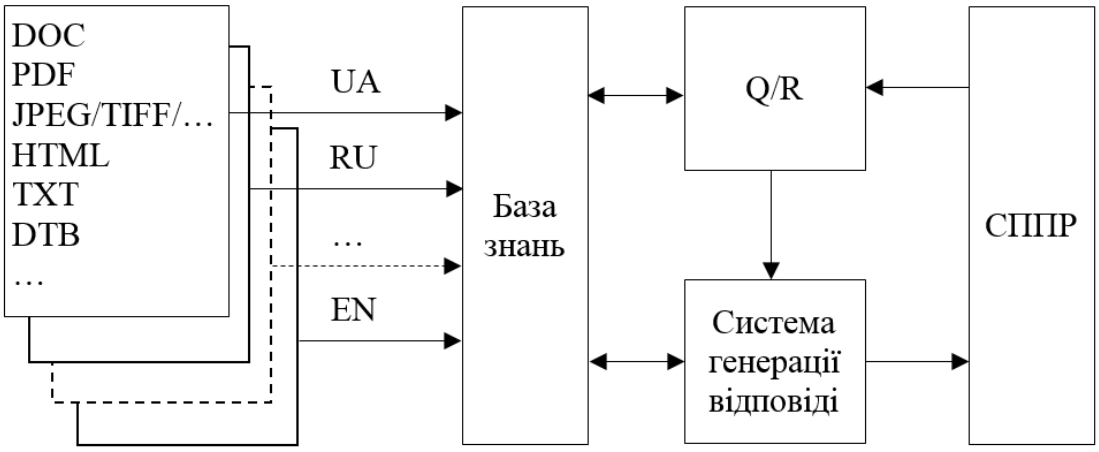



*Рис. 1. Структурна модель інформаційно-пошукової системи для підтримки прийняття рішень у військовій справі*

На вхід бази знань перманентно надходять дані у можливих форматах. Очевидно, що кожен формат потребує конкретних підходів для вирішення питань розпізнавання, впорядкування та подальшого опрацювання, що, звичайно, ускладнює систему опрацювання інформації. Обов'язкові формати, з якими працює база знань, зазначені. Для специфічних завдань їх набір може доповнюватися.

Іншою особливістю є те, що отримувані текстові дані можуть бути опрацьованими на різних мовах. Це не дивно, оскільки в Силах оборони України на озброєнні зразки ОВТ кількох поколінь і різних розробників: починаючи від зразків, які залишились від радянської армії і документація яких написана російською, через поставки від різних міжнародних партнерів, документація яких подана національними мовами, до трофейних зразків ОВТ новітніх розробок російського оборонно-промислового комплексу. Окрім того, база знань повинна передбачати розвідувальні та контррозвідувальні заходи, а без роботи із російською мовою це не вдасться втілити.

У такому разі база знань стає великим масивом підготовлених даних. Під підготовленістю розуміють, що вид даних підпорядкований цілям СППР, тобто якщо це інформаційна контент-база (на

основі текстових даних), то результатом підготовки даних є їх адаптованість до відбору та передавання до СППР за запитом.

Власне система запиту/відповіді (*Q/R* (*question/response*) подана на рис. 1 і забезпечує правильне передавання запитів від СППР до бази знань. Під правильністю розуміємо, що неоднозначності запитів у той або інший спосіб усуваються. Відповідно система генерації тексту формує результат пошуку згідно із запитом для передавання даних у СППР.

Онтологія у цьому випадку є такою:

1) множина $S$ є сукупністю впорядкованих даних у базі знань; методологію впорядкування тут не розглядаємо; даними є текстова, графічна, таблична та інша інформація, структурована та атрибутована; дані можуть бути виконані різними мовами, тож множина формується із відповідних множин іншими мовами: $S = S^{\langle \text{ua} \rangle} \cup S^{\langle \text{en} \rangle} \cup S^{\langle \text{ru} \rangle} \cup \ldots$, де верхній індекс показує атрибут мови інформації у базі знань (тут: українською, англійською, російською), для спрощення подальших викладень розглядатимемо тільки "мовну" множину українською $S^{\langle \text{ua} \rangle}$;

2) множина $S^{\langle \text{ua} \rangle}$ формується з тексту кожного цілісного об'єкта (ним є всякий електронний документ в одному з текстових форматів), що вноситься в базу знань:

$$S^{\langle \text{ua} \rangle} = S_j^{\langle \text{ua} \rangle} \cup S_k^{\langle \text{ua} \rangle}, \quad (1)$$

інакше кажучи, після об'єднання у нашому випадку двох множин вони належатимуть до множини $S^{\langle \text{ua} \rangle}$:

$$S_j^{\langle \text{ua} \rangle} \subseteq S^{\langle \text{ua} \rangle} \text{ та } S_k^{\langle \text{ua} \rangle} \subseteq S^{\langle \text{ua} \rangle},$$

де індекс $j$ показує початкову текстову множину; індекс $k$ – іншу текстову множина, що є відмінною від $S_j^{\langle \text{ua} \rangle}$, вочевидь, у такому разі $k = (1 \ldots K_{\text{ua}}]$, де $K_{\text{ua}}$ – загальна кількість множин);

3) оскільки об'єкти є самостійними документами, вони можуть не тільки "збагачувати" об'єднану множину, а і привносити дублювання текстових даних, тобто дві множини від різних документів можуть перетинатися $S^{\langle \text{ua} \rangle} \cap S_k^{\langle \text{ua} \rangle}$. Наслідком такого дублювання буде надлишковість та спотворення, але в цій статті це питання не розглядатимемо;

4) задля того, щоб множини у базі знань були подані в "єдиній системі мір" (у єдиному форматі), над об'єктами будуть виконуватися певні операції перетворення, на прикладі початкової множини це виглядає так:

$$\aleph: S_j \longrightarrow S_j'; \quad (2)$$

5) одним із видів перетворення є перетворення за мовною ознакою, тобто переклад тексту об'єкта українською, тоді вираз (1) трансформується до такого вигляду:

$$S^{\langle \text{ua} \rangle} = S_j^{\langle \text{ua} \rangle} \cup S_k^{\langle \text{en} \rangle\prime}; \quad (3)$$

вочевидь, що якщо початкова множина формується з об'єкта іншою мовою, тобто $S_j^{\langle \text{ua} \rangle} = \emptyset$, то (3) записується так:

$$S^{\langle \text{ua} \rangle} = S_j^{\langle \text{en} \rangle\prime}; \quad (4)$$

6) подання перетвореної множини $S_j'$ може здійснюватися через її зведення до одиничної матриці $1 \times N$, тобто векторного ряду даних, який формується із послідовності символів, об'єднаних у смислові одиниці з урахуванням синтаксису; смисловою одиницею є слово, символьне сполучення (абревіатура), цифрове значення, фраза, речення, абзац, розділ тощо (де $N$ – кількість елементарних символів). Приклади смислових одиниць подано нижче;

7) завданням пошуку інформації у базі знань є формування запиту у вигляді вектора $q$ розміром $1 \times M_i$, де $M_i$ – кількість елементарних символів для $i$-го запиту, $M_i \ll N$), у результаті

опрацювання якого буде отримана відповідь з генерування (виокремлення) тексту у вигляді вектора $r$ розміром $1 \times L_i$, де $L_i$ – кількість елементарних символів для $i$-ї відповіді на $i$-й запит, $L_i \ll N$, $L_i \approx M_i$):

$$\xi: q \cap S^{\langle \text{ua} \rangle} \longrightarrow \begin{cases} r \sim q \\ r \in S^{\langle \text{ua} \rangle}; \end{cases} \quad (5)$$

вочевидь, оптимізацією операції пошуку (тут: найкращою за певними критеріями відповіддю) є досягнення еквівалентності векторів $r$ та $q$ (показано символом $\sim$).

**Постановка проблеми в частковому вигляді**

Як об'єкт бази знань в цій статті розглянемо електронний документ у форматі PDF. Контентом об'єкта є сукупність вербалізованих, цифрових і символьних даних, категоризованих і структурованих, смислові ряди подано українською мовою, тобто текстом. Нижче наведено конкретний приклад такого об'єкта.

В такому разі об'єкт належить до сфери діяльності людини та сфери взаємодії "людина – машини" у царині мови та тексту. Зазвичай завдання розпізнавання, перетворення, класифікації, машинного перекладу, генерації тексту та інші вирішує інформаційна система (ІС), реалізована на основі поєднання спеціалізованих модулів для здійснення процедур із текстом або мовою (див. рис. 2).

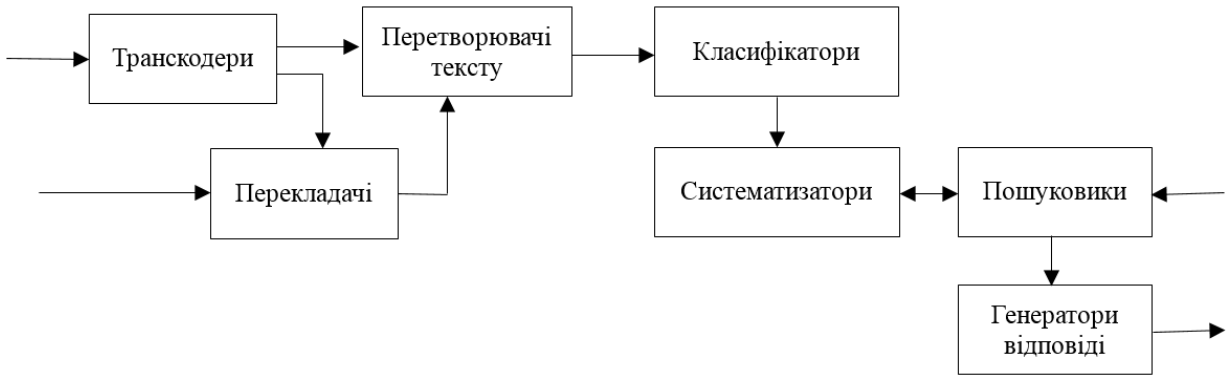



*Рис. 2. Типова функціональна модель інформаційної системи для роботи із мовою та текстом*

У реалізації такої моделі *транскодери* (ТК) забезпечують перетворення даних в отриманому форматі на робочий формат для бази знань, *перекладачі* – переклад тексту документів із вибраних іноземних мов українською мовою, *перетворювачі тексту* (ПТ) структурують дані, а також здійснюють їх очищення за визначеними алгоритмами, *класифікатори* (Кл) за визначеними ознаками зараховують текст (дані) до певних груп, встановлюють крос-текстові зв'язки, *систематизатори* впорядковують розташування тексту в базі знань (БЗ) з метою забезпечення доступу та відбору, *пошуковики запит – відповідь (ПЗВ)* призначені для підбору запитуваних даних (тексту), *генератори відповіді* надають дані у вигляді, який визначив їх споживач.

Сьогодні взаємодії "людина – машини" у царині мови та тексту є об'єктом давніх досліджень у галузі штучного інтелекту (ШІ).

Опрацювання природної мови (ОПМ) (***Natural Language Processing, NLP***) належить до однієї зі специфікацій галузі ШІ та, зокрема, глибинного навчання [6], яке вивчає та розробляє методи аналізу, розуміння та генерації людської мови комп'ютерами. Типову функціональну схему подано на рис. 3.

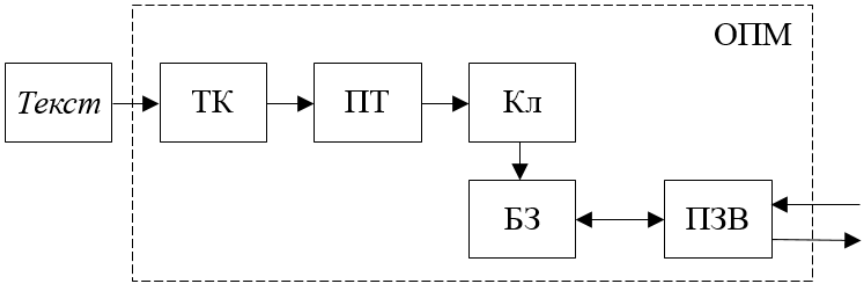



*Рис. 3. Типова функціональна модель ІС для обробки мови та тексту*

Серед методів глибинного навчання, призначених для задач ОПМ, як-от рекурентні нейронні мережі ***RNN*** [7], згорткові нейронні мережі ***CNN*** [8], виокремлено трансформери, які виявились ефективними у таких моделях, як ***GPT*** (***Generative Pre-trained Transformer***) [9] і ***BERT*** (***Bidirectional Encoder Representations from Transformers***) [10]. У задачах перекладу трансформери порівняно з архітектурою, виконаною на ***RNN*** і ***CNN***, навчаються набагато швидше і дають квазіідентичні результати, по суті, даючи змогу досягти нового мистецтва перекладу [11]. Потенційні можливості трансформерів поширюються і на завдання опрацювання великих даних, як-от аудіо, відео, зображення.

Усі продуктивні нейронні моделі для послідовного перекладу, а також для розпізнавання мови, класифікації, узагальнення, генерації тексту тощо, ґрунтуються на кодер-декодерній архітектурі [7, 11], для якої деяку вхідну послідовність символів $x_1, \dots, x_n$ кодер упорядковує (упросторовує – *термін наш*) у продовжувану послідовність $z = (z_1, \dots, z_n)$, а відтак декодер із $z$ генерує вихідну послідовність $y_1, \dots, y_m$ символів по одному за раз.

Поєднання кодер-декодерної архітектури з трансформерами відкрило нові можливості, що згадані, тож доцільно подати цю нейронну модель (див. рис. 4) деякої розмірності $d_{model}$.

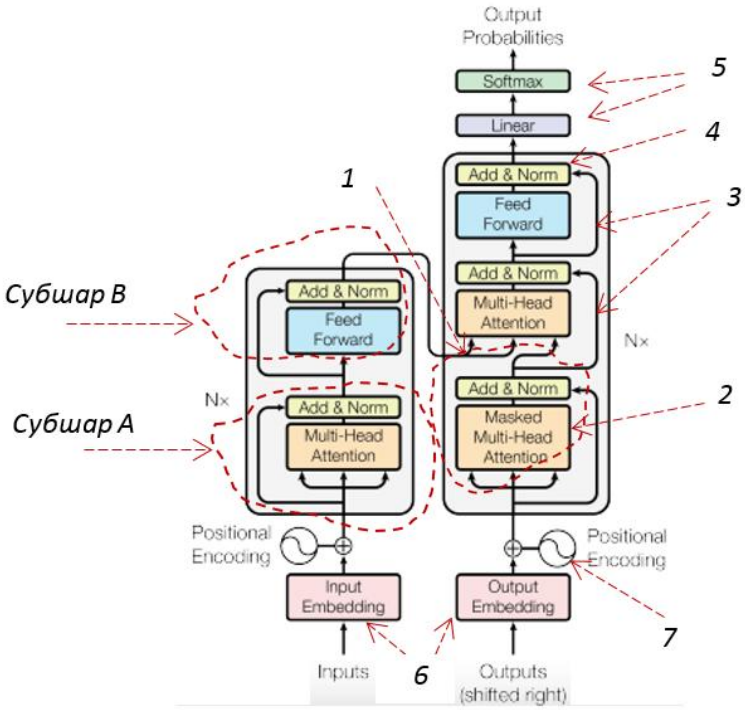



*Рис. 4. Архітектура трансформера:*
*зліва – кодер, справа – декодер ([11], розмітка – авт.)*

І кодер, і декодер складаються із шести ідентичних шарів, кожен із яких має два субшари: перший, типу *A* – мультиголовий механізм самоуваги (***multi-head self-attention mechanism***), на рис. 4 – ***Multi-Head Attention***, другий, типу *B* – повнозв'язний шар прямого поширення, на рис. 4 – ***Feed Forward***.

У декодері вводиться ще один (третій) субшар, який виконує мультиголову увагу над виходом із кодерної частини *1*.

Щоб запобігти прояву стану з попередньої позиції у наступній, механізм самоуваги декодера дещо модифіковано (див. *2* на рис. 4). У цьому разі передбачення для деякої *i*-ї позиції залежить лише від виходів на попередніх позиціях.

Також на рис. 4 позначено (по кожному одному виду із наявних):

*3* – залишковий зв'язок (***residual connection***) [12] для забезпечення авторегресивності моделі [13], для якої попередньо згенерований символ стає доповнювальним входом під час генерації наступного символу: $LayerNorm\big(x + Sublayer(x)\big)$, де $Sublayer(x)$ – функція, яку виконує субшар (розмірність вихідних даних субшарів відповідає $d_{model}$);

*4* – шарову нормалізацію, запропоновану в [14] задля зменшення часу навчання;

*5* – навчена лінійна трансформація і нормована експоненційна функція з метою конвертування даних на виході декодера в передбачувані марковані ознаки (на схемі – ***output probabilities***);

*6* – навчені вкладення (на рис. 4 – ***embedding***) для конвертування вхідних і вихідних маркерів (***token***) у вектори розмірності $d_{model}$;

*7* – вектор позиційного кодування тієї самої розмірності, що і вкладення, щоб вони додавалися між собою, тим самим привносяться ознаки місця (абсолютного чи відносного) маркерів у послідовності (чого немає в ***RNN*** і ***CNN***) [15].

Отже, ключовим у трансформерах є механізм самоуваги, завдяки якому модель зосереджується на різних (важливих так чи інакше) частинах вхідної послідовності під час генерації вихідної послідовності. Відбувається це за допомогою обчислення зваженого сумарного внеску кожного слова в контекст усієї послідовності.

На вхід модель приймає векторне представлення кожного із елементів, що дає змогу технічно розділяти кожний окремий елемент входу та його контекст.

Повнозв'язні шари із функціями активації (***ReLU*** або іншими) здійснюють опрацювання репрезентацій на рівні слова, допомагаючи моделі краще розуміти складні взаємозв'язки між словами в послідовності. У підсумку модель здатна видавати довгострокові залежності.

Подамо типову функціональну модель ІС (рис. 3) із застосуванням трансформерів.

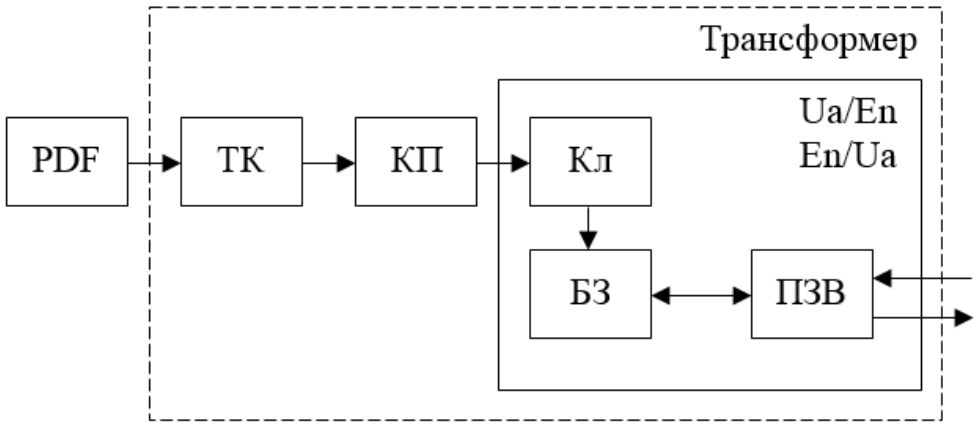



*Рис. 5. Функціональна модель ІС на трансформері для опрацювання тексту*

Оскільки особливістю трансформера є потреба у великих масивах даних для тренування, а також використання складних оптимізаторів зворотного поширення помилки типу *Adam* або *SGD* для оновлення ваг моделі, розроблення моделі трансформера є істотно ресурсо- та часозатратним.

Тож вважаємо за доцільне протестувати модель для опрацювання цільового документа, що дасть змогу налаштувати у разі доцільності розроблену модель на конкретні завдання за допомогою невеликих наборів даних.

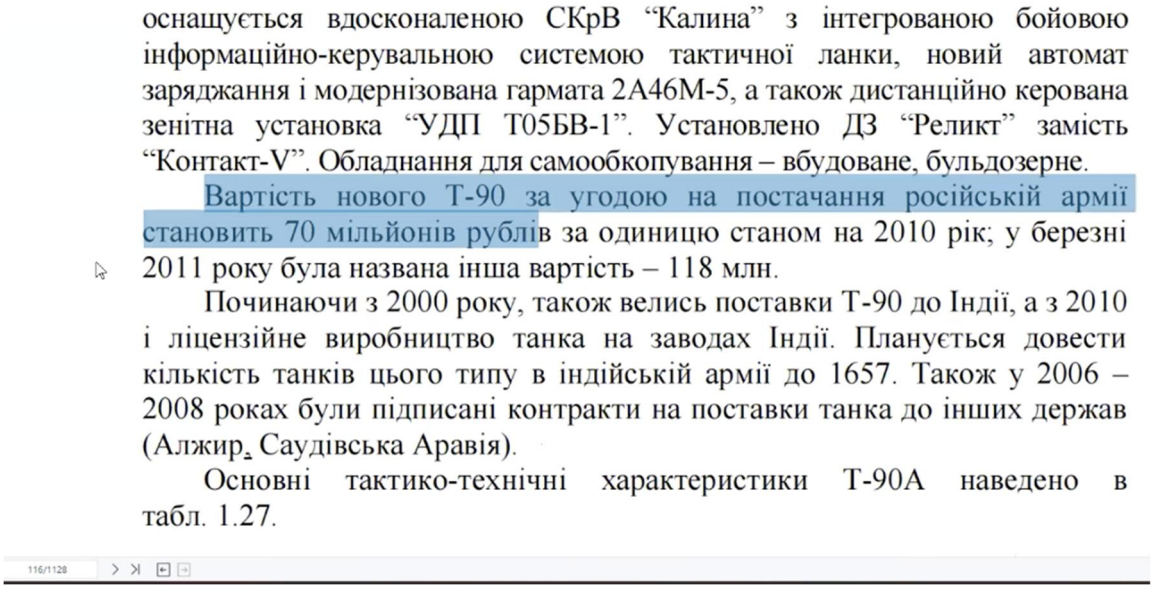
Досліджуваний текст – фрагмент із цільового електронного документа із військово-спеціалізованої бази знань, а саме з довідкового видання “Озброєння і військова техніка Російської Федерації…” [16]. Формат видання – PDF (з можливістю переходу за внутрішніми гіперпосиланнями лише від змісту).

Як зазначено в анотації до другого видання, “перше видання Довідника учасника АТО викликало зацікавленість серед фахівців у сфері національної безпеки і оборони, посадових осіб органів військового управління, з’єднань та частин Збройних сил України, інших військових формувань, що беруть участь у АТО, науково-педагогічних та наукових працівників.

Недостатня кількість надрукованих примірників попереднього видання та прийняття останнім часом міністерством оборони росії на озброєння сучасних зразків озброєння і військової техніки, участь формувань збройних сил російської федерації у конфлікті на сході України обумовлюють необхідність внесення доповнень та видання доопрацьованого примірника”.

Довідник містить інформацію про понад 600 зразків ОВТ збройних сил російської федерації за усіма видами та родами військ. Обсяг видання – понад 1100 сторінок.

Приклад сторінки (в нижньому лівому кутку номер сторінки документа (тут – 116)) подано на рис. 6.



оснащується вдосконаленою СКрВ “Калина” з інтегрованою бойовою інформаційно-керувальною системою тактичної ланки, новий автомат заряджання і модернізована гармата 2А46М-5, а також дистанційно керована зенітна установка “УДП Т05БВ-1”. Установлено ДЗ “Реликт” замість “Контакт-V”. Обладнання для самообкопування – вбудоване, бульдозерне.

Вартість нового Т-90 за угодою на постачання російській армії становить 70 мільйонів рублів за одиницю станом на 2010 рік; у березні 2011 року була названа інша вартість – 118 млн.

Починаючи з 2000 року, також велись поставки Т-90 до Індії, а з 2010 і ліцензійне виробництво танка на заводах Індії. Планується довести кількість танків цього типу в індійській армії до 1657. Також у 2006 – 2008 роках були підписані контракти на поставки танка до інших держав (Алжир, Саудівська Аравія).

Основні тактико-технічні характеристики Т-90А наведено в табл. 1.27.

116/1128

*Рис. 6. Фото сторінки цільового документа (взято з [16], фото авт.)*

Вочевидь, що у 2023 р. наступне видання цієї тематики (чи будь-якої подібної) ще зросте в обсязі й за номенклатурним рядом, адже у виданні 2017 р. немає інформації про ракетні комплекси та крилаті ракети, про БПЛА типу “Герань”/Shahed, FPV-дрони тощо.

У довіднику інформація про зразок порівняно структурована і містить призначення, коротку історичну довідку щодо розробки, оцінювану кількість у загальному складі збройних сил російської федерації (або у підрозділі чи з’єднанні відповідно бойового та кількісного складу ОВТ), наявні модифікації та значення на світовому ринку озброєнь, конкретні кількісні та якісні параметри, що стосуються зразка ОВТ (тактико-технічні характеристики, бойові можливості).

**Основні результати досліджень**

Отже, для перевірки підходів до створення пошуково-інформаційної військово-спеціалізованої бази знань для використання в СППР спершу ми вирішили розв'язати задачу на спрощеній моделі, вибравши для цього вже натреновану модель трансформера, доступну на інтернет-платформі або інших ресурсах. Можливі моделі трансформерів зведено в табл. 1.

На цьому етапі перевірки гіпотези вибір конкретної платформи неістотний, тож критеріїв простоти, відкритості, різноманіття цілком достатньо. Цим критеріям відповідає, зокрема, платформа ***Hugging Face***, що пропонує чимало моделей різного призначення, наприклад, модель ***google/flan-t5-large*** з високим рейтингом, від надійної компанії, яка працює з великими мовними моделями (***Large Language Model***). Як фреймворк для роботи з моделлю вибрали ***Llama Index***.

Суттєвим для нас обмеженням моделі google/flan-t5-large є те, що вона не розуміє української мови.

Це обмеження ми зняли через імпорт додаткових моделей перекладу тексту: ***Helsinki-NLP/opus-mt-uk-en*** та ***Helsinki-NLP/opus-mt-en-uk***, які демонструють велику точність перекладу та мають багато завантажень, що є відносним показником надійності. За допомогою моделі досліджуваний текст, а також питання користувача перекладатимуться з української мови на англійську, а другої – відповідь моделі з англійської мови на українську мову.

На вхід програми подається цільовий PDF файл. Оскільки сама модель приймає на вхід лише текстовий рядок (***string***), виникла необхідність зчитування текстової інформації з PDF файлу і перетворення його на звичайний текст. Його можна отримати за допомогою ***Python*** бібліотеки ***PyPDF2***, що і було зроблено.

*Таблиця 1*

**Характеристика відкритих натренованих моделей для обробки природної мови**

| № | Платформи з моделями | Спеціалізація та опис моделей, приклади | Фактори переваг і обмежень |
|---|---|---|---|
| 1 | *AllenNLP* | моделі з бібліотеки та ресурсу для виокремлення іменованих сутностей, семантичного розуміння та машинного перекладу | простота побудови моделі аналізу мови, легко масштабується |
| 2 | *GitHub* | репозиторій містить вихідний код проєктів для самостійного використання інструментів відкритого програмного забезпечення *Python*, наприклад, *PyTorch* чи *GenSim* для моделювання тем, порівняння подібності документів, латентно-семантичного аналізу тощо | висока швидкість опрацювання та здатність обробляти велику кількість тексту; наявність попередньо підготовлених ваг моделей, сценаріїв використання й утилітів перетворення для моделей |
| 3 | *Google Cloud AI Platform* | для обробки природної мови: *BERT* – для семантичного аналізу; *Cloud Translation* – для машинного перекладу | широка різноманітність областей і задач |
| 4 | *Hugging Face* | для класифікації тексту, машинного перекладу, семантичного аналізу тощо | простота, публічний доступ |
| 5 | *Intel AI* | бібліотека *Architector NLP* із відкритим кодом для оптимізації *NLP*, вивчення глибоких топологій навчання, спрощення робочих процесів | гнучкість щодо додавання нових моделей, компонентів НМ, методів обробки даних, легке навчання та запуск моделей |
| 6 | *OpenAI* | для генерації тексту: *GPT-3* | спеціальна ліцензія або підписка для доступу |
| 7 | *TensorFlow Hub* | для класифікації, векторного представлення слів, машинного перекладу | покращена візуалізація |

Кожен текст містить різні поєднання смислових одиниць, що потребує їх класифікації, розпізнавання та відповідного опрацювання. В цільових текстах є чимало неоднозначних смислових одиниць (див., напр., рис. 6), що можна побачити у відомостях, зведених до табл. 2.

*Таблиця 2*

**Характеристика смислових одиниць цільового тексту**

| № | Смислова одиниця | Приклади одиниць | Особливості обробки |
|---|---|---|---|
| 1 | Символьна група (абревіатура) | ОШС | потребує розкриття значення символів |
| 2 | Символьно-цифрова група | БТР-80,<br>135 мм,<br>40 од. | стандартизовані смислові одиниці, що уможливлює і робить потрібним їх внесення до бібліотек |
| 3 | Цифрова група | 70 | без контексту, наприклад, з табличних даних, є не зрозумілим |
| 4 | Змішана символьна група | FPV-дрони,<br>БПЛА ‘Shahed’,<br>ТОС-2А | переклад може спотворювати сутність через фонемічні, а подекуди і лексичні збіги |
| 5 | Лексема | Василек,<br>Зоопарк | потрібне ототожнення кодових назв, інколи і жаргонних |
| 6 | Смислова фраза | Вогневе завдання | може містити у собі набагато більше шарів відомостей і неоднозначностей |
| 7 | Таблиця | – | у різних бібліотеках може по-різному подаватися, групуватися тощо |
| 8 | Розділ | – | потребує встановлення меж – початкової та кінцевої – для закінченого висловлювання |

Як наслідок, у задачі конвертування тексту ці смислові одиниці необхідно зберігати та розпізнавати, але на цьому етапі дослідженні ми цього поки що не робили.

Зрозуміло, що після конвертування ми отримали дуже великий обсяг тексту, що перевантажує модель через зчитування. За допомогою функції токенізації тексту, імпортованої з *GitHub*, величезний нерозмічений, невпорядкований масив інформації був розділений на менші, помітно легші для опрацювання абзаци, речення та слова. Отриманий результат є готовим для роботи з моделлю текстом.

Наступним кроком стало застосування системи типу “питання – відповідь” (*QR*), які призначені для пошуку інформації безпосередньо у джерелі (документі, наборі даних тощо) з видаванням коротких відповідей. Для вирішення таких завдань існує оптимізована архітектура глибинного навчання, відома як мережа динамічної пам’яті (*Dynamic Memory Network, DNM*). *DNM* навчається на тренувальному наборі вхідних даних та питань і формує епізодичні “спогади” про них, які потім використовуються для генерації доречних відповідей.

## Описання програмної реалізації моделі пошуку відповіді у цільовому PDF-документі

Окреслені вище підходи відображено у програмному коді, який подамо нижче у вигляді етапів і кроків розв’язання задачі.

Етап 1. Підготовка моделі.

Крок 1. Встановлення та імпорт необхідних бібліотек Python (вибрану модель імпортували з бібліотеки transformers за допомогою функції pipeline):

```
! pip install -q langchain transformers sentence_transformers llama-index
from llama_index import SimpleDirectoryReader, LangchainEmbedding, GPTListIndex, PromptHelper, GPTVectorStoreIndex
from langchain.embeddings.huggingface import HuggingFaceEmbeddings
from llama_index import LLMPredictor, ServiceContext
import torch
from langchain.llms.base import LLM
from transformers import pipeline
from llama_index import Document
from tokenize_uk import tokenize_sents as tkn
```

Крок 2. Імпорт бібліотеки PDF для перетворення на string:

```
! pip install PyPDF2
from PyPDF2 import PdfReader
```

Крок 3. Налаштування моделі (змінних параметрів) для виконання функцій виклику та опрацювання тексту за допомогою фреймворку Llama Index:

```
class customLLM(LLM):
model_name = "google/flan-t5-large"
pipeline = pipeline("text2text-generation", model=model_name, device=0, model_kwargs={"torch_dtype":torch.bfloat16})
def _call(self, prompt, stop=None):
return self.pipeline(prompt, max_length=9999)[0]["generated_text"]
@property
def _identifying_params(self):
return {"name_of_model": self.model_name}
@property
def _llm_type(self):
return "custom"
llm_predictor = LLMPredictor(llm=customLLM())
```

У коді параметр device=0 відповідає за запуск моделі із використанням GPU (за недостатніх обчислювальних можливостей варто або вилучити його, або використовувати хмарний сервіс).

Крок 4. Імпортування моделей перекладу тексту з української англійською та з англійської українською, що дасть змогу забезпечити, щоб модель розуміла зміст питання користувача та надавала відповіді на нього мовою питання:

```
translator_ukr_to_en = pipeline(model = "Helsinki-NLP/opus-mt-uk-en", device=0)
translator_en_to_ukr = pipeline(model = "Helsinki-NLP/opus-mt-en-uk", device=0)
```

Крок 5. Підготовка інструментарію до перетворення тексту на зрозумілий для машини вектрний рядок, що можна виконувати також за допомогою ембедингів (як видно з коду, використані відкриті та прийнятні за точністю ембединги від Hugging Face):

```
hfemb = HuggingFaceEmbeddings()
embed_model = LangchainEmbedding(hfemb)
```

Етап 2. Опрацювання тексту моделлю.

Крок 1. Завантаження у модель цільового файлу та зчитування з нього тексту з поданням у форматі string:

```
doc = PdfReader('Task.pdf')
print(len(doc.pages))
```

Крок 2. Переклад мовою моделі тексту, попередньо розділеного на малі частини (токенізація):

```
text = ""
text_tr = ""
for i in range(100,200):
page = doc.pages[i]
text_from_page = page.extract_text()
text_tokenized = tkn(text_from_page)
for temp in text_tokenized:
a = translator_ukr_to_en(temp)
text_tr = "".join(str(k) for k in a[0]['translation_text'])
text += text_tr
print(text)
```

Як бачимо з коду, документ обмежений його десятою частиною, тобто сторінками цільового файлу від 100-ї до 200-ї, що дасть змогу перевірити працездатність моделі. Навіть для цієї часткової задачі час токенізації та з'єднання тексту становив майже 450 с.

Крок 3. Перетворення форми представлення тексту (у вигляді списку) для зчитування моделлю:

```
text_list = [text]
documents = [Document(t) for t in text_list]
```

Крок 4. Передавання тексту в модель:

```
service_context = ServiceContext.from_defaults(llm_predictor=llm_predictor, embed_model=embed_model)
index = GPTVectorStoreIndex.from_documents(documents, service_context=service_context)
```

Етап 3. Застосування моделі.

Крок 1. Підготовка до взаємодії:

```
import logging
logging.getLogger().setLevel(logging.CRITICAL)
```

Крок 2. Створення запиту до моделі (як приклад вибрано питання щодо ціни бойової машини Т-90 згідно із даними відкритих контрактів на торгівлю зброєю – див. рис. 5):

```
question_ukr = "Яка вартість нового Т-90?"
```

Крок 3. Опрацювання запиту моделі (переклад мовою моделі, аналіз моделлю тексту для віднаходження відповіді):

```
b = translator_ukr_to_en(question_ukr)
question_eng = b[0]['translation_text']
print(question_eng)
query_engine = index.as_query_engine()
response = query_engine.query(str(question_eng))
```

Крок 4. Видавання відповіді на запит мовою користувача:

```
c = translator_en_to_ukr(response.response)
c[0]['translation_text']
```

Час відпрацювання запиту – 10 с. Отриману відповідь подано на рис. 7.

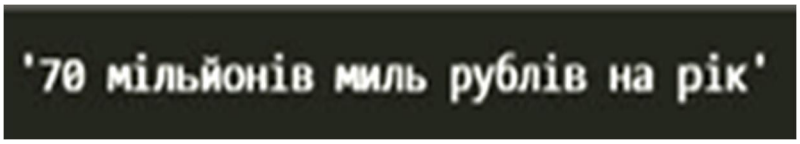



*Рис. 7. Фото інтерфейсу реалізованої ІС з відповіддю на пробний запит (фото авт.)*

Як видно з рис. 7, відповідь виявилася не повністю відповідною матеріалам, що містяться у цільовому документі, а саме з’явилася надлишкова лексема. Ймовірною причиною є або недосконаліть моделей та файлу, або потреба додаткових налаштувань.

Цього питання стосуватимуться подальші дослідження.

А загалом зазначимо: у нашому прикладі інформація, наявна у відповіді цієї інформаційно-довідкової системи, правильна.

**Висновки**

У загальному випадку онтологія щодо добування інформації із бази знань передбачає такі операції: збирання інформації, упорядкування даних у базі знань, виділення із бази знань текстової інформації, перетворення текстової інформації для векторного представлення, міжмовне перетворення текстової інформації (переклад), токенізація, генерування відповіді на сформований запит, оптимізація операції пошуку (машинне навчання).

Запропонована модель здатна самостійно зчитувати текст з PDF-файлу, розділяти його на токени, за допомогою навченого трансформера перекладати текст англійською, аналізувати, після чого видавати відповідь у перекладі українською мовою.

Модель на основі навченого трансформера характеризується такими властивостями: простотою, що зумовлено однозначністю інформаційно-пошукового запиту; стабільністю у разі використання мови-посередника – англійської мови; точністю, тобто здатна знайти правильну відповідь на запитання у великому масиві інформації.

До обмежень моделі належить високі часові вимоги на її запуск та на її компіляцію, а також наявність лексичних або інших помилок.

Напрямами подальших досліджень стануть:

1. Накопичення статистики щодо оцінювання точності моделі.

2. Розроблення складових моделі для правильної токенізації складних смислових одиниць (спеціальних цифрових, символьних і символьно-цифрових груп, а також спеціальних і жаргонних термінів).

3. Оцінювання обчислювальних і часових вимог до моделі.

4. Створення повноцінної моделі для інформаційно-пошукової системи для підтримки прийняття рішень у військовій справі.

5. Розроблення рекомендацій для ОВУ – посадовців щодо застосування інформаційних систем із упровадженими навченими моделями.

**Список літератури**

1. Вовнянка, Р., Досин, Д., Ковалевич, В. (2014). Метод видобування знань з текстових документів. *Вісник Національного університету "Львівська політехніка". Серія: "Інформаційні системи та мережі"*, № 783, 303–312.

2. Литвин, В. (2011). Бази знань інтелектуальних систем підтримки прийняття рішень. Львів: Вид-во Нац. ун-ту "Львівська політехніка". 240 с.

3. Вавіленкова, А. (2013). Аналіз методів обробки текстової інформації. *Вісник НТУ "ХПІ"*, № 39 (1012).

4. Литвин, В. (2013). Метод видобування знань з природомовних текстів для автоматизованої розбудови онтологій. *Автоматизовані системи управління та прилади автоматики*, № 164, 67–72.

5. Палагін, О., Петренко М. (2017). Розбудова абстрактної моделі мовно-онтологічної інформаційної системи. *Математичні машини і системи*, № 1, 42–50.

6. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press. URL:https://www.deeplearningbook.org/.

7. Schmidt, Robin M. (2019). Recurrent Neural Networks (RNNs): A gentle Introduction and Overview. *Computer Science. Machine Learning*. URL: https://arxiv.org/abs/1912.05911v1.

8. Rahman, M., Islam, M., Sassi, R. et al. (2019). Convolutional neural networks performance comparison for handwritten Bengali numerals recognition. *SN Appl. Sci.* 1, 1660. URL: https://doi.org/10.1007/s42452-019-1682-y.

9. Brown, T. B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A. & others (2020), 'Language models are few-shot learners'. URL: arXiv preprint arXiv:2005.14165.

10. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Vol. 1, 4171–4186. URL: https://aclanthology.org/N19-1423.pdf.

11. Gomez, A. N., Jones, L., Kaiser, Ł., Parmar, N., Polosukhin, I., Shazeer, N., Uszkoreit, J., Vaswani, A. (2017). Attention is All You Need. In *31st Conf. on Neural Information Processing Systems*. URL: arXiv:1706.03762v5.

12. He, K.; Zhang, X.; Ren, S.; Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.

13. Graves, A. (2013). Generating sequences with recurrent neural networks. URL: arXiv:1308.0850.

14. Ba, J.; Kiros, J. and Hinton, G. (2016). Layer normalization. URL: arXiv:1607.06450.

15. Gehring, J.; Auli, M.; Grangier, D.; Yarats, D. and Dauphin, Y. (2017). Convolutional sequence to sequence learning. URL: arXiv:1705.03122v2.

16. Алімпієв, А., Пєвцов, Г., Гриб Д. та ін. (2019). Озброєння і військова техніка Російської Федерації: довідник учасника АТО. За заг. ред. А. Алімпієва. Харків, 1112.

# THE METHOD AND THE MODEL FOR PROCESSING TEXTUAL INFORMATION ON A LEARNED TRANSFORMER FOR INFORMATION-RETRIEVAL SYSTEM

**Vasyl Lytvyn[1], Volodymyr Tymchuk[2]**

[1] Lviv Polytechnic National University, Information Systems and Networks Department,
12, S. Bandery str., Lviv, Ukraine
[1–2] Hetman Petro Sahaidachnyi National Army Academy, Lviv, Ukraine
[1] vasyl.v.lytvyn@lpnu.ua, ORCID0000-0002-9676-0180,
[2] v_tymchuk@yahoo.co.uk, ORCID0000-0002-3549-2813



**To form a knowledge base is complicated problem traditionally. There are a lot kind of objects that are possibly used for forming a knowledge base. These objects may have different structures, formats, ways of data representation, languages. The simple conjunction is not effective and suitable. In general case the knowledge base has got as an unordered knowledge base. There are uncategorized documents in such unordered knowledge base with different formats that causes the special and particular approaches for recognition, systematization and next processing of some textual information. It's why the complexes of automation for all stages of processing are complicated. Naturally it is a restriction for some kind of the decision support system, especially in military or other applications with key time factor (to get a quick and exact access to the knowledge base in decision support system). So, we analyzed the mentioned restrictions and conditions for forming a knowledge base in the paper. We depicted that the ontology of knowledge base both in general and specific cases includes such operations as data collection, data regularization, extraction of knowledge, data conversion for matrix representation, data language processing, tokenization, output generation for a request and machine learning for information-retrieval system optimization. There is a model of information-retrieval system for knowledge base with widely-used PDF-documents that is proposed in the paper. We made the model using open learned transformer and *Llama Index* framework to decrease the time demands in the information-retrieval system. Also, we included the language processing models for translation the specific textual information from Ukrainian into English and back. As a result, we got the method and the model for processing the textual information from PDF-document in Ukrainian that could be effective in any decision support system. The method ensures the reading, tokenization, translation, analysis and retrieve generation of the data in Ukrainian. The model showed its simple, stable and exact estimations, but there are also some disadvantages, high time installation/compilation and little language defaults are some of them. The results encourage us to continue the research and to get the statistics set to analyze the model estimation more properly.**

**Key words: deep learning machine in data-processing system; information-retrieval system; decision support system; method for processing textual information; ontology of knowledge base; extraction of knowledge.**