УДК 681.3.06

АЛГОРИТМ СПРОЩЕННЯ СИНТАКСИЧНИХ СТРУКТУР ТЕКСТУ ПРИРОДНОЮ МОВОЮ ДО СТАНДАРТИЗОВАНИХ РЕЧЕНЬ

М.М. Глибовець, О.Ю. Остапенко

Національний університет "Києво–Могилянська академія",
04070, Київ, вул.Сковороди, 2.
Тел.: (044) 463 6985, glib@ukma.kiev.ua

Розглядається проблема автоматизації обробки текстів природною мовою та надається опис запропонованого алгоритму, який може бути використано для автоматичної побудови тестових питань до тексту. Алгоритм передбачає глибинно-синтаксичний аналіз тексту, результатом використання якого є репрезентація фраз у вигляді розміченого дерева, графа концептуальних залежностей. Даний підхід передбачає спрощення синтаксичних структур до елементарних речень та використання певних правил для розбиття речень на елементарні структури.

Article considers the problems of text processing automation in human language along with the description of the algorithm that might be used for automated construction of test questions on the text. Given algorithm assumes the deep-syntactic text analysis, usage of which results in the phrases that are represented in the marked tree view, a conceptual dependency graph. Present approach assumes the simplification of the syntax structures to elementary sentences and usage of the certain rules for splitting sentences into elementary structures.

Вступ

Сучасний розвиток людства призводить до швидкого зростання кількості інформації з усіх галузей науки та бізнесу. Особливістю накопичення інформації є її хаотичність та не стандартизованість. Тому лінгвістична обробка мовних текстів стала однією з центральних проблем інтелектуалізації інформаційних технологій. Ще з середини 50-х років минулого століття значні зусилля науковців були спрямовані на розробку математичних алгоритмів та комп'ютерних програм обробки текстів природною мовою [1]. Для автоматизації аналізу та синтезу текстів створювалися різноманітні моделі процесів обробки тексту, а також відповідні алгоритми та структури представлення даних. Більшість досліджень та розробок були присвячені розумінню англійської мови, водночас як головною проблемою для нашого географічного регіону є обробка текстів українською мовою. Тому дана робота спрямована на дослідження лінгвістичної обробки саме української мови.

На даний момент все більше навчальних закладів починають використовувати електронну освіту. Саме через це багато наукових досліджень спрямовано на створення ефективних програмних систем підтримки розподіленого навчального середовища [2], зосереджених на повну або часткову автоматизацію як підсистеми управління навчального закладу, так і підсистеми керування навчальним контекстом. Особлива увага приділяється зменшенню участі та зусиль викладача на організацію навчання, створення і збереження навчальних матеріалів та перевірку знань студента.

Перевірка знань студента (тестування) є одним з найважливіших етапів навчального процесу. За останній час було зроблено багато досліджень та розробок програмних систем підтримки цього процесу, наприклад, такі, що спрямовано на побудову стандартизованого репозитарію [3], на розширення доступного класу тестових питань [4, 5] або на створення нової архітектури підтримки тестування (розподілені системи, семантичні веб сервіси) [6]. У даній роботі теж розглядається підсистема перевірки знань студента в контексті створення алгоритму автоматизації побудови тестових питань адаптивного типу.

Автоматична обробка мови

Під автоматичною обробкою мови розуміють теоретичну та прикладну лінгвістику, зв'язану з комп'ютерним опрацюванням текстів. Розрізняють наступні типи досліджень [7]:

- дослідження та моделювання механізмів аналізу та синтезу речень;
- створення автоматичних словників, лінгвістичних баз даних;
- створення лінгвістичних процесорів для опрацювання тексту;
- розроблення систем перевірки орфографії;
- побудова алгоритмів для автоматичного реферування, машинного перекладу, інформаційного пошуку та розуміння природної мови.

Центральним компонентом автоматичної обробки мови є автоматичний аналіз різноманітних мовних структур. Специфіка алгоритму в кожному конкретному випадку визначається типом лінгвістичного завдання, конкретною метою та типом оброблювальної мовної конструкції. У даній роботі розглядається підхід до

автоматизації побудови тестових питань на основі тексту природною мовою, а така задача належить до того класу, що й, наприклад автоматичне реферування тексту. Вона відноситься до типу, де переважає глибинно-семантичний аналіз фраз, тобто алгоритм передбачає оперування з семантичними одиницями [8]. Процедури, що забезпечують роботу алгоритму, – це присвоєння лексемам[1] їх морфологічних, синтаксичних, семантичних характеристик та встановлення типів семантичних зв'язків у синтаксично зв'язаних словосполученнях. Кінцевим продуктом аналізу такого типу буде семантична або глибинно-синтаксична репрезентація фраз у вигляді розміченого дерева, графа концептуальних залежностей, семантичної формули або іншого типу семантичної репрезентації.

Загальний підхід до вивчення зв'язного тексту

У всіх існуючих граматиках межею опису є синтаксис складних речень. Однак речення не існує саме по собі, воно є частиною тексту та є засобом вираження зв'язних думок. На даний час дуже багато досліджень присвячено проблемі зв'язності мовлення незважаючи на складність цієї проблеми. Мета дослідження у даній області може бути сформульовано у самому загальному випадку як опис закономірностей, якими зв'язний текст відрізняється від незв'язного набору речень, тобто одним з головних етапів роботи над цією проблемою є запропонування можливого методу аналізу мовного матеріалу.

Значні складності при вивченні тексту виникають через складність і різноманітність синтаксичних структур. Тому деякі дослідники пропонують спочатку привести текст до набору деяких спрощених стандартизованих речень, тоді засоби зчеплення фрагментів думки стануть більш очевидними.

Вивчення структури зв'язного мовлення (тексту) виходить за межі звичайного синтаксису, так як досліджуються не окремі речення, а групи речень, що поєднані єдністю змісту і визначеними структурними закономірностями. Такі закономірності відсутні у випадковому наборі нічим не зв'язаних речень.

Основною одиницею дослідження тексту може бути абзац, який звичайно складається з групи речень, які передають досить самостійний відрізок думки і характеризуються визначеними структурними закономірностями. На основі статистичних даних досліджень можна зробити висновки, що висловлювання (або складне синтаксичне ціле, або компоненти тексту) досить часто збігаються з абзацами. Саме через ці властивості абзацу його було обрано як ключову одиницю автоматичної побудови питань в даній роботі. Тобто при подальшій роботі текст оброблюється поступово, за абзацами, і саме за ними потім будуються тестові запитання, бо передбачається, що абзац передає відносно скінчений фрагмент думки.

Зрозуміло, що структуру абзацу треба вивчати як деяку семантико-синтаксичну єдність , в якому окремі елементи (речення) тісно зв'язані між собою. Через це ми стикаємося з наступною проблематикою: синтаксична різнорідність складу абзацу, ступінь самостійності його частин, спосіб об'єднання окремих частин в межі єдиного цілого. Але всі ці питання є частиною одного – передача змісту в зв'язному тексті. Тому в даній роботі особливу увагу приділено методу дослідження текстів. Але таке дослідження ускладнено тим, що речення не завжди, можна навіть сказати – досить рідко, є одиницею, тотожною до поняття «скінчена думка». Якщо з синтаксичної точки зору зрозуміло, де закінчується одне речення і починається інше, то з точки зору змісту або логіки зовсім не обов'язково, що кінець думки збігався з кінцем речення. Так само не має відповідності між складним і простим реченням у залежності від ступеня складності думок, що вони виражають. Мається на увазі, що просте речення може висловлювати складну думку. Це є досить велика проблема для нашого дослідження, бо перша вимога, висунена до тестів – це їх логічність, тобто тестове питання має відображати якесь поняття, розкрите у тексті лекцій, і відображати коректно.

Звідси випливає, що бажаний результат не буде досягнуто якщо ми будемо слідкувати за передачею змісту від речення до речення. Зрозуміло, що спочатку треба привести текст до набору деяких спрощених стандартизованих речень. Передбачимо, що такі речення передають деяку «елементарну думку». Тобто ми будемо розглядати текст як послідовність спрощених синтаксичних структур і відповідно – взаємозв'язаних «елементарних думок». Якщо ми приведемо текст до такого вигляду, то тоді засоби передачі змісту буде значно легше відслідкувати.

Зведення фраз до набору спрощених стандартизованих конструкцій може розглядатися як процес, зворотній до створення складних речень – тобто його можна назвати процесом розпізнавання ступеня складності. Тому треба зробити дослідження синтаксичних закономірностей створення складного тексту шляхом аналізу самого тексту. При вивченні тексту важливою задачею є опис типів зв'язків між його частинами. Відношення між частинами зв'язного тексту (абзацу) виражаються різними мовними засобами, але всі вони так чи інакше відображають смислову залежність цих частин одна від одної. У роботі будемо розглядати лексико-семантичні форми зв'язку. Тут треба зауважити, що головна особливість зв'язного тексту, що відрізняє його від незв'язного набору фраз, – це повторення однакових або семантично близьких понять в абзаці.

---

[1] Лексема – це множина словоформ, що відрізняються одна від одної тільки словозмінними значеннями

Алгоритм

Процес автоматизації побудови текстових питань складається з декількох етапів. Першим з них є впровадження запропонованого алгоритму, який розбиває складні синтаксичні структури на спрощені стандартизовані речення.

В якості вихідних даних ми будемо розглядати граф фрази, який повинен відповідати наступним критеріям:

- стрілки не ведуть до сполучників. Ця особливість запису графа дає можливість виносити сполучники в зв'язки (наприклад, при розбитті складного речення на ланку простих або при розбитті за однорідністю);
- якщо у декількох однорідних членів зустрічається спільний залежний член, то стрілки до нього ведуться від кожного з них;
- в конструкціях з однорідністю дозволяється перетин стрілок;
- всі однорідні члени вважаються залежними від одного керуючого (це випливає з пункту 1);
- в разі узагальнення при однорідності стрілки йдуть від нього до всіх однорідних членів.

Крім того, в процесі підготовки треба визначити ще декілька правил:

- слово А керує словом Б, якщо стрілка від слова А веде безпосередньо до слова Б;
- слово А керує словом Б, якщо є неперервний ланцюжок стрілок від А до Б;
- введемо поняття «група залежності» слова х (ГЗ-х) – повний набір слів, яким керує слово х;
- слова А і Б будемо вважати зв'язаними, якщо має місце один з випадків: А керує Б або А і Б керують словом В, причому А стоїть поміж Б та В.

Структура графа повинна відповідати наступним вимогам:

- вершиною графа будемо вважати слово, в яке не входить жодна стрілка (експерименти показали, що найчастіше це присудок). Крім того, в процесі проходження алгоритму стало зрозуміло, що треба допускати таку можливість, , що граф може мати декілька вершин (наприклад, при однорідному присудку);
- граф, що має лише одну вершину будемо вважати графом, що відповідає простому реченню. Зрозуміло, що в такому випадку просте речення – це група залежності однієї вершини.

Розглянемо правила розбиття на прості одиниці – канонічні підграфи. Визначення цього поняття отримаємо індуктивно: а) граф фрази є своїм канонічним під графом, б) результат застосування к канонічному підграфу одного зі сформульованих далі правил розбиття назвемо канонічним підграфом (далі – просто підграф).

Перше правило дозволяє розбити граф з декількома вершинами на ряд підграфів з однією вершиною, тобто отримати набір канонічних підграфів, що відповідають простим реченням. Тобто це правило може бути сформульоване як виділення групи залежності для кожної вершини.

Друге та третє правило визначає можливість розбиття графа речення з відокремленими (знаками пунктуації) словами.

Правила розбиття можуть застосовуватися до одного й того ж графа кілька разів, що дає нові підграфи. Кінцевим графом будемо вважати підграф, то якого вже не може бути застосовано жодне правило розбиття. Такий кінцевий граф будемо вважати бажаним спрощеним реченням, отримання якого і є метою роботи розробленого алгоритму.

Наступною частиною автоматизації генерації питань є розробка алгоритму, що в якості результату своєї роботи давав би перелік словосполучень, які будуть використані як основа для питання.

На першому кроці цього алгоритму для пошуку слів (тих слів, що повторюються у тексті) використовуємо поняття псевдофлексії та псевдооснови. Під псевдофлексією розуміємо останні одну, дві або три букви слова. Також вона може мати значення 0. Частина слова, що залишилась після відсічення псевдофлексії називається псевдоосновою. Наприклад, для слова «таємниця» псевдофлексіями будуть: 0, -я, -ця, -иця. Відповідно існує також і 4 псевдооснови: таємниця-, таємниц-, таємни-, таємн-.

Зауваження: псевдооснова не має включати менш ніж 2 букви.

Кожне слово абзацу побуквенно порівнюється з кожним з наступних. Якщо у пари слів збіглися всі букви за виключенням псевдофлексії, то обидва слова отримують мітку «слова, які повторюються» (тобто у разі співпадіння основ). Недоліком такого підходу є те, що іноді маркуються слова з різними коренями, але в ході експериментів було з'ясовано, що процент таких помилок – невисокий.

На другому кроці для кожного з цих слів визначається чи є воно присудком або належить до групи залежності присудка. Якщо в таку групу залежностей входить хоча б одне марковане слово або сам присудок є таким словом, то всі слова цієї групи отримують маркування. На основі цього створюється набір ланцюжків.

На третьому кроці має відбуватися відкидання ланцюжків, які дублюються. Тобто якщо один з ланцюжків є складовою частиною іншого, то менший з них видаляється. Рахувати починаємо від слова, що зустрічається у найбільшій кількості ланцюжків. Визначається характерна для даного слова синтаксична функція. Якщо слово зустрілося в тексті n разів, з них m використань відноситься до групи залежності підмету, причому $m \geq 0{,}6n$, то для цього слова характерна функція 1, аналогічно – для присудка (функція 2). Для випадку коли $m < 0{,}6n$ & $n-m < 0{,}6n$ характерною вважається функція 3.

На останньому кроці відбувається вивід отриманих словосполучень або слів. Далі вони будуть оброблятися з метою генерації питань.

Крім цього, властивістю даного алгоритму є те, що враховуються лише повнозначні слова, що повторюються. Це було зроблено з тією метою, щоб алгоритм дозволяв не тільки з'ясувати про які поняття йде мова у певній частині тексту, але й уточнював, що саме написано про ці поняття.

Зрозуміло, що даний алгоритм базується на певних правилах, відповідно до яких і відбувається розбиття речення на елементарні частини, шукається групи залежностей, на основі яких визначаються підмет та присудок. Будемо відрізняти головні та другорядні правила. Останні вступають в дію у випадку, якщо аналіз не можливо провести лише тільки на базі головних правил. Розглянемо дані правила.

Правило для побудови груп числівників застосовується до ланцюжка числівників – кількісних та порядкових. Наприклад – сорок чотири. Головним вважається останнє слово.

Правило для побудови груп з нецілими числами застосовується для ланцюжків чисел + розділовий знак. Наприклад – 12,6. Головним вважається перший числовий комплекс.

Правило для груп зі словами, що керують числівниками застосовується до груп, що містять числівник та іменник. Наприклад – стаття 154. Головним вважається іменник. Аналогічно будуються правила для дат – 1 січня 1999 року. Так само будуються словосполучення типу «сорок вісім днів». Головною умовою є множина іменника та узгодження з числівником за відмінком.

Правила для модифікаторів прикметників застосовується для групи з двох слів, де одне з них – «такий», а інше – повний прикметник. Приклад – такий гарний. Головним вважається повний прикметник.

Правило для побудови груп прислівник + прикметник застосовується до груп, що складаються з прислівника типу «дуже» + прикметник. Наприклад – вельми корисний, дуже гарний, особливо відомий. Головним вважається прикметник.

Правила для побудови груп однорідних прикметників використовується для таких випадків – n груп, головне слово яких – прикметник у повній формі; n груп, головне слово яких – прикметник порівняльного ступеня. Умова: перші n–1 груп розділені комами, а перед останньою стоїть сурядний сполучник без коми. Якщо ланцюжок повних форм прикметників – всі вони узгоджуються за відмінком (тобто множина можливих відмінків кожного головного слова мають непустий перетин), якщо ланцюжок повних форм в однині – всі головні слова узгоджуються за родом. Приклад: гарний, поганий і злий; першої і єдиної.

Правила для прислівників порівняльного ступеня використовується для груп, що складаються зі слова, яким може керувати прислівник або прикметник порівняльного ступеня. Головним вважається прикметник. Приклад: значно важче; значно більше та розумніше.

Правила для побудови груп прислівник+дієслово використовується для груп, що складаються з одного прислівника та одного дієслова, або з групи прислівників та одного дієслова, або з одного прислівника та групи дієслів. Головним вважається дієслово. Приклад: важко жити; важко розробити.

Правила для елективних груп використовується для груп, що починаються з наступних елементів: який, один, будь-який, більшість, деякий, кожен (тільки однина) або з порядкового числівника (перший з нас) чи одиночного числівника (двоє з нас) або з прикметника у формі «най-» (найкращий з нас) . Після цих елементів має стояти прийменник «з». Головним словом є головне слово елемента.

Правила для побудови груп з іменниками застосовується до груп іменник + прикметник, причому всі прикметники узгоджуються за родом, числом та відмінком з головним словом – іменником. (єдиному справжньому другу). Також може застосовуватися для відокремлювання в препозиції до особового займенника. Застосовується до груп, що складаються з іменника, прикметника, та особового займенника, узгоджених за родом, числом та відмінком. Приклад: повернувшись пізно, стомлений та щасливий, він миттєво заснув. Також застосовується до іменників та прикметників, що не узгоджуються за числом, але узгоджуються за відмінком. Наприклад: з червоною та блакитною стрічками або стомленим папі та мамі.

Правила для побудови груп прийменників застосовується до побудови груп прийменників та іменників. Можливі відмінки іменників мають непустий перетин з множиною відмінків, якими керує прийменник. Якщо прийменник однослівний, то інформація береться з результатів морфологічного аналізу, інакше – зі словника виразів.

Правила для побудови груп однорідних іменників застосовується до групи іменників. Ці іменники мають бути узгоджені один з одним за відмінком; перед останнім стоїть сурядний сполучник без коми, а інші

розділені комою; якщо є неособовий займенник і не має самотнього сурядного сполучника, то група не будується. Розглянемо роботу даного правила покроково:

Це належить до структури речення та всього абзацу.

Перед початком роботи правила о однорідних групах іменників це речення буде мати таку структуру:

Це належить до ІГ(структури речення) та ІГ(всього абзацу).

Це правило спочатку порівнює групи (структури речення) і (всього абзацу) – грамери відмінку головних слів цих груп не перетинаються, тому після цього правило буде порівнювати групи речення і (всього абзацу), проголосить їх однією групою, яка займе старе місце слова *речення*. Після чого у речення буде структура:

Це належить до ІГ(структури однорід_І(речення та всього абзацу)).

Група не будується у таких випадках: якщо ком немає і кількість сурядних сполучників більше двох (Прийшли Вася і Катя і лектор почав заняття); якщо не має сурядного сполучника і один з членів після морфологічного аналізу було визначено як «Власна назва, ім'я тощо» (Вася, колега не прийшов) або якщо немає сурядного сполучника і один з членів (займенник) не в називному відмінку (його, батька не було).

Правила для побудови груп дієслова застосовується до груп, що складається з дієслова та іменника. Головним вважається дієслово. Умовою для побудови даної групи є знахідний відмінок іменника. Крім цього, може застосовуватися до груп, що складаються лише з дієслів. У такому випадку головним вважається те дієслово, що стоїть у інфінітиві.

Правила для побудови груп однорідних членів речення і фрагментів з двоскладними сполучниками застосовується до речень, що містять у собі двоскладні або розривні сполучники. Це правило працює у реченні зліва направо, будує сполучені слова та групи.

Правило побудови іменника та дієприкметника використовується для речень типу «будинок, побудований на тому місці,...». Головним є іменникова група. Має виконуватися узгодженість за родом, числом та відмінком.

Таке саме правило застосовується і до групи, що складається з іменника та фрагменту зі словом «який» або «чий». Приклад: будинок, який побудували на тому місті.

Окрім, для даного алгоритму ще розроблено низку додаткових правил, що допомагають визначити підмет та присудок у разі неможливості їх визначення за допомогою вищенаведених правил.

1. Пошук підмету та присудку в реченні, що містить тире. Пошук починається при умові, що фрагмент має вершину «тире». Якщо в лівій частині речення від тире стоїть одиночний іменник у номінативі або іменник у номінативі, який є вершиною групи, або інфінітив, то даний іменник або інфінітив проголошуються потенційними підметами. У випадку, коли знайдені одночасно й іменник у номінативі й інфінітив, то перевага надається інфінітиву. Якщо в правій частині речення від тире стоїть одиночний іменник у номінативі або іменник у номінативі, який є вершиною групи, або інфінітив, то даний іменник або інфінітив проголошуються потенційними присудками. У випадку, коли знайдені одночасно й іменник у номінативі й інфінітив, то перевага надається інфінітиву. Група підмета та присудка будується тільки тоді, коли знайдені й потенційний підмет і потенційний присудок.

2. Пошук підмету та присудку у випадку, коли речення містить коротку форму прикметника або дієприкметника та дієслова у особовій формі. Пошук починається якщо фрагмент має вершини з такими ознаками – коротка форма прикметника або дієприкметника, або дієслова у особовій формі. Підметом вважатимемо вершину фрагмента. Визначимо підмет через П.

2.1. Якщо П – дієслово в особовій формі. Якщо П є словом «здаватися» в 3 особі однини, та перед ним є іменникова група в давальному відмінку і без називного, тоді не будуємо групу підмета. Наприклад: Васі здається, що він правий – не будуємо.

2.2. Потенційним присудком може бути будь-яка словоформа в номінативі наступних класів (в порядку пріоритету): словоформа «хто»; особовий займенник у номінативі; числівник «обидва» в номінативі; синтаксичні іменники, що мають номінатив і не є «що»; словоформа «що», якщо підмет є нетранзитивним дієсловом або з підметом було побудовано групу, що має прямий додаток; одиночний числівник з класу «обидва, двоє, четверо,...», будь-який порядковий числівник або прикметник, якщо він має номінатив; головне слово в номінативі в деяких групах.

Після знаходження гіпотези перевіряється її узгодженість з присудком.

Тобто головними етапи даного алгоритму є побудова елементарних частин речення, визначення підмету та присудку з метою побудови груп залежності, та виділення повнозначних слів, які повторюються. Саме ці слова і будуть словами, на основі яких будуються питання до навчального матеріалу.

Отримані запитання

Генерація питань відбувається на основі слів та словосполучень, отриманих завдяки алгоритму, вищеописаному, та на основі шаблонів. На даний момент створено лише 2 шаблони «Що таке <> » та

«Розкрийте поняття <>» (якщо в отриманому словосполученню є слово «поняття»). У тому випадку, коли для отриманого словосполучення застосовується шаблон <Що таке>, то головне слово даного словосполучення ставиться у називний відмінок, а відмінок залежного слова узгоджується з головним.

Наприклад, одне з запитань, отримане автоматично виглядає наступним чином: *Що таке елементарна задача?* Це питання належить до абзацу «Будемо називати задачу елементарною, якщо для неї існує явний алгоритм розв'язку. Тоді метою декомпозиції будь-якої задачі є її зведення до сукупності елементарних задач.»

Крім того, ефективність даного алгоритму було перевірено ще й на абзаці з роботи Щербини «Питання граматики і лексикології української мови». Уривок з його роботи виглядає так:

«У художній літературі, нарешті, діє найважливіша категорія, що визначає кінець кінцем естетичну цінність слова, – категорія літературного образу (персонажа, автора). Тут зв'язок слова та образу безпосередній і очевидний, бо в мовній характеристиці персонажа, у авторській мові все обумовлено, мотивовано правдою мистецтва. І якщо мати на увазі художню цінність каламбуру у новому обсязі, тобто у широкому контексті, в системі літературних образів, включаючи образ самого автора, то можна сказати, що ця цінність не залежить навіть і від якостей дотепності самих по собі, а визначається насамперед правдою життя, правдою мистецтва.»

Ключовими словосполученнями для даного уривку будуть наступні словосполучення: найважливіша категорія, категорія літературного образу, зв'язок слова і образу, система літературних образів, цінність каламбуру.

Зрозуміло, що застосування до даних словосполучень тестового шаблону «Що таке <>», дає декілька варіантів тестових запитань, декілька з яких можуть бути обрані як повноцінні питання.

Висновки

Для допоміжної автоматичної генерації питань при тестуванні запропоновано метод, який не базується на лінгвістичному аналізі тексту, а передбачає побудову питань за всіма поняттями, які явно виділені у тексті навчального курсу. Тобто поняття, що були виділені «курсивом», додаються до шаблону питань які пропонуються викладачеві як кандидати на занесення у базу тестових питань.

Даний підхід має свої переваги та недоліки.

Даний метод не базується на лінгвістичному аналізі, тому виділене означення, що складається більше ніж з одного слова (наприклад, алгоритм Дейкстри) розглядається як окремі визначення. Але цей недолік долається завдяки тому, що будь-яке автоматично генероване питання спочатку має пройти перевірку у викладача, а вже після його затвердження питання може додатися до загальної бази питань. Тому помилкові питання відкидаються відразу і не можуть вплинути на подальше тестування. Як показали експерименти, процент таких некоректних питань дуже невисокий.

Недоліком такого підходу є те, що отримані запитання покривають лише визначення, вони дозволяють перевірити лише мінімальний обсяг знань. Але з іншого боку вони можуть бути використані для тестів, що застосовуються для поточних перевірок або для звичайного адаптивного тестування як питання першого, найлегшого рівня складності.

Такий підхід вимагає деякої стандартизації подання навчальних матеріалів у цілому та визначень в них зокрема. Цей пункт не можна віднести до недоліків, бо стандартизація процесів або інструментів завжди впливала на продуктивність лише тільки позитивно.

Даний алгоритм передбачає наявність допоміжного словника, що вимагає певних людських зусиль для його створення.

Зрозуміло, що даний метод може бути застосований лише як допоміжний до методу, що базується на лінгвістичному аналізі, але і він значною мірою допомагає автоматизувати процес створення тестових запитань.

Запропонований алгоритм автоматизації побудови текстових запитань на основі лінгвістичного аналізу текстів українською мовою базується на концепції зв'язності тексту як семантичної близькості фраз. Ця концепція складається з таких припущень: закономірності зв'язного тексту можуть бути описані як способи передачі думки від речення до речення, а найбільш очевидний спосіб передачі думки – це повторення одних і тих самих семантично близьких понять у зв'язному тексті. Даний алгоритм являє собою аналіз надфразової побудови тексту, та заснований на принципі повтору лексико-синтаксичних структур. У ході аналізу та під час встановлення внутрішньо-текстових зв'язків такі структури наводяться за певними правилами до спрощеного виду (наприклад, складні речення замінюються набором простих, а відокремлені – оформлюються як незалежні стандартизовані речення). У результаті такого розбиття стає можливим реконструювати схему контекстуальних відношень між повторюваними лексемами. За допомогою таких схем визначаються повтори найбільшої довжини, що відтворюють своєрідний гіперсинтаксичний скелет тексту.

Подальший розвиток дослідження планується проводити у бік оптимізації алгоритму та кращій інтеграції різних рівнів аналізу один з одним. Крім того, планується розширити клас текстових питань, що підлягають автоматизації. Зрозуміло, що наступні дослідження будуть спрямовані у бік розвитку

морфологічного аналізу як однієї з головних складових лінгвістичного. Насамперед, бачаться перспективними дослідження алгоритму морфологічного передбачення. Власне морфологічний аналіз, а саме його автоматизація, є головними напрямками подальшого дослідження в цій галузі.

В якості розвитку системи, можна розглядати побудову спільного, алгоритмізованого словника для української мови, який можна було б використовувати у майбутньому при розробці аналогічних систем. Але такий словник можливий за умови виходу в світ словника української мови, який міг би стати основою для тезаурусів та словників, які використовуються в аналізі.

1. *Севбо И.П.*, Структура связного текста и автоматизации реферирования, – М.: Наука, 1969. – 135 с.
2. *Глибовець М.М., Данченко А.А.* Про один з підходів використання розподілених експертних систем навчальних систем у дистанційній освіті, Наукові записки НаУКМА, Комп'ютерні науки. – 2003. – Том 21. – С. 53 – 64.
3. *Глибовець М.М., Яцевський В.* Проблема організації навчальних ресурсів: побудова депозитарію, міжнародна конференція «Теоретичні та прикладні аспекти побудови програмних систем». – К.: 2005 С. 49 – 61.
4. *Аванесов В.С.* Научные основы тестового контроля знаний. – М.: Исследовательский центр, 1994. – 135 с.
5. *Weiss D.J.(Ed.)* New Horizons in Testing: Latent Trait Test Theory and Computerised Adaptive Testing. N – Y., Academic Press, 1983. – 345 p.
6. *Lord F.M.* Application of Item Response Theory to Practical Testing, Lawrence Erlbaum Associates, Inc.. 1980. – 274 p.
7. *Штерн І.Б.* Вибрані топіки та лексикони сучасної лінгвістики. – К.: 1998. – 336 с.
8. *Белецкая И.* Типы лингвистических задач и реализующие их процедуры // Знания – Діалог – Решение. – К.: – 1990. – 235 с.