УДК 004.023: 81`32

ЕВРИСТИЧНИЙ АЛГОРИТМ МОРФОЛЕКСИЧНОГО АНАЛІЗУ ДЛЯ НЕВІДОМИХ СЛІВ

В.Ю. Тарануха

Київський національний університет імені Т. Шевченко,
03680, Київ, проспект Академіка Глушкова, 4д
Тел.:+(38044)259 0427, факс: +(38044)259 0427, e-mail: ava@unicyb.kiev.ua

Розглянуто спосіб покращення евристичного алгоритму морфолексичного аналізу невідомих слів для слов'янських мов. Пропонується використовувати словник тексту як основне джерело даних для побудови гіпотез, та набір n-грам як допоміжне джерело для фільтрації.

The article deals with improvement of heuristic algorithm for morpholexical analysis of unknown words in Slavic languages. Lexicon is used as a main source of niformtion to construct hypotheses. Set of n-grams is used for filtering.

Вступ

Для ефективної взаємодії між людиною та машиною оптимальним видається застосування природної мови, та програмного забезпечення, що здатне розуміти команди природною мовою. Проте, природні мова є великою, складною та повсякчас змінюється. Одною з важливих підзадач при створення природномовних інтерфейсів між людиною та машиною є евристичний морфролексичний аналіз слів природної мови.

Слов'янські мови мають складну систему словозміни, що пов'язано з особливостями побудови синтаксичних зв'язків у реченнях. Це ускладнює визначення канонічних форм та граматичних характеристик у порівнянні з романо-германськими мовами. Багатство словотвірних моделей серйозно ускладнює морфолексичний аналіз для невідомих слів. Слова можуть бути невідомими з ряду причин, починаючи з тривіальної відсутності в словнику попри давню присутність у мові, закінчуючи випадками коли в тексті зустрілося нове для мови слово.

Є ряд методів, що дозволяють виконувати евристичний морфолексичний аналіз, кожен з яких має свої переваги та недоліки. До них входять: аналіз окремих словоформ [1], методи засновані переважно на групуванні [2, 3], використання складних комплексів різних ознак [4], та використання аналізу зв'язків між елементами тексту [5].

Мета роботи – дослідження способу, що буде простим у реалізації, не вимагатиме великих баз знань про мову та буде здатен налаштовуватися на характеристики конкретного тексту.

Огляд існуючих підходів

Введемо наступні позначення. Лема – слово мови в усіх його формах, а словоформа – конкретна форма слова з прив'язаними до неї граматичними ознаками. Канонічна форма – форма слова, що однозначно визначає слово і множину його словоформ. Машинна основа – частина слова, що не змінюється в різних словоформах одного слова, може бути порожня. Машинне закінчення – послідовність літер з кінця слова, що безпосередньо слідує за машинною основою, може бути порожнім. Машинна флективна група – сукупність машинних закінчень, що відповідають одній канонічній формі слова та описують всі словоформи для даної канонічної форми. Машинний суфікс – послідовність літер з кінця машинної основи. Словник системи – сукупність таблиць, що описують канонічні форми, флективні групи та зв'язки між ними у відповідній системі.

Алгоритм аналізу окремих словоформ, що базується на використанні машинних словників [1]. В цьому алгоритмі в якості основного джерела даних використовується таблиця відповідностей машинних суфіксів машинним закінченням. В якості основної евристики – максимізація довжини послідовності літер, що збіглась у словоформі невідомого слова та у певної комбінації машинний суфікс + машинне закінчення, отриманої з таблиць.

В роботі [1] для реалізації було вибрано такі частини мови: іменник, дієслово, прикметник та прислівник. З метою швидкої реалізації був використаний скінчений автомат, що дозволяє швидко аналізувати послідовності літер з кінця слова. Для розв'язання потенційних колізій пов'язаних із збіжністю написання різних за граматичними ознаками словоформ в межах однієї частини мови вибирається один варіант інтерпретації, що пов'язаний з найбільш продуктивним закінченням. Продуктивність визначається відповідно до частоти вживання машинної флективної групи в словнику.

Якість роботи алгоритму сягає 87 %, тобто це та частка словоформ слів, що вважаються невідомими для словника системи, для яких було коректно визначено принаймні 1 комплект ознак: канонічну форму, закінчення та всі граматичні ознаки для відповідного закінчення.

Алгоритм заснований на групуванні [2] використовує додаткову евристику засновану на сукупності словоформ тексту, а саме, те що різні словоформи, що відповідають одній канонічній формі повинні мати спільну

машинну основу та їхні машинні закінчення повинні входити до спільної флективної групи. В роботі[2] для кожної словоформи будується гіпотеза, що описується деревом, заданим формальною граматикою. Коли всі дерева побудовані виконується кореляційний аналіз між гіпотезами, з метою відкидання хибних гілок та можливо дерев.

Кореляція будується за такими ознаками: по гіпотезам основ, по значенням частини мови, по відмінам дієслів, по роду іменників, множинам флексій, що задають парадигматичні класи. Словозмінні категорії як то, наприклад, відмінок не використовуються в кореляції. При такому підході спостерігатися генерація зайвих наборів граматичних ознак для аналізованих словоформ.

Алгоритм, що використовує складні комплекси ознак [4], та орієнтований на специфічний підклас невідомих слів, а саме – на прізвища. Будується надлишкова множина гіпотез про канонічну форму та граматичні ознаки кожної словоформи, а потім фільтрується. Метод використовує додаткову інформацію за спеціальними маркерними словами: „пан", „пані", „мсье", „леді", тощо, при побудові базових гіпотез.

Фільтрація відбувається:

– на основі правил утворення прізвищ конкретної мови, наприклад, варіантів прізвищ на „ов/ин" чоловічого роду: „Скин" – „Якин". Фільтруються гіпотези по окремо взятим словоформам;

– на основі порівняння даних з одного тексту. Словоформи об'єднуються в кластери за допомогою часткового співставлення за множинами словоформ;

– на основі спеціалізованих правил щодо елементів словоформ. Наприклад: „жолі", „швілі", „іані". Фактичний список виходить довгим і вимагає підгонки під конкретну мову;

– на основі загального правила: вибирається гіпотеза, що має максимальну кількість збіжних літер кінця з відомим прізвищем/моделлю прізвища.

Загальна якість роботи висока (F_1 -міра 93 %, при точності визначення граматичних ознак 94 %; та повноті 92 %), проте очевидним недоліком є необхідність звертатися до експертів-лінгвістів, щоб настроїти фільтри алгоритму для певної мови.

Алгоритми, що використовують синтаксичний аналізатор іменних груп та приховану модель Маркова, описані в роботі [5]. Такі методи дозволяють успішно фільтрувати гіпотези про канонічну форму та граматичні ознаки кожної невідомої словоформи з високою точністю. Проте це з одного боку передбачає реалізацію та використання важких в обчислювальному сенсі алгоритмів, з другого боку є надлишковим, якщо текст, що аналізується не вимагає зняття морфологічної неоднозначності.

Базовий алгоритм

В роботі [3] досліджено можливість використання групових евристик для аналізу значно ширшої множини частин мови і зроблено спробу побудувати більш-менш універсальний алгоритм для слов'янських мов, не обмежуючись якоюсь однією мовою. З того часу як було створено першу версію було проведено ряд досліджень, що показали що ряд припущень, використаними в роботі [3] є зайвими, і алгоритм можна спростити без погіршення якості морфрлексичного аналізу.

Виявилося, що спроба застосувати спеціалізоване сортування для того, щоб збільшити імовірність послідовного об'єднання двох словоформ в гіпотезу засновану на групі нічого не дає в сенсі точності аналізу, проте ускладнює алгоритм та сповільнює його роботу. Також було внесено ряд спрощень у програмну реалізацію.

В роботі алгоритму використовуються такі фіксовані джерела інформації про очікувані флективні групи та граматичні характеристики слів:

Таблиця відповідностей машинних суфіксів машинним закінченням, має два варіанти. Варіант 1: таблиця виключно для слів зі словозміною, як то іменники, дієслова, чисельники тощо. Варіант 2: таблиця для будь-яких слів мови.

Таблиця відповідностей машинних закінчень граматичним кодам.

Таблиця правил для незмінюваних частин мови. Це єдине що вимагає роботи лінгвіста для адаптації під конкретну мову, всі інші таблиці можна згенерувати автоматично за машинним словником.

Цей підхід відрізняється від запропонованого в [2], оскільки розрахований на визначення граматичних ознак для будь-яких частин мови, а не лише обмеженої множини.

Основним джерелом даних є словник словоформ тексту T . Слово, для словоформ якого є запис у словнику системи (базі даних слів системи) D будемо називати відомим, інакше – невідомим.

Базовий алгоритм ЕА:

– скласти словник тексту T (позначимо його W);

– відсортувати словник W за алфавітом;

– розбити на блоки, по першим двом літерам. Кожен блок опрацьовується незалежно. При потребі це дозволяє зробити паралельну реалізацію алгоритму;

– в межах кожного блоку застосувати процедуру агрегації;

– якщо словоформа не агрегувалася, то застосувати процедуру аналізу одиничної словоформи.

Процедура агрегації:

1) зафіксувати стартову словоформу – першу серед наявних у блоці, якщо така є. Якщо немає – перейти на пункт 10 процедури;
2) утворити гіпотезу зі стартової словоформи;
3) перебрати слова у блоці починаючи від другого і до кінця – виконати пункти 4–6;
4) взяти слово, спробувати приєднати до гіпотези;
5) гіпотеза складається, якщо для двох чи більше словоформ можна виділити:
   - спільний початок слова довжиною більше 0 (машинну основу),
   - машинний суфікс спільній для всіх словоформ довжиною більше 0,
   - множину машинних закінчень, що точно вкладається принаймні в одну машинну флективну групу,
   - машинний суфікс допускає зв'язування з принаймні з однією машинною флективною групою визначеною для гіпотези.
6) інакше – пропустити слово, продовжити цикл;
7) всі слова, що приєдналися до гіпотези виключити із словника W ;
8) використати правила для незмінюваних частин мови над гіпотезою;
9) перейти на пункт 2 процедури агрегації;
10) гіпотези, що складаються з однієї словоформи розформувати.

В цій процедурі використовується таблиця відповідностей машинних суфіксів до машинних закінчень Варіант1.

Процедура аналізу одиничної словоформи:

1) на основі останніх символів слова перебрати варіанти машинного суфіксу та машинного закінчення;
2) вибрати найдовшу послідовність літер, для якої можна утворити послідовність „машинний суфікс" + "машинне закінчення", таку, що машинний суфікс та машинне закінчення сумісні;
3) визначити граматичні характеристики словоформи на основі машинного суфіксу та машинного закінчення;
4) використати правила для незмінюваних частин мови над гіпотезою.

В цій процедурі використовується таблиця відповідностей машинних суфіксів до машинних закінчень Варіант2.

Наприклад, після виконання базового алгоритму ЕА словник виду: {„дзвін" , „дзвінок", „дзвінком", „дзвонять"} дасть наступну множину гіпотез: {(„дзвін",((0, ім. чол. одн. наз.))),(„дзвін",((„ок", ім. чол. одн. наз.), („ком", ім. чол. одн. орудн.)), („дзвон", ((„ять", дієсл. множ. тепер.)))}

Пошук та підстановка в правило, в першу чергу виконуються для прикметників, з утворенням прислівників, для української та російської мов. Для інших частин мови це залежатиме від конкретної реалізації флективних груп у машинному словнику.

Наприклад, нехай лема „быстро" була невідома. Тоді отримавши машинну основу „быстр" для машинного закінчення „о" матиме набір ознак для середнього роду. Це буде отримано за моделлю слова „зеленый" та формою „зелено". Маючи в розпорядженні лему та канонічну форму перевіряється, чи можна отримати прислівник „быстро" за формулою: словоформа прикметника середнього роду однини, що складається з машинної основи та машинного закінчення „о" також створює лему класу прислівник.

При використанні алгоритму виникли певні міркування щодо очікуваних показників точності.

По-перше, при такому підході, порівняно з [2] значно скорочується кількість гіпотез, та втрачається частина групувань, що могли б утворитися, якби використовувався повноперебірний підхід подібний до описаного в роботі [2]. Проте це не спричинило помітних втрат точності. Це пов'язано з властивостями угадування для слов'янських мов.

Наприклад, потенційна гіпотеза для російської мови („генера"+ „ла", „генера"+ „л") може бути проаналізована як („что делала?", „что делал?") з породженням зайвих варіантів розбору. Проте примусове вилучення словоформ знищує такі гіпотези, якщо коректна гіпотеза зустрінеться раніше. З іншого боку це повинно призводити до того, що якщо неправильна гіпотеза згенерується раніше за правильну, то всі словоформи отримають неправильні граматичні характеристики.

Те, що загальна оцінка в цілому не страждає пов'язано з розподілом словоформ при алфавітному упорядкуванні словника W . Виявляється, що імовірність утворити правильну гіпотезу набагато вища за імовірність утворити неправильну гіпотезу, за умови, що словоформи зібрані з тексту, що написаний правильною мовою.

По-друге, проблеми з точністю виникнуть також у випадках, якщо в словник тексту що аналізується потраплять словоформи з грубими помилками, але для текстів без примусових спотворень імовірність незначними.

Наприклад, гіпотеза для української мови („генера"+ „ла","генера"+ „ти") буде проаналізована як („що робила?", „що робити") з породженням апріорі хибних варіантів.

Знову ж таки, незначні втрати від таких гіпотез пояснюються такими факторами:

- порівняно незначною кількістю помилок у тестовому корпусі,
- такі гіпотези поглинають незначну кількість словоформ,
- найчастіше крім втрачених словоформ невідомі леми мають інші словоформи, що коректно аналізуються.

Важливою особливістю Базового алгоритму ЕА є те, що можна регулювати вимоги до довжини машинного суфікса, тим самим отримувати різні набори граматичних ознак для однієї і тієї самої гіпотези про групування, при можливо різних машинних основах, частинах мови та множинах машинних закінчень. Це в першу чергу корисно при використанні у взаємодії з синтаксичним аналізом, оскільки може виявитися, що машинний суфікс було вибрано невірно, це дало невірні граматичні ознаки для машинні закінчення, які в свою чергу дали невірні граматичні ознаки, що спричинило неможливість коректного синтаксичного розбору. Тоді в ряді випадків на вимогу модуля синтаксичного аналізу можна переобчислити характеристики гіпотези. Проте згадане переобчислення є ознакою того, що словник тексту було проаналізовано некоректно і виникає імовірність, що в синтаксичний аналіз попередньо опрацьованої частини тексту теж потрапили помилки.

Крім того, в ряді випадків набір граматичних ознак одразу генерується надто широким, хоч і напевне покриває необхідні коректні ознаки. Виникає потреба в надбудові, що дозволить водночас максимізувати кількість коректних граматичних ознак та мінімізувати кількість зайвих ознак.

Надбудова над базовим алгоритмом

В якості додаткового джерела даних пропонується використати набір n-грам зібраних з тексту T, що аналізується, де n-грама це послідовність з n елементів, що замінюють словоформи у копії тексту. Надалі, не порушуючи загальності, зафіксуємо $n = 2$.

Введемо такі позначення. Множина комплектів граматичних ознак словоформи G (надалі – грам-множина). Наприклад, для словоформи іменника це множина комплектів, що описують комбінації роду, числа і відмінку, що відповідають заданій словоформі іменника. Грам-множини виникають через те, що різні граматичні ознаки часто прив'язуються до однієї і тієї самої словоформи. Так, наприклад, для іменників першої відміни м'якої групи однини форми родового та давального відмінків збігаються, „кого-чого" – „землі" та „кому-чому" – „землі". Омонімія в цілому є поширеним явищем в слов'янських мовах і це вимагає відображення у моделі. Грам-множина є фактичним результатом роботи морфолексичного аналізу. Грам-код g - чисельний код, що приписується кожній грам-множині G та однозначно визначає грам-множину.

Флекс-множина F – множина номерів флективних груп, що були використані для генерації грам-множини. Флекс-код f – чисельний код що приписується кожній флекс-множині F та однозначно визначає флекс-множину.

Вводиться функція $Tr_g()$, що співставляє словоформам певні елементи наступним чином:

1) для кожного відомого слова, що належить до повнозначних змінюваних частин мови (як то іменник, дієслово, прикметник, тощо) та для займенників результатом буде грам-код $g(w)$;

2) для відомого кожного слова, що належить до службових частин мови, або незмінюваного слова результатом буде відповідна словоформа;

3) для кожної невідомої словоформи результатом буде вона сама.

Вводиться функція $Tr_f()$, що співставляє словоформам певні елементи наступним чином:

1) для кожного відомого слова, що належить до повнозначних змінюваних частин мови(як то іменник, дієслово, прикметник, тощо) та для займенників результатом буде флекс-код $f(w)$;

2) для відомого кожного слова, що належить до службових частин мови, або незмінюваного слова результатом буде відповідна словоформа;

3) для кожної невідомої словоформи результатом буде вона сама.

Виконується трансформація двох копій тексту з використанням функцій $Tr_g()$ та $Tr_f()$, коли кожне входження словоформи замінюється на значення відповідної функції.

На основі трансформованої копії тексту з використанням $Tr_g()$ обчислюються n-грами. Важливо, що використовуються лише n-грами, що враховують виключно лівий контекст словоформи. Це пов'язано зі структурою словосполучень в українській та російській мові. По-перше, прийменники, що дозволяють визначити відмінок обов'язково стоять попереду іменників. По-друге, прислівники та інші слова, що модифікують значення також частіше стоять перед тим словом, що уточнюють. У складних конструкціях, виду „прийменник" + „прикметник" + „іменник" прийменник також стоїть перед відповідним прикметником, що може бути невідомим словом.

Побудова набору векторів ознак.

1) За копією тексту збирається словник W_g, де елементами є відповідні унікальні значення $Tr_g()$, де w є словоформа з тексту T. Цей словник є опорним словником.

2) На основі W_g будується сукупність векторів V_g, кожен з яких відповідає своєму унікальному значенню $Tr_g(w)$. Елементами векторів є частоти n-грам, що відповідають комбінаціям $(Tr_g(w_i), Tr_g(w))$, де $Tr_g(w_i)$ – значення, що може відповідати більш ніж одній словоформі.

3) За копією тексту збирається словник W_f, де елементами є відповідні унікальні значення $Tr_f()$, де w є словоформа з тексту T. Цей словник є довідником, для подальшої оптимізації.

Після того, як ознаки побудовані множина $\{Tr_g(w) \mid w \in T\}$ розділяється на дві підмножини: $Sk = \{Tr_g(w) \mid w \in T \;\&\; w \in D\}$ та $Sh = \{Tr_g(w) \mid w \in T \;\&\; w \notin D\}$. За побудовою, в Sh знаходяться невідомі словоформи.

Sk задає перелік, який описує у V_g сукупність правил зв'язування для відомих слів та наборів граматичних ознак. Sh задає перелік, який описує у V_g сукупність правил зв'язування для невідомих слів, та відповідних їм грам-кодів. Таким чином досягається ефект налаштування системи на текст T, та виключається потреба в залученні професійних лінгвістів для побудови додаткових правил, як то було в роботі[4]

Розширений алгоритм ЕА:

1) обчислити ознаки за алгоритмом ЕА в різних режимах, щоб отримати різні кількості грам-кодів Відповіді системи для словоформ утворюють множину A, елементами якої є трійки $(i, w, EA(w,i))$, де ***i*** - номер відповідних параметрів запуску;

2) для кожної невідомої словоформи $w \in Sh$, для всіх значень ***i,*** обчислюється елемент або сума елементів з V_g, що відповідають грам-кодам отриманим за $EA(w,i)$. Позначимо її $Sw(w,i)$;

3) вибирається $V_g(w_j)$, такий що для $w_j : g(w_j) \subseteq g(w) \;\&\; \forall k \neq j, |g(w_j)| \geq |g(w_k)|$;

4) обчислюється косинус кута між $V_g(w_j)$ та $Sw(w,i)$;

5) максимальне значення вказує на найкращий результат аналізу.

Обчислення $V_g(w_j)$ безпосередньо є досить складною задачею, оскільки в загальному випадку вимагає повного перебору або оптимізації якимось іншим чином. В реалізації для спрощення задачі застосовуються дані з W_f, що дозволяє зручно оперувати $f(w)$, відповідно підібрати оптимальний $V_g(w_j)$, за ознакою $f(w_j) \subseteq f(w)$ і уникнути повного перебору ознак.

Таким чином сформульований розширений алгоритм ЕА базується на двох припущеннях. По-перше, що текст який підлягає аналізу хоч і може бути написаний з порушенням нормативної граматики для вибраної мови, але зі збереженням єдиних локальних граматичних правил від початку до кінця тексту. По-друге, що слова тексту побудовані за спільними для всієї мови принципами та правилами словотворення, і ці правила не перевизначаються в процесі написання тексту.

Чисельний експеримент

Для експерименту було вибрано українську мову. Експерименти було проведено на текстах стенограм Верховної Ради України. Було сформовано корпус обсягом 112,5 МБ. Для цього відповідні стенограми було зібрано з сайту http://rada.gov.ua/meeting/stenogr .

На корпусі було виділено словник системи з 15,620 словоформ, всі інші слова були замінені на стоп-слово "#". Словник було пропущено через систему морфолексичного аналізу, і отримано словники канонічних форм, обсягом 3519 одиниці, та словник грам-кодів обсягом 1270 одиниць.

Зі словника було виділено вибірку в 350 словоформ, для якої виконано евристичний морфолексичний аналіз за Базовим алгоритмом ЕА. Середня кількість словоформ у групі при цьому дещо відрізняється від середньої за словником, а саме 4,31 на вибірці проти 4,43 на всьому словнику.

Для збереження порівнюваності якість роботи алгоритму визначалася двічі. Перший раз за тими самими ознаками що і в [3], а другий – за підходом описаним в [1].

Код, що відповідає допустимій комбінації граматичних ознак для відповідної частини мови будемо називати кодом ознак.

Множину кодів ознак словоформи тексту, отриману за допомогою еталонного словника або вручну з використанням правил граматики , будемо вважати множиною правильних кодів ознак.

Точність за кодами ознак – це відношення кількості правильних кодів ознак до кількості всіх кодів ознак, які мають бути отримані алгоритмом для словоформи.

$$Acc = \frac{Corr}{Corr + Miss}, \tag{1}$$

де Acc – точність за кодами ознак, $Corr$ – кількості правильних кодів ознак, $Miss$ – кількість кодів, що повинні були потрапити у результат, але не потрапили.

Множина надлишкових кодів ознак словоформи тексту становить сукупність кодів ознак отриманих алгоритмом, які не входять в множину правильних кодів ознак словоформи тексту.

Надлишковість у визначенні кодів ознак – це відношення кількості надлишкових кодів ознак до кількості кодів ознак, отриманих в результаті роботи алгоритму для словоформи в цілому.

$$Excess = \frac{Extra}{Corr + Extra}, \tag{2}$$

Де $Excess$ – надлишковість, $Corr$ – кількості правильних кодів ознак, $Extra$ – кількість зайвих кодів ознак.

Для вибірки отримані такі значення: точність 93 %, надлишковість 18 %, що порівнювано з результатами отриманими в роботі [3].

Після використання розширеного алгоритму ЕА, результати відчутно змінилися. Завдяки відкиданню частини зайвих грам-кодів надлишковість впала до 13 %. Як виявилось розширений алгоритм ЕА незначно збільшив кількість коректних грам-кодів.

На основі точності та надлишковості обчисленої таким чином можна зробити висновок про те, що якщо за евристичним морфолексичним аналізом буде слідувати синтаксичний аналіз то буде згенеровано суттєво менше неправильних та химеричних синтаксичних дерев.

При порівнянні за способом визначення точності запропонованим у [1], точність базового алгоритму склала 97 %, оскільки в роботі [1] точною вважається відповідь алгоритму на словоформу, якщо словоформа має хоч би 1 коректний код ознак, а кількість зайвих взагалі ніяк не оцінюється. За такого методу порівняння точність розширеного алгоритму також склала 97 %.

Після приведення до показників за мірою F_1, базовий алгоритм ЕА дає значення 0,87, розширений 0,9.

Висновки

В роботі проаналізовано можливість покращити евристичний морфолексичний аналіз невідомих слів без використання зайвих складаних обчислень, як то звертання до синтаксичного аналізу або інших важких алгоритмів. При цьому алгоритм не вимагає джерела даних про мову у вигляді додаткових правил граматики, але здобуває певне представлення способів зв'язування слів безпосередньо з тексту, що аналізується.

Показано, що запропонована надбудова над базовим алгоритмом ЕА покращує якість роботи в середньому, зменшуючи кількість зайвих граматичних ознак та не зменшуючи кількості коректно визначених. В такому вигляді його можна вживати для автоматизації побудови словників, та як попередній етап перед автоматичним синтаксичним аналізом. Останнє буде особливо вдалим, оскільки запропонований алгоритм за рахунок внутрішньої фільтрації в середньому гарантує зменшення кількості породжених парсером варіантів синтаксичного розбору.

Подальша робота повинна включати аналіз n-грам розмірності більш ніж 2 та можливість застосування попередньо в ручну зібраної бази векторів для ряду службових частин мови, як то прийменників. Враховуючи особливості української та інших слов'янських мов можна припустити, що це має покращити результат отриманий у цій роботи.

1. *Сокирко А.В.* Морфологические модуле на сайте aot.ru // Компьютерная лингвистика и интеллектуальные технологии: Диалог'2004, 2004 – С. 559–564.
2. *Ножов И.М.* Процессор автоматизированного морфологического анализа без словаря. Деревья и корреляция // 2000, Ел. версія: – http://www.dialog-21.ru/Archive/2000/Dialogue%202000-2/284.htm.
3. *Анисимов А.В., Романик А.Н., Тарануха В.Ю.* Эвристические алгоритмы для определения канонических форм и грамматических характеристик слов // Кибернетика и Системный анализ. –2004. – № 2. – С. 3–14.
4. *Сулейманова Е.А., Константинов К.А.* Об эвристическом методе разрешения неоднозначности при морфологическом анализе незнакомых фамилий // Машинное обучение и анализ даннях. – 2013. – Т. 1, № 5. – С. 519–525.
5. *Сокирко А.В., Тюлдова С.Ю.* Сравнение эффективности двух методик снятия лексической и морфологической неоднозначности для руского языка // 2005. Ел. версія: - http://download.yandex.ru/company/grant/2005/01_Sokirko_92802.pdf.