ПЛОВДИВСКИ УНИВЕРСИТЕТ „ПАИСИЙ ХИЛЕНДАРСКИ“ – БЪЛГАРИЯ
НАУЧНИ ТРУДОВЕ, ТОМ 50, КН. 1, СБ. В, 2012 – ФИЛОЛОГИЯ,
PAISII HILENDARSKI UNIVERSITY OF PLOVDIV – BULGARIA
RESEARCH PAPERS, VOL. 50, BOOK 1, PART C, 2012 – LANGUAGES AND LITERATURE

ЕЗИКОВИТЕ КОРПУСИ И ПРЕВОДНИТЕ ПАМЕТИ КАТО ДОПЪЛНИТЕЛЕН РЕСУРС ЗА УЧЕБНИ РЕЧНИЦИ

Руси Николов
Пловдивски университет „Паисий Хилендарски“

LINGUISTIC CORPORA AND TRANSLATION MEMORIES AS ADDITIONAL RESOURCES FOR STUDY DICTIONARIES

Roussi Nikolov
Paisii Hilendarski University of Plovdiv

TREFL – Translation REFerence Library – is intended to use as a simple, versatile, portable, effective and customizable reading, writing and translation aid tool capable of managing very large sets of data. Over the last few years it has evolved considerably, both technologically and linguistically. The performance of the search engine and the overall quantitative and qualitative characteristics of the available textual resources justify a deeper presentation of the essential aspects of the program, namely the applied search algorithm and the linguistic prioritization model.

Key words: corpora, translation memories, dictionaries; search engine, linguistic prioritization

Записите в учебните речници съдържат обикновено илюстрации на употребата на съответните думи в контекст. В съвременната практика на чуждоезиковото обучение част от специалистите поставят дори на преден план, като първостепенен обект на изучаване именно тези представителни и запомнящи се речеви единици, по-големи от думата, считайки, че функционалните описания, дефинициите и преводните еквиваленти на лексикално ниво водят по-скоро до разхищение на интелектуални ресурси. Тук ще приемем разумно, че оптимални резултати в тази област се постигат чрез подходящо дозиране на различни подходи. Следователно изчерпателността и качеството на илюстративния речеви материал в речниковите статии, както и ефективността на достъпа до него са не по-малко важни от прецизността на

дефинициите, лингвистичните описания и преводите на лексикално равнище. Информационните технологии и свободно достъпните едноезични и паралелни езикови ресурси в дигитална форма предоставят нови и несъизмеримо по-ефективни методи за обогатяване, ефективно управление и персонализиране на съдържанието на компютърните учебни речници.

TREFL – Translation REFerence Library, е компютърна програма за управление на едноезична и на паралелна текстова информация, приложима при работа с естествените езици, в частност в преводаческата дейност и в чуждоезиковото обучение. През последните години тя претърпя значително развитие, едновременно лингвистично и технологично (като често едното обуславя другото). Новите характеристики на наличните ресурси (засягащи количеството, качеството и приоритетността на данните) и подобрените технически показатели на търсачката (отнасящи се до бързодействието при изпълнение на заявките за търсене) дават основание за представяне на програмата в дълбочина, и по-конкретно на приложения модел за търсене и управление на данни (добавяне, приоритизиране, промяна и изтриване).

Програмата се основава на механизъм за разширено търсене в пълния текст чрез индексиране на данните. Въпреки че е създадена със сравнително прости програмни средства (скриптов език AHK), основните технически показатели на търсачката – бързодействие и количествени ограничения – са съизмерими с тези на най-известните бази от данни, например *SQLite*.

Нещо повече, сравнението по бързодействие, при преводна памет от близо милион записа, с експериментална версия на *TREFL* с вградена *SQLite* търсачка беше в полза на оригиналния механизъм за търсене. Освен това разпространяваната версия на *SQLite* поставя три съществени ограничения (по-точно затруднения, защото тези ограничения са по принцип преодолими, но с цената на допълнителен програмен код с големина и сложност, обезсмислящи използването на тази търсачка). Първото ограничение е свързано с търсенето без разграничение между малки и главни букви, което е възможно единствено при 26-те букви на латинската азбука. Второто ограничение на *SQLite* е невъзможността да се приложи директно модел за данните, позволяващ на администратора/потребителя да обособи сегменти в текста от една и съща колона, които да не се индексират. (Не трябва да се индексират например *HTML* етикетите.) Третото ограничение произтича от невъзможността да се реализират ефективно чрез наличната библиотека от функции на *SQLite* елементи на семантично търсене за

различни езици чрез морфологично и синонимно групиране на думи. Токенайзери за *SQLite* – програмни модули за генериране на граматически парадигми чрез морфологичен анализ и синтез на думи въз основа на правила – се създават трудно както на технологично, така и на лингвистично равнище (особено за определени езици), като същевременно не може да се очаква от тях гъвкавост и възможност за персонализиране от страна на потребителя.

В заключение един инструмент за бази данни с общо предназначение като *SQLite* по-трудно би отговорил на тези и бъдещи специфични изисквания на избрания по лексикографски, преводачески и дидактически съображения многоезичен модел за данните.

Независимо от високите технически показатели на търсачката на *TREFL*, ергономичността на програмата и успешното ѝ практическо приложение от дълги години на нея може да се гледа само като на прототип по следните две причини. Гъвкавостта на един скриптов език беше използвана за създаване на високоефективен алгоритъм за сложно търсене; следващ възможен етап е използването на скоростта на компилиран език. Друг възможен и особено желан етап е създаването на онлайн версия на *TREFL*. Това би позволило в частност да се създаде нов тип полезно взаимодействие посредством информационните технологии между преподавател и студент в областта както на дистанционното, така и на традиционното чуждоезиковото обучение. С оглед на тези перспективи и необходимостта от нови сътрудничества настоящото представяне е насочено към информатици. Това кратко и неформално описание е насочено едновременно и към филолози, в частност специалисти по чуждоезиково обучение, по лексикография и по превод, като основни потенциални потребители на програмата.

Кратките и неформални, но съдържателни описания на инструментите за бази данни са редки и непълни. Такова е описанието в статията „*Създаване и използване на индекс за подобряване на техническите показатели*“ на *Microsoft Access 2010*[1]. Трите точки в общия преглед от статията ще послужат тук като основа и ще бъдат доразвити с оглед на специфичните решения в *TREFL*.

В един текст, който е продукт на речева дейност и се разглежда като подредена последователност от думи, подредбата е лингвистично детерминирана и следователно формално непредсказуема (за разлика от заглавните думи, подредени по азбучен ред в един традиционен речник). Процесът на търсене на елементи в такова неструктури-

[1] http://office.microsoft.com/bg-bg/access-help/HA010341594.aspx

рано множество може да се ускори значително, ако, вместо да „сканираме“ данните, съпоставяйки съдържанието на всеки прочетен блок информация с това на запитването, използваме допълнителна структурирана информация, позволяваща да се определят адресите на търсените елементи. Именно такава допълнителна структурирана информация представляват индексите, които по същество не се различават от индексите в книгите.

Показателна аналогия за действието на една компютърна търсачка, използваща индекси, е решението на следната практическа задача: как да намерим страниците в една книга, всяка една от които съдържа всички имена на определени цитирани автори? Прочитането на цялата книга страница по страница съответства на „сканиране“ на данните по блокове. По-ефективно е използването на индекс на имената на цитираните автори, т.е. азбучен списък от имена, всяко от които е придружено от съответните номера на страници (т.е. адреси). Решението се свежда до намиране на общите адреси, т.е. на сечението на множествата от адреси.

Индексирайки всички думи, този метод се прилага успешно за бързо извличане на информация, въпреки че размерът на един такъв индекс е съизмерим с размера на индексираните данни. Това е така, защото възможността да се структурира информацията в индекса (по аналогия с подреждането по азбучен ред на индекса в една книга) позволява да се приложат високоефективни алгоритми за търсене. Бързодействието се компрометира обаче, когато при индексиране на данните се получават изключително големи множества от адреси, какъвто е случаят със служебните думи (предлози, съюзи, частици, и други). Показателна за това е еволюцията на търсачките. Първоначалното решение е компромисно и се състои просто в игнориране на тези думи. Това означава, че при индексиране на текст те имат на практика същия статут като пунктуацията. Намирането на оптимизационни решения за подобряване на техническите показатели на търсачките позволява на един следващ етап тези често срещани думи със сравнително ниска информационна стойност да се третират като равностойни на пълнозначните думи. Сега ставаме свидетели на трети етап в усъвършенстването на търсачките с постепенното реализиране на елементи на семантично търсене, в частност възможността една ключова думи да представлява не просто конкретна словоформа, а граматическа, синонимна или друга парадигма[2]. Ще оценим много-

[2] Дефинирането на списъци от думи (граматически или синонимни парадигми), позволяващи да се гледа на една ключова дума от заявката за търсене като пред-

кратно увеличената работа, която извършва търсачката при парадигматично търсене (и съответно нейната ефективност), ако разгледаме една реална заявка за търсене на три ключови думи, всяка от които е представител на парадигма от десет словоформи. Възможните комбинации от три словоформи, всяка от които принадлежи на различна парадигма, са 10 x 10 x 10 = 1000. Следователно една заявка за семантичното търсене на три ключови думи е еквивалентна на 1000 заявки за търсене на три конкретни словоформи. На практика семантичното търсене увеличава многократно продуктивността на записите, по аналогия със записите за една система за машинен превод, дефиниращи трансферни синтактични правила с помощта на регулярни изрази.

През същите тези три етапа премина и развитието на *TREFL*. По-долу са представени моделът на данните, критериите за приоритизиране на резултатите, възможностите за разширено търсене и технологичните решения, които обуславят бързодействието, надеждността и ергономичността на програмата.

Всеки запис (едноезичен или двуезичен) се състои от две части – изходен и целеви сегмент. Целевият сегмент при едноезичен запис може да бъде речникова или енциклопедична статия, лингвистичен или енциклопедичен коментар, референции за източника (с възможна хипервръзка, когато източникът е онлайн документ), както и произволна друга информация; полето за целевия сегмент може и да е празно. Големината и на двата сегмента – изходен и целеви – е неограничена по обхват. Възможностите за форматиране съвпадат с тези на HTML документите. Търсените езикови елементи – думи, словоформи, точни изрази, точни изрази с точните словоформи – се осветяват в резултата. В таблицата с данните се индексира единствено първата колона, съдържаща изходните сегменти. Освен възможностите за скрити коментари, предоставени от HTML езика, потребителят може да постави в средни скоби части от текста на изходните сегменти, които се показват, но не се индексират, например: *string [sb] up*. Ако вместо средни скоби се използва друг контролен символ (вж. до-

ставител на съответната парадигма, се смята само за първа (и незначителна) стъпка към същинското семантичното търсене: „…more relevant search results which are based on the semantics and meaning of the query, and not dependent upon preset keyword groupings“ (http://www.searchenginejournal.com/semantic-search-engines/9832/). Това твърдение е принципно некоректно, защото в често срещания случай, когато заявката за търсене се състои от малко на брой ключови думи, в граничния случай само една, всяка семантична интерпретация, която не се основава на съществуващи списъци от думи, дефиниращи семантични полета, би била твърде произволна.

кументацията на програмата), неиндексираните думи не се обособяват зрително като такива. Този трети начин за избягване на индексирането е подходящ например при собствени имена като подчертаните в следната преводна единица: *They say in death, all things become clear. Tokugen Numataka now knew it was true. ≈ Казват, че смъртта изяснява всичко. Токуген Нуматака вече знаеше, че това е самата истина*[3].

Резултатите се подреждат автоматично по нарастваща дължина на изходните сегменти (измервана с броя на индексираните думи), но оценката по важност на всеки запис може да се контролира. В един изходен сегмент например, който съдържа термин, придружен от лингвистична информация за думата и прагматична информация за понятието, може да се индексира само заглавната дума. По този начин, когато се търси този термин (или семантично асоциирани с него словоформи), тази преводна единица (в случая речникова), ще се извежда на екрана като резултат с максимален приоритет.

Възможните режими на търсене в *TREFL* могат да се систематизират в скала по степен на размитост на търсенето.

1. Търсене на думи. (В този режим на търсене всяка словоформа в заявката се разглежда като представител на група от думи, например граматическа и/или синонимна парадигма.)
2. Търсене на словоформи.
3. Търсене на секвенция от думи, т.е. израз. (Като режим 1, но думите се търсят в дадената последователност.)
4. Търсене на секвенция от словоформи, т.е. точния символен низ. (Като режим 2, но словоформите се търсят в дадената последователност.)
5. Търсене на секвенция от словоформи, съвпадаща по дължина с изходен сегмент. (Прилага се ексклузивно при редактиране на запис за избягване на дубликатите в данните, а с оглед на критерия за сортиране на резултатите режим 5 се съдържа фактически като приоритетен в режим 4. При импортиране на данни има опция за допускане или изключване на дубликати на изходните сегменти.)

Специфичните оптимизационните решения, които допринасят за бързодействието на програмата, са следните:

1. Адресите за всяка дума се записват в отделен файл. Тази многофайлова организация на индекса, която се противопоставя например на базата данни *SQLite*, където данните и индексите са организирани в един-единствен файл, позволява: 1) бърза синхронизация

[3] Dan Brown – Digital Fortress, 1998. Превод от английски: Иван Златарски, 2004. <http://chitanka.info/text/8807/130>

при създаване на резервни копия, тъй като се синхронизират само данните и евентуално само малка част от файловете на индекса, където са отразени промените; 2) бързо редактиране на записите, тъй като се променя единствено малка част от файловете, където са отразени промените; 3) бързо прочитане на адресите, защото се ползват единствено мощните ресурси на самата операционна система.

2. Адресите за всяка дума се групират в индекса по броя на думите в изходния сегмент за всеки запис. Това позволява да се работи с малки и малко на брой подмножества и да се ускорят значително операциите при намиране на сеченията на такива подмножества. (При търсене на един изходен сегмент от 30 думи например с цел редактиране на записа интерес ще представляват единствено подмножествата от адреси на сегменти от 30 думи.)

3. При намиране на сеченията на няколко такива подмножества думите се класират по брой адреси за всеки брой думи. Това позволява при търсене на няколко думи, част от които са пълнозначни, а другата част – служебни, да се търсят първо сеченията на подмножествата на пълнозначните думи, които от своя страна са най-често с празни или сравнително малки множества, като с това се спестяват многобройни операции с дългите списъци от адреси за служебните думи.

4. Операциите на търсене съответстват по брой на ограничението по брой (избран от потребителя) на извежданите на екрана резултати. С други думи, ако изберем разумно да покажем първите 10 най-релевантни от възможните 100 000 резултата (например при търсене на служебни думи), това ще спести, грубо казано, 99,99% от времето за обработка на заявката за търсене. (Аналогична зависимост може да се констатира, разбира се, и при интернет търсачките.)

В посочения документ за *Microsoft Access* се прави констатацията, че „индексите забавят променянето на вашите данни. Решението от точка 1 по-горе (многофайлова организация на индекса) минимизира този отрицателен ефект, който, заедно с необходимостта от по-голям капацитет на носителя на информация, е „цената“ на бързодействието при търсене. *SQLite* илюстрира противоположната крайност чрез решението да се обединят потребителските данни и индексът в един-единствен файл, което води и до количествени ограничения: „*Because SQLite puts everything into a single file (and thus, a single file system), very large data sets can stress the capability of the operating system or file system design*“ (Крейбих 2010: 14).

Експериментирането с различни схеми на индекса позволява да се направи една по-обща и по-любопитна констатация, макар и с по-

малка степен на достоверност: концептуални промени в съдържанието и структурата на индекса, които водят до допълнително ускоряване на търсенето, водят обикновено и до допълнително забавяне на променянето на основните данни. Ако индексът съдържа например данни за честотата на употреба на всяка дума, това ще ускори търсенето (точка 3 по-горе), но ще забави допълнително променянето на данните. При *TREFL* статистически данни не се записват, а се изчисляват при всяко търсене. Съществуват следователно резерви за ускоряване на търсенето, но това ускоряване би било за сметка на забавяне на променянето на данните. (Както най-често се случва при инженерните задачи, алгоритмични или други, всички решения са компромисни и зависят от конкретните практически условия и цели.) Тази зависимост е любопитна с оглед на аналогията, макар и повърхностна на този етап, която би могла да се направи между информатика и психолингвистика, свързвайки например понятията индексация и търсене съответно с изучаване и достъп до вътрешната лексика: по-продължителното изучаване на един език ускорява достъпа до вътрешната лексика.

Освен типични за търсачките характеристики *TREFL* притежава и функции, характерни за системите за преводна памет, а именно:

1. Импортиране и експортиране на данни в стандартния формат TMX с цел обмен на преводни памети между различни потребители и различни системи за преводна памет.

2. Автоматично (но по необходимост контролирано от човек) сегментиране на паралелни текстове на изреченско ниво, генериране на преводни единици чрез подравняване (т.е. поставяне на изреченията в съответствие) и в крайна сметка бързо създаване на големи и качествени преводни памети.

Благодарение на тези функции данните, с които работи *TREFL,* могат да обогатяват и персонализират чрез поглъщане и филтриране на съществуващите неизброими свободно достъпни електронни езикови ресурси. При определени условия могат да се използват индивидуално и текстове със запазени авторски права. Възможността за персонализиране и пълен контрол върху данните при работа с тази програма (за разлика от други локални търсачки или онлайн приложения като *linguee.com*, например) е особено важна, когато се импортират данни с ниска информативна стойност. Като се има предвид голямата редундантност на информацията в компилираните преводни памети, предоставени от Генералната дирекция „Писмени преводи“ на Европейската комисия, може да се твърди, че те са с безспорно ниска гло-

бална преводаческа стойност. Преводният сегмент *Член X, алинея Y*, например се повтаря стотици или дори хиляди пъти с различни числени стойности на *X* и *Y*. Следователно голямото количество на тези или други данни и престижът на издателя им не гарантират сами по себе си ефективното им практическо приложение в суров вид в програмите за управление на езикови ресурси, използвани за търсене на езикова, преводаческа или енциклопедична информация, в частност при четене, писане или превод.

Ръководство за работа с *TREFL* с описание на многобройните ѝ функции и хипервръзка за изтеглянето ѝ могат да се намерят на сайта на програмата[4].

Литература

Крейбих 2010: Kreibich, Jay A. *Using SQLite*, O'Reilly. o'reilly.com, 2010.

[4] http://web.uni-plovdiv.bg/rousni/