

Janusz S. Bień

# Język hebrajski w słowniku Lindego

## Analiza przypadku

16.03.2015, 10.10.2018

Tekst na otwartej licencji Creative Commons Uznanie Autorstwa, źródła dostępne w repozytorium <https://bitbucket.org/jsbien/ilindecsv>.

Indeksy zostały przygotowane w 2015 i w tym samym czasie opisane nieco zbyt skrótowo w niniejszym tekście. W 2018 r. dodano uzupełnienia i objaśnienia w formie przypisów lub uwag w inny sposób wyróżnionych typograficznie.

### 1. Wstęp

W pliku `1h.csv` znajduje się przykładowy indeks do korekty wyrazów pisanych alfabetem hebrajskim występujących w pierwszym tomie słownika. Może służyć do testowania narzędzi.<sup>1</sup>

W pliku `2h.csv` znajduje się przykładowy indeks do korekty wyrazów pisanych alfabetem hebrajskim występujących w pierwszym tomie słownika.

*Udostępnione są również robocze indeksy, w tym również dla innych tomów.*[2018]

Indeks można w zasadzie weryfikować i poprawiać za pomocą `djview4poliqarp`<sup>2</sup>.

Można wykorzystywać również odpowiedni edytor tekstowy, ale wymaga to nabrania wprawy w edytowaniu tekstów o mieszanym kierunku pisma.

Planowany tryb korekty w programie `djview4poliqarp` powinien być najlepsza metodą<sup>3</sup>.

Indeksy tego typu powinny być tworzone automatycznie. Opisana dalej procedura utworzenia tego indeksu ręcznie stanowi punkt wyjścia do sformułowania odpowiednio szczegółowego algorytmu<sup>4</sup>.

### 2. Wyszukiwanie słów w alfabecie hebrajskim

Wyszukiwanie słów było kłopotliwe, ponieważ wyszukiwarka działa obecnie tylko na starych skanach<sup>5</sup>. Punktem wyjścia była kwerenda `Syr`, trafienia za pomocą programu `djview4poliqarp` zostały zapisane w formie pliku `csv`. Z pliku tego usunięto niepotrzebne kolumny w celu nadania mu formy indeksu. Po utworzeniu indeksu w `djview4poliqarp` wszystkie hasła zostały przejrane. Hasła odnoszące się do przytoczeń w alfabecie hebrajskim zostały zmodyfikowane:

- dopasowane zaznaczenia do wyrazu w alfabecie hebrajskim,
- hasło zostało zastąpione numerem strony.

<sup>1</sup> Planowane narzędzie nie powstało.

<sup>2</sup> Patrz np. <https://www.slideshare.net/jsbien/jsb-i-linde181001ipi-117452985>

<sup>3</sup> Częściowo nieaktualny tekst *Tryb korekty w djview4poliqarp* z 2015 r. jest obecnie dostępny w repozytorium <https://bitbucket.org/jsbien/linde-info>. Proponowane tam rozszerzenia programu `djview4poliqarp` nie wydają się obecnie tak bardzo istotne.

<sup>4</sup> W aktualnej wersji `djview4poliqarp` tworzenie indeksów jest nieco łatwiejsze, swoją drogą zmianie uległy też priorytety.

<sup>5</sup> Nieaktualne. Stare skany to inaczej wersja 2010, patrz <https://szukajwsłownikach.uw.edu.pl/słownik-lindego/>.

Dodatkowo utworzono hasła dla innych wyrazów w alfabecie hebrajskim znajdującym się w sąsiedztwie.

W trakcie pracy ujawniła się wada programu `djview4poliqarp` polegająca na tym, że dla wyświetlonego hasła nie ma metody wyświetlenia odpowiedniej strony za pomocą `djview`, co pozwoliłoby zlokalizować stronę w strukturze słownika za pomocą „konspektu” (w „starych skanach” cały słownik stanowił formalnie jeden dokument)<sup>6</sup>. W konsekwencji numery tomów zostały ustalone przez posortowanie w Emacsie wynikowego pliku według URL (z niejasnych powodów dało to kilka błędów, które zostały poprawione ręcznie).

Na podstawie numeru tomu i numeru strony możliwe było zlokalizowanie szukanych słów w nowych skanach<sup>7</sup> (wygodnie było to robić równolegle z oglądaniem trafień na starych skanach).

Niestety w ten sposób udało się znaleźć tylko niewielką część słów pisanych tym alfabetem.<sup>8</sup>

### 3. Lokalizowanie i wycinanie słów w alfabecie hebrajskim

Początkowo program `djview4poliqarp` nie pozwalał indeksować dowolnego dokumentu DjVu<sup>9</sup>. Najpierw został stworzony „ślepy” indeks zawierający w charakterze hasła tylko numer tomu i numer strony (w razie potrzeby powtórzone odpowiednią liczbę razy), a potem za pomocą `djview` były tworzone URL dopisywane do pliku. Okazało się jednak to niewygodne, dla drugiego tomu utworzono ślepy indeks zawierający bezpośredni kontekst w komentarzu.

W ogólnym wypadku wycinki powinny być robione automatycznie na podstawie URL<sup>10</sup>, i pobierane od razu z binarnej maski<sup>11</sup>.

Obecnie konieczne było stosowanie funkcji programu `djview` zapisywania zaznaczenia w pliku graficznym — stosowany był format `png` (bez konkretnego powodu)<sup>12</sup>. Niestety wydaje się, że tak utworzone pliki mają rozdzielczość zależną od jakichś przypadkowych czynników, nie stanowią więc rzeczywistych wycinków oryginału.

### 4. OCR słów w języku hebrajskim

Wybrałem najprostszą, choć może nie najlepszą drogę — uzyskane wycinki wykorzystałem jako dane do FineReadera. Większość liter FineReader — jak się wydaje — rozpoznał poprawnie. Kilka razy niewątpliwie się pomylił. Kilka razy źle określił gabaryty słów, po ich ręcznej modyfikacji rozpoznał słowa lepiej (a w każdym razie inaczej). Niektórych słów nie rozpoznał w ogóle - były to słowa nadmiernie powiększone przy robieniu wycinków.

Wyniki zostały zapisane w formacie PDF i w formie czystego tekstu. PDF został skonwertowany do DjVu, aby mieć jednocześnie dostęp do skanu i tekstu. Wydaje się jednak, że `pdf2djvu` zmienił kolejność znaków hebrajskich.

Dla tomu drugiego zapisano wyniki w formie czystego tekstu i tam kolejność znaków była prawidłowa.

Obecnie można rozważyć wykorzystanie Tesseracta. [2018]

<sup>6</sup> Wada ta została usunięta dopiero 2 2018 r. (wersja 3 programu).

<sup>7</sup> Chodzi o skany w wyższej rozdzielczości i w skali szarości, a nie czarno białe, jak wersja 2010. Jest to tzw. wersja 2016, patrz <https://szukajwsloownikach.uw.edu.pl/slownik-lindego-nowy/>.

<sup>8</sup> Stwierdzenie niejasne, do zweryfikowania przez powtórzenie wyszukiwania bezpośrednio na nowych skanach.

<sup>9</sup> Niejasne jest użycie czasu przeszłego, do ewentualnej weryfikacji w historii programu.

<sup>10</sup> Patrz <https://sourceforge.net/p/djvu/feature-requests/95/>.

<sup>11</sup> Ma to sens tylko wtedy, gdy mamy pewność, że binaryzacja jest poprawna

<sup>12</sup> Funkcję taką ma również `djview4poliqarp`, widocznie nie był stosowany ze względu na problem wspomniany wcześniej/

## 5. Nanoszenie wyników OCR na indeks

Zadanie to powinno być oczywiście wykonywane automatycznie, robienie tego ręcznie okazało się tak niewdzięczne, że próbka została ograniczona tylko do pierwszych dwóch tomów<sup>13</sup>.

Istotnym problemem jest kierunek pisma:

- na jakim etapie przetwarzania zmieniała się kolejność liter — sprawa ta wymaga wyjaśnienia,
- na litery hebrajskie Emacs reaguje przejściem w tryb pisania od prawe do lewej, co jest mocno dezorientujące i sprzyja pomyłkom.

Kształt znaków hebrajskich w domyślnym foncie (DejavuSans?) różni się znacznie od ich kształtów w słowniku Lindego. Wskazane jest używanie fontu bardziej przypominającego oryginalny (Cardo?, Linux Libertine?). Jednak obecnie indeks w `djview4poliqarp` wyświetlany jest za pomocą fontu systemowego.

## 6. Zakończenie

Ręczne przygotowanie indeksu okazało się trudniejsze, niż się wydawało.

Trudność brała się z nieznaności alfabetu hebrajskiego i konieczności wykorzystywania automatycznego rozpoznawania znaków, oraz z braku wprawy w edycji tekstów o dwóch kierunkach pisma. Całkowicie ręczne wpisywanie haseł do indeksu w programie `djview4poliqarp` byłoby chyba znacznie łatwiejsze.

---

<sup>13</sup> Dla pozostałych tomów dostępne są tylko indeksy z lokalizacją, ale bez treści haseł.