INVESTIGATIONES LINGUISTICAE VOL. XXVI, 2012

AL. NIEPODLEGŁOŚCI 4, 60-874, POZNAŃ – POLAND

Współczesne trendy w lingwistyce komputerowej a problem automatycznego tłumaczenia języka arabskiego

Current trends in computer linguistics and problem of the machine translation of Arabic

Jerzy Łacina

ZAKŁAD ARABISTYKI I ISLAMISTYKI UAM
AL. NIEPODLEGŁOŚCI 24, 61-714 POZNAŃ

jlacinar@wp.pl

Abstract

The aim of this paper is to present some problems concerning the machine translation of Arabic in the context of the chosen NLP theories and their evolution. First attempts of electronic machine translations in Europe started only a little more than fifty years ago. It is enough time to perceive some aspects of the evolution? Although a lot of the concepts are still valid, the situation in A.D. 2012 is quite different than even twelve years ago. We still see useful old works of N. Chomsky, D. Cohen, but CFG seems to be supported with some new theories which also have got some disadvantages. Some interesting problems occur in the process of automatic translation when the number of grammatical cases is smaller in the source language than that in the output language.

1 Wstęp

Historia eksperymentów z tłumaczeniem automatycznym języka naturalnego przy użyciu komputera sięga roku 1948, kiedy próby takie zostały podjęte w Wielkiej Brytanii i USA. W 1954 rozpoczęto prace nad tym w ZSRR, w 1959 roku we Włoszech . a pod koniec roku 1959 we Francji, gdzie utworzono l'ATALA (l'Association pour l'étude et le développement de la Traduction Automatique et de la Linguistique Appliquée), w 1959 r. CETA (Centre d'études pour la Traduction Automatique), w grudniu 1959 roku powstał instytut Blaise Pascala (l'Institut Blaise Pascal) składający się z dwóch sekcji: paryskiej, kierowanej przez Aimé Sestier i w Grenoble pod kierunkiem Bernarda Vauquois, W maju 1960 powstał trzeci ośrodek w Nancy.

Trudno dziś uwierzyć, że jeszcze w 1954 roku nie było we Francji ani jednego komputera. W Wielkiej Brytanii były w tym czasie dwa, a w Niemczech jeden (Léon 1998).
Jednym z prekursorów analizy automatycznej języka arabskiego był David Cohen, który w 1961 r. przedstawił swój „essai d'une analyse automatique de l'arabe". W chwili gdy piszę te słowa, przygotowywana jest do druku 300 stronicowa książka A. Jaccarini *Approche algorithmique de la grammaire arabe*. Francja rozpoczynając z pewnym opóźnieniem, działalność na polu tłumaczeń komputerowych, dzisiaj zajmuje czołowe miejsce w dziedzinie komputerowej analizy języka arabskiego. Jest to pewnym pocieszeniem dla arabistyki polskiej, która jeszcze nawet nie raczkuje w tym kierunku. Podjęte przeze mnie w latach dziewięćdziesiątych ubiegłego wieku próby stworzenia analizatora i generatora arabskich czasowników zaowocowały w 1994 r. dwoma programami komputerowymi: zespolonym z modułem tłumaczącym analizatorem *Muhallil* i generatorem *Musarrif*. Dzisiaj niczym specjalnym te aplikacje nie zadziwiają, w tamtych latach jednakże mało było na świecie ośrodków mogących się pochwalić gotowymi programami tej jakości. Przyszłość analizy komputerowej języka arabskiego w Polsce dopiero przed nami, Mam nadzieję, że artykuł niniejszy będzie pewnym przyczynkiem do przyspieszenia kroków i zintensyfikowania działalności w tym kierunku i Polska, podobnie jak Francja, startując z pewnym opóźnieniem, znajdzie się pewnego dnia tak jak ona w ścisłej czołówce światowej. W artykule przedstawiam wybrane aspekty związane z problematyką tłumaczenia komputerowego w aspekcie ogólnym i uszczegółowionym do języka arabskiego. Połączenie tych dwóch aspektów: ogólnego i szczegółowego tworzy poziom meta, który również jest przy okazji swoistego rodzaju wynikiem badawczym, odrębnym od obydwu aspektów: ogólnego i szczegółowego analizy języka naturalnego.

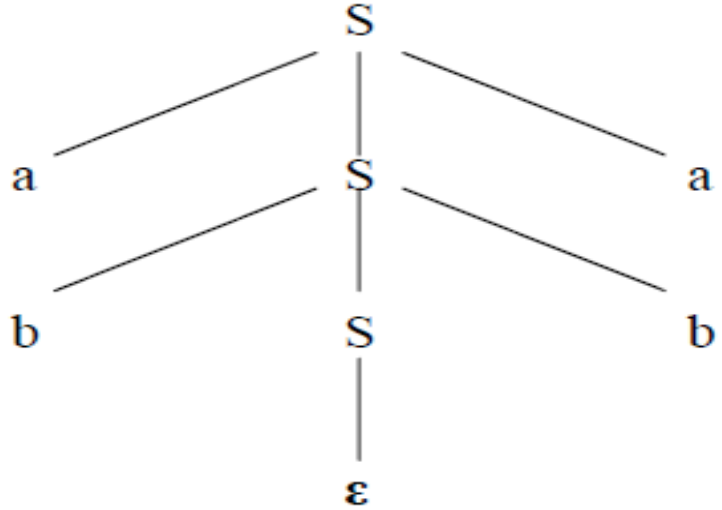
1.1 Czy każda relacja jest rodzajem tłumaczenia?

W roku 2000 Pineda i Garza przedstawili relację między językiem naturalnym i terminami graficznymi jako podobną do tej, która zachodzi między tłumaczeniem jednego języka naturalnego na drugi. Poddano pod dyskusję relacje między reprezentacjami multimodalnymi i odniesieniami przestrzennymi z jednej strony, oraz myśleniem multimodalnym i inferencją deiktyczną z drugiej Na tym etapie nie udało się ostatecznie rozwiązać tego problemu, zwrócono jednak uwagę na to, że ma on charakter semiotyczny (Pineda Luis i Gabriela Garza 2000).
Marchand i Damper rozwijając problematykę *PbA* (Pronunciation by analogy), dostrzegli problem translatorski na poziomie fonologicznym. PbA zostało użyte to rozwiązania trzech trudności związanych z mapowaniem ważnych łańcuchów języka mówionego: litery na jej tłumaczenie fonemiczne, fonemu na literę i litery na konwersję akcentu. Analiza okazała się efektywna, błędów było jednak sporo. Najwięcej

kłopotów sprawiły znaki samogłoskowe, dlatego Marchand i |Damper uczciwie przyznali, że znalezienie odpowiednich algorytmów dla przetłumaczenia ich na wartość dźwiękową pozostawiają przyszłości (Marchand i Damper 2000). Zaznaczając istnienie powyższego problemu, w naszych dalszych rozważaniach skupimy się jednak na tym, co dzieje się na poziomach morfologii i składni.

2 Gramatyka palindromiczna i drzewo parsowania

Idea wykorzystania systemu komputerowego do analizy języków jest prawie tak stara jak idea systemów komputerowych. W 1949 roku Warren Weaver wyróżnił trzy poziomy problemów w komunikacji:
1) Precyzja transmisji symboli komunikacyjnych (problem techniczny)
2) Zgodność transmitowanych symboli z oczekiwanym znaczeniem (problem semantyczny)
3) Efektywność otrzymanego znaczenia w odniesieniu do pożądanej reakcji (problem pragmatyczny). (Weaver 1949).
W roku 1988 Brown zasugerował, że możliwe jest skonstruowanie automatycznego systemu tłumaczeniowego (Brown i in, 1988). Zaproponowano techniki tworzenia takich systemów przy wykorzystaniu procesu uczenia maszynowego. Model Browna wykorzystywał modele statystyczne do ustalenia translacyjnej ekwiwalencji, przy czym ekwiwalencja była relacją między dwoma wyrażeniami o tym samym znaczeniu, przynależącymi jednak do różnych języków.
Zdaniem Nederhofa (2000), gramatyka nie generuje regularnego języka i wynika to z następującej definicji Chomskyego:



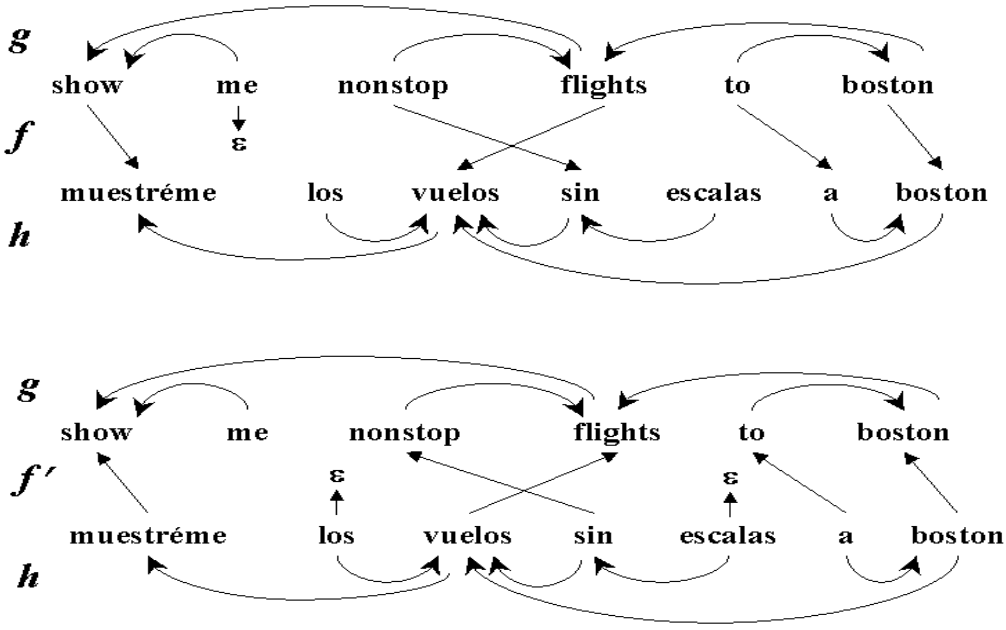
S ☐ a S a
S ☐ b S b
S ☐ **ε**

Rys. 1. Gramatyka palindromiczna i drzewo parsowania (Nederhof 2000)

Gramatyka jest samozanurzona[1] jeżeli istnieje **A** spełniające warunek: **A** ∈ **N**, przy czym **A** □ * **αAβ**, gdzie **α ≠ ε** oraz **β ≠ ε.**
Jeśli gramatyka nie jest samozanurzona, nie pojawią się symbole gramatyczne po lewej i po prawej stronie spinu drzewa parsowania. Nie jest wówczas możliwa nieograniczona łączność między obu stronami spinu drzewa np. w gramatyce palindromicznej.
(Spinem jest tu pionowa ścieżka parsowania)[2].

3 Model generatywnie probabilistyczny

Kiedy mamy do czynienia z translacją, modele mogą być generatywnie probabilistyczne. Eksperymentalne wyniki zastosowania takich modeli użytych do tłumaczenia z angielskiego na hiszpański i japoński przedstawił w 2000 r. Alshawi. Zależności są traktowane jako hierarchiczne, a derywacja jest probabilistyczna i generuje drzewka zależności jako pary. Para taka to np. hierarchicznie zsynchronizowane dwa łańcuchy (*strings*). Mają zastosowanie cztery podstawowe funkcje. Pierwsze dwie przyporządkowują wyrazy źródłowe wyrazom docelowym (wliczając w to łańcuchy puste), jest to odwrotne mapowanie.
Dwie pozostałe funkcje przyporządkowują wyrazy zależne wyrazom głównym.



Rys. 2. Przyporządkowanie hierarchiczne (Alshawi i in. 2000).

Alshawi wykorzystał do tego celu zasady działania maszyny skończenie stanowej, jednakże w jego modelu możliwe było wykorzystanie ruchu

1) Angielski termin: self-embedded constructions został przetłumaczony na: konstrukcje samozanurzone w: Noam Chomsky.1982. Zagadnienia teorii składni.Wrocław-Warszawa-Kraków-Gdańsk-Łódź: Zakład Narodowy im. Ossolińskich Wydawnictwo.

2 N.Chomsky przedstawił w roku 1959 gramatykę regularną G jako taką, która posiada jedynie reguły A □ a oraz A □ BC, gdzie B ≠ C, i jeśli A □ φ1Bφ2 oraz A □ ψ1Bψ2 są regułami G, wówczas φi = ψi (gdzie i =1,2) (Chomsky 1959).

wydłużonego i zmniejszeniu liczby stanów w stosunku do innych maszyn skończenie stanowych, gdyż głowica mogła rozpoczynać swój ruch od miejsca oznaczonego odpowiednim symbolem.
Podczas procesu tłumaczenia maszyny wykorzystywały łańcuchy wejściowe i wyjściowe, jako sekwencje zależności odpowiadających sobie wyrazów w językach: źródłowym i docelowym. Rozpoznawane były również łańcuchy puste, co jednak przyczyniało się do uproszczeń modelowych również w drzewie języka źródłowego. Alshawi zdawał sobie doskonale sprawę z wielu niedoskonałości swojego modelu, co oczywiście nie umniejsza zalet.
Rubinoff, przedstawiając generator IGEN wskazuje na trudności, które pojawiają się przy tłumaczeniu automatycznym dosłownym z francuskiego na angielski, np. francuskiemu czasownikowi *faire* odpowiadały angielskie: *make / do*. W ten sposób francuskie zdanie: *Le temps fait chaud* było tłumaczone na: *the weather makes / does warm*. (Rubinoff 2000)[3].

4 Model Melameda

Wg Melameda najlepsze modele tłumaczeniowe, to takie, których parametry odpowiadają najlepiej źródłom przy różnorodności danych, modele probabilistyczne, których parametry odzwierciedlają powszechne translacyjne równoważności i/lub odzwierciedlają istniejącą wiedzę o poszczególnych językach i parach języków, a także korzystają z najlepszych doświadczeń tradycji empirycznych i racjonalnych.
Melamed przedstawił modele, wraz z metodami efektywnej oceny ich parametrów oparte na podstawowych trzech założeniach:

1. Większość tokenów przekłada się tylko na jeden token.
2. Segment tekstu przeważnie nie jest tłumaczony dosłownie.
3. Różne obiekty językowe statystycznie różnią się swoim zachowaniem w procesie tłumaczenia.

Zdaniem Melameda te trzy metody znacząco udoskonaliły dokładność tłumaczenia automatycznego (Melamed 2000).
Również w 2000 roku zespół pod przewodnictwem Andreasa Stolcke przedstawił koncepcję analizy dialogu mówionego gdzie wprowadzono pojęcie *aktów dialogu dialog acts*- (DA). Wykorzystano podczas analizy tagowanie elementów dialogu mówionego[4]. Tagi okazały się bardzo przydatne, gdyż agent konwersacji powinien wiedzieć czy zadano mu

3 Podczas opracowywania analizatora arabskich czasowników Muhallil, z wbudowaną opcją tłumaczenia na angielski, natrafiałem na trudności związane z tym na jaki czas, tryb, liczbę i rodzaj angielski oddać arabskie osobowe formy perfektywne i imperfektywne. (ŁACINA, Jerzy. 1994. Muhallil – komputerowy analizator czasowników arabskich. http://www.staff.amu.edu.pl/~lacina/page4.html)

4 Ogólny podział algorytmów tagowania wygląda następująco (wg Dębowski 2001):

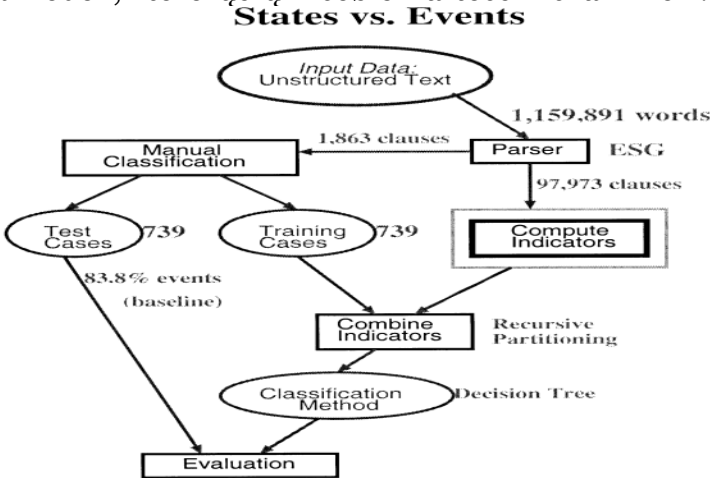
- oparte na modelach probabilistycznych (na niestacjonarnych modelach Markowa)
  - oparte na ukrytych modelach Markowa
  - oparte na modelowaniu maksimum entropii
- _ nieoparte na modelach probabilistycznych (jawnie)
  - oparte na pamięci
  - oparte na transformacjach

Algorytmy tagerów opartych na transformacjach posiłkują się wnioskowaniem ilościowym na podstawie danych wyłącznie w czasie uczenia się a algorytmy tagerów oparte na modelach probabilistycznych i na pamięci zarówno w czasie uczenia się i tagowania nowych tekstów.

pytanie czy polecono coś wykonać, a np. sumaryzator potrzebuje zachować ścieżkę kto do kogo mówi. Tagi okazały się całkiem skuteczne przy ustaleniu „głębokiego sensu" dialogu.
Daniel Marcu zauważył, że spójne teksty nie są tylko proste sekwencje fraz i zdań lecz raczej skomplikowane artefakty o bardzo rozwiniętej strukturze retorycznej i bada w jakim stopniu dobrze uformowane retoryczne struktury mogą być automatycznie uzyskane za pomocą algorytmów przygotowanych do analizy form powierzchniowych (Marcu 2000).
Marcu optymistycznie przedstawił swoje podejście, jednakże również realistycznie stwierdza: „how much of an impact the rhetorical parser presented here can have on solving these problems, of course, remains an empirical question". Wydaje się, że nie może być inaczej, jeśli do retoryki mielibyśmy podchodzić używając tylko algorytmów przygotowanych do analizy form powierzchniowych.

5 Klasyfikacja aspektowa

Klasyfikacja aspektowa odwzorowuje czasowniki na mały zbór cech pierwotnych aby wyrazić czas. Klasyfikacja jest niezbędna do zinterpretowania modyfikatorów czasowych (*temporal modifiers*) (Siegel i McKeown 2000).
Aspekt czasownika można przewidzieć na podstawie częstotliwości współwystępowania czasownika i niektórych modyfikatorów. Metody uczenia maszynowego zostały wykorzystane do automatycznego generowania modeli, które łączą w sobie wartości wskaźników.



Rys. 3. Model Siegela – McKeown (SIEGEL, Eric V. I Kathleen R. McKeown 2000).

Lingwiści w XX w. byli świadomi tego jak ważny jest czasownik (Dorr 1997) i że relacja: predykat-argument jest ważnym wyzwaniem stojącym przed lingwistyką komputerową, gdyż łączy się z wiedzą o pozycji i strukturze argumentów (Merlo i Stevenson 2001).

Stephen G. Pulman zajął się problemem rozpoznawania kontekstowego, dwukierunkowego. Uznał przy tym, że na wyjściu (output)

procesu tworzenia zdania gramatycznego mamy do czynienia z *formą quasi-logiczną* (QLF), przy czym jeśli gramatyka jest bardziej skomplikowana tym więcej napotkamy form QLF. Niezależne kontekstowo znaczenia zdań, do których się odnosimy są *rozwiązanymi formami logicznymi* (RLF) i są wyrażane przy pomocy naturalnych podzbiorów języka QLF. Im bardziej więc będą rozwiązane RLF, tym mniej pozostanie w nich QLF. Konteksty są modelowane przy pomocy zdań RLF (Pulman 2000).,

Rozwijając myśl Willego (1982), Bateman w 2001 r. idzie tropem analizy konceptualno-formalnej - *Formal Concept Analysis* (*FCA*). (Bateman i in. 2001) Metoda ta jest wykorzystywana przez matematykę stosowaną, która zakłada formalne ujmowanie znaczeń i hierarchii znaczeń, zezwalając jednocześnie na wykorzystywanie pojęć matematycznych do opisu i przetwarzania danych konceptualnych.

FCA zakłada istnienie kontekstu formalnego (***G, M, I***) reprezentującego zbiory danych, gdzie ***G*** jest zbiorem obiektów, ***M*** jest zbiorem atrybutów, a ***I*** tworzy binarną relację między tymi dwoma zbiorami. I(g,m) należy czytać: obiekt g posiada właściwość m, przy czym g ∈ **G** oraz m ∈ **M**. Bateman nawiązuje również do Manna (Mann i Thompson 1986), adaptując *RST* (rhetorical structure theory),

6 Model Hobbsa

W roku 1978 Hobbs opracował prosty algorytm, który okazał się skuteczny również przy rozpoznawaniu zaimków anaforycznych w tekście hiszpańskim. Skuteczność wyniosła 81,8% (Palomar i in. 2001). Algorytm analizuje drzewko w ustalonej kolejności i wyszukuje frazy nominalne w określonym rodzaju i liczbie. Hobbs testował algorytm na angielskich zaimkach: *he*, *she*, *it*, i *they*.

Podczas każdego tłumaczenia istnieje zagrożenie omijania pewnych niuansów znaczeniowych lub uwzględnienia jakiegoś teoretycznie możliwego, lecz niewłaściwego niuansu, dlatego dobre tłumaczenie wymaga wyrafinowanego procesu doboru leksykalnego w celu ustalenia, które z wyrazów bliskoznacznych w jednym języku są znaczeniowo najbliższe ich odpowiednikom najbardziej odpowiednie w konkretnej sytuacji wyrażonej w innym języku.

7 Synonimia bliska i absolutna

(Edmonds i Hirst 2002) wprowadzają pojęcie *synonimii bliskiej* (*near-synonymy*) oraz *synonimii absolutnej* (*absolute synonymy*), przy czym sami przyznają, że ta ostatnia jeśli w ogóle istnieje, jest bardzo rzadka, o czym wspominali zresztą wcześniej Quine (1951) i Goodman (1952). Z drugiej strony, jeśli chodzi o synonimię bliską, nawet dla rodzimych użytkowników języka, zwykle nie jest sprawą łatwą odnalezienie jednoznacznych reguł wykazujących istnienie różnic znaczeniowych w każdym z możliwych kontekstów (Edmonds i Hirst 2002).

Innym problemem jaki pojawia się przy tłumaczeniu komputerowym jest identyfikacja imion własnych (Mikheev 2002). Imiona te bowiem powinny być raczej transliterowane, aniżeli tłumaczone na drugi język, aby uniknąć np. sytuacji przetłumaczenia Jacka Londona na Jacka Londyna, nazwiska White na „Biały" itp. Z drugiej strony należałoby jednak znaleźć reguły odróżniania nazwiska od innej nazwy własnej, która ma swój odpowiednik w języku tłumaczenia, np. nazwę miasta London, nie tylko można, lecz wręcz należy tłumaczyć na Londyn, podczas gdy nazwisko London należałoby pozostawić w oryginalnej postaci.

8 Dyzambiguacja w analizie automatycznej

Problem *ujednoznacznienia* (*disambiguation*) był znany już w latach pięćdziesiątych. Młody wówczas doktor fizyki, specjalizujący się w promieniowaniu kosmicznym Victor Yngve, po zapoznaniu się z Claudem Shannonem i jego teorią komunikacji, zainteresował się możliwościami tłumaczenia automatycznego jednego języka na drugi. Właśnie będąc u Shannona w Bell Telephone Laboratories w roku 1952 dowiedział się, że w czerwcu owego roku miała być organizowana konferencja na temat tłumaczenia maszynowego. Dosyć szybko, bo już w roku 1953, po odejściu z MIT Bar Hillela, został mianowany przez Jerome'a Wiesnera szefem Research Laboratory for Electronics (RLE) (Hutchins 2012). I tak rozpoczęła się kariera jednego z największych autorytetów w dziedzinie lingwistyki komputerowej. W roku 1955 Yngve opublikował artykuł w którym poruszył chyba jako pierwszy problem ujednoznacznienia. (YNGVE 1955).

Stevenson i Wilks (2001) wyróżnili siedem etapów analizy:

1 Analiza wstępna
2 Wydzielenie części mowy
3 Optymalizacja definicji słownikowych
4 Preferencje selekcyjne
5. Kody podmiotowe (w nawiązaniu do Yarowsky 1992)
6. Ekstrakcja kollokacji
7. Kombinacja modułów ujednoznaczniających

Modele probabilistyczne, w większym lub mniejszym stopniu mogą generować nieprawidłowe wyniki. Nową próbę zminimalizowania ryzyka niesionego przez gramatyki probabilistyczne przedstawili ostatnio Cohen i Smith (2012). Ocena gramatyk probabilistycznych, w wielu przypadkach, zaczyna się od oceny maksymalnego prawdopodobieństwa (MLE). W ujęciu ogólnym gramatyka probabilistyczna (G, θ) określa prawdopodobieństwo warunkowe pojawienia się łańcucha x w kontekście gramatycznej derywacji z:

$$q(x, z \mid \theta, G) = \prod_{k=1}^{K} \prod_{i=1}^{N_k} \theta_{k,i}^{\psi_{k,i}(x,z)} = \exp \sum_{k=1}^{K} \sum_{i=1}^{N_k} \psi_{k,i}(x, z) \log \theta_{k,i}$$

gdzie ϕ k, i jest funkcją, która „oblicza" ile razy przy wartości k pojawiło się w derywacji zdarzenie i. θ oznacza szereg wielomianów K ($\theta_1, \ldots, \theta_K$)

którym przy wartości *k* odpowiada *Nk* zdarzeń przeciwnych. Dla θ k = (θ k,1, ..., θ *k,Nk*), dla każdego θ k,i zachodzi prawdopodobieństwo:

$$\forall k, \forall i, \qquad \theta_{k,i} \geq 0$$

$$\forall k, \qquad \sum_{i=1}^{N_k} \theta_{k,i} = 1$$

(Cohen i Smith 2012)

9 Model centrujący

Jako pierwsi wykorzystali model centrujący[5] Brennan, Walker-Friedman i Pollard (1987). Przyjmując za podstawę algorytm rozpoznania anafory (Grosz i Sidner 1986, Grosz, Joshi i Weinstein 1983). W modelu tym przyjęto następujące założenia:

1.Segment dyskursu składa się z sekwencji wypowiedzi, U_1,. . . , U_n.

2. Dla każdej wypowiedzi jest ustalana lista rankingowa (Cf) jednostek dyskursu.

3. Najwyżej oceniony element listy Cf nazwany jest centrum preferowanym (Cp).

4.Najwyżej oceniona jednostka listy Cf utworzonej dla sekwencji U_{i-1}, realizowanej w U_i jest wstecznie centrująca (Cb).

Istnieje kilka rodzajów przejść z jednego topika[6] wypowiedzi do drugiego w zależności od tego, czy Cb jest zachowana w dwóch kolejnych wypowiedziach U_{n-1} i U_n oraz czy Cb jest również Cp w U_n.

Z innym problemem spotykamy się w wypadku interpretacji i tłumaczenia konstrukcji rzeczownikowych, np. angielskie *satellite observation* może być rozumiane zarówno jako „obserwowanie satelity" jak i „obserwowanie przez satelitę". Ujednoznaczenie konstrukcji rzeczownikowych jest zatem ważne, jeśli nie konieczne w procesie tłumaczenia automatycznego (Lapata 2002). Ważne jest uwzględnienie kontekstu, zastosowanie może mieć również statystyka i prawdopodobieństwo pojawienia się konstrukcji w jakimś znaczeniu bardziej niż w innym.

10 TSG i STSG

Wiele metod mieści się obecnie w ramach (STSG - synchronous tree substitution grammars). W odróżnieniu od gramatyk bezkontekstowych (CFG), w gramatykach TSG (tree substitution grammars), każda reguła tworzy znacznie większy fragment drzewka. Zsynchronizowane TSG mogą generować fragmenty drzewek w języku źródłowym i docelowym równolegle, przy czym każda reguła generuje wówczas fragment drzewka w innym języku (Gildea 2012).

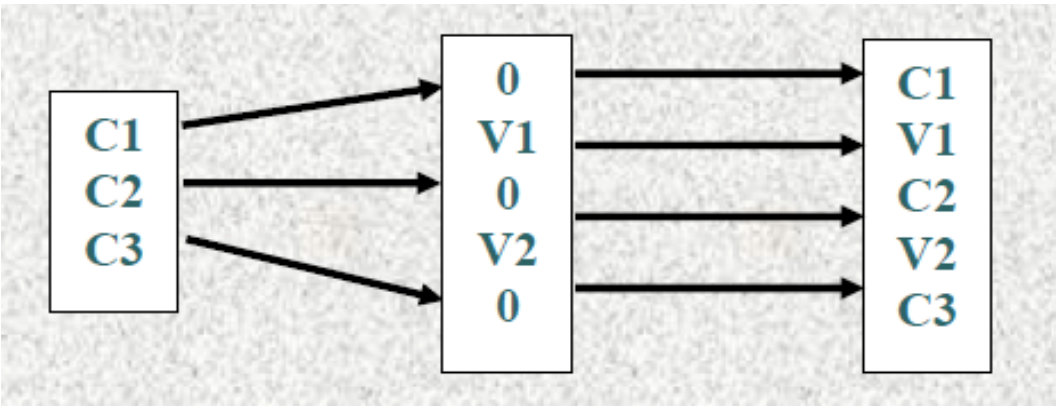
11 Analiza komputerowa języka arabskiego

5 centering model (wg Miltsakaki 2002).

6 Topik (topic) oznacza tutaj tę część wypowiedzi, na której koncentruje się uwaga – jednostkę centralną (central entity).

Analiza komputerowa języka arabskiego w świecie arabskim prowadzona jest głównie w krajach Maghrebu – Tunezji, Maroku i Algierii. Aktywność Egiptu na tym polu, aczkolwiek dość prężna, jak się wydaje - ma charakter bardziej komercyjny niż naukowy. Poza światem arabskim problematyką tą zajmują się naukowcy m.in. w USA, Francji, Wielkiej Brytanii i niektórych innych ośrodkach, w tym również w Pradze, Bergen, Nijmegen i in.
Poniżej przykład analizy języka arabskiego opracowany w Tunezji przez Munira Zrigui. Analizator ten ma również pewne możliwości tłumaczenia arabskiego tekstu na francuski. Możliwości jego analiza ograniczają się zasadniczo do morfologii z wykorzystaniem schematów tematycznych.

11.2 Charakterystyka arabskiego języka pisanego:



Rdzeń Schemat Wyraz

11.3 Przykład derywacji od rdzenia كتب

فعل	حمل	Notion de porter
فاعل	حَامِل	porteur
فعل	حَمَلَ	a porté
مفعل	مَحمَل	brancard
فعل	حُمِلَ	a été porté

11.4 Przykład derywacji od rdzenia حمل

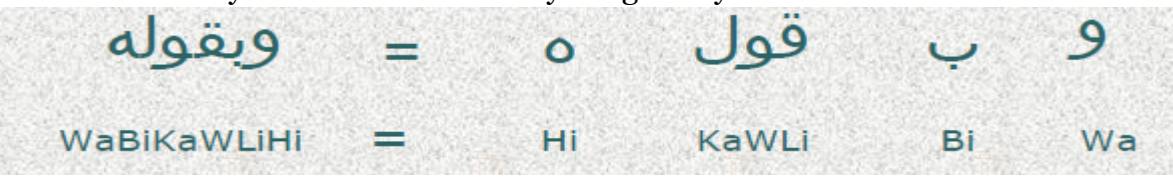
Strukturę czasownika arabskiego Zrigui przedstawia jako łańcuch składający się z pięciu ogniw7:

Enklityka	Sufiks	Korpus schematyczny	Prefiks	Proklityka
نَا	ونَ	تَذْكَّرُ	تَ	أ

11.5 Kierunek odczytu

Kierunek odczytu biegnie od strony prawej ku lewej lewej. Łańcuch składa się z następujących ogniw:

- Proklityka: partykuła pytajna أ
- Prefiks impf. ت
- Korpus schematyczny: تَذَكَّر derywowany od rdzenia : ذكر wg schematu . تَفَعَّل
- Sufiks: ونwerbalny, mn. m.
- Enklityka : نا zaimek osobowy sufigowany 1. os. mn.



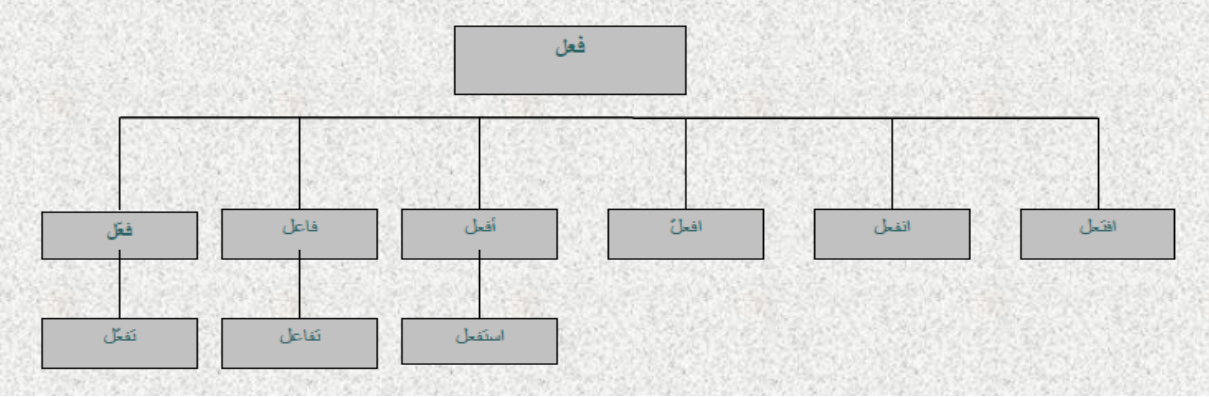
i wg jego słów = jego słowa wg i

Dla analizatora przedstawionego przez Zrigui, trudność stanowi brak wokalizacji arabskich wyrazów i ich wieloznaczność.

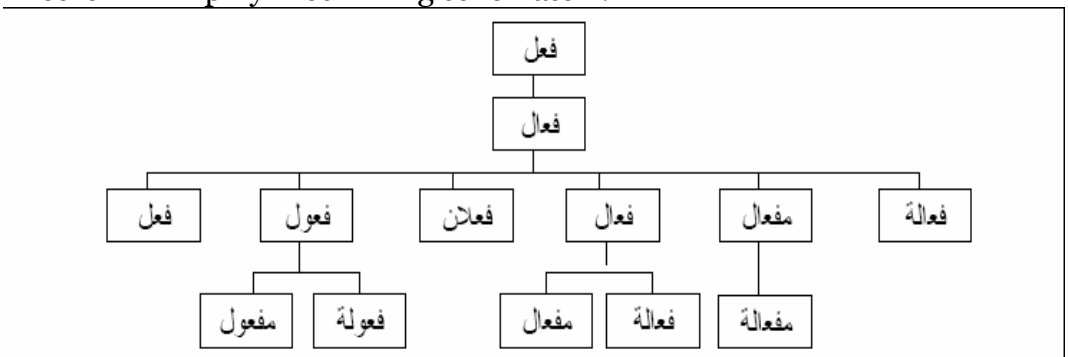
11.6 Podstawowe schematy analizy

7 Ogniw w rzeczywistości może być więcej niż pięć.

Czasowniki są analizowane wg podstawowych schematów



Rzeczowniki i przymiotniki wg schematów:



Jest to analizator dość prosty i pomimo pewnych dodatkowych algorytmów i ograniczeń, nie jest w stanie poradzić sobie z każdym tekstem arabskim, zwłaszcza że wbrew temu, co twierdzi Zrigui, ani kolejność ogniw struktury czasownika arabskiego nie musi być taka jak podaje wyżej, ani też liczba tych ogniw nie może być ograniczona tylko do pięciu. Jeśli pozostawimy pięć ogniw, a czasownik będzie w stronienie biernej, to podmiana nie może być tylko paradygmatyczna, wymaga ona również uwzględnienia syntaktycznej współzależności: prefiks-korpus schematyczny (ale z innymi ogniwami już nie). Schemat taki powinien zatem wyglądać tak:

5	4	3	2	1
Enklityka	Sufiks	Korpus schematyczny	Prefiks	Proklityka
نَا	ونَ	تَذَكَّرُ	تَـ	أ
		tadakkaru	**ta**	

gdzie istnieje zależność syntaktyczna o charakterze paradygmatycznym między ogniwami 2 i 3, tzn. jeśli podstawiany jest prefiks w pozycji 2, to podstawiony musi być również korpus schematyczny w pozycji 3. Przy czym podmiana paradygmatyczna nie musi wówczas następować w pozycjach 1, 4, 5. We wspomnianym wyżej wypadku schemat klasy piątej korpusu jest identyczny w stronie czynnej i biernej, natomiast prefiks

strony biernej różni się wokalizacją od prefiksu strony czynnej. Sytuacja może być odwrotna, np kiedy mamy do czynienia z czasownikiem w drugiej klasie tematycznej. Poniżej przykład słowoformy z tematem w klasie drugiej w stronie czynnej:

5	4	3	2	1
Enklityka	Sufiks	Korpus schematyczny	Prefiks	Proklityka
نَا	ونَ	ذَكَّرُ	تُ	أ
		dakkiru	**tu**	

i w stronie biernej:

5	4	3	2	1
Enklityka	Sufiks	Korpus schematyczny	Prefiks	Proklityka
	ونَ	ذَكَّرُ	تُ	أ
		dakkaru	**tu**	

W tym wypadku w obydwu stronach różnią się wokalizacją tematy, a nie ma różnicy między prefiksami.
Trzecia możliwość jest taka, że będzie różnica w stronie czynnej i biernej zarówno w wokalizacji tematów, jak i w wokalizacji prefiksów. Poniżej taki właśnie przykład w pierwszej klasie tematycznej czasownika arabskiego w stronie czynnej:

5	4	3	2	1
Enklityka	Sufiks	Korpus schematyczny	Prefiks	Proklityka
نَا	ونَ	ذَكُرُ	تَ	أ
		dkuru	**ta**	

i w stronie biernej:

5	4	3	2	1
Enklityka	Sufiks	Korpus schematyczny	Prefiks	Proklityka
	ونَ	ذَكَرُ	تُ	أ
		dkaru	**tu**	

Jak widać na powyższych przykładach w wypadku pojawienia się strony biernej ogniwo 5 z zasady powinno nosi wartość 0. Uwzględnienie tego faktu jest możliwe w procesie analizy, jednak Zrigui o tym nie wspomina.
Schemat Zrigui nie zadziała również, kiedy pojawia się element syntaktyczny nie mieszczący się w ramach pięciu ogniw, np. partykuła, bądź spójnik jednoliterowy, przykładem może być konstrukcja:

أَفَتَتَذَكَّرُونَنَا afatatad̲akkarūnanā: W takich wypadkach pięcioogniwowy schemat Zrigui już sobie nie radzi z napotkaną formą czasownikową. Najprawdopodobniej nie będzie sobie również mógł poradzić często z odróżnieniem form imiennych od czasownikowych, nie wspominając już o wieloznaczności wynikającej z braku wokalizacji.
Te problemy do dzisiaj napotykają znacznie bardziej rozwinięte systemy niż prosty system Zrigui.
Podczas automatycznej analizy, sprowadzenie formy tekstowej do jednoznacznej formy hasłowej odbywa się za pomocą *stemmera*. Stemmery bazują na regułach tworzonych ”ręcznie” albo wygenerowanych automatycznie. Jednak algorytmy stemmerów mogą podawać błędne odpowiedzi (Korzycki 2008). Bez odpowiednich taggerów, stemmer może np. poinformować nas np. że arabski czasownik VIII klasy tematycznej انتهى o rdzeniu نهى (nhy) jest czasownikiem klasy VII o rdzeniu تهى (thy), co jest teoretycznie również możliwe, w praktyce jednak mało prawdopodobne.
Pomimo zatem ogromnej regularności morfologicznej języka arabskiego, nie istnieje możliwość stworzenia działającego bezbłędnie analizatora, który nie uwzględniałby kontekstu, a czasami również prawdopodobieństwa wystąpienia danej konstrukcji.
Pomimo niezwykłej regularności arabskiej koniugacji i struktury morfologicznej arabskich wyrazów, konieczne jest uwzględnienie zależności wertykalnych na poziomie składni i tekstu oraz horyzontalnych między poszczególnymi wyrazami. W tym ostatnim wypadku potrzebny jest korpus słowoform arabskich[8].

11.7 DIINAR.1

DIINAR.1 (***DIctionnaire INformatisé de l'ARabe, version 1***)[9] to opracowany przy współudziale ośrodków w Lyonie i Tunisie zaawansowany słownik sprzężony z bazą danych arabskich słowoform, wspomagany analizatorem i generatorem form wyrazowych.
Słownik ten działa w oparciu o założenie, że słowoforma w języku arabskim to łańcuch znaków składający się formatywu jądrowego (*formant-noyau* - Fn) mogącego być rozszerzonym o formatywy zewnętrzne (*formant-extensions* - Fe), dodawane w lewo lub w prawo Fn. Formatywy aFe są formatywami przed Fn, formatywy pFe, to formatywy po Fn. Na arabską słowoformę składają się:
– proklityka (PCL), wliczając w to jednospółgłoskowe spójniki
– prefiks (PRF).

8 Korpus dostępny na Brigham Young University: (http://arabicorpus.byu.edu/) zawiera ponad 173 miliony nieotagowanych arabskich słowoform. Można też skorzystać z pomocy LDC w USA (Linguistic Data Consortium, University of Pennsylvania, 3600 Market Street, Suite 810, Philadelphia, PA 19108, USA), a w Europie ELRA (European Language Resources Association, 55, rue Brillat-Savarin – 75013 Paris, France) i projekt NEMLAR (Network for Euro-Mediterranean Language Resources - Center for Sprogteknologi (CST), Copenhagen)
9 Arabski akronim معالي ("معجم العربية العالي")

– temat (BAS) reprezentowany przez RDZEŃ (RAC) (trój- lub czteroradykałowy) oraz SCHEMAT (SCH) składający się ze spółgłosek i samogłosek występujących w temacie obok spółgłosek rdzennych. W ten sposób temat takabbar, 'wywyższać się', składa się z 3-spółgłoskowego RAC /k-b-r/ i SCH: /taR1aR2R2aR3/, gdzie R1, R2 and R3 są zmiennymi spółgłosek rdzennych k-t-b.
Rzeczowniki, które nie mogą być analizowane przez RAC i SCH są traktowane jako quasi-tematy (*pro-bases* - PBA), np. اسماعيل, 'Ismail', يونيسكو 'UNESCO', itd,
– sufiks (SUF) będący końcówką koniugacyjną, deklinacyjną, przyrostkiem żeńskim, itd. .;
– enklityka (ECL) rozumiana w zasadzie jako zaimek sufigowany (ABBÈS 2004).
Idea DIINAR wyraźnie nawiązuje do myśli Josepha Dichy (1997), który uprościł koncepcję Davida Cohena o *mot maximal – mot minimal* do postaci Fe – Fn – Fe i ustalił pięcioogniwową koncepcję słowoformy arabskiej, z której jak się wydaje, skorzystał później Zrigui. Ta koncepcja ma swoje wady, gdyż dobry stemmer analizując tekst arabski, zwłaszcza niewokalizowany, generuje dużą liczbę potencjalnych rdzeni, które albo nie mają zastosowania w konkretnym tekście, albo w ogóle nie występują w języku arabskim. To ostatnie zagrożenie można zredukować poprzez użycie taggera i konsultację ze słownikiem. Problem rdzeni i słowoform istniejących w języku, jednakże nie pasujących do kontekstu jest dużo trudniej rozwiązać. Na wspomnianym wyżej modelu Siegela – McKeown widać, że *parser* uwzględnia *manual classification*, co nas nie satysfakcjonuje i satysfakcjonować nie powinno. Stemmer opierając się na pięcioogniwowym schemacie arabskiej słowoformy jest skazany na problemy afiksami o wartości zerowej, np. prefiks o wartości zerowej napotkamy w takich formach jak وكتب wktb czy ومحمد wmHmd. Może być również i tak, że w niektórych słowoformach jest możliwe zarówno jego wystąpienie, jak i jego brak, np. ويبس wybs. Bez uwzględnienia składniowych relacji kontekstowych skuteczna analiza morfologiczna w tym systemie nie jest możliwa.

12 Transducery MBOT

Ostatnio Daniel Gildea (2012) przedstawił wyniki swoich badań nad o skutecznością transducerów MBOT (*multi bottom–up tree transducers*) wychodząc z założenia, że aczkolwiek transducery drzew są definiowane jako stosunki między drzewami, to przy tłumaczeniu maszynowym opartym na składni interesujące są przede wszystkim relacje zachodzące między łańcuchami którymi owocują drzewa otrzymywane na wejściu i wyjściu. Synchroniczne gramatyki STSG (*synchronous tree substitution grammars*), są często przyjmowane jako podstawa do badań nad statystycznym tłumaczeniem maszynowym opartym na składni, tym jednak co jest naprawdę ważne w tłumaczeniu jest nie tyle gramatyka, co

łańcuchy (*strings*) tekstu generowane przez drzewa gramatyk. Gramatyki TSG (*tree substitution grammars*) są dosyć wygodne, rozwijają możliwości gramatyk bezkontekstowych CFG (*context-free grammars*), gdyż są w stanie generować odpowiednio duży fragment drzewa. Synchroniczne gramatyki TSG (STSG) generują te fragmenty drzew równolegle w językach źródłowym i wyjściowym. Gramatyki STGS są zatem dość wygodne i mają sporo możliwości, mają jednak i tę wadę, że operacji STGS nie da się złożyć w większą całość. Złożone operacje STGS nie będą już działały jako gramatyka STGS (Gildea 2012). Przydatne w takim wypadku może być zastosowanie MBOT.

Zasadę działania MBOT można w skrócie opisać tak, że jest to a system (S, Σ, Δ, F, R), gdzie:

- S, Σ i Δ oznaczają odpowiednio: stany, symbole wejścia, symbole wyjścia.
- $F \subset S$ jest zbiorem stanów akceptacji
- R jest skończonym zbiorem reguł $l \rightarrow r$ gdzie, korzystając ze zbioru zmiennych X,

$l \in T_\Sigma(S(X))$ i $r \in S(T_\Delta(X))$ tak że:

- każdy $x \in X$ występujący w l występuje dokładnie raz w r i odwrotnie

oraz:

- $l \square S(X)$ lub $r \square S(X)$.

Jeden krok tranducera MBOT dokonywany jest poprzez przepisanie lokalnego fragmentu drzewa zgodnie z zasadami ustalonymi w R. Fragment l zastępowany jest przez r, poddrzewo (*subtree*) dla każdej zmiennej l jest kopiowane do odpowiadającego mu miejsca zmiennej w r. Reguły transducera działają od dołu do góry - od liści (*leaves*) drzewa wejściowego i kończą się w stanie akceptacji. Stany transduktora są oznaczone subskryptem w celu odróżnienia ich od symboli alfabetów wejściowego i wyjściowego.

Tłumaczenie jest tu zdefiniowane jako pary łańcuchów, **owoce** *(yield)* M (MBOT) to zbiory par łańcuchów (s, t) takie, dla których istnieje drzewo $s' \in T_\Sigma$ mające swój owoc s,

drzewo $t' \in T_\Delta$ mające swój owoc t, oraz transdukcja z s' do t' taka, która jest akceptowana przez M.

13 Zakończenie

Zastosowanie MBOT w procesie tłumaczenia automatycznego z pewnością pozwoli uniknąć wielu pomyłek tłumaczeniowych, w dalszym ciągu jednak proces tłumaczenia maszynowego jest niedoskonały. Modyfikujemy systemy i ciągle są jakieś systemowe braki.

Niektóre ograniczenia tłumaczenia maszynowego wydają się być wręcz nie do rozwiązania. Przedstawione wyżej przykłady nie uwzględniały sytuacji, w której językiem wejściowym będzie język z trzema przypadkami gramatycznymi, jak arabski, a wyjściowym z siedmioma przypadkami gramatycznymi jak polski. Efekt tłumaczenia maszynowego może

przynieść w takiej sytuacji rezultat nieoczekiwany, gdyż nie ma prostej reguły przypisania przypadka w języku źródłowym do odpowiadającego mu przypadka w języku wyjściowym. Np. przyimek [bi] w języku arabskim wymusza na rzeczowniku wystąpienie tylko w jednym przypadku gramatycznym: الجر. Przyimek ten wskazuje zwykle na relację lokatywną lub instrumentalną, czemu z reguły odpowiadają polskie formy narzędnika lub miejscownika. Formy narzędnikowe i miejscownikowe nie mogą być dowolnie zamieniane. Co innego znaczy bowiem „być w Egipcie", a co innego „być Egiptem". Arabska konstrukcja **ليس بمصر** [laysa bi-miṣra] jak się zdaje może znaczyć tylko: „nie jest w Egipcie", natomiast identyczną pod względem składniowym konstrukcję **ليس بطالب** [laysa bi-ṭālibin] przetłumaczymy na: „nie jest studentem". Niezły skądinąd model Melameda nie wyjaśnia co zrobić w sytuacjach kiedy jednemu tokenowi wejściowemu odpowiada więcej niż jeden token na wyjściu, lub w sytuacjach, kiedy brakowi tokena (np. łącznika lub czasownika posiłkowego w arabskim zdaniu imiennym), odpowiada token (łącznik lub czasownik posiłkowy) w zdaniu polskim. Zaproponowany przez Gildeę MBOT zbliża nas nieco do rozwiązania tego problemu, ale go nie rozwiązuje.

Bibliografia

Abbès, Ramzi, Joseph Dichy, Mohamed Hassoun. 2004. „The Architecture of a Standard Arabic lexical database: some figures, ratios and categories from the DIINAR.1 source program". *Semitic '04: Proceedings of the Workshop on Computational Approaches to Arabic Script-based Languages*, ss.. 15-22 .

Alshawi, Hiyan, Srinivas Bangalore, Shona Douglas .2000. „Learning Dependency Translation Models as Collections of Finite-State Head Transducers". *Computational Linguistics*, Vol. 26, N. 1. ss. 45-60.

Brennan, Susan, Marilyn Walker-Friedman, and Carl Pollard. 1987. „A centering approach to pronouns." *Proceedings of the 25th Annual Meeting of the Association for Computational Linguistics*, ss. 155–162, Stanford, California.

Brown, Peter F., John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Fredrick Jelinek, Robert L. Mercer, and Paul Roossin. 1988. „A statistical approach to language translation". *Proceedings of the 12th International Conference on Computational Linguistics*, ss. 71–76, Budapest, Hungary.

Chomsky, Noam. 1959. „On certain formal properties of grammars". *Information and Control*, 2:137–167.

Chomsky, Noam.1982. *Zagadnienia teorii składni*.Wrocław-Warszawa-Kraków-Gdańsk-Łódź: Zakład Narodowy im. Ossolińskich Wydawnictwo.

Clark, Alexander , Chris Fox. Shalom Lappin (ed) .2010. *The Handbook of Computational Linguistics and Natural Language Processing*. Blackwell Publishing Ltd.

Cohen, David. 1961. „Essai d'une analyse automatique de l'arabe". T.A. informations. Repr: D. Cohen, *Études de linguistique sémitique et arabe*, Paris, Mouton, 1970.

Cohen, Shay B, Noah A, Smith. 2012. „Empirical Risk Minimization for Probabilistic Grammars: Sample Complexity and Hardness of Learning". *Computational Linguistics*, Vol. 38, N. 3. pp. 479-526

Dębowski, Łukasz 2001. *Tagowanie i dezambiguacja morfosyntaktyczna. Przegląd metod i oprogramowania*. Warszawa. http://www.ipipan.waw.pl/~ldebowsk/docs/raporty/kropka934.pdf

Dichy, Joseph. 1997. „Pour une lexicomatique de l'arabe : l'unité lexicale simple et l'inventaire fini des spécificateurs du domaine du mot". *Meta* 42, juin 1997, Québec, Presses de l'Université de Montréal: 291-306.

Dorr, Bonnie. 1997. „Large-scale dictionary construction for foreign language tutoring and interlingual machine translation". *Machine Translation*, 12(4):1–55.

Edmonds, Philip, Graeme Hirst .2002. „Near-Synonymy and Lexical Choice". *Computational Linguistics*, Vol. 28, N. 2. ss. 105-144.

Engelfriet, J., E. Lilin, and A. Maletti. 2009. „Extended multi bottom–up tree transducers". *Acta Informatica*, 46(8):561–590.

Engelfriet, J., E. Lilin, and A. Maletti. „Composition and Decomposition of Extended Multi Bottom-Up Tree Transducers".Acta Informatica–manuscript: http://www.ims.uni-stuttgart.de/~maletti/pub/englilmal08b.pdf

Gildea, Daniel. 2012. „On the String Translations Produced by Multi Bottom–Up Tree Transducers". *Computational Linguistics*, Vol. 38, N. 3. pp. 673-693

Goodman, Nelson. 1952. „On likeness of meaning". L. Linsky, editor, Semantics and the Philosophy of Language. University of Illinois Press, ss. 67–74.

Grosz, Barbara, Aravind Joshi, and Scott Weinstein. 1983. „Providing a unified account of definite noun phrases in discourse". *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, ss. 44–50, MIT Press, Cambridge, Massachusetts.

Grosz, Barbara and Candace Sidner. 1986. „Attentions, intentions and the structure of discourse". *Computational Linguistics,* Vol 12. N. 3. ss. 175–204.
Hobbs, Jerry R. 1978. „Resolving pronoun references". *Lingua*, 44:311–338.
Hutchins, W. John .2012. „Victor H. Yngve". *Computational Linguistics* (Obituary). Volume 38, Number 3. ss. 461-467.
Korzycki, Michał. 2008. *Transducer skończenie stanowy jako narzędzie rozpoznawania form tekstowych wyrazów polskich* - rozprawa doktorska napisana pod kierunkiem
profesora Wiesława Lubaszewskiego. Kraków, 2008
Léon, Jacqueline. 1998. "Les débuts de la traduction automatique en France (1959-1968): à contretemps?". *Modèles Linguistiques*. T. XIX, fascicule 2 ss.55-86.
Lapata, Maria .2002. „The Disambiguation of Nominalizations". *Computational Linguistics*, Vol. 28, N. 3. ss. 357-388.
Łacina, Jerzy. 2002. „Rola ogranicznika w automatycznym tłumaczeniu arabskiego tekstu koranicznego na język polski" w: Adnan Abbas (ed) *Palestyna dawniej i dziś - Palestine Past and Present - materiały interdyscyplinarnej Konferencji Naukowej, zorganizowanej w Poznaniu 19-20 listopada 2001*.Instytut Orientalistyczny Uniwersytet im. A. Mickiewicza . Poznań.
Mann, William C. and Sandra A. Thompson. 1986. „Relational propositions in discourse". *Discourse Processes*, 9(1):57–90, January-March.
Marchand, Yannick, Robert I. Damper .2000. „A Multistrategy Approach to Improving Pronunciation by Analogy". *Computational Linguistics*, Vol. 26, N. 2. ss. 195-219.
Marcu, Daniel .2000. „The Rhetorical Parsing of Unrestricted Texts: A Surface-based Approach". *Computational Linguistics*, Vol. 26, N. 3. ss. 395-448.
Melamed, Dan .2000. „Models of Translational Equivalence among Words". *Computational Linguistics*, Vol. 26, N. 2. ss. 221-249.
Merlo, Paola, Eva Esteve Ferrer .2006. „The Notion of Argument in Prepositional Phrase Attachment". *Computational Linguistics*, Vol. 32, N. 3. ss. 341-377.
Mikheev, Andrei .2002. „Periods, Capitalized Words, etc". *Computational Linguistics*, Vol. 28, N. 3. ss. 289-318.
Nederhof, Mark-Jan .2000. „Practical Experiments with Regular Approximation of Context-Free Languages". *Computational Linguistics*, Vol. 26, N. 1. pp. 18-44.
Palomar, Manuel, Antonio Ferrandez, Lidia Moreno, Patricio Martinez-Barco, Jesus Peral, Maximiliano Saiz-Noeda, Rafael Munoz .2001. „An Algorithm for Anaphora Resolution in Spanish Texts". *Computational Linguistics*, Vol. 27, N. 4. ss. 545-567.
Pineda, Luis, Gabriela Garza .2000. „A Model for Multimodal Reference Resolution". *Computational Linguistics*, Vol. 26, N. 2. ss. 139-193.
Pulman, Stephen G. .2000. „Bidirectional Contextual Resolution". *Computational Linguistics*, Vol. 26, N. 4. ss. 497-537.
Quine, W. V. O. 1951. „Two dogmas of empiricism". *Philosophical Review*, 60:20–43.
Rubinoff, Robert .2000. „Integrating Text Planning and Linguistic Choice Without Abandoning Modularity: The IGEN Generator". *Computational Linguistics*, Vol. 26, N. 2. ss. 108-138.
Siegel, Eric V., Kathleen R. McKeown .2000. „Learning Methods to Combine Linguistic Indicators: Improving Aspectual Classification and Revealing Linguistic Insights". *Computational Linguistics*, Vol. 26, N. 4. ss. 595-628.

Stevenson, Mark, Yorick Wilks .2001. „The Interaction of Knowledge Sources in Word Sense Disambiguation". *Computational Linguistics*, Vol. 27, N. 3. ss. 321-349.
Stolcke, Andreas, Klaus Ries, Noah Coccaro, Elizabeth Shriberg, Rebecca Bates, Daniel Jurafsky, Paul Taylor, Rachel Martin, Carol Van Ess-Dykema, Marie Meteer .2000. „Dialogue Act Modeling for Automatic Tagging and Recognition of Conversational Speech". *Computational Linguistics*, Vol. 26, N. 3. ss. 339-373.
Weaver, W. 1949. "The Mathematics of Communication", *Scientific American*, 181. ss. 11-15.
Wille, Rudolf. 1982. „Restructuring lattice theory: an approach based on hierarchies of concept". *I. Rival, editor, Ordered Sets*. Reidel, Dordecht/Boston, ss. 445–470.
Yarowsky, David. 1992. „Word-sense disambiguation using statistical models of Roget's categories trained on large corpora". *Proceedings of the 14th International Conference on Computational Linguistics* (COLING-92), ss. 454–460, Nantes.
Yngve, Victor.1955. „Syntax and the problem of multiple meaning". William N. Locke and A. Donald Booth, editors, *Machine Translation of Languages: Fourteen Essays*. Technology Press of the Massachusetts Institute of Technology and Wiley, Cambridge, MA, and New York, ss. 208–226.
Zrigui, Mounir. 2007. „Traitement automatique de la langue arabe". Unitéé de recherche RIADI, faculté de Sciences de des Monastir, Tunisi.