

POLFIE: gramatyka LFG dla języka polskiego

Agnieszka Patejuk i Adam Przepiórkowski



INSTYTUT PODSTAW INFORMATYKI
POLSKIEJ AKADEMII NAUK
ul. Jana Kazimierza 5, 01-248 Warszawa



Seminarium ZIL, Warszawa, 22 lutego 2016



- krótki wstęp do teorii LFG
- implementacja: polskie LFG w XLE
- INESS i bank struktur LFG

POLFIE

<http://zil.ipipan.waw.pl/LFG/>

Gramatyka POLFIE w XLE-Web (testowanie online)

<http://iness.mozart.ipipan.waw.pl/iness/xle-web>



Gramatyka leksykalno-funkcyjna:

- ang. *Lexical Functional Grammar* (LFG; Bresnan 1982, 2000, Dalrymple 2001),
- teoria **generatywna** (w szerokim sensie tego terminu),
- ale **nie transformacyjna**,
- silnie **zleksykalizowana**,
- **sformalizowana**,
- **implementowalna**,
- obejmuje **różne poziomy** wiedzy lingwistycznej,
- użyta do opisu **wielu** typologicznie różnorodnych języków,
- aktywnie **rozwijana** (liczne ośrodki, coroczna międzynarodowa konferencja – w tym roku w Warszawie).

Poziomy reprezentacji



Dwa podstawowe poziomy reprezentacji:

- **c-struktura:**

- struktura składnikowa w formie drzewa,
- oparta na **kategoriach składniowych**,
- struktura **powierzchniowa**,
- **zależna od języka**;

- **f-struktura:**

- struktura funkcyjna w formie struktury atrybutów,
- oparta na **funkcjach gramatycznych**,
- struktura **głęboka**,
- **dość uniwersalna**,
- bliska **semantycy** (ale to nie semantyka).

Pozostałe: s(ematic)-structure, a(rgument)-structure,
i(nformation)-structure. . .



Reguła (uproszczona) dla zdania *Antek idzie*:

- $S \rightarrow N \quad V$
(\uparrow SUBJ)= \downarrow \uparrow = \downarrow
- S (zdanie): lewa strona reguły (matka)
- N (rzeczownik) i V (czasownik): prawa strona reguły (dzieci)
- N ma anotację mówiącą, że jest to podmiot
- V ma anotację mówiącą, że przekazuje cechy matce



- reguła zdaniowa (uproszczona):

$$S \rightarrow \quad N \quad V$$

$$(\uparrow \text{SUBJ})=\downarrow \quad \uparrow=\downarrow$$

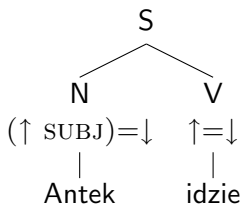
- wpis dla słowa *Antek*:

Antek N $(\uparrow \text{PRED})='ANTEK'$
 $(\uparrow \text{NUM})= \text{SG}$
 $(\uparrow \text{CASE})= \text{NOM}$
 $(\uparrow \text{GEND})= \text{M1}$

- wpis dla słowa *idzie*:

idzie V $(\uparrow \text{PRED})='IŚĆ<(\uparrow \text{SUBJ})>'$
 $(\uparrow \text{SUBJ NUM})=_{\text{c}} \text{SG}$
 $(\uparrow \text{SUBJ CASE})=_{\text{c}} \text{NOM}$
 $(\uparrow \text{TENSE})= \text{PRES}$

Powstałe struktury



PRED	'IŚĆ$\langle \mathbb{1} \rangle$'	
SUBJ	$\mathbb{1}$	PRED 'ANTEK'
		CASE NOM
		GEND M1
		NUM SG
TENSE	PRES	

Funkcje gramatyczne



- **podstawowe pojęcie** w LFG,
- określa relację między **nadrzędnikiem a podrzędnikiem**,
- **niezależne od pozycji** w drzewie,
- **mogą być** w pewnym stopniu **skorelowane z semantyką**,
- funkcje są **uniwersalnie dostępne**,
- jednak nie każdy język korzysta z całego zestawu.

Podstawowy zestaw funkcji



- SUBJ: podmiot (*Antek idzie*),
- OBJ: dopełnienie bliższe (*Eryk lubi Antka*),
- OBJ_θ: dopełnienie dalsze (*Antek dał Erykowi płytę*),
- OBL: dopełnienie przyimkowe (*Eryk czeka na Antka*),
- COMP: dopełnienie zdaniowe (*Antek mówi, że idzie*),
- XCOMP: dopełnienie bezokolicznikowe (*Antek chce iść*),
- XCOMP-PRED: dopełnienie predykatywne (*Antek jest miły*).

Przykład: dopełnienie bliższe (OBJ)



Eryk lubi *Antka*.

PRED	‘LUBIĆ⟨ 1 , 2 ⟩’				
SUBJ	1 <table style="border-collapse: collapse; display: inline-table; vertical-align: middle;"> <tr> <td style="border-right: 1px solid black; padding: 5px 5px 5px 5px;"> PRED </td> <td style="padding: 5px 5px 5px 5px;"> ‘ERYK’ </td> </tr> <tr> <td style="border-right: 1px solid black; padding: 5px 5px 5px 5px;"> CASE </td> <td style="padding: 5px 5px 5px 5px;"> NOM </td> </tr> </table>	PRED	‘ERYK’	CASE	NOM
PRED	‘ERYK’				
CASE	NOM				
OBJ	2 <table style="border-collapse: collapse; display: inline-table; vertical-align: middle;"> <tr> <td style="border-right: 1px solid black; padding: 5px 5px 5px 5px;"> PRED </td> <td style="padding: 5px 5px 5px 5px;"> ‘ANTEK’ </td> </tr> <tr> <td style="border-right: 1px solid black; padding: 5px 5px 5px 5px;"> CASE </td> <td style="padding: 5px 5px 5px 5px;"> ACC </td> </tr> </table>	PRED	‘ANTEK’	CASE	ACC
PRED	‘ANTEK’				
CASE	ACC				
TENSE	PRES				

Przykład: dopełnienie dalsze (OBJ_θ)

Antek dał *Erykowi* płytę.

PRED	‘DAĆ$\langle 1,2,3 \rangle$’
SUBJ	$\left[\begin{array}{l} \text{PRED} \quad \text{‘ANTEK’} \\ \text{CASE} \quad \text{NOM} \end{array} \right]$
OBJ	$\left[\begin{array}{l} \text{PRED} \quad \text{‘PŁYTA’} \\ \text{CASE} \quad \text{ACC} \end{array} \right]$
OBJ _θ	$\left[\begin{array}{l} \text{PRED} \quad \text{‘ERYK’} \\ \text{CASE} \quad \text{DAT} \end{array} \right]$
TENSE	PRES

Przykład: dopełnienie przyimkowe (OBL)



Eryk czeka *na Antka*.

PRED	‘CZEKAĆ<[1],[2]>’						
SUBJ	[1] <table style="border-collapse: collapse; display: inline-table;"> <tr> <td style="padding: 5px;">PRED</td> <td style="padding: 5px;">‘ERYK’</td> </tr> <tr> <td style="padding: 5px;">CASE</td> <td style="padding: 5px;">NOM</td> </tr> </table>	PRED	‘ERYK’	CASE	NOM		
PRED	‘ERYK’						
CASE	NOM						
OBL	[2] <table style="border-collapse: collapse; display: inline-table;"> <tr> <td style="padding: 5px;">PRED</td> <td style="padding: 5px;">‘ANTEK’</td> </tr> <tr> <td style="padding: 5px;">CASE</td> <td style="padding: 5px;">ACC</td> </tr> <tr> <td style="padding: 5px;">PFORM</td> <td style="padding: 5px;">NA</td> </tr> </table>	PRED	‘ANTEK’	CASE	ACC	PFORM	NA
PRED	‘ANTEK’						
CASE	ACC						
PFORM	NA						
TENSE	PRES						

Przykład: dopełnienie zdaniowe (COMP)



Antek mówi, że idzie.

PRED	‘MÓWIĆ<[1],[2]>’												
SUBJ	[1]	<table style="border-collapse: collapse;"> <tr> <td style="border-right: 1px solid black; padding-right: 10px;">PRED</td> <td style="padding-left: 10px;">‘ANTEK’</td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 10px;">CASE</td> <td style="padding-left: 10px;">NOM</td> </tr> </table>	PRED	‘ANTEK’	CASE	NOM							
PRED	‘ANTEK’												
CASE	NOM												
COMP	[2]	<table style="border-collapse: collapse;"> <tr> <td style="border-right: 1px solid black; padding-right: 10px;">PRED</td> <td style="padding-left: 10px;">‘IŚĆ<[3]>’</td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 10px;">SUBJ</td> <td style="padding-left: 10px;">[3]</td> <td style="border-left: 1px solid black; padding-left: 10px;"> <table style="border-collapse: collapse;"> <tr> <td style="border-right: 1px solid black; padding-right: 10px;">PRED</td> <td style="padding-left: 10px;">‘PRO’</td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 10px;">TENSE</td> <td style="padding-left: 10px;">PRES</td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 10px;">COMP-FORM</td> <td style="padding-left: 10px;">ŻE</td> </tr> </table> </td> </tr> </table>	PRED	‘IŚĆ<[3]>’	SUBJ	[3]	<table style="border-collapse: collapse;"> <tr> <td style="border-right: 1px solid black; padding-right: 10px;">PRED</td> <td style="padding-left: 10px;">‘PRO’</td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 10px;">TENSE</td> <td style="padding-left: 10px;">PRES</td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 10px;">COMP-FORM</td> <td style="padding-left: 10px;">ŻE</td> </tr> </table>	PRED	‘PRO’	TENSE	PRES	COMP-FORM	ŻE
PRED	‘IŚĆ<[3]>’												
SUBJ	[3]	<table style="border-collapse: collapse;"> <tr> <td style="border-right: 1px solid black; padding-right: 10px;">PRED</td> <td style="padding-left: 10px;">‘PRO’</td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 10px;">TENSE</td> <td style="padding-left: 10px;">PRES</td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 10px;">COMP-FORM</td> <td style="padding-left: 10px;">ŻE</td> </tr> </table>	PRED	‘PRO’	TENSE	PRES	COMP-FORM	ŻE					
PRED	‘PRO’												
TENSE	PRES												
COMP-FORM	ŻE												
TENSE	PRES												

Przykład: dopełnienie bezkolicznikowe (XCOMP)



Antek chce *iść*.

PRED		‘CHCIEĆ<1,2>’				
SUBJ	1	<table border="0"> <tr> <td>PRED</td> <td>‘ANTEK’</td> </tr> <tr> <td>CASE</td> <td>NOM</td> </tr> </table>	PRED	‘ANTEK’	CASE	NOM
PRED	‘ANTEK’					
CASE	NOM					
XCOMP	2	<table border="0"> <tr> <td>PRED</td> <td>‘IŚĆ<1>’</td> </tr> <tr> <td>SUBJ</td> <td>1</td> </tr> </table>	PRED	‘IŚĆ<1>’	SUBJ	1
PRED	‘IŚĆ<1>’					
SUBJ	1					
TENSE		PRES				

Antek kazał Erykowi *iść*.

PRED		‘KAZAĆ<1,2,3>’				
SUBJ	1	<table border="0"> <tr> <td>PRED</td> <td>‘ANTEK’</td> </tr> <tr> <td>CASE</td> <td>NOM</td> </tr> </table>	PRED	‘ANTEK’	CASE	NOM
PRED	‘ANTEK’					
CASE	NOM					
OBJ _θ	2	<table border="0"> <tr> <td>PRED</td> <td>‘ERYK’</td> </tr> <tr> <td>CASE</td> <td>DAT</td> </tr> </table>	PRED	‘ERYK’	CASE	DAT
PRED	‘ERYK’					
CASE	DAT					
XCOMP	3	<table border="0"> <tr> <td>PRED</td> <td>‘IŚĆ<2>’</td> </tr> <tr> <td>SUBJ</td> <td>2</td> </tr> </table>	PRED	‘IŚĆ<2>’	SUBJ	2
PRED	‘IŚĆ<2>’					
SUBJ	2					
TENSE		PRES				

Przykład: dopełnienie bezkolicznikowe (XCOMP)



Antek chce *iść*.

[PRED	‘CHCIEĆ<1,2>’]				
	SUBJ	1 [<table border="0"> <tr> <td>PRED</td> <td>‘ANTEK’</td> </tr> <tr> <td>CASE</td> <td>NOM</td> </tr> </table>		PRED	‘ANTEK’	CASE	NOM
PRED	‘ANTEK’						
CASE	NOM						
	XCOMP	2 [<table border="0"> <tr> <td>PRED</td> <td>‘IŚĆ<1>’</td> </tr> <tr> <td>SUBJ</td> <td>1</td> </tr> </table>	PRED	‘IŚĆ<1>’	SUBJ	1	
PRED	‘IŚĆ<1>’						
SUBJ	1						
	TENSE	PRES					

Antek kazał Erykowi *iść*.

[PRED	‘KAZAĆ<1,2,3>’]				
	SUBJ	1 [<table border="0"> <tr> <td>PRED</td> <td>‘ANTEK’</td> </tr> <tr> <td>CASE</td> <td>NOM</td> </tr> </table>		PRED	‘ANTEK’	CASE	NOM
PRED	‘ANTEK’						
CASE	NOM						
	OBJ _θ	2 [<table border="0"> <tr> <td>PRED</td> <td>‘ERYK’</td> </tr> <tr> <td>CASE</td> <td>DAT</td> </tr> </table>	PRED	‘ERYK’	CASE	DAT	
PRED	‘ERYK’						
CASE	DAT						
	XCOMP	3 [<table border="0"> <tr> <td>PRED</td> <td>‘IŚĆ<2>’</td> </tr> <tr> <td>SUBJ</td> <td>2</td> </tr> </table>	PRED	‘IŚĆ<2>’	SUBJ	2	
PRED	‘IŚĆ<2>’						
SUBJ	2						
	TENSE	PRES					

Przykład: dopełnienie predykatywne (XCOMP-PRED)



Antek uważa Eryka za *miłego*.

Antek jest *miły*.

[PRED	‘BYĆ<1,2>’]	
	SUBJ	1 [PRED ‘ANTEK’
		CASE		NOM
	XC-P	2 [PRED ‘MIŁY<1>’
SUBJ		1 [CASE	NOM
TENSE	PRES			

[PRED	‘UWAŻAĆ<1,2,3>’]	
	SUBJ	1 [PRED ‘ANTEK’
		CASE		NOM
	OBJ	2 [PRED ‘ERYK’
CASE		ACC		
XC-P	3 [PRED ‘MIŁY<2>’		
	SUBJ	2 [CASE	ACC
	PFORM	ZA		
TENSE	PRES			

Przykład: dopełnienie predykatywne (XCOMP-PRED)



Antek uważa Eryka za *miłego*.

Antek jest *miły*.

[PRED	‘BYĆ<	[1,2]’]	
	SUBJ	[1]	[
		PRED	‘ANTEK’]
		CASE	NOM]
		[
	XC-P	[2]	[
		PRED	‘MIŁY<]	
		SUBJ	[1]]	
		CASE	NOM]	
	TENSE	PRES			

[PRED	‘UWAŻAĆ<	[1,2,3]’]		
	SUBJ	[1]	[
		PRED	‘ANTEK’]	
		CASE	NOM]	
		[
	OBJ	[2]	[
		PRED	‘ERYK’]		
		CASE	ACC]		
		[
	XC-P	[3]	[
		PRED	‘MIŁY<]		
		SUBJ	[2]]		
		CASE	ACC]		
		PFORM	ZA]		
	TENSE	PRES				

Struktura gramatyki LFG



Dwie podstawowe części:

- **leksykon** (wpisy dla poszczególnych słów):
 - ograniczenia morfoskładniowe,
 - ograniczenia walencyjne,
 - ograniczenia zleksykalizowane;
- **reguły**:
 - umożliwiają łączenie słów we frazy,
 - mogą wprowadzać dodatkowe ograniczenia.

Poza tym:

- tokenizator,
- morfologia,
- makra,
- deklaracja cech.

O gramatyce POLFIE



Gramatyka POLFIE:

- podejście **LFG**,
- implementacja w **XLE** (platforma dedykowana LFG),
- stworzona w oparciu o **wcześniejsze implementowane gramatyki języka polskiego**:
 - GFJP (DCG): Świdziński 1992, Woliński 2004,
 - FOJP (HPSG): Przepiórkowski i in. 2002;
- **korzysta** z informacji:
 - walencyjnych z **Walentego**,
 - fleksyjnych z **Morfeusza**,
- **testowana** na zdaniach z:
 - banku struktur Składnica: Świdziński i Woliński 2010,
 - korpusu NKJP (szczególnie NKJP1M);
- intensywnie **rozwijana** (m.in. semantyka).

Analizator fleksyjny Morfeusz



Morfeusz (Woliński 2006, 2014):

- **najlepszy** analizator dla języka polskiego,
- **szybki, stabilny**, wygodny w użyciu,
- bardzo **duże pokrycie**,
- tagset podobny do NKJP,
- **publicznie dostępny** na licencji swobodnej (*open source*):
<http://sgjp.pl/morfeusz/>.

Źródło:

- informacji o charakterystyce fleksyjnej,
- informacje są konwertowane na wpisy leksykalne.

Analizator fleksyjny Morfeusz



Morfeusz (Woliński 2006, 2014):

- **najlepszy** analizator dla języka polskiego,
- **szybki, stabilny**, wygodny w użyciu,
- bardzo **duże pokrycie**,
- tagset podobny do NKJP,
- **publicznie dostępny** na licencji swobodnej (*open source*):
<http://sgjp.pl/morfeusz/>.

Źródło:

- informacji o charakterystyce fleksyjnej,
- informacje są konwertowane na wpisy leksykalne.

Przykład konwersji informacji z Morfeusza



Wybrana interpretacja z Morfeusza:

- kot, kot, subst:sg:nom:m2
- forma: kot
- lemat: kot
- znacznik:
 - część mowy: rzeczownik (subst),
 - liczba: pojedyncza (sg),
 - przypadek: mianownik (nom),
 - rodzaj: męsko-zwierzęcy (m2).

Po konwersji:

kot N * (^ PRED)= 'kot'
 (^ NUM)= sg
 (^ CASE)= nom
 (^ GEND)= m2

Przykład konwersji informacji z Morfeusza



Wybrana interpretacja z Morfeusza:

- kot, kot, subst:sg:nom:m2
- forma: kot
- lemat: kot
- znacznik:
 - część mowy: rzeczownik (subst),
 - liczba: pojedyncza (sg),
 - przypadek: mianownik (nom),
 - rodzaj: męsko-zwierzęcy (m2).

Po konwersji:

kot N * (^ PRED)='kot'
 (^ NUM)= sg
 (^ CASE)= nom
 (^ GEND)= m2

Słownik walencyjny Walenty



Walenty (Przepiórkowski i in. 2014b,c,a):

- **elektroniczny** słownik walencyjny (dla ludzi i maszyn),
- **duży i szczegółowy**,
- obecnie głównie **powierzchniowoskładniowy**,
- niebawem będzie dodawany poziom **semantyczny**,
- główne źródło empiryczne: NKJP (Przepiórkowski i in. 2012),
- **publicznie dostępny** na licencji swobodnej (*open source*):
<http://walenty.ipipan.waw.pl/>.

Źródło:

- informacji o wymaganiach walencyjnych predykatów,
- informacje są konwertowane na wpisy leksykalne.

Słownik walencyjny Walenty



Walenty (Przepiórkowski i in. 2014b,c,a):

- **elektroniczny** słownik walencyjny (dla ludzi i maszyn),
- **duży i szczegółowy**,
- obecnie głównie **powierzchniowoskładniowy**,
- niebawem będzie dodawany poziom **semantyczny**,
- główne źródło empiryczne: NKJP (Przepiórkowski i in. 2012),
- **publicznie dostępny** na licencji swobodnej (*open source*):
<http://walenty.ipipan.waw.pl/>.

Źródło:

- informacji o wymaganiach walencyjnych predykatów,
- informacje są konwertowane na wpisy leksykalne.

Przykład konwersji informacji z Walentego



Wybrany schemat z Walentego:

- chrapać: pewny: _: : imperf: subj{np(str)}
- predykat: chrapać
- ocena schematu: pewny
- aspekt: niedokonany (imperf)
- argumenty: podmiot nominalny (subj{np(str)})

Po konwersji (wersja uproszczona):

(^ PRED)='chrapać<(^ SUBJ)>'

(^ SUBJ CASE)=c nom

Przykład konwersji informacji z Walentego



Wybrany schemat z Walentego:

- chrapać: pewny: _: : imperf: subj{np(str)}
- predykat: chrapać
- ocena schematu: pewny
- aspekt: niedokonany (imperf)
- argumenty: podmiot nominalny (subj{np(str)})

Po konwersji (wersja uproszczona):

(^ PRED)='chrapać<(^ SUBJ)>'

(^ SUBJ CASE)=c nom

Jak działa POLFIE?



- tekst jest dzielony na zdania
- dla każdego słowa w zdaniu tworzony jest wpis leksykalny:
 - informacja fleksyjna z Morfeusza
 - informacja walencyjna z Walentego
- parser XLE generuje struktury (c- i f-) zgodne z gramatyką
- struktury można wyeksportować do pliku (Prolog, XML)
- niejednoznaczne struktury trzeba ujednoznaczyć:
 - w XLE: od razu, ale mało wygodnie
 - w INESS: dedykowane zaawansowane narzędzie

Krótkie wprowadzenie



Bank struktur to:

- zbiór **wzorcowych** rozkładów (c- i f-struktur)
- tworzony **ręcznie lub automatycznie**

Potencjalne zastosowania:

- **prezentacja** wyników
- możliwość **przeszukiwania**
- pomoc przy **rozwoju zasobów** (szczególnie gramatyki)
- **trenowanie** parsera stochastycznego
- na podstawie struktur można tworzyć **nowe zasoby**



- stworzony i aktywnie **rozwijany** na uniwersytecie w **Bergen** (Rosén i in. 2007, Rosén i in. 2012)
- dostępny **przez sieć**, działa w różnych przeglądarkach:
 - instalacja norweska: <http://iness.uib.no/>
 - instalacja lokalna ZIL:
<http://iness.mozart.ipipan.waw.pl/>
- bardzo **elastyczny, wygodny i szybki** (można wybrać poprawny rozbiór spośród 5000 w 8 krokach)
- daje możliwość tworzenia **różnych** banków struktur, m.in.:
 - LFG
 - zależnościowe
- zintegrowany z XLE-Web: możliwość **parsowania online**

Jak działa INESS?



- struktury (c- i f-) są analizowane przez **INESS**:
 - znajdowane są miejsca, gdzie struktury się różnią,
 - **wyróżniki** służą do wybierania rozbioru na podstawie różnic,
 - wyróżniki dotyczą różnych poziomów struktury (c- i f-);
- przykładowe wyróżniki:
 - **liczba argumentów** wymaganych przez predykat,
 - miejsce **przyłączenia** (relacja nadrzędnik–podrzędnik);
- wybieranie wyróżników:
 - **pozytywne**: wybiera rozkłady spełniające warunek,
 - **negatywne**: wybiera rozkłady, które nie spełniają warunku,
 - swoboda: **dowolna** kolejność wyboru wyróżników,
- można **przenosić** wyróżniki między wersjami:

INESS: struktury przed ujednoznacznieniem



Discriminants

Selected solutions: 2 of 2 | gold no good finished

spurious amb. bad source

Order by: type/anchor frequency disc. power

Jak wygląda przepiórka ?

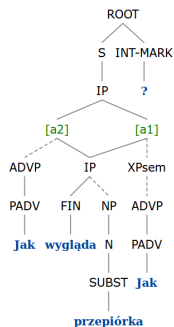
F-structure discriminants | show all

0:5	_TOP 'wyglądać<[],[]>'	1	compl (1)
0:5	_TOP 'wyglądać<[]>'	1	compl (1)
5:1	'wyglądać<[],[]>' OBL-MOD 'jak'	1	compl (1)
5:14	'wyglądać<[],[]>' SUBJ 'przepiórka'	1	compl (1)
5:1	'wyglądać<[]>' ADJUNCT \$ 'jak'	1	compl (1)
5:14	'wyglądać<[]>' SUBJ 'przepiórka'	1	compl (1)

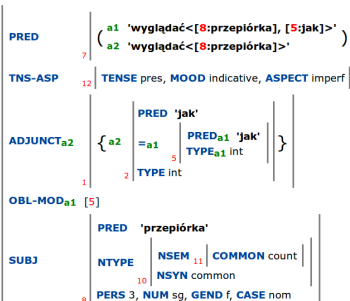
C-structure discriminants

1	Jak wygląda przepiórka		
	IP -> XPsem IP	1	compl (1)
	IP -> ADVP IP	1	compl (1)

C-structure



F-structure



INESS: ujednoznacznione struktury



Discriminants

Selected solutions: 1 of 2 | gold no good finished spurious amb.

bad source

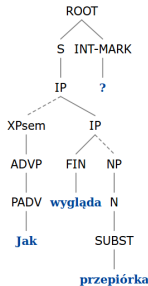
Order by: type/anchor frequency disc. power

Jak wygląda przepiórka ?

F-structure discriminants | show all

5:1 'wyglądać<[],[]>' OBL-MOD 'jak'

C-structure



F-structure

PRED	'wyglądać<[8:przepiórka], [5:jak]>'
TNS-ASP	TENSE pres, MOOD indicative, ASPECT imperf
OBL-MOD	PRED 'jak' TYPE int
SUBJ	PRED 'przepiórka' NTYPE NSEM COMMON count NSYN common PERS 3, NUM sg, GEND f, CASE nom

Strategia ujednoznaczniania



Strategia przyjęta przy tworzeniu polskiego banku struktur LFG:

- preferować wyróżniki dotyczące **f-struktury**
- wybierać tylko **niekontrowersyjne** wyróżniki
- **częściowe** ujednoznacznienie jest dopuszczalne (w sytuacji prawdziwej niejednoznaczności konieczne)
- dodawać **komentarze** do problematycznych zdań



- f-struktura jako **przybliżenie semantyki**
- **ekstrakcja informacji** z f-struktur
- rozszerzenie o **poziom semantyczny**:
 - lepsza **ekstrakcja informacji**
 - rozpoznawanie relacji **wynikania** między tekstami
 - **wyszukiwanie semantyczne**

Podsumowanie



Przedstawiliśmy:

- krótki wstęp do teorii LFG
- polskie LFG i jego strukturę
- polski bank struktur LFG i INESS
- zastosowania parsowania LFG

POLFIE

<http://zil.ipipan.waw.pl/LFG/>

Gramatyka POLFIE w XLE-Web (testowanie online)

<http://iness.mozart.ipipan.waw.pl/iness/xle-web>

Pytania?



"Dziękujemy za uwagę."

```

[PRED      'dziękować<[6-SUBJ:pro], [10:uwaga]>'
SUBJ      [PRED 'pro'
           [CASE nom, NUM pl, PERS 1]
OBL      [PRED      'uwaga'
          [CHECK [_CAT subst]
           [NTYPE [NSEM [COMMON count]
                    [NSYN common]
                    10 [CASE acc, GEND f, NUM sg, PERS 3, PFORM za, PTYPE nose]]]
          [CHECK [_CAT fin]
           6 [TNS-ASP [ASPECT imperf, MOOD indicative, TENSE pres]]]]]]]]

```

Dziękujemy za uwagę.

Podsumowanie



Przedstawiliśmy:

- krótki wstęp do teorii LFG
- polskie LFG i jego strukturę
- polski bank struktur LFG i INESS
- zastosowania parsowania LFG

POLFIE

<http://zil.ipipan.waw.pl/LFG/>

Gramatyka POLFIE w XLE-Web (testowanie online)

<http://iness.mozart.ipipan.waw.pl/iness/xle-web>

- Bresnan J. (red.) 1982: *The Mental Representation of Grammatical Relations*, The MIT Press, Cambridge, MA.
- Bresnan J. 2000: *Lexical-Functional Syntax*, Blackwell.
- Calzolari N., Choukri K., Declerck T., Loftsson H., Maegaard B., Mariani J., Moreno A., Odijk J., Piperidis S. (red.) 2014: *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014, Reykjavík, Iceland*. ELRA.
- Dalrymple M. 2001: *Lexical Functional Grammar*, Academic Press.
- Przepiórkowski A., Kupść A., Marciniak M., Mykowiecka A. 2002: *Formalny opis języka polskiego: Teoria i implementacja*, Akademicka Oficyna Wydawnicza EXIT, Warszawa.
- Przepiórkowski A., Bańko M., Górski R. L., Lewandowska-Tomaszczyk B. (red.) 2012: *Narodowy Korpus Języka Polskiego*, Wydawnictwo Naukowe PWN, Warsaw.
- Przepiórkowski A., Hajnicz E., Patejuk A., Woliński M. 2014a: *Extended phraseological information in a valence dictionary for NLP applications*, [w:] *Proceedings of the Workshop on Lexical and Grammatical Resources for Language Processing (LG-LP 2014)*, Association for Computational Linguistics and Dublin City University, Dublin, Ireland, s. 83–91.
- Przepiórkowski A., Skwarski F., Hajnicz E., Patejuk A., Świdziński M., Woliński M. 2014b: *Modelowanie własności składniowych czasowników w nowym słowniku walencyjnym języka polskiego*, *Polonica XXXIII*, s. 159–178.
- Przepiórkowski A., Hajnicz E., Patejuk A., Woliński M., Skwarski F., Świdziński M. 2014c: *Walenty: Towards a comprehensive valence dictionary of Polish*, [w:] *Calzolari i in. (2014)*, s. 2785–2792.

- Rosén V., Meurer P., Smedt K. D. 2007: Designing and implementing discriminants for LFG grammars, [w:] M. Butt i T. H. King (red.): The Proceedings of the LFG'07 Conference, CSLI Publications, Stanford, CA, s. 397–417.
- Rosén V., De Smedt K., Meurer P., Dyvik H. 2012: An open infrastructure for advanced treebanking, [w:] J. Hajič, K. De Smedt, M. Tadić, i A. Branco (red.): META-RESEARCH Workshop on Advanced Treebanking at LREC2012, Istanbul, Turkey, s. 22–29.
- Świdziński M. 1992: Gramatyka formalna języka polskiego, Wydawnictwa Uniwersytetu Warszawskiego, Warszawa.
- Świdziński M., Woliński M. 2010: Towards a bank of constituent parse trees for Polish, [w:] Text, Speech and Dialogue: 13th International Conference, TSD 2010, Brno, Czech Republic, Lecture Notes in Artificial Intelligence, Springer-Verlag, Berlin, s. 197–204.
- Woliński M. 2004: Komputerowa weryfikacja gramatyki Świdzińskiego, praca doktorska, Instytut Podstaw Informatyki Polskiej Akademii Nauk, Warszawa.
- Woliński M. 2006: Morfeusz — a Practical Tool for the Morphological Analysis of Polish, [w:] M. Kłopotek, S. T. Wierchoń, i K. Trojanowski (red.): Intelligent information processing and web mining, Springer-Verlag, s. 503–512.
- Woliński M. 2014: Morfeusz reloaded, [w:] Calzolari i in. (2014), s. 1106–1111.