Strojový překlad na ÚFALu

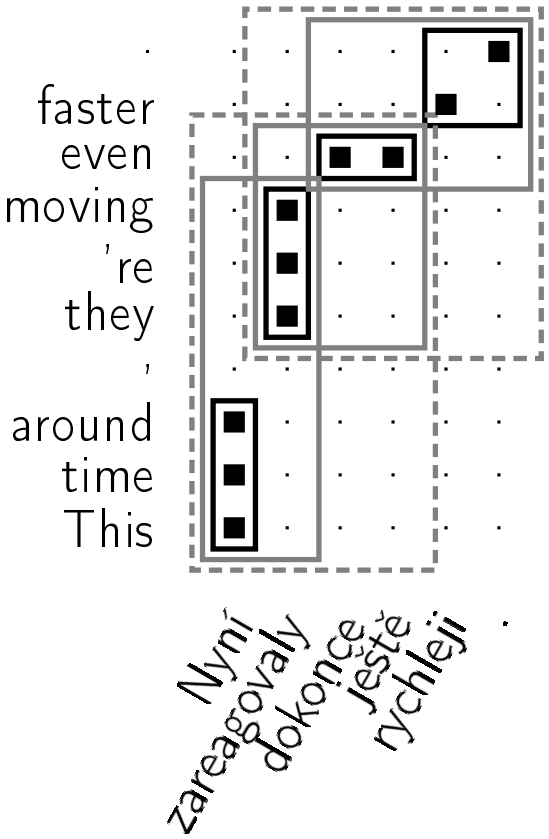
Ondřej Bojar
bojar@ufal.mff.cuni.cz
Ústav formální a aplikované lingvistiky
Matematicko-fyzikální fakulta
Univerzita Karlova v Praze

Obsah prezentace

- Frázový překlad a překlad o více faktorech.
- Strukturní překlad (stromový transfer a srovnání s TectoMT).
- Postřehy závěrem:
  - Žhavé novinky.
  - Problémy BLEU.
  - Problémy symetrizace a extrakce frází.
- Shrnutí a pokus o nadhled.

Reklama úvodem: nepropásněte týdenní soustředění v Praze, zdarma.
`http://ufal.mff.cuni.cz/euromatrix/mtmarathon/`

Frázový překlad



This time around	=	Nyní
they 're moving	=	zareagovaly
even	=	dokonce ještě
...	=	...
This time around, they 're moving	=	Nyní zareagovaly
even faster	=	dokonce ještě rychleji
...	=	...

Ve frázovém překladu hledáme:

- takovou segmentaci vstupní věty na úseky („fráze“)
- a takové překlady frází

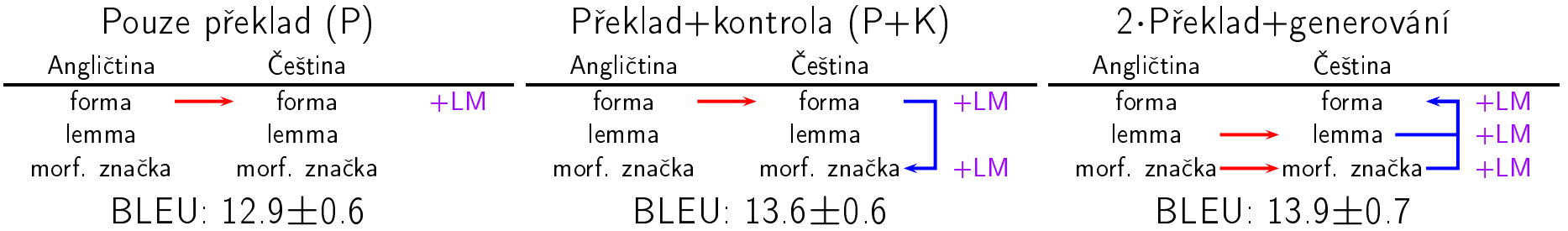
aby byl výstup co nejpravděpodobnější.

Volně šiřitelná implementace: www.statmt.org/moses

Frázový překlad o více faktorech

- Místo prostých slovních forem je každé slovo zapsáno jako n-tice faktorů.
- Možnost zapojit víc jazykových modelů (LM): koherence forem, morf. značek, ...

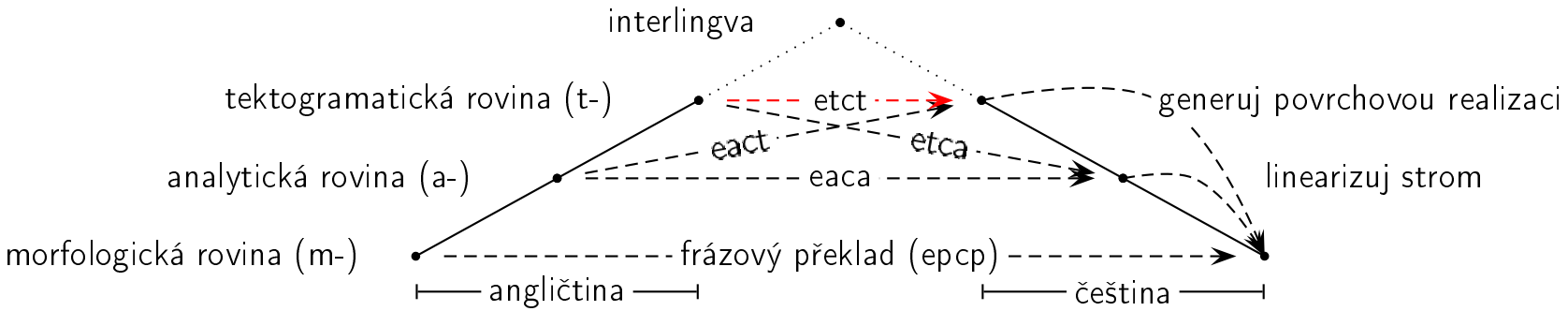
Provedené experimenty: konfigurace kroků v postupném vyplňování cílových faktorů:



⇒ Explicitní kontrola morfologie pomáhá.

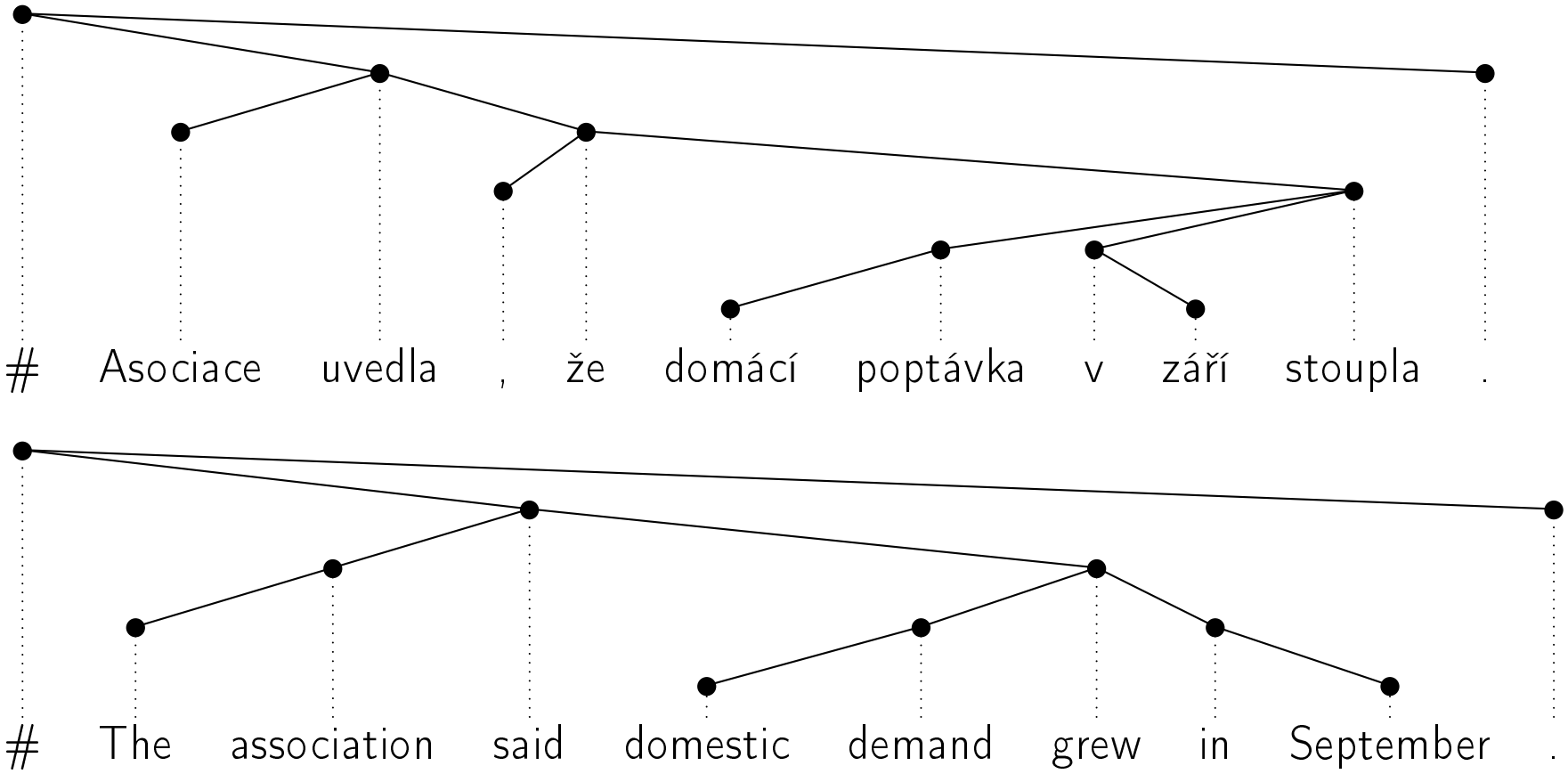
⇒ Složitější scénáře již nejsou výrazně výhodnější.

Strukturní překlad

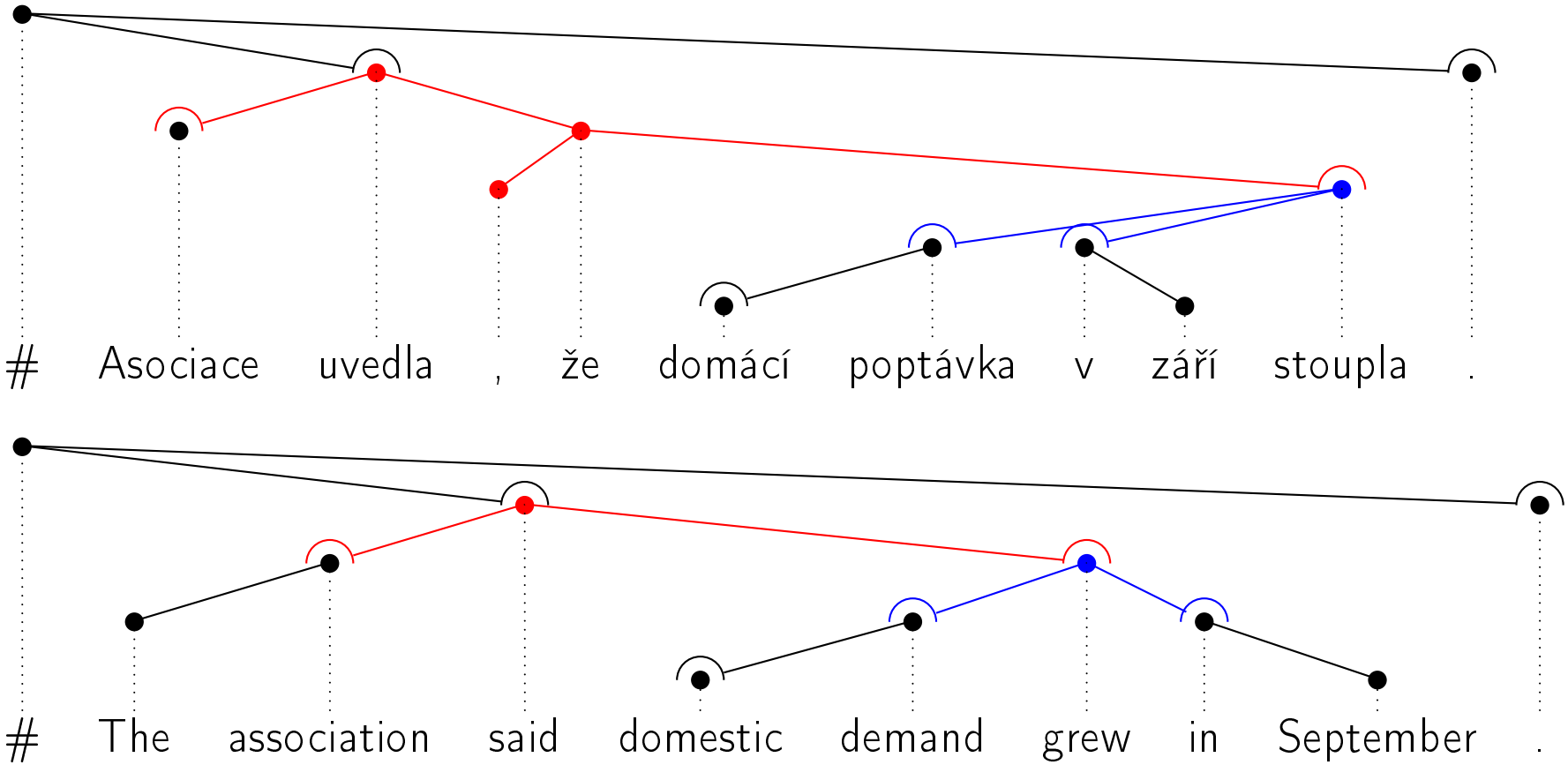


- Vstupní i cílová věta reprezentována jako závislostní strom.
- Hlubší roviny jsou si podobnější.
- Formalismus **Synchronous Tree Substitution Grammar** (Čmejrek, 2006) pro převod stromu na strom.
- Implementovaný dekodér lze užít na kterékoli rovině i napříč rovinami.

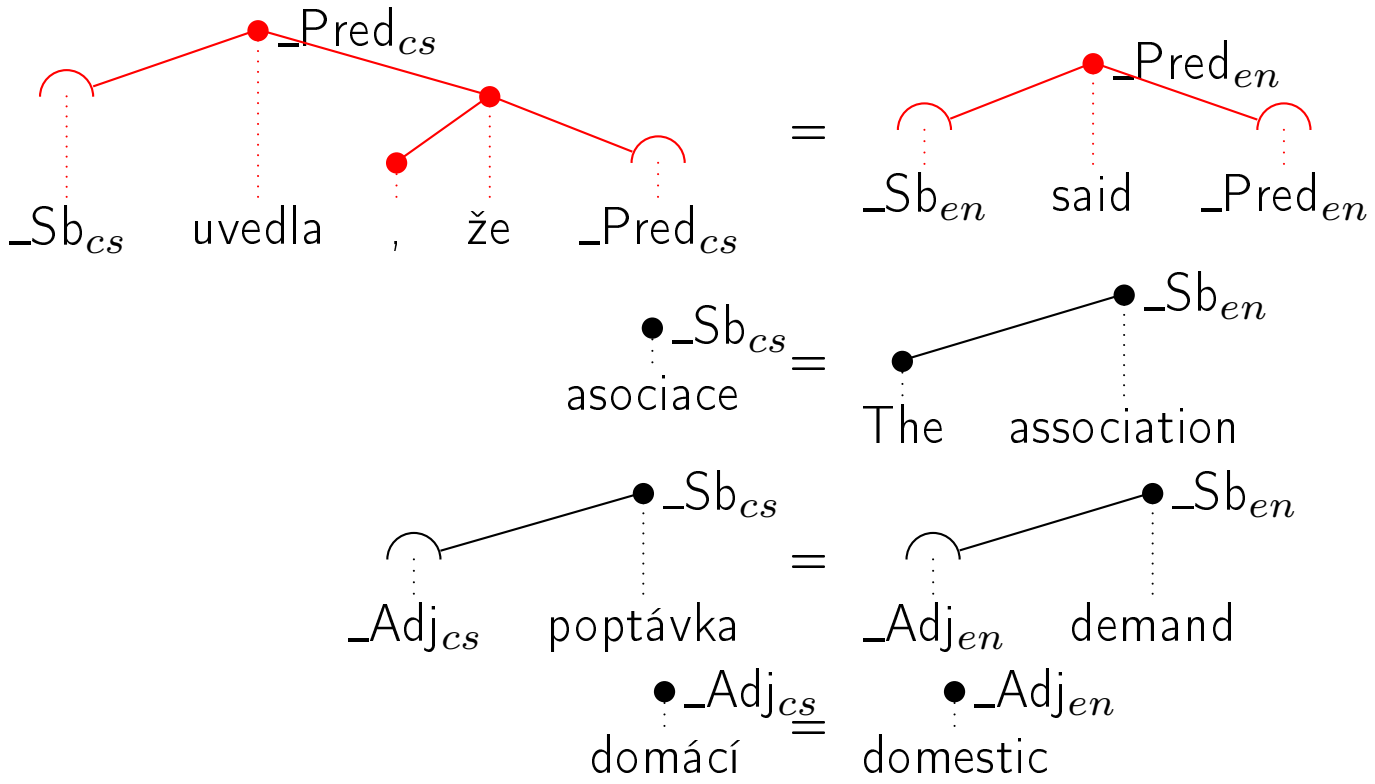
Ilustrace: Dvojici analytických stromů...



...rozložíme na stromečky...



...a sebereme slovník překladů stromečků.

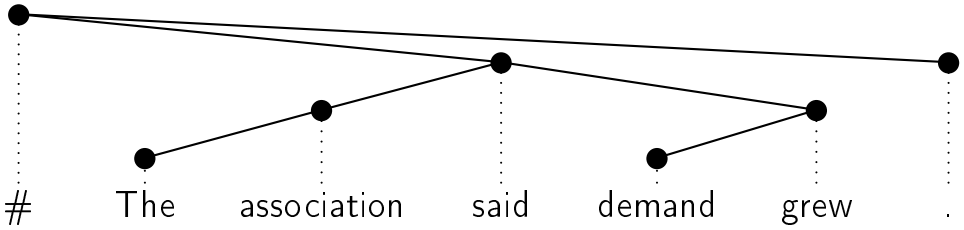


Překlad (decoding) pomocí STSG

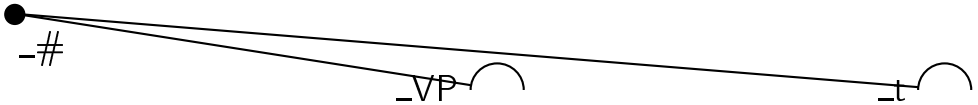
- Ke zdrojovému stromu hledáme dekompozici a cílový závislostní strom, aby jejich synchronní derivace δ měla maximální pravděpodobnost.
- Implementováno ve dvou krocích:

1. Příprava tabulky **možností překladu**:
   - Pro každý vstupní uzel studuji všechny stromečky, které zde mohou začínat.
   - Pokud ke zvolenému stromečku existuje cílový, našli jsme možnost překladu.
   - Uchováváme jen τ nejlepších možností překladu pro každý uzel.
2. Postupné **budování částečných hypotéz**:
   - Od kořene dolů zdrojový strom pokrýváme překladovými možnostmi.
   - Uchováváme jen σ nejlepších částečných hypotéz dané velikosti (počet vstupních uzlů pokrytých vnitřními uzly)

Ukázka možností překladu



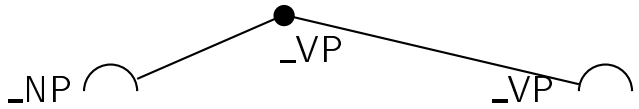
Možnosti překladu v kořeni:



⇒ # _Pred _AuxK

⇒ # _Pred .

Možnosti překladu v uzlu „said“:

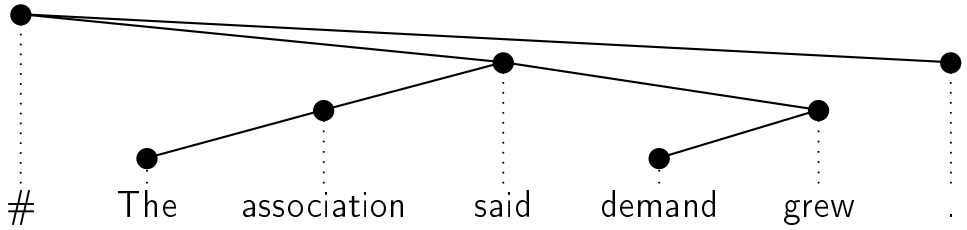


⇒ _Sb uvedla , že _Pred

Možnosti překladu v uzlu „.“:

⇒ .

Postupné budování hypotéz



Ukázková derivace:

Linearizovaný výstup:

h_0 _# ⇒ _#

h_1 _# _VP ⇒ # _Pred .

h_2 _# _NP _VP ⇒ # _Sb uvedla , že _Pred .

h_3 _# _NP _NP ⇒ # _Sb uvedla , že _Sb stoupla .

Překlady stromečků z paralelního treebanku

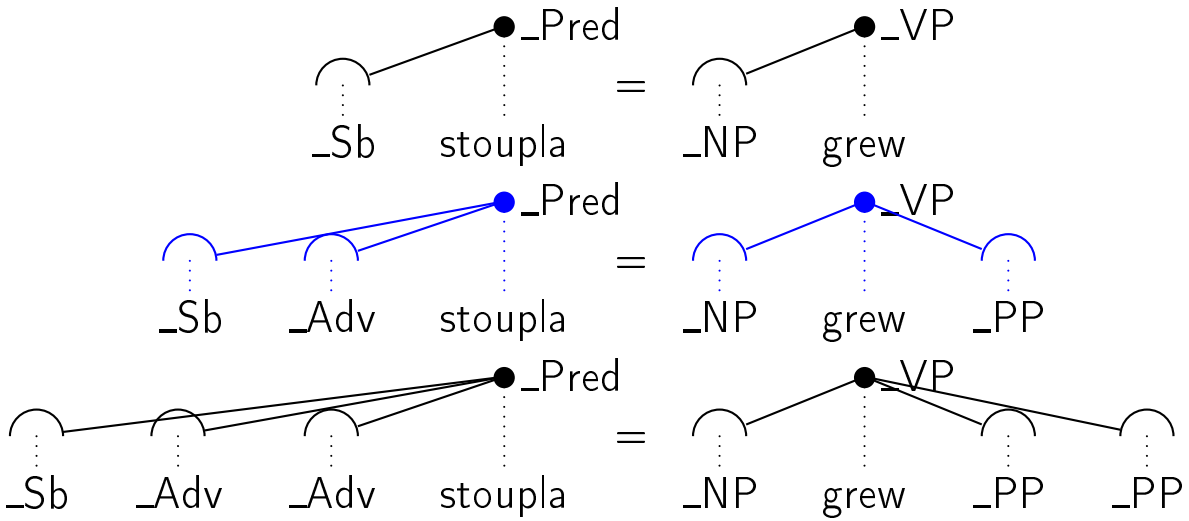
- Disertace Martina Čmejrka nabízí algoritmus zarovnávání stromu na strom.
- Já zatím používám jednoduchou heuristiku:

1. Získej **zarovnání uzel na uzel** (GIZA++ na linearizovaných stromech).
2. Extrahuj všechny dvojice stromečků splňujících všechny tyto podmínky:
   - ne více než i vnitřních uzlů a f slotů,
   - **kompatibilní se zarovnáním uzlu na uzel**,
     např. překlad žádného uzlu nesmí ležet mimo cílový stromeček a sloty si musí být překladem
   - stromečky splňují **podmínku STSG**:
     Všichni následníci vnitřního uzlu musí být rovněž součástí extrahovaného stromečku (ať už jako vnitřní uzly nebo sloty), tj. pro vybudování stromu nebyla třeba operace adjunkce.
3. Podílem frekvencí odhadni pravděpodobnosti, např. $p(t_1, t_2|\text{kořen}_1, \text{kořen}_2)$

Rizika ředění dat (1)

Počet a stavy slotů pro volná doplnění:

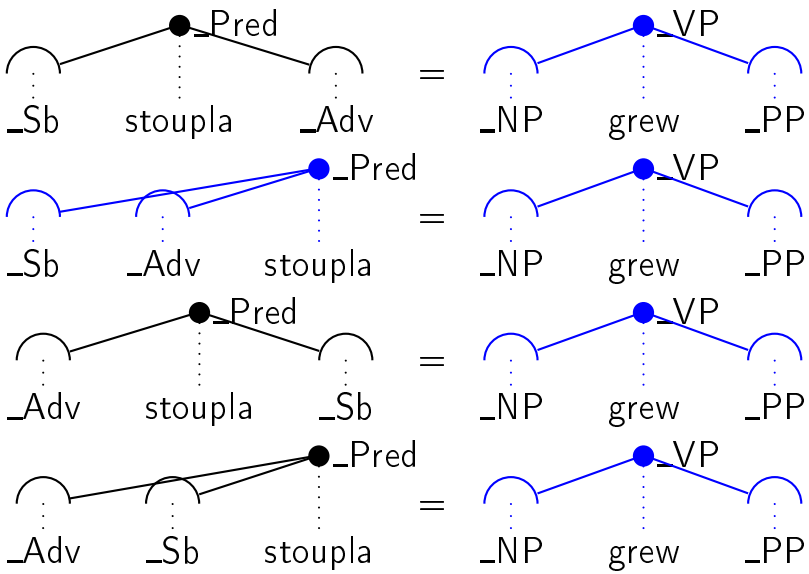
- Podmínka STSG: Jakmile je uzel použit jako vnitřní, musí být vyrobeni i všichni jeho následníci. (Neexistuje operace adjunkce, připojení dalších synů.)



Rizika ředění dat (2)

Pořadí uzlů:

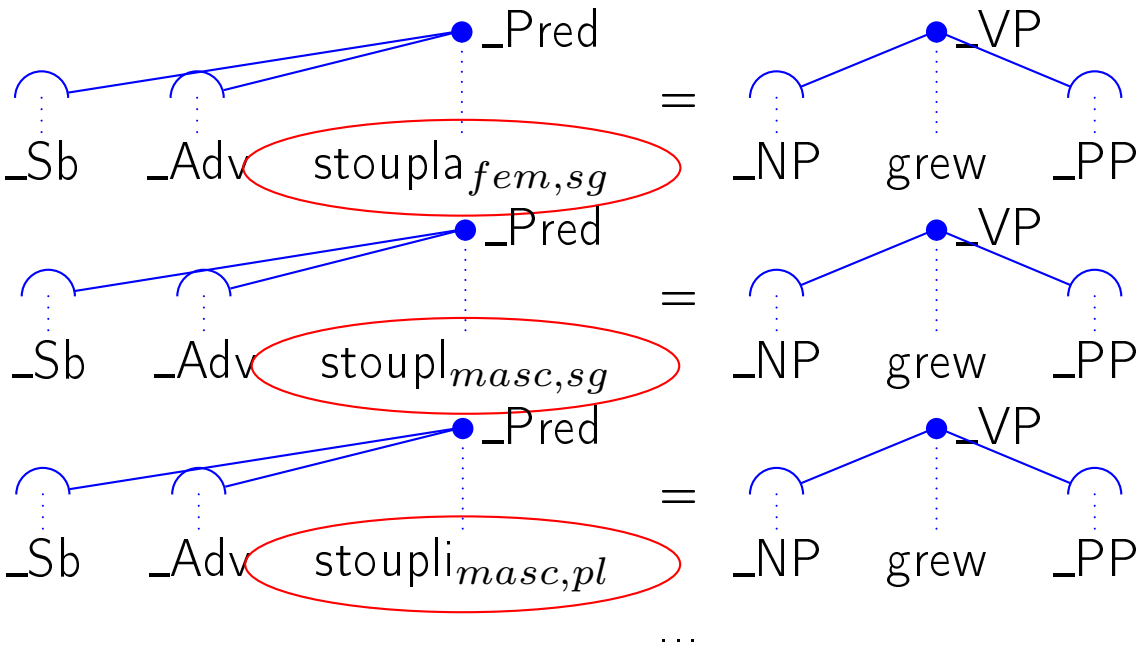
- Volný slovosled češtiny pro jednu anglickou variantu nabízí mnoho překladů.
- Na t-rovině problém můžeme odložit až do generátoru povrchového vyjádření.



Rizika ředění dat (3)

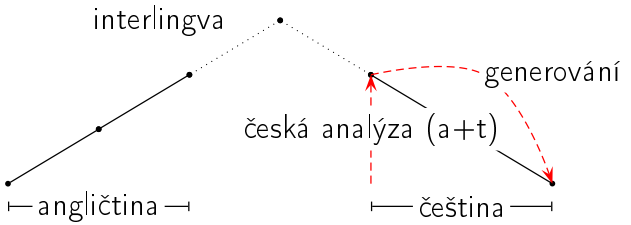
Morfologická bohatost:

- na t-rovině neměla být problémem, ale t-uzly mají množství atributů...



Horní mez kvality překladu (BLEU)

- Analyzuj české věty až na t-rovinu.
- Případně ignoruj některé atributy uzlů.
- Generuj zpátky českou větu.
- Vyhodnoť BLEU proti původní české větě.

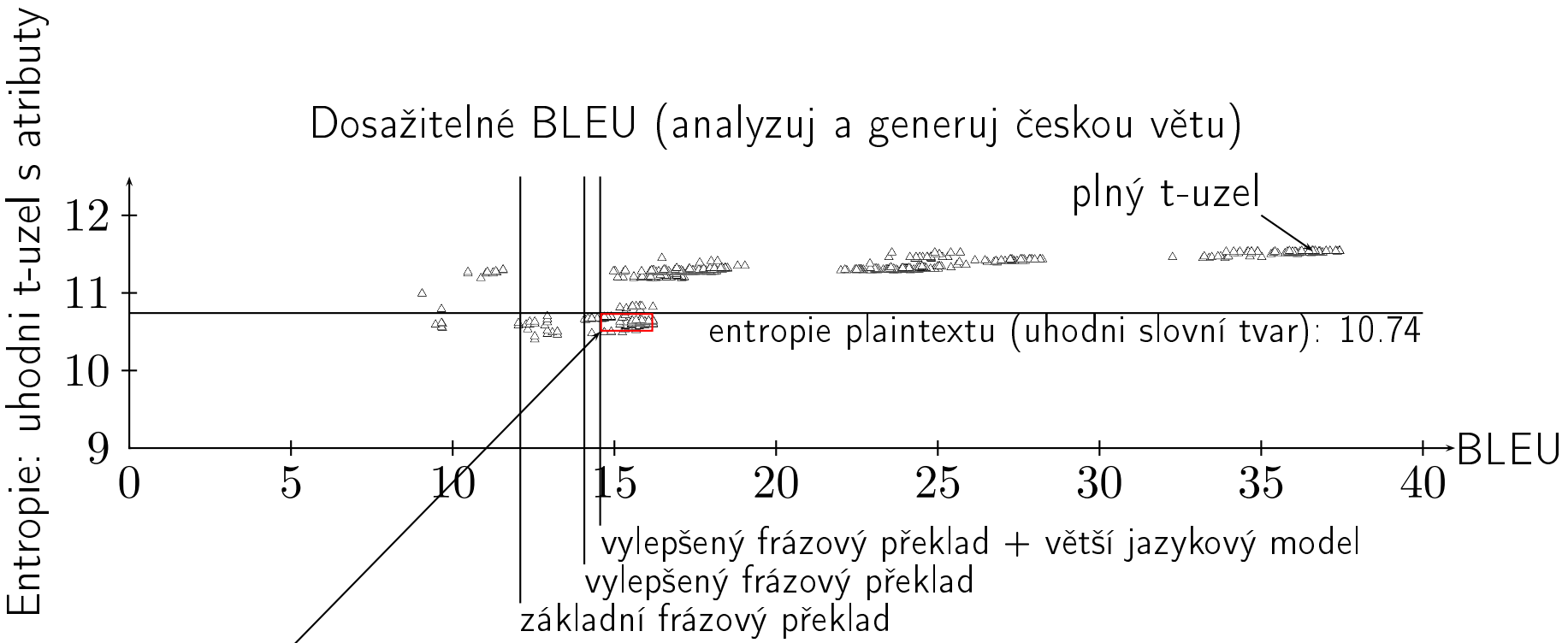


	Horní mez BLEU
Plná automatická t-rovina, žádné atributy neignorovány	36.6±1.2
Ignoruj typ věty (všechny pokládej za oznamovací)	36.6±1.2
Ignoruj podrobné gramatémy sloves (resultativita, ...)	36.6±1.2
Ignoruj slovesný čas, rod, ...	24.9±1.1
Ignoruj všechny gramatémy	5.3±0.5

⇒ Atributy t-uzlů jsou zásadní pro správné generování.

⇒ Lze najít vhodnou rovnováhu mezi bohatostí slovníku a dosažitelným BLEU?

Konec pohádek o snížení bohatosti tvarů



Prostor pro zlepšení za předpokladu:

- t-uzly jsou atomické (a omezujeme množinu atributů)
- chceme zůstat pod entropii plaintextu

$\Rightarrow$ I s bezchybným transferem je prostor pro zlepšení zanedbatelný.

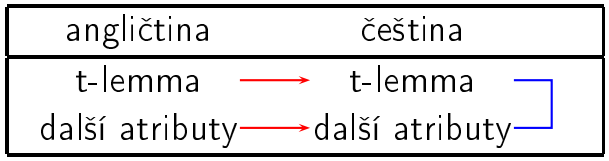
Důsledek: potřebujeme víc faktorů

- t-rovina sama o sobe *zvyšuje* složitost výběru popisku uzlu

⇒ nemůžeme brát popisky uzlů jako nedělitelné jednotky: `go.V.past.third.sg...`

Naprosté minimum:

- dva faktory, pro překlad lematu a gramatických atributů odděleně
- kontrola slučitelnosti lematu a gramatémů (příp. s více jednojazyčnými daty)



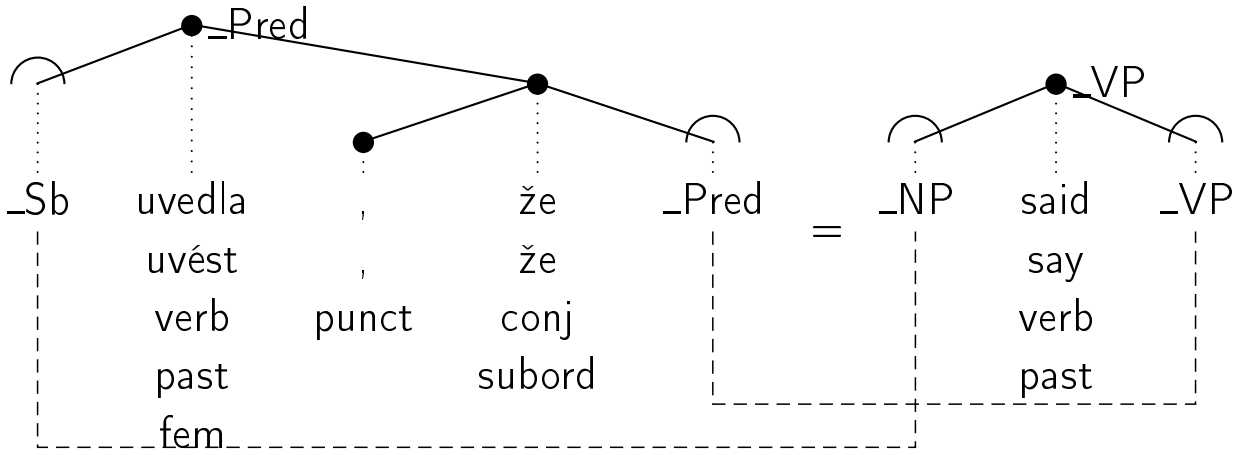
Faktorový překlad je v konfliktu s klíčovou silnou stránkou STSG:

- STSG umožňuje měnit tvar (a velikost) stromu,
- při faktorovém překladu musím vědět, který uzel odpovídá kterému.

⇒ otevřená otázka: jak vhodně spojit stromečky a více faktorů?

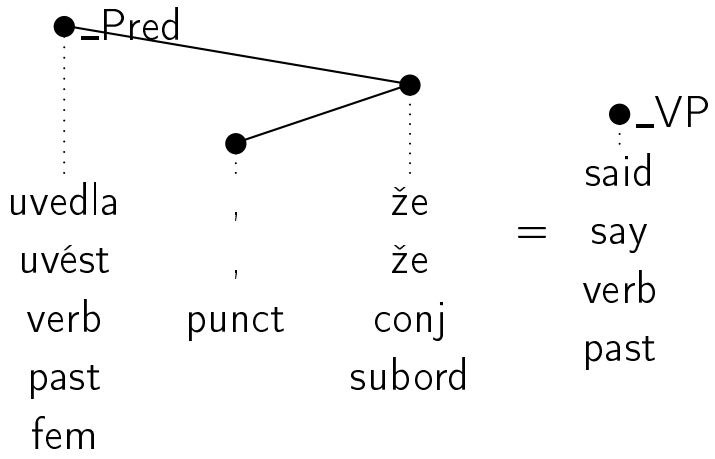
Konstrukce stromečků 1: věrný převod

- Základní metoda STSG, zachovává:
  - tvary stromků, pořadí uzlů
  - všechny faktory vnitřních uzlů i stavy slotů



K2: Přelož vnitřní uzly, dogeneruj sloty

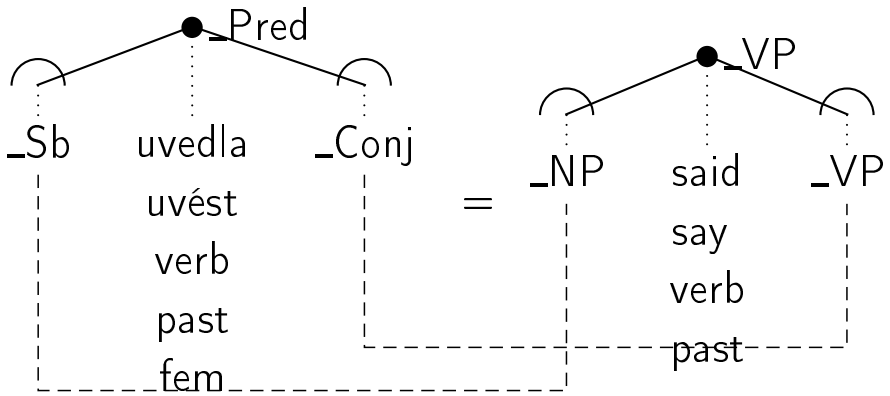
- Zachováváme: tvar stromků, všechny faktory vnitřních uzlů.
- Zdrojové sloty odstraníme, přeložíme a umístíme každý zvlášť.
  - Pro jednoduchost jen v případě okamžité linearizace (netřeba rekonstruovat strukturu).
  - Umisťujeme sloty jen před nebo za vnitřní uzly, nikoli mezi ně.



K3: Překlad uzel po uzlu

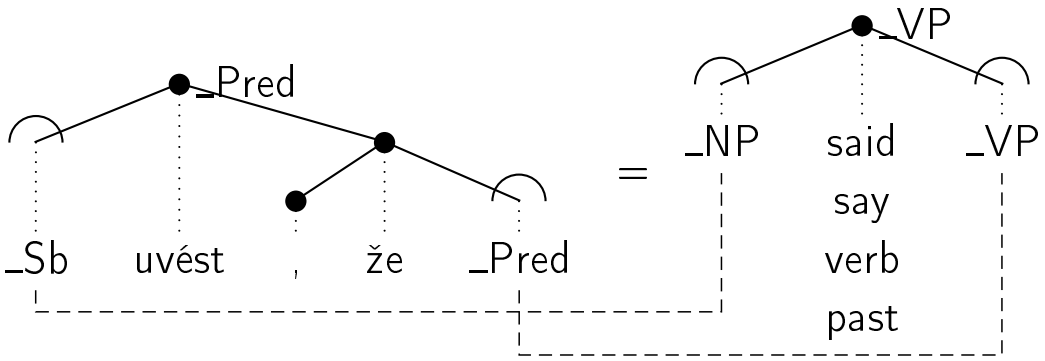
- Stromky omezeny na jeden vnitřní uzel, všechny faktory zachovány.
- Sloty přeložíme každý zvlášť, pro jednoduchost zachováváme pořadí.
- Lze užít i pro generování struktury (otec slotů je evidentní).
- Pokud použijeme jen tuto metodu, nelze měnit počet uzlů.

  ⇒nevhodné pro překlad na analytické rovině



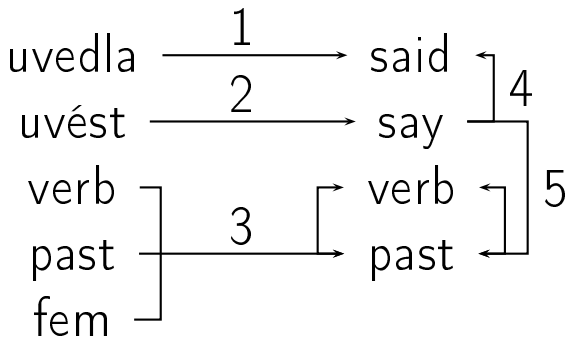
K4: Ignoruj některé vstupní faktory

- Odhlédneme od některých vstupních atributů ⇒ omezíme bohatost vstupních struktur.
- Výstupní faktory generujeme všechny (tj. hádáme hodnoty dle kontextu).



K5: Vícefaktorový překlad (uzel po uzlu)

- Analogie frázového překladu o více faktorech (Koehn and Hoang, 2007).
- Konfigurací dána posloupnost kroků:
  - **Překladové kroky** převádějí vstupní faktory na výstupní.
  - **Generující kroky** svazují výstupní faktory mezi sebou.
  - Pořadí důležité s ohledem na omezený zásobník částečných hypotéz.
- Zatím uplatňujeme jen v překladu uzel po uzlu, nevzniká konflikt struktur.



Kombinace modelů

- Pravděpodobností model STSG rozšířen do log-lineární kombinace modelů:

$$\text{nejlepší derivace } \hat{\delta} = \underset{\delta \in \Delta(T_1)}{\operatorname{argmax}} \exp\Big(\sum_{m=1}^{M} \lambda_m h_m(\delta)\Big) \tag{1}$$

$$\text{místo } \hat{\delta} = \underset{\delta \in \Delta(T_1)}{\operatorname{argmax}}\, p(t^0_{1:2}|Start_{1:2}) * \prod_{i=1}^{k} p(t^k_{1:2}|q^k_{1:2}) \tag{2}$$

- Konfigurace určuje, které modely v jakém pořadí zapojit.
  - Např. přednostně věrný překlad stromků, nelze-li překládej uzel od uzlu.
- Váhy λ_m pro souběžně užité komponenty volíme pro nejlepší skóre (MERT).
  - Aktuálně jen zkouším několik málo bodů.
  - Připraveno hledání optima dvěma metodami: (Och, 2003) a (Smith and Eisner, 2006)

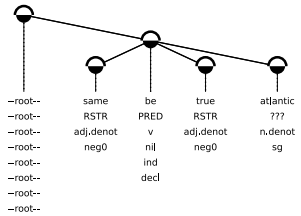
Poznámky k implementaci

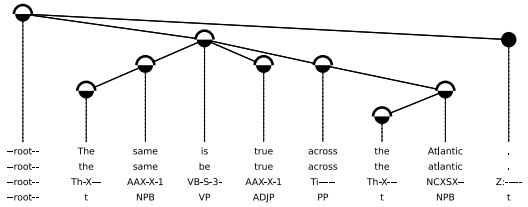
- Externí hašování překladových slovníků stromečků (pomocí Tinycdb, rychlejší implementace GDBM).
- Nově i externí sběr frekvencí (mergesort) ⇒ neomezená velikost slovníku.
- Struktura cílového stromu může být rovnou linearizována.
  - Lze užít n-gramový jazykový model už při budování hypotéz (užívám IrstLM).
  - Při generování struktury užívám hranový jazykový model:
    ⇒ preferuje pravděpodobnější kombinace otec-syn (konfigurovatelné faktory).

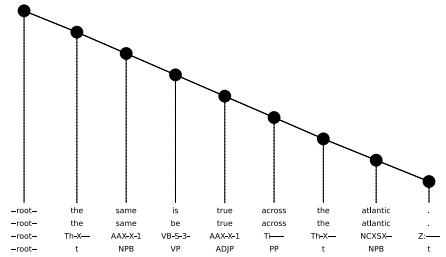
- Implementováno v Mercury (Somogyi *et al.*, 1995).
- Paralelní sběr četností i překlad na Sun Grid Engine.

Dosavadní výsledky (BLEU)

Roviny \ Jazykové modely	žádný	n-gram/*binode*
epcp bez faktorů	8.65±0.55	10.90±0.63
eaca bez faktorů	6.59±0.52	8.75±0.61
etca bez faktorů	-	6.30±0.57
etct s faktory, zachovává strukturu	5.31±0.53	*5.61±0.50*
eact, faktory jen na vstupu, výstup atomický	-	*3.03±0.32*
etct, základní STSG (bez faktorů), všechny atributy	1.61±0.33	*2.56±0.35*
etct, základní STSG (bez faktorů), jen t-lemata	0.67±0.19	-

t:


a:


p:


Diskuse: proč mi t-rovina nepomáhá

- **Kumulace chyb** každého kroku analýzy:
  - Např. $93\,\% * 85\,\% * 93\,\% * 92\,\% = 67\,\%$.
- **Výrazná ztráta dat** kvůli neparalelním strukturám:
  - Stačí jedna chyba v českém či anglickém parsingu nebo slovním zarovnání $\Rightarrow$ nelze extrahovat.
  - Volný překlad „včerejší jednání“ místo „meeting yesterday“ $\Rightarrow$ nutno extrahovat po slovech.
- **Kombinatorická exploze** při generování výstupních atributů:
  - Nejprve stavím celé stromečky, včetně všech atributů. Spojuji později.
  - Bez kontextu okolních uzlů těžko rozhodovat detailní atributy, kombinací je moc.

  $\Rightarrow$ lexikální varianty často odsunuty ze zásobníku.

  $\Rightarrow$ n-best list je pestrý v nepodstatných atributech.
- **Deterministické generování**:
  - Připraveno pro ruční stromy, přeložené automatické mají spoustu chyb.
  - Nevyužívá n-gramový jazykový model.

Vztah k TectoMT

- TectoMT je obecná platforma, všechny analýzy dat jsem dělal v TectoMT.

- Můj transfer má jako jednotku dvojici stromečků se stejnými sloty.
- V TectoMT zatím povinně zachována struktura t-stromu.

- Můj transfer obecně převádí závislostní strom na závislostní strom.
- TectoMT specializováno na t-rovinu, spousta věcí zadrátována.

- Můj transfer chce být uniformní, v TectoMT řada dílčích heuristik.

  Možná víc ctím poučku, že je lepší specifické podrobnosti uchovávat v datech než v kódu.

Srovnání strukturního a frázového překladu

Metoda	Jazykový model	BLEU
Moses P+K, CzEng	4-gramy slov + 7-gramy značek, SYN2006	15.3±0.9
Moses P+K, CzEng	3-gramy slov + 7-gramy značek	14.2±0.7
Moses P+K	3-gramy slov + 7-gramy značek	13.9±0.9
Moses P	3-gramový	12.9±0.6
epcp bez faktorů	3-gramový	10.9±0.6
epcp bez faktorů	žádný	8.7±0.6
nejlepší etct	binodový	5.6±0.5

- Moses lepší než „epcp“:
  - „epcp“ neumožňuje měnit pořadí frází.
  - Moses má řádně implementován MERT (optimalizace vah modelu na BLEU).
- n-gramový LM očividně pomáhá, ale i „epcp“ bez LM > „etct“.

Shrnutí hloubkového překladu

- Hloubkový syntaktický rozbor nabízí naději dělat překlad lépe.
- Složitější systém má však více volných parametrů:
  - Jak přesně zadefinovat jednotlivé mezistupně.
  - Jaké komponenty zvolit pro dosažení mezistupňů, a jak je nakonfigurovat.

  $\Rightarrow$ (Příliš) velký prostor pro experimentování.
- Chyby jednotlivých komponent se kumulují.
  - Bylo by nutné optimalizovat přes celý řetízek.

  $\Rightarrow$ Komponenty by musely předávat ne jednu, ale více variant výstupu.
- Přijetí nerealistických předpokladů vede k zředění dat.
- Cena za replikaci je velmi vysoká, přidat např. němčinu by bylo hodně práce.

$\Rightarrow$ Zatím vede frázový překlad.
(Optimisté ještě úplně nevymřeli.)

ACL WMT08: Lidská hodnocení překladu

Procento vět, kdy byl daný systém hodnocen stejně nebo lépe než ostatní systémy:

Systém	Commentary (v doméně)	News (mimo doménu)
Můj P+K (cu-bojar)	**71.4 %**	63.4 %
PC Translator	66.3 %	**71.5 %**
TectoMT (cu-tectomt)	48.8 %	49.4 %
Moses, bez faktorů (uedin)	48.6 %	50.2 %

- TectoMT (Žabokrtský *et al.*, 2008) srovnatelné s prostým frázovým překladem.
  - Transfer na t-rovině jako posloupnost ručně vyladěných kroků, nikoli uniformní prohledávání.

  ⇒ Tektogramatická rovina stále dává dobrou naději (nebo je salát jako salát).
- Frázový překlad o více faktorech (a s více daty) podstatně lepší.
  - Ve známé doméně zvítězil nad komerčním systémem, v obecném překladu horší.

ACL WMT09: Žhavé novinky aj→čj

Systém	BLEU	NIST
Moses T	14.24	5.175
Moses T+C	13.86	5.110
Google	13.59	4.964
U. of Edinburgh	13.55	5.039
Moses T+C+C&T+T+G 84k	10.01	4.360
Eurotran XP	09.51	4.381
PC Translator	09.42	4.335
TectoMT	07.29	4.173

- Filmové titulky (68 % vět, 50 % tokenů!) nestály za nárůst náročnosti.
- Na novinách a s velkým českým LM vícefaktorový překlad už nepomáhá.

Problémy BLEU

- Málo koreluje s lidmi (Kos and Bojar, 2009; Callison-Burch *et al.*, 2008)
- Velmi citlivé na tokenizaci:

  Zejména pozor na nesprávnou tokenizaci češtiny cizími skripty.
- Různé implementace „délky reference“:

  Papineni *et al.* (2002) neříká nic. Lze volit nejkratší, nejdelší, průměrnou, nejbližší (větší či menší!).

- Neporovnatelné napříč jazyky.
- Neporovnatelné při různém počtu referenčních překladů.
- Neporovnatelné při různých implementacích.
- Neporovnatelné při různých množinách vět.

I v rámci jednoho systému a jedné testovací sady vět:

- Zjistěte si empirický interval spolehlivosti (Koehn, 2004).

  Dá představu o výkyvech BLEU vlivem volby konkrétních vět v dané doméně.
- Opatrně s testy signifikance, někdy rozporné (Chiang *et al.*, 2008).

Co zlepšuje BLEU v překladu čj→aj

Souhrn starších experimetů, podrobněji viz Bojar *et al.* (2006) nebo Bojar (2006)

vhodné zarovnání po slovech	+1.5 až +2.0
morfologické předzpracování (stemming)	+1.0
morfologické předzpracování (plná lemmatizace)	+1.5
přidání nepředzpracovaného slovníku	+0.2
dodatečné paralelní texty, použity i v jazykovém modelu	+0.7 až +1.7
větší jazykový model v doméně	+2.1 až +3.4
ještě větší, ale obecný jazykový model	+4.6
dodatečné paralelní texty, ale jazykový model (větší) v doméně	+5.0 až +6.0
pravidlové zpracování číselných výrazů	+0.5
umělé zvětšování trénovacích dat na základě syntaktické struktury	+0.5
oprava evidentních prohřešků proti referenčním překladům	**+1.0 až +1.5**
sjednocení tokenizace v hypotéze a referenčních překladech	**+10.0**

Problémy symetrizace a extrakce

Teoretické: Extrakce nesouvisí přímo s nasazením v překladu.

- Kde mám příkladové věty lámat na fráze?
- Proč budovat obří slovník, když použiju zlomek?
  - Omezená slovní zásoba v testovací množině.
  - Spousta jednoduše nesmyslných frází, ale „občas se mohou hodit“.
- STSG s EM trénováním tímto netrpí, ale předpokládá nerealistické:
  - Paralelní konstrukce stromů v trénovacích datech.
  - Zákaz adjunkce $\Rightarrow$ ředění dat volnými doplněními.

Praktické:

- Průnik vypadá nejspolehlivěji, a asi nejvhodnější pro extrakci slovníku.
- Čj$\rightarrow$aj: gdf nebo sjednocení místo průniku: +1.5 až +2.0 BLEU
- Aj$\rightarrow$hi: gdfa místo gdf: **+5.0 až +6.0 BLEU**
  14.97$\pm$1.46 $\rightarrow$ 21.01$\pm$2.18; 13.82$\pm$1.46 $\rightarrow$ 18.88$\pm$2.05
- Extrahovat fráze konzistentní, nebo jen neodporující alignmentu?

Shrnutí a pokus o nadhled

Při snaze o strojový překlad se naučíte:

- Rozdělit složité úlohy na částečky a přispět částečkami (např. do TectoMT),
- Počítat, abyste nehledali jehly v horách sena (Pravděpodobnost a statistika!),
- Navrhovat datové struktury, abyste zvládli terabajty dat,

  Text na českém webu ∼1.5 TB, jeden experiment s frázovým překladem 1-2 GB ale třeba i 10 GB.
- Programovat, abyste zvládli stovky počítačů najednou,
  - Unix/Linux je naprosto nutný, Sítě a Internet velmi užitečné.
  - ÚFAL sám má 160 CPU, 40 počítačů s 32 GB RAM.
- Podstatná je koncová metrika:
  - „Lepší slovník“ je vachrlatý pojem. Víc se líbí lidem? Lexikografům?
  - Zlepšení alignment error rate (AER, tj. proti ručnímu alignmentu) nevede k lepšímu BLEU.
  - Zlepšení BLEU nemusí vést k lepšímu lidskému hodnocení.
- Ďábel může být v detailu (tokenizace pro BLEU, ale i gdf/gdfa pro Hindí).
- Jednodušší je lepší: Méně parametrů ⇒ snadněji optimalizovatelné.

Zamyšlení: Statistik vs. lingvista

Modelový lingvista usiluje o popis jazyka, vysvětlení toho, co se děje, když si lidé rozumějí.

Modelový statistik usiluje o řešení dané úlohy s co nejmenší chybou.

- statistik potřebuje úlohu
- statistik potřebuje metriku
- statistik ctí princip Occamovy břitvy
- statistik zohledňuje zákon klesajícího zisku

Motto: Od začátku pracuj od konce.

Reklama závěrem: `http://ufal.mff.cuni.cz/euromatrix/mtmarathon/`

Literatura

Ondřej Bojar, Evgeny Matusov, and Hermann Ney. Czech-English Phrase-Based Machine Translation. In *FinTAL 2006*, volume LNAI 4139, pages 214–224, Turku, Finland, August 2006. Springer.

Ondřej Bojar. Strojový překlad: zamyšlení nad účelností hloubkových jazykových analýz. In *MIS 2006*, pages 3–13, Josefův Důl, Czech Republic, January 2006. MATFYZPRESS.

Chris Callison-Burch, Cameron Fordyce, Philipp Koehn, Christof Monz, and Josh Schroeder. Further meta-evaluation of machine translation. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pages 70–106, Columbus, Ohio, June 2008. Association for Computational Linguistics.

David Chiang, Steve DeNeefe, Yee Seng Chan, and Hwee Tou Ng. Decomposability of translation metrics for improved evaluation and efficient algorithms. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 610–619, Honolulu, Hawaii, October 2008. Association for Computational Linguistics.

Martin Čmejrek. *Using Dependency Tree Structure for Czech-English Machine Translation*. PhD thesis, ÚFAL, MFF UK, Prague, Czech Republic, 2006.

Philipp Koehn and Hieu Hoang. Factored Translation Models. In *Proc. of EMNLP*, 2007.

Philipp Koehn. Statistical Significance Tests for Machine Translation Evaluation. In *Proceedings of EMNLP 2004*, Barcelona, Spain, 2004.

Kamil Kos and Ondřej Bojar. Evaluation of Machine Translation Metrics for Czech as the Target Language. *Prague Bulletin of Mathematical Linguistics*, 90, 2009. in print.

Franz Josef Och. Minimum Error Rate Training in Statistical Machine Translation. In *Proc. of the Association for Computational Linguistics*, Sapporo, Japan, July 6-7 2003.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a Method for Automatic Evaluation of Machine

Literatura

Translation. In *ACL 2002, Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, 2002.

David A. Smith and Jason Eisner. Minimum-Risk Annealing for Training Log-Linear Models. In *Proceedings of the International Conference on Computational Linguistics and the Association for Computational Linguistics (COLING-ACL), Companion Volume*, pages 787–794, Sydney, July 2006.

Zoltan Somogyi, Fergus Henderson, and Thomas Conway. Mercury: An Efficient Purely Declarative Logic Programming Language. In *Proceedings of the Australian Computer Science Conference*, pages 499–512, Glenelg, Australia, February 1995.

Zdeněk Žabokrtský, Jan Ptáček, and Petr Pajas. TectoMT: Highly Modular Hybrid MT System with Tectogrammatics Used as Transfer Layer. In *Proc. of the ACL Workshop on Statistical Machine Translation*, pages 167–170, Columbus, Ohio, USA, 2008.