Machine Translation 3:
Linguistics or Neural Networks

Ondřej Bojar
bojar@ufal.mff.cuni.cz
Institute of Formal and Applied Linguistics
Faculty of Mathematics and Physics
Charles University, Prague

December 2016
Outline of Lectures on MT

1. Introduction
   • Why is MT difficult.
   • MT evaluation.
   • Approaches to MT.
   • First peek into phrase-based MT, word alignment.

   • Phrase-based MT.
   • Hierarchical MT.

3. Linguistics in MT; or not.
   • Factored phrase-based MT.
   • Syntactic MT.
   • Neural MT.
Outline of MT Lecture 3

1. Linguistic features for tokens.
   - Factored phrase-based MT.
2. Linguistic structure to organize search.
   - Non-projectivity.
   - TectoMT: transfer-based deep-syntactic model.
3. Combination to make it actually work.
4. Neural Networks.
   - End-to-end continuous-space modelling.
   - Little linguistics (yet).
# Morphological Richness (in Czech)

<table>
<thead>
<tr>
<th>Czech</th>
<th>English</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rich morphology</td>
<td>$\geq 4,000$ tags possible</td>
</tr>
<tr>
<td></td>
<td>$\geq 2,300$ tags seen</td>
</tr>
<tr>
<td>Word order</td>
<td>free</td>
</tr>
</tbody>
</table>

### News Commentary Corpus

<table>
<thead>
<tr>
<th>Czech</th>
<th>English</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentences</td>
<td>55,676</td>
</tr>
<tr>
<td>Tokens</td>
<td>1.1M</td>
</tr>
<tr>
<td>Vocabulary (word forms)</td>
<td>91k</td>
</tr>
<tr>
<td>Vocabulary (lemmas)</td>
<td>34k</td>
</tr>
</tbody>
</table>

Czech tagging and lemmatization: Hajič and Hladká (1998)

English tagging (Ratnaparkhi, 1996) and lemmatization (Minnen et al., 2001).

December 2016

MT3: Linguistics or Neural Networks
Morphological Explosion in Czech

MT chooses output words in a form:

- Czech nouns and adjs.: 7 cases, 4 genders, 3 numbers, . . .
- Czech verbs: gender, number, aspect (im/perfective), . . .

<table>
<thead>
<tr>
<th>I</th>
<th>saw</th>
<th>two</th>
<th>green</th>
<th>striped</th>
<th>cats</th>
</tr>
</thead>
<tbody>
<tr>
<td>já</td>
<td>pila</td>
<td>dva</td>
<td>zelený</td>
<td>pruhovaný</td>
<td>kočky</td>
</tr>
<tr>
<td></td>
<td>pily</td>
<td>dvě</td>
<td>zelená</td>
<td>pruhovaná</td>
<td>koček</td>
</tr>
<tr>
<td>. . .</td>
<td>dvou</td>
<td>zelené</td>
<td>pruhované</td>
<td>kočkám</td>
<td></td>
</tr>
<tr>
<td>viděl</td>
<td>dvěma</td>
<td>zelení</td>
<td>pruhovaní</td>
<td>kočkách</td>
<td></td>
</tr>
<tr>
<td>viděla</td>
<td>dvěmi</td>
<td>zeleného</td>
<td>pruhovaného</td>
<td>kočkami</td>
<td></td>
</tr>
<tr>
<td>. . .</td>
<td></td>
<td>zelených</td>
<td>pruhovaných</td>
<td></td>
<td></td>
</tr>
<tr>
<td>uviděl</td>
<td></td>
<td>zelenému</td>
<td>pruhovanému</td>
<td></td>
<td></td>
</tr>
<tr>
<td>uviděla</td>
<td></td>
<td>zeleným</td>
<td>pruhovaným</td>
<td></td>
<td></td>
</tr>
<tr>
<td>. . .</td>
<td></td>
<td>zelenou</td>
<td>pruhovanou</td>
<td></td>
<td></td>
</tr>
<tr>
<td>viděl jsem</td>
<td></td>
<td>zelenými</td>
<td>pruhovanými</td>
<td></td>
<td></td>
</tr>
<tr>
<td>viděla jsem</td>
<td></td>
<td>. . .</td>
<td>. . .</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Morphological Explosion Elsewhere

Compounding in German:

- Rindfleischetikettierungsüberwachungsaufgabenübertragungsgesetz.
  “beef labelling supervision duty assignment law”

Agglutination in Hungarian or Finnish:

- istua “to sit down” (istun = “I sit down”)
- istahtaa “to sit down for a while”
- istahdan “I’ll sit down for a while”
- istahtaisin “I would sit down for a while”
- istahtaisinko “should I sit down for a while?”
- istahtaisinkohan “I wonder if I should sit down for a while”
Possible translations differing in morphology:

<table>
<thead>
<tr>
<th>two</th>
<th>green</th>
<th>striped</th>
<th>cats</th>
</tr>
</thead>
<tbody>
<tr>
<td>dvou</td>
<td>zelená</td>
<td>pruhovaný</td>
<td>kočkách</td>
</tr>
<tr>
<td>dva</td>
<td>zelené</td>
<td>pruhované</td>
<td>kočky</td>
</tr>
<tr>
<td>dvě</td>
<td>zelené</td>
<td>pruhované</td>
<td>kočky</td>
</tr>
<tr>
<td>dvěma</td>
<td>zeleným</td>
<td>pruhovaným</td>
<td>kočkám</td>
</tr>
</tbody>
</table>

- 3-gram LM too weak to ensure agreement.
- 3-gram LM possibly already too sparse!
Explicit Morphological Target Factor

- Add morphological tag to each output token:

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>two</td>
<td>green</td>
<td>striped</td>
<td>cats</td>
</tr>
<tr>
<td>dvou</td>
<td>zelená</td>
<td>pruhovaný</td>
<td>kočkách</td>
</tr>
<tr>
<td></td>
<td>fem-loc</td>
<td>neut-acc</td>
<td>masc-nom-sg</td>
</tr>
<tr>
<td>dva</td>
<td>zelené</td>
<td>pruhované</td>
<td>kočky</td>
</tr>
<tr>
<td></td>
<td>masc-nom</td>
<td>masc-nom</td>
<td>masc-nom</td>
</tr>
<tr>
<td></td>
<td>fem-nom</td>
<td>fem-nom</td>
<td>fem-nom</td>
</tr>
<tr>
<td>dvě</td>
<td>zelené</td>
<td>pruhované</td>
<td>kočky</td>
</tr>
<tr>
<td></td>
<td>fem-nom</td>
<td>fem-nom</td>
<td>fem-nom</td>
</tr>
<tr>
<td></td>
<td>fem-acc</td>
<td>fem-acc</td>
<td>fem-acc</td>
</tr>
<tr>
<td>dvěma</td>
<td>zeleným</td>
<td>pruhovaným</td>
<td>kočkám</td>
</tr>
<tr>
<td></td>
<td>fem-dat</td>
<td>fem-dat</td>
<td>fem-dat</td>
</tr>
</tbody>
</table>
Advantages of Explicit Morphology

- LM over morphological tags generalizes better.
  - \( p(\text{dvě kočkách}) < p(\text{dvě kočky}) \ldots \) surely
    
    But we would need to see all combinations of \text{pruhovaný} and \text{kočka}!
    
    \( \Rightarrow \) Better to ask if \( p(\text{fem-nom fem-loc}) < p(\text{fem-nom fem-nom}) \)
    
    which is trained on any feminine adj+noun.

- But still does not solve everything.
  - \( p(\text{dvě zelené}) \gtrless p(\text{dva zelené}) \ldots \) bad question anyway!
    
    Not solved by asking if \( p(\text{fem-nom fem-nom}) \gtrless p(\text{masc-nom masc-nom}) \).

- Tagset size smaller than vocabulary.
  
  \( \Rightarrow \) can afford e.g. 7-grams:
  
  \( p(\text{masc-nom fem-nom fem-nom}) < p(\text{fem-nom fem-nom fem-nom}) \)
Factored Phrase-Based MT

- Both input and output words can have more factors.
- Arbitrary number and order of:

  **Mapping steps** (→)
  
  Translate (phrases of) source factors to target factors.
  
  two green → dvě zelené

  **Generation steps** (↓)
  
  Generate target factors from target factors.
  
  dvě → *fem-nom*; dva → *masc-nom*
  
  ⇒ Ensures “vertical” coherence.

  **Target-side language models** (+LM)
  
  Applicable to various target-side factors.
  
  ⇒ Ensures “horizontal” coherence.  

(Koehn and Hoang, 2007)
Factored Phrase-Based MT

See slides by Philipp Koehn (Fri Jan 30, 2009, pp.28–75):

- Example
- Model and Training
- Decoding
- Experiments
  - Alternative Decoding Paths
## Translation Scenarios for En → Cs

### Vanilla

<table>
<thead>
<tr>
<th>English</th>
<th>Czech</th>
</tr>
</thead>
<tbody>
<tr>
<td>form</td>
<td>form</td>
</tr>
<tr>
<td>lemma</td>
<td>lemma</td>
</tr>
<tr>
<td>morphology</td>
<td>morphology</td>
</tr>
</tbody>
</table>

### Translate+Check (T+C)

<table>
<thead>
<tr>
<th>English</th>
<th>Czech</th>
</tr>
</thead>
<tbody>
<tr>
<td>form</td>
<td>form</td>
</tr>
<tr>
<td>lemma</td>
<td>lemma</td>
</tr>
<tr>
<td>morphology</td>
<td>morphology</td>
</tr>
</tbody>
</table>

### Translate+2·Check (T+C+C)

<table>
<thead>
<tr>
<th>English</th>
<th>Czech</th>
</tr>
</thead>
<tbody>
<tr>
<td>form</td>
<td>form</td>
</tr>
<tr>
<td>lemma</td>
<td>lemma</td>
</tr>
<tr>
<td>morphology</td>
<td>morphology</td>
</tr>
</tbody>
</table>

### 2·Translate+Generate (T+T+G)

<table>
<thead>
<tr>
<th>English</th>
<th>Czech</th>
</tr>
</thead>
<tbody>
<tr>
<td>form</td>
<td>form</td>
</tr>
<tr>
<td>lemma</td>
<td>lemma</td>
</tr>
<tr>
<td>morphology</td>
<td>morphology</td>
</tr>
</tbody>
</table>
## Factored Attempts (WMT09)

<table>
<thead>
<tr>
<th>Sents</th>
<th>System</th>
<th>BLEU</th>
<th>NIST</th>
<th>Sent/min</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.2M</td>
<td>Vanilla</td>
<td>14.24</td>
<td>5.175</td>
<td>12.0</td>
</tr>
<tr>
<td>2.2M</td>
<td>T+C</td>
<td>13.86</td>
<td>5.110</td>
<td>2.6</td>
</tr>
<tr>
<td>84k</td>
<td>T+C+C&amp;T+T+G</td>
<td>10.01</td>
<td>4.360</td>
<td>4.0</td>
</tr>
<tr>
<td>84k</td>
<td>Vanilla MERT</td>
<td>10.52</td>
<td>4.506</td>
<td>–</td>
</tr>
<tr>
<td>84k</td>
<td>Vanilla even weights</td>
<td>08.01</td>
<td>3.911</td>
<td>–</td>
</tr>
</tbody>
</table>

- In WMT07, T+C worked best.  
  + fine-tuned tags helped with small data (Bojar, 2007).
- In WMT08, T+C was worth the effort (Bojar and Hajič, 2008).
- In WMT09, our computers could handle 7-grams of forms.  
  ⇒ No gain from T+C.
- T+T+G too big to fit and explodes the search space.  
  ⇒ Worse than Vanilla trained on the same dataset.
Factored models are "synchronous", i.e. Moses:
1. Generates fully instantiated “translation options”.
2. Appends translation options to extend “partial hypothesis”.
3. Applies LM to see how well the option fits the previous words.

There are too many possible combinations of lemma+tag.
⇒ Less promising ones must be pruned.
  Pruned before the linear context is available.

Hieu Hoang wasted a year on trying asynchronous factors.
  Pruning hard to design (no clear comparison for partial translation options).
A different system (Bojar and Týnovský, 2009) uses “delayed factors”.
  The final value generated only after the full hypothesis is ready.
Tamchyna et al. (2016) use a discriminative classifier.
**A Fix: Reverse Self-Training**

Goal: Learn from monolingual data to produce **new** target-side word forms in **correct** contexts.

<table>
<thead>
<tr>
<th>Source English</th>
<th>Target Czech</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Para 126k</strong></td>
<td></td>
</tr>
<tr>
<td>a cat chased. . .</td>
<td>kočka honila. . .</td>
</tr>
<tr>
<td></td>
<td>kočka honit... (lem.)</td>
</tr>
<tr>
<td>I saw a cat</td>
<td>viděl jsem kočku</td>
</tr>
<tr>
<td></td>
<td>vidět být kočka (lem.)</td>
</tr>
<tr>
<td><strong>Mono 2M</strong></td>
<td></td>
</tr>
<tr>
<td>?</td>
<td>četl jsem o kočce</td>
</tr>
<tr>
<td></td>
<td>číst být o kočka (lem.)</td>
</tr>
</tbody>
</table>

Use reverse translation

I read about a cat  ← backed-off by lemmas.

⇒ New phrase learned: “about a cat” = “o kočce”.

December 2016 MT3: Linguistics or Neural Networks
The Back-off to Lemmas

- The key distinction from self-training used for domain adaptation (Bertoldi and Federico, 2009; Ueffing et al., 2007).
- We use simply “alternative decoding paths” in Moses:
  
  \[
  \begin{array}{|c|c|c|}
  \hline
  \text{Czech} & \text{English} & +LM \\
  \text{form} & \rightarrow & \text{form} \\
  \hline
  \end{array}
  \quad \text{or} \quad
  \begin{array}{|c|c|c|}
  \hline
  \text{Czech} & \text{English} & +LM \\
  \text{lemma} & \rightarrow & \text{form} \\
  \hline
  \end{array}
  \]

- Other languages (e.g. Turkish, German) need different back-off techniques:
  - Split German compounds.
  - Separate and allow to ignore Turkish morphology.
Small Para, Increasing Mono

Monolingual data (mils of sents.)

BLEU

Monolingualse data (mils of sents.)

Mono LM and TM

Mono LM

December 2016 MT3: Linguistics or Neural Networks
Increasing Para, Fixed Mono

![Graph showing BLEU scores for different types of data. The graph compares % Test Forms Covered against Parallel data (mils of sents.). The x-axis represents Parallel data in millions of sentences, while the y-axis shows BLEU scores. There are different lines representing: Mono LM, Mono LM and TM, Parallel, and Parallel and Mono. The graph indicates improvements in BLEU scores with increased parallel data.]
Remaining Morphological Errors

• What are the remaining errors?
• Analyzed 77 Verb-Modifier pairs in 15 source sentences:

<table>
<thead>
<tr>
<th>Translation of</th>
<th>Verb</th>
<th>Modifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>... preserves meaning</td>
<td>56%</td>
<td>79%</td>
</tr>
<tr>
<td>... is disrupted</td>
<td>14%</td>
<td>12%</td>
</tr>
<tr>
<td>... is missing</td>
<td>27%</td>
<td>1%</td>
</tr>
<tr>
<td>... is unknown (not translated)</td>
<td>0%</td>
<td>5%</td>
</tr>
</tbody>
</table>

• Verbs are often dropped.
• Even when Verb&Mod correct (line 1), 56% of cases are non-grammatical or meaning-disrupted relations.
Sample Errors

<table>
<thead>
<tr>
<th>Input:</th>
<th>Keep on investing.</th>
</tr>
</thead>
<tbody>
<tr>
<td>MT output:</td>
<td>Pokračovalo investování. (grammar correct here!)</td>
</tr>
<tr>
<td>Gloss:</td>
<td>Continued investing. (Meaning: The investing continued.)</td>
</tr>
<tr>
<td>Correct:</td>
<td>Pokračujte v investování.</td>
</tr>
</tbody>
</table>

⇒ language model misled us ⇒ need to include source valency information.

<table>
<thead>
<tr>
<th>Input:</th>
<th>brokerage firms rushed out ads . . .</th>
</tr>
</thead>
<tbody>
<tr>
<td>MT Output:</td>
<td>brokerské firmy vyběhl reklamy</td>
</tr>
<tr>
<td>Gloss:</td>
<td>brokerage firms ran ads</td>
</tr>
<tr>
<td>Correct:</td>
<td>brokerské firmy vychršly reklamy</td>
</tr>
<tr>
<td>Comprehensible:</td>
<td>brokerské firmy vyběhly s reklamami</td>
</tr>
</tbody>
</table>

Target-side data may be rich enough to learn: vyběhnout–s–instr
Not rich enough to learn all morphological and lexical variants:

Summary So Far

• Target-side rich morphology causes data sparseness.
• Factored setups compact the sparseness. . .  
  . . .  but the search space is likely to explode at runtime.
• Explosion contained thanks to pruning. 
  . . .  but the pruning happens without linear context 
  ⇒ high risk of search errors.

One of possible promising techniques for handling sparseness and avoiding the explosion:

• Reverse self-training (Bojar and Tamchyna, 2011).

What remains are errors in expressing sentence structure. 
⇒ Let’s try handling syntax properly.
Constituency vs. Dependency Trees

Constituency trees (CFG) represent only bracketing: which adjacent constituents are glued tighter to each other.

Dependency trees represent which words depend on which. Usually, some agreement/conditioning happens along the edge.

Constituency

John (loves Mary)
John \textit{VP}(loves Mary)

\begin{tikzpicture}
  \node (S) at (0,0) {\textit{S}};
  \node (NP) at (-2,-1) {\textit{NP}} edge (S);
  \node (VP) at (0,-1) {\textit{VP}} edge (S);
  \node (John) at (-2,-2) {John} edge (NP);
  \node (loves) at (0,-2) {loves} edge (VP);
  \node (NP) at (-1,-2) {\textit{NP}} edge (loves);
  \node (Mary) at (-1,-3) {Mary} edge (NP);
\end{tikzpicture}

Dependency

\begin{tikzpicture}
  \node (loves) at (0,0) {loves};
  \node (John) at (-2,0) {John} edge (loves);
  \node (Mary) at (0,0) {Mary} edge (loves);
\end{tikzpicture}
What Dependency Trees Tell Us

Input: The **grass** around your house should be **cut** soon.

Google: **Trávu** kolem vašeho domu by se měl **snížit** brzy.

- **Bad lexical choice** for *cut* = *sekat/snížit/krájet/řezat/...*
  - Due to long-distance dependency with *grass*.
  - One can “pump” many words in between.
  - Could be handled by full source-context (e.g. maxent) model.
- **Bad case of tráva.**
  - Depends on the chosen active/passive form:
    
    \[
    \begin{array}{ll}
    \text{active} \Rightarrow \text{accusative} & \text{passive} \Rightarrow \text{nominative} \\
    \text{trévu} \ldots \text{by} \text{ste} \text{ se měl posekat} & \text{tréva} \ldots \text{by} \text{ se měla posekat} \\
    \text{tréva} \ldots \text{by} \text{ měla být posekána} & \\
    \end{array}
    \]

Examples by Zdeněk Žabokrtský, Karel Oliva and others.
Tree vs. Linear Context

The grass around your house should be cut soon

- Tree context (neighbours in the dependency tree):
  - is better at predicting lexical choice than $n$-grams.
  - often equals linear context:
    Czech manual trees: 50% of edges link neighbours,
    80% of edges fit in a 4-gram.

- Phrase-based MT is a very good approximation.
- Hierarchical MT can even capture the dependency in one phrase:

$$X \rightarrow <\text{the grass } X \text{ should be cut, trávu } X \text{ byste měl posekat }>$$
“Crossing Brackets”

- Constituent outside its father’s span causes “crossing brackets.”
  - Linguists use “traces” (□) to represent this.
- Sometimes, this is not visible in the dependency tree:
  - There is no “history of bracketing”.
  - See Holan et al. (1998) for dependency trees including derivation history.

Despite this shortcoming, CFGs are popular and “the” formal grammar for many. Possibly due to the charm of the father of linguistics, or due to the abundance of dependency formalisms with no clear winner (Nivre, 2005).
Non-Projectivity

= a gap in a subtree span, filled by a node higher in the tree.

Ex. Dutch “cross-serial” dependencies, a non-projective tree with one gap caused by *saw* within the span of *swim*.

\[
\begin{align*}
\ldots & \text{dat Jan kinderen zag zwemmen} \\
\ldots & \text{that John children saw swim} \\
\ldots & \text{that John saw children swim.}
\end{align*}
\]

- 0 gaps $\Rightarrow$ projective tree $\Rightarrow$ can be represented in a CFG.
- $\leq 1$ gap & “well-nested” $\Rightarrow$ mildly context sensitive (TAG).

See Kuhlmann and Möhl (2007) and Holan et al. (1998).
Why Non-Projectivity Matters?

- CFGs cannot handle non-projective constructions:
  Imagine John *grass saw being-cut*!
- No way to glue these crossing dependencies together:
  - Lexical choice:
    \[ X \rightarrow \langle \text{grass} \ X \text{ being-cut, trávu} \ X \text{ sekat} \rangle \]
  - Agreement in gender:
    \[ X \rightarrow \langle \text{John} \ X \text{ saw, Jan} \ X \text{ viděl} \rangle \]
    \[ X \rightarrow \langle \text{Mary} \ X \text{ saw, Marie} \ X \text{ viděla} \rangle \]
- Phrasal chunks can memorize fixed sequences containing:
  - the non-projective construction
  - and all the words in between! (⇒ extreme sparseness)
Is Non-Projectivity Severe?

Depends on the language.

In principle:

- Czech allows long gaps as well as many gaps in a subtree.

In treebank data:

- 23% of Czech sentences contain a non-projectivity.
- 99.5% of Czech sentences are well nested with \( \leq 1 \) gap.

Proti odmítnutí se zítra Petr v práci rozhodl protestovat

Against dismissal aux-refl tomorrow Peter at work decided to object

Peter decided to object against the dismissal at work tomorrow.
Tectogrammatics: Deep Syntax Culminating

Background: Prague Linguistic Circle (since 1926).

Materialized theory — Treebanks:
- Czech: PDT 1.0 (2001), PDT 2.0 (2006)
- Czech-English: PCEDT 1.0 (2004), PCEDT 2.0 (2012)

Practice — Tools:
- parsing Czech to surface: McDonald et al. (2005)
- parsing Czech to deep: Klimeš (2006)
- parsing English to surface: well studied (+rules convert to dependency trees)
- parsing English to deep: heuristic rules (manual annotation in progress)
- generating Czech surface from t-layer: Ptáček and Žabokrtský (2006)
Analytical vs. Tectogrammatical

- #45 To it změnit should change punct
- #45 to změnit should Generic Actor

- hide auxiliary words, add nodes for “deleted” participants
- resolve e.g. active/passive voice, analytical verbs etc.
- “full” t-layer resolves much more, e.g. topic-focus articulation or anaphora
This should be changed.
Czech and English T-Layer

Predicate-argument structure: change_{should}(ACT: someone, PAT: it)
The Tectogrammatical Hope

Transfer at t-layer should be easier than direct translation:

- Reduced structure size (auxiliary words disappear).
- Long-distance dependencies (non-projectivities) solved at t-layer.
- Word order ignored / interpreted as information structure (given/new).
- Reduced vocabulary size (Czech morphological complexity).
- Czech and English t-trees structurally more similar \( \Rightarrow \) less parallel data might be sufficient (but more monolingual).
- Ready for fancy t-layer features: co-reference.

The complications:

- 47 pages documenting data format (PML, XML-based, sort of typed)
- 1200 pages documenting Czech t-structures
  “Not necessary” once you have a t-tree but useful understand or to blame the right people.
“TectoMT Transfer” (1/3)

- **ANALYSIS**
  - deep syntax: tectogrammatical layer
  - shallow syntax: analytical layer
  - morphological layer
  - source language (English)

- **TRANSFER**
  - t-layer

- **SYNTHESIS**
  - a-layer
  - m-layer
  - w-layer
  - target language (Czech)
“TectoMT Transfer” (2/3)

**ANALYSIS**
- tectogrammatical layer
  - fill formems
  - grammatemes
  - build t-tree
  - mark edges to contract
- analytical layer
  - analytical functions
  - parser (McDonald's MST)
  - tagger (Morce)
- morphological layer
  - lemmatization
  - tokenization
  - segmentation

**TRANSFER**
- source language (English)
  - query dictionary
  - use HMTM
- target language (Czech)
  - t-layer
  - fill morphological categories
  - impose agreement
  - add functional words
  - a-layer
    - generate wordforms
  - m-layer
    - concatenate
- w-layer

**SYNTHESIS**
- rule based & statistical blocks

December 2016
MT3: Linguistics or Neural Networks
To learn more: Slides 6–28 by Martin Popel (2009):

- Illustration of TectoMT transfer.
- Analysis of translation errors.
- Hidden Markov Tree Model (HMTM).

**Bad news:** TectoMT alone performs poorly.

- Errors cummulate.
- T-layer does bring its independence assumptions.
- No means for plain copy-paste.
Poor Man’s System Combination

- Translate input with TectoMT.
- Align translation back to source.
- Extract phrases.
- Add as a separate phrase table.
- MERT to find weights of both phrase tables.
TectoMT Brings Phrases

Input  I saw two green striped cats.

TectoMT Output  Viděl jsem dvě zelené pruhované kočky.

Phrases extracted:

I saw = Viděl jsem
I saw two = Viděl jsem dvě
  . . .
two = dvě
two green = dvě zelené
two green striped = dvě zelené pruhované
two green striped cats = dvě zelené pruhované kočky
  . . .
The output of TectoMT covers (most of) the source.

- Long and short phrases, one form only.

<table>
<thead>
<tr>
<th>I saw</th>
<th>two</th>
<th>green</th>
<th>striped</th>
<th>cats</th>
</tr>
</thead>
<tbody>
<tr>
<td>já</td>
<td>pila</td>
<td>dva</td>
<td>zelený</td>
<td>pruhovaný</td>
</tr>
<tr>
<td>pily</td>
<td>dvě</td>
<td>zelená</td>
<td>pruhovaná</td>
<td>koček</td>
</tr>
<tr>
<td>. . .</td>
<td>dvou</td>
<td>zelené</td>
<td>pruhované</td>
<td>kočkám</td>
</tr>
<tr>
<td>viděl</td>
<td>dvěma</td>
<td>zelení</td>
<td>pruhovaní</td>
<td>kočkách</td>
</tr>
<tr>
<td>viděla</td>
<td>dvěmi</td>
<td>zeleného</td>
<td>pruhovaného</td>
<td>kočkami</td>
</tr>
<tr>
<td>. . .</td>
<td></td>
<td>zelených</td>
<td>pruhovaných</td>
<td></td>
</tr>
<tr>
<td><strong>viděl jsem</strong></td>
<td></td>
<td>zelenými</td>
<td>pruhovanými</td>
<td></td>
</tr>
<tr>
<td>viděla jsem</td>
<td></td>
<td>. . .</td>
<td>. . .</td>
<td></td>
</tr>
</tbody>
</table>
TectoMT Brings Phrases

The output of TectoMT covers (most of) the source.

• Long and short phrases, one form only.

<table>
<thead>
<tr>
<th>I</th>
<th>saw</th>
<th>two</th>
<th>green</th>
<th>striped</th>
<th>cats</th>
</tr>
</thead>
<tbody>
<tr>
<td>já</td>
<td>pila</td>
<td>dva</td>
<td>zelený</td>
<td>pruhovaný</td>
<td>kočky</td>
</tr>
<tr>
<td>pily</td>
<td>dvě</td>
<td>zelená</td>
<td>pruhovaná</td>
<td>kočky</td>
<td></td>
</tr>
<tr>
<td></td>
<td>dvě</td>
<td>zelené</td>
<td>pruhované</td>
<td>koček</td>
<td></td>
</tr>
<tr>
<td>viděl</td>
<td>dvou</td>
<td>zelené</td>
<td>pruhované</td>
<td>kočkám</td>
<td></td>
</tr>
<tr>
<td>viděla</td>
<td>dvěma</td>
<td>zelení</td>
<td>pruhovaní</td>
<td>kočkách</td>
<td></td>
</tr>
<tr>
<td></td>
<td>dvěmi</td>
<td>zeleného</td>
<td>pruhovaného</td>
<td>kočkami</td>
<td></td>
</tr>
<tr>
<td>viděl jsem</td>
<td></td>
<td>zelených</td>
<td>pruhovaných</td>
<td></td>
<td></td>
</tr>
<tr>
<td>viděl jsem</td>
<td></td>
<td>zelenými</td>
<td>pruhovanými</td>
<td></td>
<td></td>
</tr>
<tr>
<td>viděla jsem</td>
<td>dvě zelené</td>
<td>pruhované kočky</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>dvě zelené pruhované kočky</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Chimera: Complex Combination

- Input:
  - Famous cases also relate to graphic elements.
- TectoMT translates using deep syntax:
  - Slavné případy se být týkají grafické prvky.
- PBMT adds 200M en-cs sents and 3,6G cs words:
  - Slavné případy se týkají také grafické prvky.
- Automatic error correction for agreement or negation:
  - Slavné případy se týkají také grafických prvků.

(Google: Slavné případy týkat i grafické prvky.)
## Chimera’s Winning Years

<table>
<thead>
<tr>
<th>System</th>
<th>BLEU</th>
<th>TER</th>
<th>Manual</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>WMT13</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+ + +</td>
<td>20.0</td>
<td>0.693</td>
<td>0.664</td>
</tr>
<tr>
<td>+ +</td>
<td>20.1</td>
<td>0.696</td>
<td>0.637</td>
</tr>
<tr>
<td></td>
<td>19.5</td>
<td>0.713</td>
<td>–</td>
</tr>
<tr>
<td><strong>Google Translate</strong></td>
<td>18.9</td>
<td>0.720</td>
<td>0.618</td>
</tr>
<tr>
<td></td>
<td>14.7</td>
<td>0.741</td>
<td>0.455</td>
</tr>
<tr>
<td><strong>WMT14</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+ + +</td>
<td>21.1</td>
<td>0.670</td>
<td>0.373</td>
</tr>
<tr>
<td>UEDIN-UNCONSTR.</td>
<td>21.6</td>
<td>0.667</td>
<td>0.357</td>
</tr>
<tr>
<td>+ +</td>
<td>20.9</td>
<td>0.674</td>
<td>0.333</td>
</tr>
<tr>
<td><strong>Google Translate</strong></td>
<td>20.2</td>
<td>0.687</td>
<td>0.168</td>
</tr>
<tr>
<td></td>
<td>15.2</td>
<td>0.716</td>
<td>–0.177</td>
</tr>
<tr>
<td><strong>WMT15</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+ + +</td>
<td>18.8</td>
<td>0.715</td>
<td>0.686</td>
</tr>
<tr>
<td>+ +</td>
<td>18.7</td>
<td>0.717</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>17.6</td>
<td>0.730</td>
<td>–</td>
</tr>
<tr>
<td><strong>Google Translate</strong></td>
<td>16.4</td>
<td>0.750</td>
<td>0.515</td>
</tr>
<tr>
<td></td>
<td>13.4</td>
<td>0.763</td>
<td>0.209</td>
</tr>
</tbody>
</table>
And Now for Something...

English-German by UEDIN

- 2013: phrase-based SMT = 20.3, syntax-based SMT = 19.4
- 2014: phrase-based SMT = 20.9, syntax-based SMT = 20.2
- 2015: phrase-based SMT = 20.8, syntax-based SMT = 22.0, neural MT = 18.9
...Completely Different

English-German by UEDIN

- Phrase-based SMT
- Syntax-based SMT
- Neural MT

2013 2014 2015 2016

December 2016 MT3: Linguistics or Neural Networks
Neural Machine Translation

... see slides 2-20 and 42-51 by Rico Sennrich.

or

... see the 234 slides (ACL 2016 tutorial, 58MB):

https://sites.google.com/site/acl16nmt/

The basics of NMT are here:

• slides 14-19, 24-25: NMT for one word, overview.
• slides 47-53: Recurrent neural LM.
• slides 84-95: Encoder-decoder, decoding.
• slides 130-140: Encoder-decoder with attention.
• ... but also the basics of NN, e.g. GRU (slides 72-79).
Linguistic features added:
• as factors (word-level annotations) to phrase-based MT
• (as syntactic labels last week)
• as deep syntax, organizing the whole process.

... always very high risk of paying for unjustified assumptions.

Neural networks:
• get rid of most of the assumptions.
• but are very expensive to train.
References


