Lab Outline

• Brief overview of phrase-based machine translation.
• Set up your own environment for experiments.
• Inspect an existing MT experiment.
Phrase-Based Machine Translation

• essential ingredient: parallel data
• obtain a dictionary of phrase translations, with probabilities (phrase table):
  \[ P(\text{ahoj} \mid \text{hello}) = 0.2 \]
  \[ P(\text{“nazdar vespolek”} \mid \text{“hello there”}) = 0.05 \]
• use a language model to ensure that outputs look like fluent Czech:
  \[ P(\text{přišel} \mid \text{“Petr včera zase”}) \gg P(\text{přijdou} \mid \text{“Petr včera zase”}) \]
• translation (decoding) = search for the most probable combination of possible phrasal translations
PBMT in More Detail

Preprocessing: tokenization, tagging...

- Word alignment
- Phrase extraction

- Language Model (LM)
- Translation M. (TM)
- Reordering M. (RM)

Basic model

Parameter optimization (MERT)

Optimized model

Translate
Essential Tools

- Moses
  - [http://statmt.org/moses/](http://statmt.org/moses/)
  - open-source toolkit for statistical machine translation
  - includes:
    - decoder (the actual translation program)
    - language modeling toolkit (KenLM)
    - tools for training translation models from data
    - tools for model parameter optimization
    - many helper scripts (pre-processing etc.)

- GIZA++
  - word alignment

- Eman
  - our experiment manager (not “essential” but you may find it useful)
Eman – Intro

• Training an MT system requires a pipeline of relatively complex tools

• Eman can help because it:
  – provides wrappers for all these steps
  – keeps track of existing experiments, their steps, their state etc.
  – handles dependencies, re-uses existing steps
  – lets users assign meaningful tags to experiments
  – keeps track of experimental results
  – allows sharing of experiments between users
PBMT in Eman's View

Monolingual corpus

Parallel corpus

Devset corpus

Input corpus

Processing, tokenization, tagging...

Word alignment

Phrase extraction

Im

Trans tm (TM)

Reorder rm

B model

Parameter optimization (MERT)

Opt mert model

translate
Eman – Installation

- Simply source this installation script:
  . ~tamca7am/install-all.sh
- If everything worked, this should display eman's man page:
  eman --man
- You should get two new directories:
  - eman ... eman lives here in bin/eman
  - ufal-smt-playground ... place to run experiments
Your First MT Experiment

• Go to your experimental “playground”:
  
  cd ufal-smt-playground/playground

• Experiments (playgrounds) can be shared between users. We'll take advantage of this:

  eman addremote \\n  ~tamca7am/ufal-smt-playground/playground \\
  tamchyna ;
  eman reindex ; ./corpman --reindex
Obtaining Moses (and GIZA++)

- Just use my mosesgiza step:
  
  s.mosesgiza.c7daa783.20150218-1514

- ...or compile your own
  
  - get a local version of Boost, e.g. mine:
    
    ~tamca7am/boost_1_56_0/install
  
  - also, we need CMPH in Moses for our experiments:
    
    ~tamca7am/cmph-2.0/install
  
  - create your own mosesgiza:
    
    BOOSTPATH=/boost/here/ CMPHPATH=/cmph/here/ \ 
    eman init --start mosesgiza