Memory Based MT

Antal van den Bosch
Peter Berck

Tilburg centre for Creative Computing
Tilburg University
The Netherlands

http://ilk.uvt.nl/mbmt
Outline

• MBMT
• Evaluation
• Software
• Conclusions
What is Memory Based MT?

- Example based MT
- Take bits of source text, map to bits of target text
- Recombine the target bits into a sentence
Details

- Use (GIZA++) aligned sentences
De stemming vindt vanavond plaats.
The vote will take place this evening.
Details

- Use (GIZA++) aligned sentences
- Use trigrams
Trigram Mapping

\[
\begin{array}{ccc}
\text{De} & \text{stemming} & \text{vindt} \\
\text{stemming} & \text{vindt} & \text{vanavond} \\
\text{vindt} & \text{vanavond} & \text{plaats} \\
\text{vanavond} & \text{plaats} & . \\
\end{array}
\]

\[
\begin{array}{ccc}
\text{The} & \text{vote} & \text{will} \\
\text{vote} & \text{will} & \text{take} \\
\text{this} & \text{evening} & . \\
\text{take} & \text{place} & \text{this} \\
\end{array}
\]
Details

• Use (GIZA++) aligned sentences
• Use trigrams
• Decision tree based k-NN classifier
Details

- Use (GIZA++) aligned sentences
- Use trigrams
- Decision tree based k-NN classifier
- Decoder
Recombine
Details

- Use (GIZA++) aligned sentences
- Use trigrams
- Decision tree based k-NN classifier
- Decoder
- LM assigns perplexity score to sentence
NGOs are good within the European Union. Perplexity = 198.326

NGOs are good the EU, within the European. Perplexity = 243.701
Details

- Use (GIZA++) aligned sentences
- Use trigrams
- Decision tree based k-NN classifier
- Decoder: when more than one answer, use language model
- LM assigns perplexity score to sentence
- No explicit linguistic knowledge
deze zin kan nooit vertaald worden.
## Experiments

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>OpenSub</td>
<td>2 million</td>
</tr>
<tr>
<td>JRC-Aquis</td>
<td>14 million</td>
</tr>
<tr>
<td>EMEA</td>
<td>9 million</td>
</tr>
<tr>
<td>LM: Reuters</td>
<td>37 million</td>
</tr>
</tbody>
</table>
# Results: OpenSub

<table>
<thead>
<tr>
<th></th>
<th>WER</th>
<th>PER</th>
<th>BLEU</th>
<th>METEOR</th>
<th>NIST</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moses</td>
<td>53.2878</td>
<td>46.9615</td>
<td>0.3289</td>
<td>0.5407899</td>
<td>5.9035</td>
</tr>
<tr>
<td>MBMT</td>
<td>68.3948</td>
<td>61.3335</td>
<td>0.1631</td>
<td>0.4015985</td>
<td>4.2428</td>
</tr>
<tr>
<td>Google</td>
<td>50.0984</td>
<td>45.0847</td>
<td>0.3056</td>
<td>0.5223539</td>
<td>5.7893</td>
</tr>
<tr>
<td>Systran</td>
<td>60.7691</td>
<td>54.6135</td>
<td>0.1749</td>
<td>0.4500350</td>
<td>4.5828</td>
</tr>
</tbody>
</table>

MT Marathon 2009 Prague
## Results: JRC-Aquis

<table>
<thead>
<tr>
<th></th>
<th>WER</th>
<th>PER</th>
<th>BLEU</th>
<th>METEOR</th>
<th>NIST</th>
</tr>
</thead>
<tbody>
<tr>
<td>MBMT</td>
<td>58.5586</td>
<td>36.7447</td>
<td>0.4513</td>
<td>0.6336529</td>
<td>7.8306</td>
</tr>
<tr>
<td>Google</td>
<td>48.4244</td>
<td>32.8729</td>
<td>0.4713</td>
<td>0.6511708</td>
<td>8.2668</td>
</tr>
<tr>
<td>Systran</td>
<td>60.8488</td>
<td>43.0711</td>
<td>0.3321</td>
<td>0.5549924</td>
<td>6.7365</td>
</tr>
</tbody>
</table>
## Results: EMEA

<table>
<thead>
<tr>
<th></th>
<th>WER</th>
<th>PER</th>
<th>BLEU</th>
<th>METEOR</th>
<th>NIST</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moses</td>
<td>46.5543</td>
<td>39.3553</td>
<td>0.4701</td>
<td>0.6501440</td>
<td>7.0593</td>
</tr>
<tr>
<td>MBMT</td>
<td>72.7873</td>
<td>63.6633</td>
<td>0.2633</td>
<td>0.4801131</td>
<td>5.1145</td>
</tr>
<tr>
<td>Google</td>
<td>57.5692</td>
<td>50.4385</td>
<td>0.3918</td>
<td>0.5829913</td>
<td>6.3772</td>
</tr>
<tr>
<td>Systran</td>
<td>63.2418</td>
<td>55.1430</td>
<td>0.2895</td>
<td>0.5366058</td>
<td>5.4716</td>
</tr>
</tbody>
</table>
## Results: Performance & Speed

<table>
<thead>
<tr>
<th></th>
<th>WER</th>
<th>PER</th>
<th>BLEU</th>
<th>METEOR</th>
<th>NIST</th>
<th>Train</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>MBMT</td>
<td>72.7</td>
<td>63.6</td>
<td>0.238</td>
<td>0.460</td>
<td>4.97</td>
<td>20:17</td>
<td>0:08</td>
</tr>
<tr>
<td>Moses</td>
<td>46.6</td>
<td>39.4</td>
<td>0.470</td>
<td>0.650</td>
<td>7.06</td>
<td>3:10:06</td>
<td>2:51</td>
</tr>
</tbody>
</table>
Software

Timbl

Wopr

MBMT

MT Marathon 2009 Prague
MBMT: Memory-based machine translation

MBMT is a software package for training and running a machine translation system. It is based on the 8-nearest neighbor classifier as implemented in TMBL, and also features a memory-based target language model based on WOPRE, a memory-based language model. It assumes a word-aligned bilingual parallel training corpus, such as produced by GIZA++. 

Features
- Generates a machine translation model from a word-aligned parallel corpus
- Fast training and translation

MBMT is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation.

Written by
MBMT is written by Antal van den Bosch, with contributions from Peter Buick and Ko van der Sluis.

Download and installation
mbmt-0.1.tar.gz (1.39 Kb)

To install, please follow these basic instructions:
- MBMT relies on an installation and availability of SVN of the following two packages:
  - TMBL version 6.14 (or higher)
  - WOPRE version 1.4.0 (or higher)
- The script will unpack tar files (mbmt-0.1.tar.gz) in a directory called mbmt-0.1.
- In the mbmt-0.1 directory, issue a ‘configure’ command, followed by ‘make’.
- If you want to install the software elsewhere, issue a ‘configure –prefix<install-dir>’, followed by ‘make’ and ‘make install’.

Quick start
- The ‘runmbt.sh’ script included in the package runs a full training and translation process, based on a GIZA++-aligned ‘A3.final’ file, and a target-language text (one sentence per line).
- For example, run the script by issuing ‘mbmt.sh IRC-Acquis.sample.A3.final IRC-Acquis.source.text.nl’ (Dutch-English files extracted from the IRC-Acquis multilingual parallel corpus).

MBMT has been compiled successfully with gcc (4.0 - 4.2), on Intel platforms running several versions of Linux and the Mac OS X platform.

References
For more information and background on MBMT, see
- MT Marathon paper

Sponsor
MBMT is developed as part of the Implicit Linguistics project, funded by NWO, the Netherlands Organisation for Scientific Research.

http://ilk.uvt.nl/mbmt
WOPR

Memory-based word prediction and language modeling

WOPR: Memory-based word prediction and language modeling

WOPR is a wrapper around the k-nearest neighbor classifier in TMBL, offering word prediction and language modeling functionalities. Trained on a text corpus, WOPR can predict missing words, report perplexities at the word level and the text level, and generate spelling correction hypotheses.

The WOPR name is obviously a blligant cultural reference to the mainframe computer WOPR, "War Operation Plan Response", a key role player in the 1983 US movie War Games. Through a hacked phone dialup connection, WOPR's owner (playing games with a teenager played by a young Matthew Broderick) almost causes a full nuclear war. Image from Wikipedia.

Features

- Generates language models
- Tests language models on new text, reporting perplexities, prediction distributions, word-level entropies and perplexities
- Optionally exports ABN's formatted language model files
- Optionally filters its output for spelling correction candidates

WOPR is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation.

Written by

WOPR is written by Peter Broek, with input from Aart van den Bosch and Ko van der Shoot.

Download and Installation

wopr-1.4.7.tar.gz (120 Kb)

To install, please follow these basic instructions:

- Please make sure you have installed TMBL version 6.1.4 (or higher).
- Follow the HOWTO.
- For general information on Wopr, see here.

Wop has been tested on:

- Intel platform running several versions of Linux
- AMD64 platform running pristine Linux
- Mac OS X platform

WOPR incorporated

WOPR is used in MBMT, our memory-based machine translation software.

Sponsor

WOPR is developed as part of the Implicit Linguistics project, funded by NWO, the Netherlands Organisation for Scientific Research.

References

For more information and background on WOPR, see:


http://ilk.uvt.nl/wopr
Specs

mbmt.sh script which uses:

- MBMT
  - C programs to make instances and recombine
- Woppr
  - C++ language model
- Timbl
  - C++ instance based learner
- Tested on Linux and OS X
Quickstart

• Install Timbl, Wopr and MBMT
• Run the mbmt.sh script which:
  – Takes an aligned file
  – Creates instances
  – Trains translation and language models
  – Runs test set
Conclusions: Cons

• Inferior to Moses
• Because:
  – No smoothing
  – No fertility or “null” model
  – Limited to trigrams
Conclusions: Pro

• Speed
  – training and decoding

• Memory footprint low
Memory & Speed

Number of training tokens (x million)

Megabytes

Seconds

Training
Testing

MT Marathon 2009 Prague
Future work

• Constraint Satisfaction Inference (CSI)
  – Integration of fertility model and “null model”

• Parameters
  – Beam in search
  – Classifier parameters (speed-accuracy trade-off)
  – Higher n in n-grams

• Reliance on word aligner
  – Test BerkeleyAligner
Thank you

http://ilk.uvt.nl/mbmt