Class-Based Language Modeling for Translating into Morphologically Rich Languages

Arianna Bisazza & Christof Monz
Phrase-based SMT

SRC: \( \text{word}_{s_1} \) \( \text{word}_{s_2} \) \( \text{word}_{s_3} \) \( \text{word}_{s_4} \) \( \text{word}_{s_5} \) \( \text{word}_{s_6} \) \( \text{word}_{s_7} \)

TRG: \( \text{word}_{t_1} \) \( \text{word}_{t_2} \)

\( \text{Disto. scores} \)

\( \text{TM scores} \)

\( \text{LM scores} \)
Phrase-based SMT

 SRC: \[\text{word}_{s_1} \ \text{word}_{s_2} \ \text{word}_{s_3} \ \text{word}_{s_4} \ \text{word}_{s_5} \ \text{word}_{s_6} \ \text{word}_{s_7}\]

 TRG: \[\text{word}_{t_1} \ \text{word}_{t_2} \ \text{word}_{t_3} \ \text{word}_{t_4} \ \ldots\]

Disto. scores

TM scores

LM scores
Phrase-based SMT

\[
\begin{align*}
\alpha_{TM-d} & \quad + \quad \alpha_{TM-i} & \quad + \quad \alpha_{DM} & \quad + \quad \ldots & \quad + \quad \alpha_{LM} \\
\log P_{TM-d}(f|e) & \quad + \quad \log P_{TM-i}(e|f) & \quad + \quad \log P_{DM}(f_{t-1}, f_t) & \quad + \quad \ldots & \quad + \quad \log P_{LM}(e)
\end{align*}
\]
Phrase-based SMT

TRG: \( \text{word}_{T_1} \text{ word}_{T_2} \text{ word}_{T_3} \text{ word}_{T_4} \ldots \)

\[
\alpha_{\text{TM-d}} \log P_{\text{TM-d}}(f|e) \quad \alpha_{\text{TM-i}} \log P_{\text{TM-i}}(e|f) \quad \alpha_{\text{DM}} \log P_{\text{DM}}(f_{t-1}, f_t) \quad \ldots \quad \alpha_{\text{LM}} \log P_{\text{LM}}(e)
\]
N-gram language modeling
N-gram language modeling

- English:

  freedom of movement must be **encouraged**

  \[ P_{LM} \approx \frac{\# \text{(must be encouraged)}}{\# \text{(must be *)}} \]
N-gram language modeling

- English:
  
  freedom of movement must be **encouraged**

  \[ P_{LM} \approx \frac{\# \text{(must be encouraged)}}{\# \text{(must be *)}} \]

- Morphologically rich language:
  
  la libertà di movimento deve essere **incoraggiata**

  \[ P_{LM} \approx \frac{\# \text{(deve essere incoraggiata)}}{\# \text{(deve essere *)}} \]
N-gram language modeling

• English:

  freedom of movement must be **encouraged**

\[
P_{\text{LM}} \approx \frac{\text{# (must be encouraged)}}{\text{# (must be *)}}
\]

• Morphologically rich language:

  la libertà di movimento deve essere **incoraggiata**

\[
P_{\text{LM}} \approx \frac{\text{# (deve essere incoraggiata)}}{\text{# (deve essere *)}}
\]

<table>
<thead>
<tr>
<th>deve essere incoraggiato</th>
<th>120</th>
</tr>
</thead>
<tbody>
<tr>
<td>devono essere incoraggiati</td>
<td>54</td>
</tr>
<tr>
<td>dovrebbe essere incoraggiata</td>
<td>3</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>deve essere <strong>incoraggiata</strong></td>
<td>0</td>
</tr>
</tbody>
</table>

Must backoff to shorter history!
N-gram language modeling

- English:
  
  freedom of movement must be **encouraged**

  \[ P_{LM} \approx \frac{\# (\text{must be encouraged})}{\# (\text{must be } \ast)} \]

- Morphologically rich language:
  
  la libertà di movimento deve essere **incoraggiata**

  \[ P_{LM} \approx \frac{\# (\text{deve essere incoraggiata})}{\# (\text{deve essere } \ast)} \]

  Must backoff to shorter history!
N-gram language modeling

- English:

  freedom of movement must be encouraged

  \[ P_{LM} \approx \frac{\# \text{(must be encouraged)}}{\# \text{(must be *)}} \]

- Morphologically rich language:

  la libertà di movimento deve essere incoraggiata

  \[ P_{LM} \approx \frac{\# \text{(deve essere incoraggiata)}}{\# \text{(deve essere *)}} \]

  Must backoff to shorter history!
N-gram language modeling

- **English:**

  freedom of movement  must be **encouraged**

  \[ P_{LM} \approx \frac{\# \text{(must be encouraged)}}{\# \text{(must be *)}} \]

- **Morphologically rich language:**

  (sing.fem.)

  la libertà di movimento  deve essere **incoraggiata**

  \[ P_{LM} \approx \frac{\# \text{(deve essere incoraggiata)}}{\# \text{(deve essere *)}} \]

  Long dependencies important for inflection!

  Must backoff to shorter history!
Class-based language modeling

- IDEA: group words with similar distributional behaviour into equivalence classes (Brown et al. 1992)

\[ P_{\text{class}}(w_i|w_{i-n+1}^{i-1}) = \]
\[ p_0(C(w_i)|C(w_{i-n+1}^{i-1})) \cdot p_1(w_i|C(w_i)) \]
Class-based language modeling

- IDEA: group words with similar distributional behaviour into equivalence classes (Brown et al. 1992)

\[ P_{\text{class}}(w_i|w_{i-n+1}^{i-1}) = \]
\[ \cdot p_0(C(w_i)|C(w_{i-n+1}^{i-1})) \cdot p_1(w_i|C(w_i)) \]

\[ P_{\text{class}} \approx \frac{\#(C_x C_y C_z)}{\#(C_x C_y \ast)} \cdot \frac{\#(\text{incoraggiata})}{\#(C_z)} \]
Class-based language modeling

- IDEA: group words with similar distributional behaviour into equivalence classes (Brown et al. 1992)

\[ P_{\text{class}}(w_i|w_{i-n+1}^{i-1}) = \]
\[ p_0(C(w_i)|C(w_{i-n+1}^{i-1})) \cdot p_1(w_i|C(w_i)) \]

### Diagram

![Diagram](image.png)

\[ P_{\text{CLASS}} \approx \frac{\# (\text{Cx Cy Cz})}{\# (\text{Cx Cy *})} \cdot \frac{\# (\text{incoraggiata})}{\# (\text{Cz})} \]

Enable use of longer history ✅

Capture long dependencies for inflection ❓
Goal of this work
Goal of this work

Compare in a unified SMT setting:
- different kinds of classes
- different class-based model forms
- different combining frameworks

for translation into a morphologically rich language
Goal of this work

Compare in a unified SMT setting:

- different kinds of classes
- different class-based model forms
- different combining frameworks

for translation into a morphologically rich language

Working language pair:

- English to Russian (Russian type/token ratio two times higher than English)
Kinds of classes

- Data-driven: partition vocabulary into given nb. of clusters by maximizing likelihood of training corpus
Kinds of classes

- Data-driven: partition vocabulary into given nb. of clusters by maximizing likelihood of training corpus

- Linguistic:
  - annotation-based: POS, lemma, morphological tag ...
    (possible issue: non-deterministic class mapping)
  - shallow: simple rule-based suffixes, φ most frequent suffixes, orthographic features
Kinds of classes

- Data-driven: partition vocabulary into given nb. of clusters by maximizing likelihood of training corpus

- Linguistic:
  - annotation-based: POS, lemma, morphological tag ...
    (possible issue: non-deterministic class mapping)
  - shallow: simple rule-based suffixes, $\phi$ most frequent suffixes, orthographic features

- Hybrid suffix/class mappings (Müller et al. 2012):

$$C(w) = \begin{cases} 
    w & \text{if } #(w) > \theta \\
    \text{suff}(w) & \text{otherwise}
\end{cases}$$

NEW for SMT

la libertà di movimento deve essere incoraggiata
[la] [-à] [di] [-imento] [deve] [essere] [-ata]
Class-based model forms

- Class-based LM originally proposed for ASR:

\[ P_{\text{class}}(w_i|w_{i-n}^{i-1}) = p_0(C(w_i)|C(w_{i-n}^{i-1})) \cdot p_1(w_i|C(w_i)) \]
Class-based model forms

- Class-based LM originally proposed for ASR:

\[
P_{\text{class}}(w_i|w_{i-n+1}^{i-1}) = p_0(C(w_i)|C(w_{i-n+1}^{i-1})) \cdot p_1(w_i|C(w_i))
\]

- “Stream”-based LM: no class-to-word emission probability

\[
P_{\text{stream}}(w_i|w_{i-n+1}^{i-1}) = p_0(C(w_i)|C(w_{i-n+1}^{i-1}))
\]
Class-based model forms

- **Class-based LM originally proposed for ASR:**
  \[
P_{\text{class}}(w_i|w_{i-n+1}^{i-1}) = p_0(C(w_i)|C(w_{i-n+1}^{i-1})) \cdot p_1(w_i|C(w_i))
  \]

- “*Stream*”-based LM: no class-to-word emission probability
  \[
P_{\text{stream}}(w_i|w_{i-n+1}^{i-1}) = p_0(C(w_i)|C(w_{i-n+1}^{i-1}))
  \]

- **Fullibm LM: context-sensitive emission probability**
  (Goodman 2001)
  \[
P_{\text{fullibm}}(w_i|w_{i-n+1}^{i-1}) = p_0(C(w_i)|C(w_{i-n+1}^{i-1})) \cdot p_1(w_i|C(w_{i-n+1}^{i-1}))
  \]
Model combining frameworks
Model combining frameworks

- Log-linear interpolation (model level):

\[ p(x|h) = \prod_m p_m(x|h)^{\alpha_m} \]

\[ \alpha_{TM-d} \cdot \log P_{TM-d}(f|e) + \alpha_{TM-i} \cdot \log P_{TM-i}(e|f) + \alpha_{DM} \cdot \log P_{DM}(f_{t-1},f_t) + \ldots + \alpha_{LM} \cdot \log P_{LM}(e) + \alpha_{CLM} \cdot \log P_{CLM}(e) \]
Model combining frameworks

- Log-linear interpolation (model level):

\[ p(x|h) = \prod_m p_m(x|h)^{\alpha_m} \]

\[ \alpha_{TM-d} \cdot \log P_{TM-d}(f|e) + \alpha_{TM-i} \cdot \log P_{TM-i}(e|f) + \alpha_{DM} \cdot \log P_{DM}(f_{t-1}, f_t) + \alpha_{LM} \cdot \log P_{LM}(e) + \alpha_{CLM} \cdot \log P_{CLM}(e) \]

- Linear interpolation (word level):

\[ P_{mixLM}(e) = \prod_i^n \left( \sum_q \lambda_q p_q(e_i|h_i) \right) \]

\[ \alpha_{TM-d} \cdot \log P_{TM-d}(f|e) + \alpha_{TM-i} \cdot \log P_{TM-i}(e|f) + \alpha_{DM} \cdot \log P_{DM}(f_{t-1}, f_t) + \lambda_{LM} \cdot \log P_{LM}(e) + \lambda_{CLM} \cdot \log P_{CLM}(e) \]
Log-linear interpolation (model level):

\[ p(x|h) = \prod_m p_m(x|h)^{\alpha_m} \]

\[ \alpha_{TM-d} \cdot \log P_{TM-d}(f|e) + \alpha_{TM-i} \cdot \log P_{TM-i}(e|f) + \alpha_{DM} \cdot \log P_{DM}(f_{t-1}, f_t) + \alpha_{LM} \cdot \log P_{LM}(e) + \alpha_{CLM} \cdot \log P_{CLM}(e) \]

Linear interpolation (word level):

\[ p_{\text{mixLM}}(e) = \prod_{i=1}^n \left( \sum_q \lambda_q p_q(e_i|h_i) \right) \]

\[ \alpha_{TM-d} \cdot \log P_{TM-d}(f|e) + \alpha_{TM-i} \cdot \log P_{TM-i}(e|f) + \alpha_{DM} \cdot \log P_{DM}(f_{t-1}, f_t) + \alpha_{\text{mixLM}} \cdot \log P_{\text{mixLM}}(e) \]

\[ \lambda_{LM} \cdot \log P_{LM}(e) + \lambda_{CLM} \cdot \log P_{CLM}(e) \]
Model combining frameworks

- Log-linear interpolation (model level):

\[ p(x|h) = \prod_m p_m(x|h)^{\alpha_m} \]

\[ \alpha_{TM-d} \cdot \log P_{TM-d}(f|e) + \alpha_{TM-i} \cdot \log P_{TM-i}(e|f) + \alpha_{DM} \cdot \log P_{DM}(f_{t-1},f_t) + \cdots + \alpha_{LM} \cdot \log P_{LM}(e) + \alpha_{CLM} \cdot \log P_{CLM}(e) \]

- Linear interpolation (word level):

\[ p_{\text{mixLM}}(e) = \prod_{i=1}^n \left( \sum_q \lambda_q p_q(e_i|h_i) \right) \]

\[ \alpha_{TM-d} \cdot \log P_{TM-d}(f|e) + \alpha_{TM-i} \cdot \log P_{TM-i}(e|f) + \alpha_{DM} \cdot \log P_{DM}(f_{t-1},f_t) + \cdots + \alpha_{\text{mixLM}} \cdot \log P_{\text{mixLM}}(e) \]

\[ \lambda_{LM} \cdot p_{LM}(e) + \lambda_{CLM} \cdot P_{CLM}(e) \]

\[
\frac{\lambda_{LM}}{\lambda_{CLM}} = \frac{P_{LM}(e)}{P_{CLM}(e)}
\]

\[ \frac{\lambda_{CLM}}{\lambda_{LM}} = \frac{P_{CLM}(e)}{P_{LM}(e)} \]

\[ \frac{\lambda_{LM}}{\lambda_{CLM}} = \frac{P_{LM}(e)}{P_{CLM}(e)} \]

α weights can be optimized for translation quality...

... but λ weights cannot (even so, works well for standard LM interp.)
Model combining frameworks

- Linear interpolation with one lambda:

\[
\lambda \cdot P_{\text{class}}(w_i|w_{i-n+1}^{i-1}) + (1 - \lambda) \cdot P_{\text{word}}(w_i|w_{i-n+1}^{i-1})
\]
Model combining frameworks

- Linear interpolation with one lambda:

\[ \lambda \cdot P_{\text{class}}(w_i|w_{i-n+1}^{i-1}) + (1 - \lambda) \cdot P_{\text{word}}(w_i|w_{i-n+1}^{i-1}) \]

- Linear interpolation with class-specific lambdas
  (Müller et al. 2012; Bahl et al. 91):

\[ \lambda_{C(w_{i-1})} \cdot P_{\text{class}}(w_i|w_{i-n+1}^{i-1}) + (1 - \lambda_{C(w_{i-1})}) \cdot P_{\text{word}}(w_i|w_{i-n+1}^{i-1}) \]
Experiments
Experiments

- Task: English to Russian news translation (WMT 2013)
  - 2M parallel sentences
  - 21M Russian sentences (390M tokens)
  - WMT 2013 official test set
Experiments

- **Task:** English to Russian news translation (WMT 2013)
  - 2M parallel sentences
  - 21M Russian sentences (390M tokens)
  - WMT 2013 official test set

- **Baseline:** state-of-the-art phrase-based SMT system
  - includes hierarchical lexicalized reordering model (Galley et al. 2008)
  - 5-gram word LM trained on all data
  - BLEU-tuned with pairwise ranking optimization (Hopkins & May 2011)
Perplexity results

<table>
<thead>
<tr>
<th>LM type</th>
<th>smoothing</th>
<th>vocab.</th>
<th>PP</th>
</tr>
</thead>
<tbody>
<tr>
<td>words</td>
<td>Knese-Ney</td>
<td>2.7M</td>
<td>270</td>
</tr>
<tr>
<td>Brown clusters</td>
<td>Witten-Bell</td>
<td>600</td>
<td>588</td>
</tr>
<tr>
<td>suffix</td>
<td>Witten-Bell</td>
<td>968</td>
<td>2455</td>
</tr>
<tr>
<td>suffix/word hybrid (θ=5000)</td>
<td>Witten-Bell</td>
<td>8530</td>
<td>460</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Linear interp.</th>
<th>PP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>generic λ</td>
</tr>
<tr>
<td>words + clusters</td>
<td>225</td>
</tr>
<tr>
<td>words + suffixes</td>
<td>266</td>
</tr>
<tr>
<td>words + hybrid</td>
<td>243</td>
</tr>
</tbody>
</table>

- Class LM perplexity much higher, but linear interpolation outperforms simple word LM
- No significant effect by class-specific lambdas
SMT results (1)

- Log-linear combination (one more SMT feature), data-driven clusters (600)

<table>
<thead>
<tr>
<th>Additional LM</th>
<th>surface BLEU</th>
<th>Δ</th>
</tr>
</thead>
<tbody>
<tr>
<td>★ none [baseline]</td>
<td>18.8</td>
<td>—</td>
</tr>
<tr>
<td>★ 5g stream-based</td>
<td>19.1</td>
<td>+0.3*</td>
</tr>
<tr>
<td>7g stream-based</td>
<td>19.1</td>
<td>+0.3*</td>
</tr>
<tr>
<td>★ 5g class-based</td>
<td>18.9</td>
<td>+0.1</td>
</tr>
<tr>
<td>7g class-based</td>
<td>18.8</td>
<td>±0.0</td>
</tr>
<tr>
<td>5g fullibm</td>
<td>19.4</td>
<td>+0.6*</td>
</tr>
<tr>
<td>7g fullibm</td>
<td>19.3</td>
<td>+0.5*</td>
</tr>
</tbody>
</table>

- Stream-based LM: small significant improvement
- Class-based original form: no improvement
- Fullibm model with context-sensitive emission probabilities works best
- No visible gains with higher order N-gram (7)
SMT results (2)

- Log-linear combination (one more SMT feature), hybrid suffix/word classes:

<table>
<thead>
<tr>
<th>Additional LM</th>
<th>surface BLEU</th>
<th>Δ</th>
</tr>
</thead>
<tbody>
<tr>
<td>* none [baseline]</td>
<td>18.8</td>
<td>—</td>
</tr>
<tr>
<td>* 5g stream-based</td>
<td>18.9</td>
<td>+0.1</td>
</tr>
<tr>
<td>7g stream-based</td>
<td>18.9</td>
<td>+0.1</td>
</tr>
<tr>
<td>* 5g class-based</td>
<td>19.0</td>
<td>+0.2°</td>
</tr>
<tr>
<td>7g class-based</td>
<td>19.1</td>
<td>+0.3°</td>
</tr>
<tr>
<td>5g fullibm</td>
<td>19.1</td>
<td>+0.3*</td>
</tr>
<tr>
<td>7g fullibm</td>
<td>19.2</td>
<td>+0.4*</td>
</tr>
</tbody>
</table>

- Shallow morphology classes: computationally cheaper but overall smaller improvements
SMT results (3)

- Linear interpolation (combine wordLM & classLM in one SMT feature)
- Lambda weights optimized for likelihood of held-out data

### Data-driven classes

<table>
<thead>
<tr>
<th>Additional LM</th>
<th>surface BLEU</th>
<th>surface Δ</th>
</tr>
</thead>
<tbody>
<tr>
<td>* none [baseline]</td>
<td>18.8</td>
<td>—</td>
</tr>
<tr>
<td>* 5g class, log-linear comb.</td>
<td>18.9</td>
<td>+0.1</td>
</tr>
<tr>
<td>* 5g class, linear (global λ)</td>
<td>18.5</td>
<td>−0.3</td>
</tr>
<tr>
<td>5g class, linear (class λ’s)</td>
<td>18.6</td>
<td>−0.2</td>
</tr>
</tbody>
</table>

### Hybrid suffix/word

<table>
<thead>
<tr>
<th>Additional LM</th>
<th>surface BLEU</th>
<th>surface Δ</th>
</tr>
</thead>
<tbody>
<tr>
<td>* none [baseline]</td>
<td>18.8</td>
<td>—</td>
</tr>
<tr>
<td>* 5g class, log-linear comb.</td>
<td>19.0</td>
<td>+0.2°</td>
</tr>
<tr>
<td>* 5g class, linear (global λ)</td>
<td>18.9</td>
<td>+0.1</td>
</tr>
<tr>
<td>5g class, linear (class λ’s)</td>
<td>18.6</td>
<td>−0.1</td>
</tr>
</tbody>
</table>

- Linear interpolation worse than log-linear in all conditions
- using class-specific lambdas doesn’t help
SMT results (4)

- Back to first table, looking at stem-level BLEU scores
  (log-linear combination, data-driven clusters)

<table>
<thead>
<tr>
<th>Additional LM</th>
<th>surface BLEU</th>
<th>stem BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>None [baseline]</td>
<td>18.8</td>
<td>23.9</td>
</tr>
<tr>
<td>5g stream-based</td>
<td>19.1 (+0.3*)</td>
<td>24.0 (+0.1)</td>
</tr>
<tr>
<td>7g stream-based</td>
<td>19.1 (+0.3*)</td>
<td>24.1 (+0.2)</td>
</tr>
<tr>
<td>5g class-based</td>
<td>18.9 (+0.1)</td>
<td>23.8 (-0.1)</td>
</tr>
<tr>
<td>7g class-based</td>
<td>18.8 (±0.0)</td>
<td>23.9 (±0.0)</td>
</tr>
<tr>
<td>5g fullibm</td>
<td>19.4 (+0.6*)</td>
<td>24.3 (+0.4*)</td>
</tr>
<tr>
<td>7g fullibm</td>
<td>19.3 (+0.5*)</td>
<td>24.3 (+0.4*)</td>
</tr>
</tbody>
</table>

- Surface-level gains bigger than stem-level gains
  => suggests effect on choice of word inflections
Conclusions
Conclusions

- First systematic comparison of different class-based LMs for SMT into a morphologically rich language:
  - class-to-word emission probabilities matter for translation quality
  - biggest improvement with fullibm LM
  - hybrid suffix/word classes not as good as data-driven
  - linear interpolation (known to work well for standard LMs) not a good choice for class-based LMs
Conclusions

- First systematic comparison of different class-based LMs for SMT into a morphologically rich language:
  - class-to-word emission probabilities matter for translation quality
  - biggest improvement with fullibm LM
  - hybrid suffix/word classes not as good as data-driven
  - linear interpolation (known to work well for standard LMs) not a good choice for class-based LMs

- More work needed to properly model morphologically rich languages, going beyond the constraints of n-gram LM
Grazie per l’attenzione!

Thanks for your attention!

Спасибо за внимание!