Retrieval in text collections with historic spelling

Andrea Ernst-Gerlach

University Duisburg-Essen (Germany)

Dagstuhl Seminar ’Digital Historical Corpora’
Schloss Dagstuhl, Germany
December 3-8, 2006
Overview

1. Search in text collections with non standard spelling
2. Generation of transformation rules
3. Layered architecture for the historic search engine
4. Conclusion and future work
Our approach: Generation of search term variants at retrieval time

- Query expansion (flexible)
- Mapping necessary:
  - search term $\rightarrow^{(1)}$ contemporary inflections (or derivations)
  - $\rightarrow^{(2)}$ spelling variants

  (1) morphological variations
  (2) transformation rules
## Example rules for English and German

<table>
<thead>
<tr>
<th>Contemp. spelling</th>
<th>19th century</th>
<th>rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>always</td>
<td>alwaies</td>
<td>y → ie</td>
</tr>
<tr>
<td>sudden</td>
<td>suddain</td>
<td>e → ai</td>
</tr>
<tr>
<td>publicly</td>
<td>publikely</td>
<td>c → ke</td>
</tr>
<tr>
<td>wiedergaben</td>
<td>widergaben</td>
<td>wieder → wider</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ie → i</td>
</tr>
<tr>
<td>akzeptieren</td>
<td>acceptieren</td>
<td>kz → cc</td>
</tr>
<tr>
<td></td>
<td></td>
<td>k → c ∧ z → c</td>
</tr>
<tr>
<td>seht</td>
<td>sehet</td>
<td>t → et</td>
</tr>
</tbody>
</table>
Overview

1. Search in text collections with non standard spelling
2. Generation of transformation rules
3. Layered architecture of the historic search engine
4. Conclusion and future work
Generation of transformation rules

1. Retrieve evidences
2. Generate rule core
3. Generate rule candidates
4. Rule pruning
Generate rule core

- Training set of triplets
  - Contemporary word form
  - Historic word form
  - Collection frequency

- Find necessary transformations e.g.
  unnütz - unnuts
  rule cores: ^unn(ü,u)t t(z,s)$
Generate rule candidates

- Successively adding context to rule cores
  - e.g. unnütz - unnuts  
    rule core: ^unn(ü,u)t 
    ü → u  nü → nu  üt → ut  nüt → nut

- Abstraction of context
  - Consonants (C) / Vowels (V)
    - e.g. Cü → Cu
  - Word beginning (∧) / -ending ($) 
    - e.g. z$ → s$
Rule Pruning: PRISM Algorithm

- PRISM
  - Classifies set of instances into set of classes
  - Instances are fixed sets of attributes
  - Tries to generate high precision values for each class C by identifying instances belonging to C

- Extension necessary
  - Perfect rules on this data set do not generalise to unseen words
  - Generalisation / specialisation relationships between rule antecedents
Pruning: Generate negative examples

- Applying rule candidates on contemporary words of evidences
- Negative examples are generated word mappings which are not included in evidences
- e.g.

  document terms:  \( aab, azb, azz \)
  evidences:  \( az \rightarrow azz \)
  \( ab \rightarrow aab \)

  rule candidate:  \( a \rightarrow az \)
  ⇒ negative example:  \( ab \rightarrow azb \)
Pruning: PRISM Extension

- Sort instances by rules
- Calculate
  - precision $p_i$
  - occurrence frequency $q_i$
- Remove all instances where $p_i < p_{\text{min}} \lor q_i < q_{\text{min}}$
Evaluation based on evidences
Overview

1. Search in text collections with non standard spelling
2. Generation of transformation rules
3. **Layered architecture for the historic search engine**
4. Conclusion and future work
Layered architecture for historic search engine

- **Layer 0**: DL
- **Layer 1**: Search Engine
- **Layer 2**: Generating Historic Word Forms
- **Layer 3**: Generating Full Word Forms
- **Layer 4**: User Interface
Generation of full word forms

- English search engines use stemming algorithms
e. g. *comput* → *compute, computer, computation, ...*

- German is highly inflected
e. g. *haben* → *hab, hat, hätten, ...*
 ⇒ utilisation of basic forms

- German vocabulary database for full word forms
Search engine PIRE

- Uses probabilistic methods for indexing and retrieval
- Outperforms competing systems
- Rather flexible and extensible
- Integration into other systems simple
- Probabilistic weighting of search terms
# Evaluation for words from the whole collection

<table>
<thead>
<tr>
<th>Approach</th>
<th>Recall</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conventional search engine</td>
<td>0.36</td>
<td>1.0</td>
</tr>
<tr>
<td>Generation of full forms</td>
<td>0.91</td>
<td>1.0</td>
</tr>
<tr>
<td>Generation of full forms + Rule application</td>
<td>0.99</td>
<td>0.89</td>
</tr>
</tbody>
</table>
Evaluation restricted to historic forms

<table>
<thead>
<tr>
<th>Approach</th>
<th>Recall</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generation of full forms</td>
<td>not calculated</td>
<td>1.0</td>
</tr>
<tr>
<td>Generation of full forms + Rule application</td>
<td>0.70</td>
<td>0.93</td>
</tr>
</tbody>
</table>
Overview

1. Search in text collections with non standard spelling
2. Generation of transformation rules
3. Layered architecture of the historic search engine
4. Evaluation
5. Conclusion and future work
Conclusion

- Generation of historic variants of search terms
- German vocabulary database for generating full word forms
- Machine learning method for generating transformation rules
- Layered architecture for the historic search engine
- Probabilistic weights of rules can be used for weighting retrieved documents
- Work in progress
Future work

- User driven generation of rule sets
- Support for XML Retrieval
- Adaptation and enhancement of search engine PIRE
- Generation of context dependent rules
Thank you for your attention!

Andrea Ernst-Gerlach
ernst@is.inf.uni-due.de