Gothic loans in Hungarian?
Towards a framework for computer-aided borrowing detection
Dissertation project at Uni Wien

• Broader sense: Historical question: When and where could Goths and Hungarians meet? (~Middle Ages, Eurasian steppe?)
• More narrow sense: Is there a possibility of Gothic loans in Hungarian?
• Arbitrary hypothesis, case study, theoretical experiment
• Focus on methodology: How to detect potential loanwords?
• Introducing a new computer-aided framework for Python 3.8.
In: Hungarian word

Out: potential loans
Sources

Uralic sound changes: uralonet.nytud.hu

Gothic words: wikiling.de

In: Any Hungarian word

Sound substitution: LägLoS

phonetic Match
Extracting sound Changes from Etymological dictionary

• Webscrape uralonet (http://www.uralonet.nytud.hu/)
  • Etymological Dictionary published by the Hungarian Academy of Sciences
  • Convert dictionary into a dataframe of three columns:
    • Reflexes, Roots, Name of Proto-Language
• Split words into consonant and vowel clusters
• Match reflex and root sound clusters
• Add <¹>: word initial sound cluster, <²>: word final, <³>: medial.
• Create csv that shows all sound changes, including all examples
• qfysc()
How The Proto Form Generator Works

• Use of combinatorics to generate new proto forms:
• For example:

<table>
<thead>
<tr>
<th>f¹</th>
<th>ü³</th>
<th>l²</th>
</tr>
</thead>
<tbody>
<tr>
<td>p</td>
<td>ü</td>
<td>kl</td>
</tr>
<tr>
<td>f</td>
<td></td>
<td>j</td>
</tr>
</tbody>
</table>

• In: fül -> Out: pükl, püj, fükl, füj
• Throw out words that „violate“ phonotactic rules
Measuring the likeliness of etymologies

• Hypothesis: The more examples per sound change, the more credible the etymology
• Introducing a new measurement method:

NSE (Normalised sum of examples)

• In how many other words does the same sound change appear in total? (\(-\)Sum of Examples)
• Divided by the number of sound changes with in a word. (\(-\)Normalised)
Measuring semantic similarity

- Two approaches: nltk, and gensim
- nltk is a dictionary that maps concepts as hypo- and hypernyms
- It calculates the similarity of two words by counting how many steps connect one concept with another within the dictionary
- gensim works differently: Words of a text are converted into vectors via machine learning. Word similarity is the cosinedistance of two vectors
- It seems gensim (based on google news corpus) works better than nltk.
- Most efficient: Get synonyms of both words with nltk, calculate the semantic similarity of all pairs with gensim, display only the most similar pair.
Outlook

• This dissertation:
  • Remove rows from Gothic dataframe that „violate” phonotactic rules
  • Tackle speed issues: Optimise code, make it faster or move to C++ or R?
  • Add more complex nuances to substitutions:
    • trV>tVr (LägLoS)
    • word final <r,l,m,n, rs, ls, ms, ns> after consonant is syllabic, thus substituted by vowel?
  • Add more paradigms to Gothic dictionary entries (morphological generator)?
  • Add Borrowability according to Haspelmath 2009
  • Add Finno-Ugric, Ugric, and in-between time-layers
  • Make code publicly available #replicability
  • Test also with words that already have well-established etymologies
  • Analyse and interpret results

• Possible future projects:
  • Add more time layers, e.g. Turkic, Western German Dialects, Indo-Iranian etc.
  • Reconstruct Gothic etc. words from Proto-Germanic?
  • Add other Uralic and Germanic languages
  • Make algorithm more dynamic, so it can handle any given language pair?
  • Base line tests: How much is coincidence?
    • Test language pairs that historically weren’t in contact
    • E.g. Proto-Austronesian & Proto-Uralic: How many false positives?