Projects

Alexander Fraser
fraser@cis.uni-muenchen.de

CIS, Ludwig-Maximilians-Universität München

Computational Morphology and Electronic Dictionaries
SoSe 2017
2017-06-26

Outline

1. Course Requirements
2. How Projects Work
3. Project Topics
4. Forming Groups

Proposed Schedule Change

Exercise at 8:30

- Can we start the Exercise 15 minutes later, at 8:30 (rather than 8:15), so that it runs from 8:30 to 10:00?

Outline

1. Course Requirements

2. How Projects Work

3. Project Topics

4. Forming Groups

Course Requirements

- To pass this course ...
  - Exercises and assignments
  - Regular attendance
  - Course project: implementation of a small project including extensive documentation; presentation
    - Roughly last 4-5 weeks of semester
    - Programming and data analysis intensive
    - Short presentation

Outline

1. Course Requirements

2. How Projects Work

3. Project Topics

4. Forming Groups

Projects in Computational Morphology and Electronic Dictionaries

- Projects will be done in groups of about 3 people
- Procedure will be to send me a ranking of possible projects and teams (we will come back to this later)
- Please send the email at 19:00 this evening; emails sent earlier (even 1 minute earlier) will be looked at last

Evaluation

- Project code/analysis
- Write a project abstract, which includes what was done and who did what
- Project presentation
- Questions to individual group members

Schedule

- Today: presentation of topics (and later, your ranking)
- Wednesday: Project topics/groups announced, work starts (in class!)
- Several exercises over the next weeks: report on work in progress, interaction with Fraser and Berlanda

⇒ this is a chance to ask questions and indicate problems, but also to meet with your group (you'll need to meet outside as well)

⇒ will also allow us to adjust topics (particularly if too hard or too easy)

- More information on polishing abstract and on presentation in the exercise next week
- Abstract due Thursday July 20th at 8pm
- Presentations/questions in last two or three meetings (we'll use classrooms with a beamer for this, not Kalahari)

Outline

1. Course Requirements
2. How Projects Work
3. Project Topics
4. Forming Groups

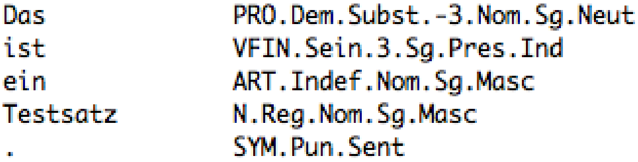
Introduction

Topics defined in terms of:

- Summary of what needs to be done
- Resources
- Programming Language (if applicable)
- Outcome
- Details of abstract (including whether German or English)
- What will be covered in the presentation

Problem: German Tagging and Lemmatization Difficult

- Summary: run German Marmot/Lemming (CIS Tagger/Lemmatizer) on two German corpora, provide a semi-automatic error analysis



```
Das          PRO.Dem.Subst.-3.Nom.Sg.Neut
ist          VFIN.Sein.3.Sg.Pres.Ind
ein          ART.Indef.Nom.Sg.Masc
Testsatz     N.Reg.Nom.Sg.Masc
.            SYM.Pun.Sent
```

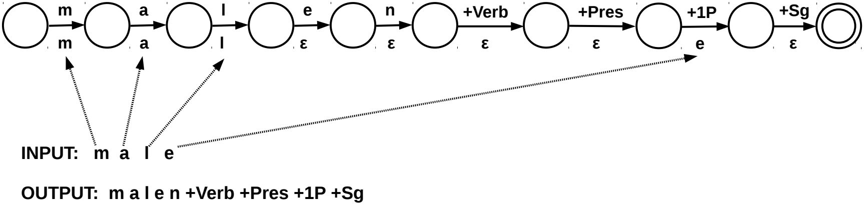
(example from RFTagger homepage, Schmid)

Project: Running German Tagger/Lemmatizer

- Summary: run German Marmot/Lemming (CIS Tagger/Lemmatizer) on two German corpora, provide a semi-automatic error analysis
- Resources: Two German corpora, SMOR (for manual disambiguation), Marmot/Lemming (see Thomas Mueller's web page)
- Programming Language: Python (for the semi-automatic analysis)
- Outcome: Error analysis pointing to strengths and weaknesses of Marmot/Lemming in two domains, python scripts for error analysis
- Abstract and Presentation: German or English, brief presentation on tagging/lemmatization, quantitative and qualitative discussion of results

Problem: German Verbs Have Complex Morphology

- Summary: Create SFST transducers which can be composed to analyze and generate German verbs (regular and irregular)



Project: German Verbs in SFST

- Summary: Create SFST transducers which can be composed to analyze and generate German verbs (regular and irregular)
- Resources: List of German verbs and their inflected forms, SFST
- Programming Language: SFST
- Outcome: Working transducers for analyzing and generating a large list of German verbs including both regulars and irregulars
- Abstract and Presentation: German or English, presentation of basic design of transducers including two examples (both regular and irregular verbs)

Project: English adjectives in SFST

- Summary: Create SFST transducers which can be composed to analyze and generate English adjectives
- Resources: List of English adjectives and their inflected forms, SFST
- Programming Language: SFST
- Outcome: Working transducers for analyzing and generating a large list of English adjectives
- Abstract and Presentation: German or English, presentation of basic design of transducers including examples

Problem: Rule-Based Machine Translation Highly Dependent on Morphology

- “Apertium is a shallow-transfer machine translation system, which uses finite state transducers for all of its lexical transformations, and hidden Markov models for part-of-speech tagging or word category disambiguation.” (source: Apertium Project)
- Summary: look at extending the system, probably the morphologies in the English/German pair

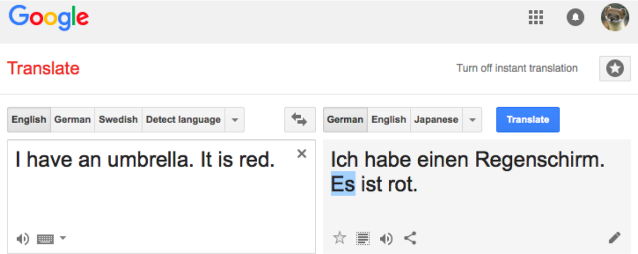
Project: Apertium Rule-Based Machine Translation

- Summary: look at extending the rule-based transfer Apertium system (open source), probably the morphologies in the English/German pair
- Resources: open-source Apertium software, Apertium manual, possibly German/English parallel data provided later
- Programming Language: Python (for checking coverage on corpus, possibly for error analysis, maybe for working with parallel data)
- Outcome: Extension of Apertium data in the English/German language pair
- Abstract and Presentation: English or German, basic presentation of how Apertium works, English and German morphologies, extensions carried out by the group

Analysing Machine Translation Output

The Problem

- Machine translation (e.g. Google Translate) is far from perfect
- For example in English → German translation
  - Incorrect verb inflections
  - Incorrect choice of pronoun
  - etc.



I have an umbrella. It is red.

Ich habe einen Regenschirm. Es ist rot.

Analysing Machine Translation Output

The Task: Find and categorise morphology errors in MT

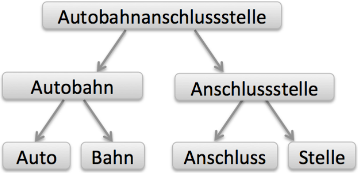
- **Preparation**: select a set of English texts
- **Translation**: translate the texts into German using a translation tool of your choice
- **Analysis**: identify errors in the German translations
- **Categorise**: construct a hierarchy / hierarchies of error categories
- **Write**: prepare guidelines for annotators to follow to label errors according to the categories
- **Assess**: follow the guidelines and annotate the translation of a test file
- **Assess**: assign a severity score to each error category
- **Code**: calculate document stats based on number of errors for each category: counts, average score over words in document, etc.

(also available: German to English translation)

Compound Splitting

The Problem

- German has many compound words, such as:
  - Bananenbrot (Banana bread)
  - Autobahnanschlussstelle (Motorway junction)
  - Donaudampfschiffahrtsgesellschaftskapitän (Danube steamship company captain)
- Long compound words may occur infrequently in text
- In NLP we often want to split them into shorter words to make them easier to handle (e.g. Machine Translation)



Compound Splitting

The Task: Design and build a compound splitter

- **Analysis**: examine a corpus of text and identify some compound words (test set)
- **Research**: read grammar books / look up existing compound splitters
- **Planning**: devise a set of compound splitting rules (or a method of your choice)
- **Development**: code up the method
- **Testing**: apply the method to a corpus of text and analyse the output

- Possible corpus resources:
  - TED Talk corpus: https://wit3.fbk.eu (XML format)
  - Europarl corpus: http://www.statmt.org/europarl/ (text format)

Text Generation

The Task: implement a text generation system

- Create a text generation system which is morphologically aware for German
- The idea is to create interactive narratives for use on touchscreen, and allow systematic changes to the narrative
- For instance, animals referred to by pronouns should be consistent; singular and plural groups should model subject-verb agreement
- Resources: SMOR
- Outcome: basic prototype of narrative generation system with clear morphological components
- Abstract and Presentation: English or German, basic presentation of challenges, how the system works and interaction with German morphological system

Outline

1. Course Requirements
2. How Projects Work
3. Project Topics
4. Forming Groups

Initial Group Discussions

- People discuss three times what to do in groups, grouped left-to-right and forwards-backwards and one move (front- row left, back-row right, forwards-backwards)
  - Please introduce yourselves, and then decide on a topic you could do together
- Email at 19:00 should contain TWO PARTS!:
  - PART ONE: Three teams (with team members!) and topics, in sorted order (preferred to least preferred)
  - PART TWO: Ranking of all 8 topics as an individual (preferred to least preferred)
- I reserve the right to completely ignore your preferences and just assign people however I want, sorry in advance

All Projects

- Tagging/Lemmatization
- SFST German verbs
- SFST English adjectives
- Apertium English-German
- MT Error Analysis English-German
- MT Error Analysis German-English
- Compound Splitting
- Text Generation

Thank you for your attention.