

CL3.101 Computational Linguistics 1

Assignment 2: Morph Analyser and Generator

Course Instructor: Parameswari Krishnamurthy

Deadline: 18th February 2024, 5:00 PM

Instructions

- Your assignment must be implemented using Apertium Ittoolbox.
- Metadata should be prepared as explained in Task 1 and Task 2.
- Tasks should be implemented for Two languages: English and any other language.
- Reverse the morph analyser into generator. Submit both analyser and generator models along with your metadix.
- Dictionary size should be at least 100.
- Test the models before submission.
- Make sure the submitted assignment is your original work. Do not copy any part of the assignment from your friends. Do not refer any AI systems to generate the code.
- No deadline extension will be possible. Please start early in order to finish it on time.
- Make sure to follow the submission format properly. You will be penalised for not following the naming and submission format.

1 Morph Analyser and Generator

Morph Analyser and Generators are important tools for text analysis.

Morph Analyser is a tool to analyse given word forms into root, category and other grammatical features.

Morph Generator is a tool to generate word forms from given root, category and other grammatical features.

1.1 Task 1: Building a paradigm

Choose a base word, perhaps a verb or a noun, and create paradigms for different grammatical features.

For instance, if your base word is "play" create paradigms for various tenses (past, present, future), aspects (perfect, participle, progressive), and persons (first, second, third).

Refer this to understand various grammatical features: <https://unimorph.github.io/doc/unimorph-schema.pdf>

Example:

- Paradigm: RUN
- play

```
<root="run",cat="v",tense="present">
```

- plays

```
<root="run",cat="v",tense="present",gnp="3,sg">
```

- played

```
<root="run",cat="v",tense="past">
```

- played

```
<root="run",cat="v",aspect="participle">
```

- playing

```
<root="run",cat="v",aspect="progress">
```

Similarly, build paradigms for other morphs-phonemically different paradigms.

English morph analyser and generator should cover all possible paradigms.

For other languages, minimum FIVE paradigms should be built.

1.2 Task 2: Building a dictionary

Build a dictionary with root word, category and the related paradigm, Minimum 100 entries are required

Example:

- ROOT,CAT,PDGM
- play,v,play
- work,vplay
- kill,v,play

1.3 Task 3: Using LTToolBox

Write a code to convert Task 1 and Task2 to metadix (.xml format)

A guide to building lttoolbox is given here: <https://wiki.apertium.org/wiki/Lttoolbox>

2 Submission Guidelines

Submit a single zip file on Moodle. Submit the codes and outputs for all the 3 tasks separately, in 3 separate files. You have to submit models also.

2.1 Assignment Evaluation

This assignment is evaluated based on:

- Metadata preparation as in Task 1 and Task 2
- Implementation in lttoolbox
- Testing the models with challenge set of word forms for its accuracy

The name of the main file should be in the format [**RollNo_FirstName_MorphAnalysisAssignment.zip**]