Hunmorph-foma morphological analyzer and generator for Hungarian.

Hunmorh-foma is a 2-way tool. Given a conjugated word it finds the word stem and the used suffices, and given a word stem and the name of the suffices it creates a conjugated word. It is possible to communicate with Hunmorph-foma using client-server model, its working can be therefore fully automatized.

Hunmorph-foma is based on foma, (http://code.google.com/p/foma), which is an open source implementation of the lexc/xfst grammar tool of the Xerpx laboratory. The handbook of the grammar tools of the Xerox laboratory (also known as fsmbook) can be found under: https://victorio.uit.no/langtech/tags/DIVVUN-NO_1_0_RELEASE/gt/doc/book.pdf_1.pdf .

Known open source grammar tools for Hungarian:

hunmorph tool (http://mokk.bme.hu/resources/hunmorph/) , that contains multiple components. It works only in analysis direction, that is, it can find the stem „ablak" and the acc. suffix, if we enter „ablakot", but it is incapable to generate „ablakot" if we enter „ablak ACC". Its dictionary can not be easily modified, since the necessary tools are not well documented. Its dictionary is a little fraction of Hunmorph-foma-s dictionary. Its automatisation is not provided per default, the user has to set up a server-client model. I found its working both slow and memory intensive, which is never an advance.

hunspell (http://sourceforge.net/projects/hunspell/) is primarily a fine spell checker tool. In grammar mode it can only handle about 60% of the possible Hungarian suffixes. Its dictionary is poor, lots of used Hungarian words are missing from it. Its function in grammar mode is instable, with other words, it crashes quite often in that mode. Support for grammar mode is poor, error messages pertaining grammar are typically left unanswered or deleted. List of error messages of hunspell for Hungarian can be found on the http://bug.openscope.org system, the errors there are documented and illustrated.

The above mentioned status made it necessary to provide a primarily grammar handling tool for the Hungarian language.

Pre-requisite to use Hunmorph-foma is the download and successful installation of foma. (http://code.google.com/p/foma)

Usage:

1. Download the Hunmorph-foma tree and untar/unzip it.
2. In the directory, where the Makefile exists:
   sh tools/chksys.sh
   If the answer confirms existence of foma and make,
   make

Command make results in the compilation of Hunmorph-foma, compilation takes about 5-6 minutes on my system.

The first testing:

$ foma -l hfnnum.foma
27.3 MB. 599957 states, 1788091 arcs, Cyclic.

Foma, version 0.9.16alpha

Type "help" to list all commands available.
Type "help <topic>" or help "<operator>" for further help.

foma[1]: up
apply up> kilincs
kilincs+Noun+Nom
apply up> hajókra
hajó+Noun+Plur+Sub
apply up> kellet
kellet+Verb+IndefSg3
apply up> hibás
hibás+Noun+Nom
hibás+Adj+Nom
apply up> és
és+Con
apply up> (Ctrl-D lenyomása)
foma[1]: down
apply down> kilincs+Noun+Posss3s+Dat
Kilincsének
kilincsének
apply down> (Ctrl-C lenyomása)

Of course, it is possible to communicate with Hunmorph-foma using client-server model, its working can be therefore fully automatized. In more detail about this you can read under http://code.google.com/p/foma/wiki/FlookupDocumentation and later on here in Client-server mode part.

Hunmorph-foma's has multiple application possibilities like:

- spell checking
- translation support
- word stemming
- tool that helps creating language corpora
- re-forming of texts
- ...

Hunmorph-foma handles the following word arts:

- nouns
- adjectives
- verbs
- numerals
- pronouns
- adverbs, conjunctives, post-positions
- other word arts....

Conjugation

Noun conjugation is in Hungarian:

					Example
word	Owned suffix.	Familiar suffix	Owner suffix	suffices	sógoromékéra
word	Plural suffix		Owner suffix	suffices	sógorokénak
word		Familiar suffix	Owner suffix	suffices	sógorékéitól

Hunmorph-foma's nomenclatur:

					Example
word	Owned suffix. (Poss)	Familiar suffix (Fam)	Owner suffix (Gen)	suffices	sógoromékéra
word	Plural suffix (Plur)		Owner suffix (Gen)	suffices	sógorokénak
word		Familiar suffix (Fam)	Owner suffix (Gen)	suffices	sógoréktól

For the suffices Hunmorph-foma uses following abbreviations, similarly to Szeged-Korpusz (http://www.inf.u-szeged.hu/projectdirs/hlt/), a Szószablya or Hunspell:

Used name	Suffix
Abl	tól, től
Acc	t, ot, et, öt
Ade	nál, nél
All	hoz, hez, höz
Cau	ért
Dat	nak, nek
Ela	ból, ből
Ess	ul, ül
Fac	vá, vé
For	ként
Forp	képp
Forpen	képpen
Ill	ba, be
Ine	ban, ben

Used name	Suffix
Ins	val, vel
Nom	
Sub	ra, re
Sup	on, en, ön, n
Ter	ig
Tem	kor
Dis	anként, enként
Soc	astul, estül

Dis and Soc-t are used only directly after the unconjugated word, not after Poss, Plur or Fam.

Gen is used only once, even though theoretically it could be used endless deep, (kutyáéé..), however, I have not found any example for that in the used corpora (Szószablya korpusz and Szeged korpusz)

Adjectives

Adjective conjugation matches 100% noun conjugation. Adjectives can be elevated, and they get a prefix in high and highest level.

Adjective elevation:

Ground level	Medium level	High level	Aggravated level
word (noun suffices)	abb, ebb, obb, bb (noun suffices)	leg .. abb, ebb, obb, bb (noun suffices)	legesleg .. abb, ebb, obb, bb (noun suffices)

Hunmorph-foma signs the existenceof elevation using the „Mid” word, in high level and aggravated level there is a prefix leg or legesleg there.

Examples:
szebbek: szebbek szép+Adj+Mid+Plur+Nom
legokosabbak: legokosabbak legokos+Adj+Mid+Plur+Nom
legeslegcsúnyábbak: legeslegcsúnyábbak legeslegcsúnya+Adj+Mid+Plur+Nom

Verbs

Verb suffices go from single, first person...third person to plural, first person to third person.
A verb can stand in infinitive form, which can go from single, first person...third person to plural, first person to third person.
Before the suffix expressing single or plural and the person a verb can have modifications expressing imperative (tat,tet) or repetitive (gat,get) action.
A verb suffix can be objective or objectless, present or past tense, conditional or conjunctive mode.

Example:

						Example
word	repetition	imperative	repetition	imperative	number, person, objectless	írogattatgattatok
word	repetition	imperative	repetition		number, person, objectless	írogattatgatok
word	repetition	imperative			number, person, objectless	írogattatok
word	repetition				number, person, objectless	írogatok
word				imperative	number, person, objectless	íratok
word		imperative	repetition		number, person, objectless	íratgatok
word					number, person, objectless	írok
word					number, person, object	írom
word					number, person, object, conditional	írnám
word					number, person, objectless, conditional	írnék
word					number, person, object, conjunctive	írjam
word					number, person, object, conjunctive	írjak
word	repetition				number, person, object, past	írogattam
word		imperative			number, person, objectless, past	írattál

Hunmorph-foma nomenclatur

- repetition sign is Rep,
- imperative's sign is Imper.
- Objective: Def Objectless: Indef
- Conditional: Cond
- Indirect: Conj
- Past tense: Past

Numerals
Conjugation of numerals is identical to conjugation of nouns

Pronouns

Pronouns can be singular, first...third person and plural first...third person.
Pronouns can be connected to any noun suffix (rám, rád, ...), and also to postfixes (alattam, alattad, ...).

Examples:

- enyém
- mellettünk
- tőletek
- ....

Adverbs, Conjunctives, etc...
Adverbs, Conjunctives and other word sorts have no conjugations.

Irregular nouns

Classes of irregular nouns

1	bokor, bokrot, bokrok	Drop a letter
2	Leave out of a letter	Drop a letter, change o to a
3	hal, halat, halak	o-a hangzóváltás
4	csík csíkot csíkok	Words of high vocals or last sillable high, that receive deep suffices
5	írt írtat írtak	Words of high vocals or last sillable high, that receive deep suffices, change o to a
6	levél, levelet,levelek	Loosing accent when conjugated
7	derék, derekat,derekak	Loosing accent when conjugated, words of high vocals or last sillable high, that receive deep suffices
8	team, teamet teamek	High foreign words, that are conjugated
9	hó, havat, havak	Words changing stem
10	ló lovat lovak	Words changing stem
11	erő, erejét erők	changing stem after poss suffices words
12	fi fiat fiak	fi type words
13	ín inat inak	ín type words
14	bakter, bakterek, bakterok	Alternating suffices, both high and low endings are ok.
15	öcs, báty néne	Words öcs, báty, néne
16	szülő szüleim szülőim	Word szülő
17	tűz tüzet tüzek	Words tűz, fűz, szűz, loosing accents
18	y-i	For certain names y at the end of word is to be handled as i
19	y-y	For certain names y in word is to be handled as i, like Ghyczy
20	ae	Generally to sign poss, both ja/je and a/e are ok. For certain words only ae is ok.
21	jaje	Generally to sign poss, both ja/je and a/e are ok. For certain words

		only jaje is ok.
22	pronouns	Words enyém.. övéik magam.. maguk

In case of irregular words, all compound words, whose last word is irregular, are to be sorted into the .lexc fájl, that handles that irregular word. For example hintaló belongs into the same ,lexc fájl, as ló.

Irregular verbs.

There are about 25 types of verbs. For simplicity, in each groups verbs using ik at the end are in an extra group. For example enhuige7.lexc and enhuige7ik.lexc. Regular verbs are in file enhuige lexc.

Irregular adverbs

1	bátor, bátrabb
2	sok-több, lassú-lassabb, nehéz-nehezebb, könnyű-könnyebb, szép-szebb, hosszú-hosszabb, bő-bővebb, ifjú-ifjabb

Irregular numerals

1	Ezer, ezrek	Drop a letter

Verb prefix

Verb prefixes are not only part of verbs, but also of nouns and adjectives (kinézés, benézés, kilátó, belátó, stb...)

Hunmorph-foma adds Verb prefixes using _LEXICON Prefigekoto before LEXICON Verb, like:

```
LEXICON Root
Verb;
Prefigekoto;

LEXICON Prefigekoto
+Pref+:keresztül Verb;
...
+Pref+:be   Verb;

LEXICON Verb
abajgatásoz AddVerbmelyik;
...
```

This arrangement guarantees, that each verb modifier gets combined with each verb. Such structures are also in files enhuadjigem.lexc, enhuadjmelyige.lexc, enhuadjfnmn.lexc, enhufnige.lexc, in all verb .lexc files, and also in other files, where combination with verb prefixes

is needed.

Adding new words.

Find out the word type, noun, verb, adjective, numeral, pronoun, adverb, other

Find words with similar ending

The word is to be put besides the most similar word with the same Add..; command, as the similar word.

If the word uses verb prefixes, it must be put into a fájl, that handles verb prefixes.

Example:
Let's assume, the facsart word is missing. It is an adjective, and can use verb prefixes. (kifacsart, etc..)

grep "art " lexc/adj/*
enhuadjigem.lexc is a good candidate, because it handles verb prefixes.

word csavart is in that fájl like:
csavart AddAdjboeet;

We add after csavart our word like:
facsart AddAdjboeet;

And after compilation facsart is part of our dictionary.

Two level Add...; commands

In the adj tree there are two leve Add...; commandsm e.g. in enhuadjfnmn.lexc:
AddAdjeEnd
AddAdjeEndat
AddAdjeEndnyi
AddAdjeEndnyiat

These command modify the adjectives, adding the endings szerű, féle, fajta, nyi, i. The modified words will then be conjugated, We save a lot of space using this solution, however, analysis will be a bit less readable, because for the word próbaszerű or próbaféle the ground word is próba and not próbaszerű or próbaféle.

Example:
apply up> próbaszerűt
próba+Adj+Acc
apply up>

Handling compound words.

Oroginally I wanted to handle compound words by adding a listing of simple words to another

listing of simple words. The list of simple words had to contain about 30 thousand words. This way is no good, since a simple compilation takes more than 30 minutes.

Therefore I decided to use words collected from corpora. The word collection is in file lexc/ noun/enhuossz3.lexc. Non regular words had to be transferred from this into the files that handle the non regular words. Testing was done using the word collection itself, the Szeged corpus and the Szószablya corpus. In all cases I used Hunspell a s a filter.

Because translation of enhuossz3.lexc takes on my system ca. 3 minutes, that slows down evaluations on word system, I introduced the usage of the „big" and „small" dictionary. The „big" form is the final forma, the „small" one is without corpus compounds. Switching between the two forms is done automatically using script „smake.sh". Smake.sh usage:
./smake.sh k
Kicsi (small)

or:
.smake.sh n
Nagy (big)

After .smake.sh n or smake.sh k make will compile the selected version.

Advantage of the method is,that switching is simple, fast and avoids errors.
Disadvantage of the method is, that there exist enhu2.foma.kicsi and an enhu2.foma.nagy fájl, that has to be updated whenever enhu2.foma gets updated. Likewise exists Makefile.kicsi and Makefile nagy fájl, that has to be updated each time Makefile gets updated.

Idea for further development:

In analysis case it is useful, if Hunmorph-foma recognizes any word form. In case of generation, however, the most often used word form generation is desirable. In order to achieve the most often used form, the dicionary needs to be reduced and also the .lexc files need some simplification to avoid to generate seldom used word forms. For example in case of the irnánk/irnók word pair clearly the írnánk forma is much more often used, than the írnók form, however, for a correct analysis we can not dismiss the írnók form altogether.

History of Hunmorph-foma writing.

Hunmorph-foma writing started with the tool SFST, (http://www.ims.uni-stuttgart.de/projekte/gramotron/SOFTWARE/SFST.html) because SFST is a simply to overview, well documented analyser/generator, and using it there exists a complete Turkish morphology, trmorph. http://www.let.rug.nl/~coltekin/trmorph/. Turkish is rather similar to Hungarian, the relation German:English corresponst to the relation Hungarian:Turkish. Turkish noun suffixation is a bit simpler than Hungarian one, even though Turkish contains almost all Hungarian irregularities. Turkish verb conjugation is slightly more sophisiticated, than Hungarian, which does cause difficulties for the morphology tool writer. Since Turkish does not use verb prefixes or compound words, this greatly simplifies the writing of a Turkish morphology analyser.

Working with SFST i succeeded to find out the way, how to get SFST work with acceptable memory usage. However, SFST files were that hard to read and maintain, that I felt, it was necessary, to search for a more suitable morphology tool, and I found foma. Foma is perfectly

capable to handle the complexities of the Hungarian language and it does that very effectively.

Utilities

LEXC check:

In order to check the lexc files I wrote a little check program tools/fomachk/fomasyn2.pl, that speedyly checks the integrity of the lexc files- The program, as all utilities is written in perl, and to work with it on needs a perl interpreter, that is on all modern systems readyly available.

Before first usage the $root variable of the fomasyn2.pl program must be set up to show onto the directory, where the Makefile is to be found.

Fomasyn2.pl needs one parameter, the word type to be checked, that is noun, adj, verb, num, fxpp, or misc.

The usage of fomasyn2.pl from the main directory:
$ perl tools/fomachk/fomasyn2.pl verb
$

In case of the fxpp parameter it complaints about unused elements, this is harmless. All other error reports need to be checked, since they can refer to serious problems.

Check of foma files:

/tools/fomachk/fomafom2.pl checks the foma files in the $root directory.
Usage from the root directory:
$ perl tools/fomachk/fomafom2.pl verb
$

fomafom2.pl checks the comments, the closing of the definition lines, checks, if all used definitions are really defined, and if all definitions are actually used.

Client-server mode

An example to using of the client-server mode is in the tools/chkwds directory. Its usage:
E
1. In the do_testup.sh fájl the root directory must be set (on my system the root directory is: /home/en/program/foma/tktest1)
2. sh do_testup.sh
The system's answer: Started flookup server on 127.0.0.1 port 6062
3. perl chkwdlistup.pl szolista.txt
kapunak: kapunak kapu+Noun+Dat
ajtótól: ajtótól ajtó+Noun+Abl
megírják: megírják +Pref+ír+Verb+DefPl3
megírják +Pref+ír+Verb+ConjDefPl3
szebbek: szebbek szép+Adj+Mid+Plur+Nom
legokosabbak: legokosabbak legokos+Adj+Mid+Plur+Nom
legeslegcsúnyábbak: legeslegcsúnyábbak legeslegcsúnya+Adj+Mid+Plur+Nom
jó: jó jó+Adj+Nom

laskagombáért: laskagombáért laskagomba+Noun+Cau
send: at chkwdlistup.pl line 28, <FILE> line 9.

Other utilities

Files in directory tools/perl :

Programs to maintain lexc files:

kivon.pl – called with one parameter, which is a file name
kivon.pl reads the given file.
Reads stdin, and lists all words of it, except the words, that are in the parameter file.

kivon1.pl – called with one parameter, which is a file name
kivon1.pl reads the given file.
Reads stdin, and lists the words of it, that are in the parameter file.

addadj.pl adds the proper Add.. tag to each element of an adjective word list on the way, enhuadjmnevek.lexc requires.

Melyi2.pl selects words, that need a deep ending, even though the last vocal is an i.

Links:

- https://gitorious.org/hunmorph-foma/hunmorph-foma/trees/master
- https://gitorious.org/hunmorph-foma/pages/Home
- https://gitorious.org/hunmorph-foma
- http://code.google.com/p/foma
- https://victorio.uit.no/langtech/tags/DIVVUN-NO_1_0_RELEASE/gt/doc/book.pdf_1.pdf
- http://mokk.bme.hu/resources/hunmorph/
- http://sourceforge.net/projects/hunspell/
- http://bug.openscope.org
- http://code.google.com/p/foma/wiki/FlookupDocumentation
- http://www.ims.uni-stuttgart.de/projekte/gramotron/SOFTWARE/SFST.html
- http://www.inf.u-szeged.hu/projectdirs/hlt/
- http://www.let.rug.nl/~coltekin/trmorph/