**Automatic Generation of Finnish Example Sentences**

This paper is modeled after Jelena Kallas’ description on how to generate a list of headwords and examples for the Estonian Collocations Dictionary semi-automatically from the Estonian National Corpus. The target number of headwords in their project is about 10,000. The goal of the present paper is to update or compile from scratch a dictionary of contemporary Finnish that has about 100,000–110,000 headwords.

As a basis, the current list of 100,000 headwords in the dictionary "Kielitoimiston sanakirja" can be used. The next step is to recognize outdated headwords and add new headwords into the list in a sensible way. One obvious method would be to check the current wordlist against frequencies in a sufficiently large corpus. Since such a corpus is not available, the validity of the items is evaluated intuitively and by Google searches and by looking up available terminological sources. The corpus frequencies may not count as the only definite criterion in selection.

# GDEX for Finnish

In the Estonian project, the words must have absolute frequency not less than 5 not be excluded from examples and headwords. Since the Finnish general purpose dictionary is aimed at native speakers, lower frequencies can be accepted. In the list of parameters below, I have struck through the items in Estonian parameters not be taken in the Finnish parameters.

# 1. Parameters

**GDEX parameters for Finnish module (15-05-15)**

1. whole sentence starting with capital letter and ending with (.), (!) or (?);
2. sentences longer than 5 words;
3. sentences shorter than 20 words;
4. penalize sentences which contain words with frequency of less than 3;
5. penalize sentences with words longer than 20 characters;
6. penalize sentences with more than two commas, brackets, colon, semicolon, hyphen, quotation marks, dash;
7. penalize sentences with proper nouns (tags "N\_Prop\_.\*" or "N\_A\_Prop.\*") ~~and~~  ~~abbreviations~~. Should be tested if it’s useful. For abbreviations, there is no tag so they cannot be excluded but by listing them one by one if necessary.
8. penalize sentences with words from "greylist"? It may be that the character of a large general purpose dictionary would not be at ease with limitations on lexical choice (esp. taboo words); however, some post-edition is then needed. I would like to proceed by testing first with a short greylist with unwanted sentence-initial adverbs.
9. ~~penalize sentences  with pronouns~~*~~minä, sinä, hän, tämä, tuo~~*~~and adverbs~~*~~siellä, tuolla~~*~~.~~ In current example sentences in Kielitoimiston sanakirja, such deictic words are used widely. Especially personal pronouns.
10. penalize sentences which start with a conjunction (tags "CC" and "CS");
11. penalize sentences where ~~lemma~~ a word form is repeated;
12. penalize sentences with tokens containing mixed symbols (e.g. letters and numbers) , URL and email addresses; To be considered: in the morphological analysis, a code "NON-TWOL” stands for unanalysed forms. Often, these forms are colloquial, e.g. "mult", instead of standard form "minulta" ('from me'). Still, I don’t know if such sentences should be outright excluded. There are also many other messy analyses: the tagger does not recognize word forms like “ettei” (= “että ei” ‘that not’) that are crucial for sentence structure.
13. penalize sentences without a finite verb (if possible; the tag for a finite verb is "V").

min([word\_frequency(w, 250000000) for w in words]) >5

formula: >

 (50 \* all(is\_whole\_sentence(), length > 5, length < 20, max([len(w) for w in words]) < 20, blacklist(words, illegal\_chars), 1-match(lemmas[0], adverbs\_bad\_start), min([word\_frequency(w, 250000000) for w in words]) > 3)

 + 50 \* optimal\_interval(length, 10, 12)

 \* greylist(words, rare\_chars, 0.05) \* 1.09

 ~~\* greylist(lemposs, anaphors, 0.1)~~

 ~~\* greylist(lemmas, bad\_words, 0.25)~~

 ~~\* greylist(tags, abbreviation, 0.5)~~

 \* (0.5 + 0.5 \* (tags[0] != conjunction))

 \* (1 - 0.5 \* (tags[0]==verb) \* match(featuress[0], verb\_nonfinite\_suffix))

 ) / 100

variables:

 illegal\_chars: ([<|\]\[>/\\^@])

 rare\_chars: ([A-Z0-9'.,!?)(;:-])

 conjunction: CC, CS

 ~~abbreviation: Y~~

 ~~anaphors: ^(mina-p|sina-p|tema-p|see-p|too-p|siin-d|seal-d)$~~

 adverbs\_bad\_start: ^(kuten|edellä|toiseksi|lisäksi|siksi|näin|esimerkiksi|siis)$

 verb: V

 ~~verb\_nonfinite\_suffix: ^(mata|mast|mas|maks|des)$~~

 ~~bad\_words: ^(loll|jama|kurat|kapo|kanep|tegelt|sitt|pätt|in|mats|homo|pagan|joodik|idioot|nats|point|kesik|aa|neeger|veits|jurama|narkomaan|jobu|siuke|õps|perse|tibi|riist|aint|tiss|pask|raisk|raisk|värdjas|prostituut|pedofiil|mupo|gei|suli|porno| – – häbedus|häbemepilu|jobama|kuselema|kagebeelane|munapiiks|oolrait|beibe|jobutus|sigarijunn|sitavedaja|dolbajoob|jobisema|pipravitt|türahiinlane|perseklile|tindinikkuja)$~~

# For comparison, the original parameters and scripts:

**GDEX parameters for Estonian module (1.07.14)**

1. whole sentence starting with capital letter and ending with (.), (!) or (?);
2. sentences longer than 5 words;
3. sentences shorter than 20 words;
4. penalize sentences which contain words with frequency of less than 5;
5. penalize sentences with words longer than 20 characters;
6. penalize sentences with more than two commas, also with brackets, colon, semicolon, hyphen, quotation marks, dash;
7. penalize sentences with words starting with a capital letter. Penalize sentences with H (=Proper noun) and Y (=abbreviation) POS-tag.;
8. penalize sentences with words from "greylist";
9. penalize sentences  with pronouns  *mina*, *sina*, *tema*,*see*, *too*and adverbs*siin*, *seal*, sentences shouldn’t start with pronouns *mina, sina, tema* and local adverbs *siin*, *seal.*
10. penalize sentences which start with conjunction. Penalize sentences which start with punctuation mark (typical informal texts) and with J (=conjuction) POS-tag.
11. penalize sentences where lemma is repeated;
12. penalize sentences with tokens containing mixed symbols (e.g. letters and numbers) , URL and email addresses;
13. sentence should contain verb as predicate, otherwise sentence is very elliptical.

min([word\_frequency(w, 250000000) for w in words]) >5

formula: >

 (50 \* all(is\_whole\_sentence(), length > 5, length < 20, max([len(w) for w in words]) < 20, blacklist(words, illegal\_chars), 1-match(lemmas[0], adverbs\_bad\_start), min([word\_frequency(w, 250000000) for w in words]) > 3)

 + 50 \* optimal\_interval(length, 10, 12)

 \* greylist(words, rare\_chars, 0.05) \* 1.09

 \* greylist(lemposs, anaphors, 0.1)

 \* greylist(lemmas, bad\_words, 0.25)

 \* greylist(tags, abbreviation, 0.5)

 \* (0.5 + 0.5 \* (tags[0] != conjunction))

 \* (1 - 0.5 \* (tags[0]==verb) \* match(featuress[0], verb\_nonfinite\_suffix))

 ) / 100

variables:

 illegal\_chars: ([<|\]\[>/\\^@])

 rare\_chars: ([A-Z0-9'.,!?)(;:-])

 conjunction: CC, CS

 abbreviation: Y

 anaphors: ^(mina-p|sina-p|tema-p|see-p|too-p|siin-d|seal-d)$

 adverbs\_bad\_start: ^(kuten|edellä|toiseksi|lisäksi|siksi|näin|esimerkiksi|siis)$

 verb: V

 verb\_nonfinite\_suffix: ^(mata|mast|mas|maks|des)$

 bad\_words:^(loll|jama|kurat|kapo|kanep|tegelt|sitt|pätt|in|mats|homo|pagan|joodik|idioot|nats|point|kesik|aa|neeger|veits|jurama|narkomaan|jobu|siuke|õps|perse|tibi|riist|aint|tiss|pask|raisk|raisk|värdjas|prostituut|pedofiil|mupo|gei|suli|porno|

* –

häbedus|häbemepilu|jobama|kuselema|kagebeelane|munapiiks|oolrait|beibe|jobutus|sigarijunn|sitavedaja|dolbajoob|jobisema|pipravitt|türahiinlane|perseklile|tindinikkuja)$