SWEGRAM
Guidelines to Annotation and Analysis of English and Swedish Texts

---

Beáta Megyesi and Rex Ruan

Department of Linguistics
Stockholm University

Preface

This document describes the tool SWEGRAM[1], with the help of which you can apply automatic annotation and linguistic analysis of English or Swedish texts. With SWEGRAM you can also create your own annotated collection of texts. We will be presenting the components of the tool and offering suggestions for how you can use the tool to conduct large-scale, empirical linguistic analysis.

SWEGRAM was initially developed to enable large-scale linguistic analysis of Swedish texts and has since been adapted to process English texts as well. The long-term goal is to make language-based text materials accessible as primary research data for the humanities and social sciences, supported by advanced tools for the processing of both written and transcribed spoken language.

SWEGRAM was originally developed through a collaboration between the Department of Linguistics and Philology and the Department of Scandinavian Languages at Uppsala University, Sweden, as part of the Swe-Clarin project[2], the Swedish node of the European CLARIN infrastructure. The project has received long-term support from the Swedish Research Council. The tool is currently hosted and maintained by the Department of Linguistics at Stockholm University.

These guidelines update our previously published SWEGRAM guidelines Megyesi et al., 2019, originally released in Swedish and based on an earlier version of the tool that applied exclusively to Swedish texts. Since 2019, we have enhanced SWEGRAM with new functionalities, including extended syntactic features, improved visualization and export options, and an upgraded back-end. Additionally, the tool now supports automatic annotation and analysis of English texts.

We would like to express our gratitude to everyone who contributed to the development of the tool. In particular, we thank Anne Palmér from the Department of Scandinavian Languages at Uppsala University for initiating the development of SWEGRAM and her significant contributions to its first version. Special thanks go to Jesper Näsman for implementing the first online version of SWEGRAM and Shifei Chen for upgrading the tool's back-end to enhance processing speed. Lastly, we are thankful to the students of the Master's program in Language Technology at Uppsala University for their valuable feedback, debugging efforts, and evaluations over the years.

[1] https://swegram.ling.su.se
[2] https://sweclarin.se

Table of contents

1 Introduction

SWEGRAM is a tool that enables the annotation and analysis of Swedish and English texts. Users can upload one or more texts for linguistic analysis, including morphological and syntactic features. These annotated texts can then be used for quantitative linguistic analysis. For example, the tool provides statistics on sentence length, word count, various readability metrics, part-of-speech (PoS) distribution, and the frequency of lemmas, PoS tags, or misspelled words. It also visualizes syntactic relationships between words in sentences and offers detailed insights into the distribution of syntactic functions and relations within the text. Figure 1 illustrates the two main components of the tool—annotation and analysis—which are described in detail in this manual.

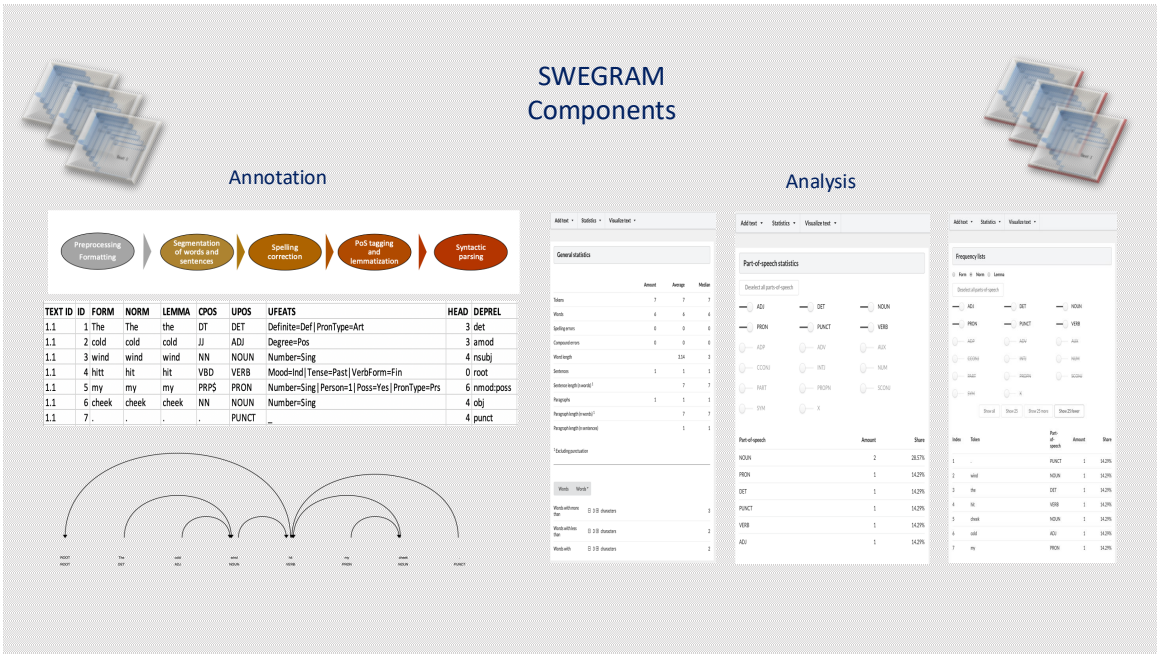


Figure 1: The SWEGRAM components.

With SWEGRAM, you can easily create your own linguistically annotated text collection, or corpus, and generate statistics on the linguistic features of your texts. The tool is user-friendly and requires no prior knowledge of automated text processing or programming skills. SWEGRAM is freely accessible to all users through its website. Uploaded texts are stored for one week before being automatically deleted from the server. Users remain anonymous, as no personal information is collected. However, this means you cannot resume work on the same file later without re-uploading it.

This manual explains how to use SWEGRAM and its various functions for analyzing both English and Swedish texts. The tool also includes a help function that provides a summary of its components and features. If you need additional information not covered in the manual or have any questions, please feel free to contact us using the information provided on the SWEGRAM website.

2 Choosing Language

SWEGRAM can be used to annotate and analyze both Swedish and English texts. On the main page, you can select between the two languages. SWEGRAM automatically applies the Swedish annotation pipeline to Swedish texts and the English pipeline to English texts. To help

distinguish between them, the navigation bar has different background colors: red for English and yellow for Swedish. If the wrong language pipeline is used, the annotation and analysis will not be accurate.

3 Upload Texts

After selecting the language, you can upload your texts. In this section, we explain the required file formats for unannotated texts and for previously annotated texts if you wish to conduct further analysis. Additionally, we outline the available annotation settings.

3.1 Upload unannotated texts

To analyze a text, you can either paste it directly into the browser or upload one or more files. For unannotated texts, the system accepts only plaintext (.txt) files, preferably encoded in the universal character standard, Unicode UTF-8. You can easily save your file in UTF-8 format using any standard text editor.

You can either write or paste a text directly into the provided field, or select a file by clicking **Upload** and choosing a text from your local device, as shown in Figure 2. To upload multiple texts, combine them in a single file, separated by metadata lines (see Section 3.2). Once uploaded, the files will appear in the text selection area under **Statistics** or **Visualize**, as illustrated in Figure 3. s

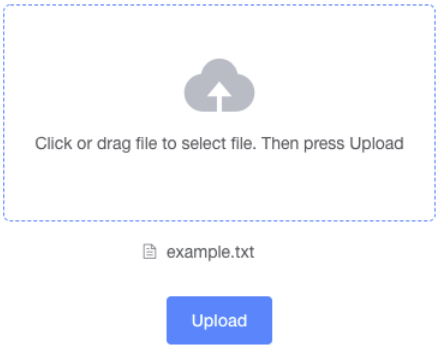


Figure 2: To upload file for annotation.

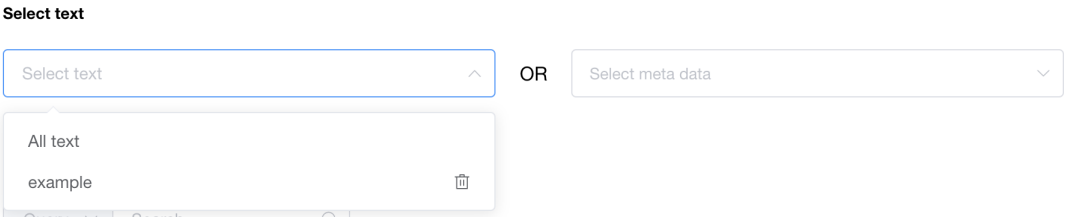


Figure 3: Uploaded file for annotation.

3.2 Text with metadata

The text file may include metadata that provides information about the text, such as the author or its origin. This feature is particularly useful for comparing texts based on metadata, such as examining differences between male and female authors, or texts produced across different times and locations.

A file can contain one or multiple metadata entries, each of which must follow a specific format to be processed correctly by the tool. Each metadata line should start with < and end with >. Within the metadata line, information is structured as FEATURE:VALUE pairs, where a colon (:) separates the feature from its value, and multiple pairs are separated by semicolons (;). The metadata format is illustrated in Figure 4.

Please note that each feature must be unique within a metadata line, and each feature can have only one value. However, you can include as many features as needed, in any order.

If you wish to mark a new text in the file without including FEATURE:VALUE pairs, you can simply use an empty metadata line, marked with just the start and end symbols, <>.

(a) Plain Text

(b) Visualized text with metadata

Figure 4: Labels from plain text to SWEGRAM.

Metadata can indicate the beginning of a new text when the file contains multiple texts. These metadata lines can be used for text selection, as shown in Figure 5. Including metadata is optional and can be omitted from the uploaded file if not needed.

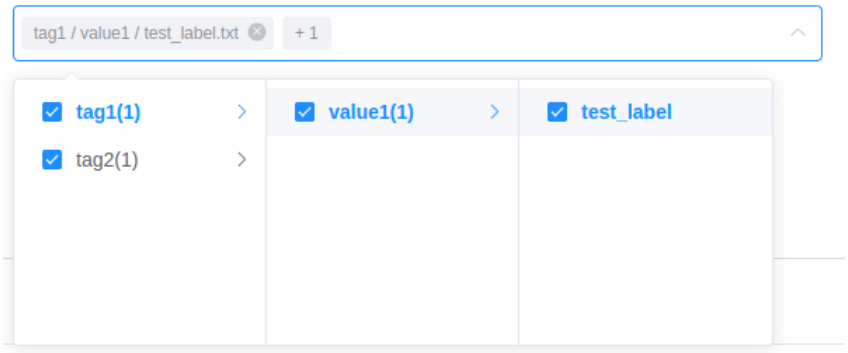


Figure 5: Metadata for selection.

If your texts include metadata, SWEGRAM will read and process this data. However, please note that SWEGRAM can only interpret metadata if the file begins with properly structured metadata, as outlined above. We give detailed example of how to use metadata in Section 7.

3.3 Upload annotated texts

If you wish to upload an already annotated text in the format required by SWEGRAM for further analysis, you can do so using a CSV file, as described in detail in Section 4.1. This option is useful if you want to correct a previously processed file and use the updated version for further analysis. In such cases, you can activate the button that indicates the file has already been annotated. When this button is enabled, the file will bypass the standard annotation process, and the system will run an annotation checker instead to verify the file's format.

If the uploaded file is in CSV format, it will be converted into plaintext. Any errors detected by the annotation checker will be displayed with an index reference, indicating the type of errors found. A list of possible error messages is provided in the Appendix.

Uploading an annotated file can be time-consuming, as the system not only checks if the text meets the formatting requirements but also inserts the text into the internal database used by SWEGRAM.

3.4 Annotation setting

SWEGRAM offers a series of tools to annotate sentences at various linguistic levels. These include tokenization for segmenting words and sentences, normalization for correcting misspelled words, and PoS tagging/parsing for morphosyntactic analysis, as shown in Figure 6. By default, *Tokenization* and *PoS tagging/Parsing* are enabled. If you need to correct misspelled words, you should activate the *Normalization* option.

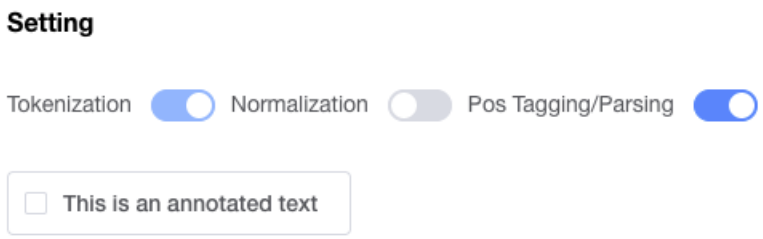


Figure 6: Choosing annotation components.

If you are uploading an already annotated text in the required format, be sure to check the box labeled *This is an annotated text*.

Once your text(s) has been uploaded, it will be processed using the linguistic annotation tool chain developed for each language.

Since text files can be large and may take considerable time to upload and annotate, we have set file size limits: 10 MB for non-annotated, raw texts.

4 Annotation

Once uploaded by pressing the *Upload* button, each text is processed linguistically by a series of components of SWEGRAM, as illustrated in Figure 7.

For each component, we employ state-of-the-art tools from the field of natural language processing to automatically process and annotate texts with proven accuracy. The annotation process is based on tools specifically designed for the analysis of English and Swedish languages, respectively.

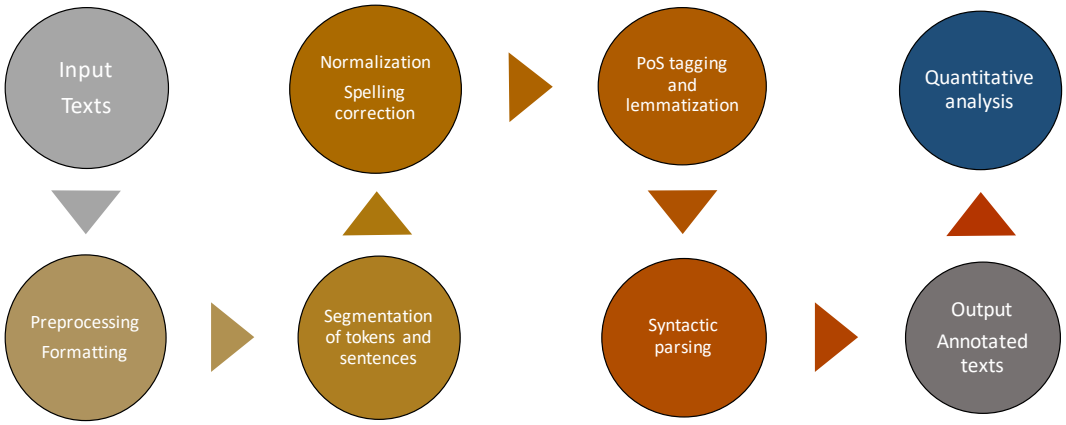


Figure 7: The SWEGRAM components for linguistic annotation.

The first stage of the annotation process reformats the uploaded file into UTF-8. Words are then separated from punctuation and segmented into linguistic units through tokenization, placing each word on its own line, with sentences separated by an empty line. Misspelled words are identified and corrected by the normalizer. The corrected, normalized text is then analyzed using a PoS tagger, which annotates each word and punctuation mark with its part-of-speech (PoS) tag and morphological information. The lemma (base form) of each word is also identified. Finally, sentences undergo syntactic analysis by a parsing module, which determines the relationships between words and their functions within the sentence. The components of the annotation process are described in more detail in Section 4.2.

If a file contains any personal comments that you do not want to be annotated or analyzed by SWEGRAM, you should start each comment line with a hashtag (#). This ensures that the lines containing additional information will not be treated as part of the text for analysis.

You can view the results of your analyzed texts directly on the SWEGRAM window under the *Statistics* and *Visualize* tabs, where various linguistic characteristics of your uploaded texts are displayed. You also have the option to download your annotated texts, with or without accompanying statistics, in text (.txt), CSV (.csv), or Excel (.xlsx) formats for saving on your computer. The following section provides an overview of the annotation format, followed by a step-by-step explanation of the annotation process and the resulting linguistic analysis.

4.1 Annotation format

Texts annotated using the tool are represented in a unique format, with annotations applied at various linguistic levels according to a specific standard. We use Unicode (UTF-8) for character encoding. To represent these annotations and align with international standards, we have adopted the tab-separated CoNLL-U format. In this format, each token (i.e., each word and punctuation mark) is placed on its own line along with its analysis, and each new sentence begins with an empty line.

Each token is analyzed at multiple linguistic levels, with all analyses represented on the same line as the token. These analyses are displayed in columns, separated by tabs, and include the ID number for the paragraph, sentence, and token, as well as the linguistic analysis. The linguistic analysis is performed at both the word level—through PoS annotation and morphological information—and at the sentence level, through syntactic analysis. PoS annotation includes two types: universal PoS tagsets (Nivre et al., 2016) and language-specific tagsets. The Stockholm-Umeå Corpus tagset (Ejerhed et al., 1992; Gustafson-Capková & Hartmann, 2006) is used for Swedish, and the Penn Treebank tagset (Marcus et al., 1993) is used for English. These are discussed in more detail in Section 4.2.4 on part-of-speech analysis and Section 4.2.5 on syntactic analysis.

Table 1 summarizes the annotation representation. On the left-hand side, the names of the features listed for each token in the columns are provided, followed by a description of each feature. It is important to note that the order of the columns for specific features may vary between languages, particularly the arrangement of universal and language-specific PoS and morphological features (`UPOS`, `XPOS`, `CFEATS`, and `XFEATS`).

`TEXT-ID`	Paragraph and sentence index, integer starting at 1 for each new paragraph and sentence.
`ID`	Token index, integer starting at 1 for each new sentence; may be a range for originally multiword tokens that have been split due to misspelling.
`FORM`	Word form or punctuation mark, so called token.
`NORM`	Corrected/normalised token for misspelled words.
`LEMMA`	Lemma or stem of word form.
`UPOS`	Part-of-speech based on Universal PoS tagset.
`XPOS`	Part-of-speech based on the Stockholm-Umeå Corpus PoS tagset for Swedish and Penn Treebank PoS tagset for English
`CFEATS`	List of morphological features from the Stockholm-Umeå Corpus; "_" if feature is missing. This is only valid for Swedish text analysis.
`UFEATS`	List of morphological features from the Universal PoS tagset; "_" if feature is missing.
`HEAD`	Head of the current word, which is either a value of `ID` or zero (0) if the word is the `ROOT` of the sentence.
`DEPREL`	Dependency relation to the HEAD based on the Universal dependency relations.
`MISC`	Any other annotation.

Table 1: Annotation representation.

Figure 8 illustrates the output from the linguistic annotation for the sentence *The cold wind hit my cheek.*, which includes the misspelled token *hitt* that SWEGRAM's normalization module corrected to *hit*. The comment beginning with a hashtag (# Analyzed sentence 21.06.2020) is not processed by the tool. The bolded line following the comment in the first raw of the example was added to clarify the type of annotation each column represents.

Column 1 states the `TEXT-ID` number for the paragraph followed by the sentence. In the above example, the relevant sentence is in the first paragraph and is the first sentence in the paragraph (1.1). The second column `ID` shows the position of each token in the sentence. The third column `FORM` renders the writer's original text token for token, both words and punctuation. The fourth column `NORM` renders the normalised text, with any misspellings corrected. The fifth column `LEMMA` states the base form of each token. This is followed by the linguistic

#Analyzed sentence 21.06.2020.

TEXT ID	ID	FORM	NORM	LEMMA	UPOS	XPOS	FEATS	HEAD	DEPREL
1.1	1	The	The	the	DET	DT	Definite=Def|PronType=Art	3	det
1.1	2	cold	cold	cold	ADJ	JJ	Degree=Pos	3	amod
1.1	3	wind	wind	wind	NOUN	NN	Number=Sing	4	nsubj
1.1	4	hitt	hit	hit	VERB	VBD	Mood=Ind|Tense=Past|VerbForm=Fin	0	root
1.1	5	my	my	my	PRON	PRP$	Number=Sing|Person=1|Poss=Yes|PronType=Prs	6	nmod:poss
1.1	6	face	face	face	NOUN	NN	Number=Sing	4	obj
1.1	7	.	.	.	PUNCT	.	_	4	punct

Figure 8: Output – linguistic annotation.

analysis regarding PoS and syntactic information. The UPOS column lists the PoS for each token according to the universal PoS tagset as defined in Universal Dependencies (UD). The XPOS column also lists the PoS annotation, but this time based on the Penn Treebank tagset (Marcus et al., 1993) for English texts and Stockholm-Umeå Corpus (SUC) 3.0 for Swedish texts (Gustafson-Capková & Hartmann, 2006). The next two columns state the morphological features; the CFEATS column (which in Figure 8 is left out since it is only valid for Swedish text) according to SUC tags and the UFEATS column according to UD tags. The HEAD and DEPREL columns show the syntactic analysis based on UD. HEAD shows the head word in the sentence, while DEPREL shows the syntactic function to the head word of the sentence. To explain the annotation, we give a detailed explanation of the sentence below.

- The sentence appears in the first sentence of the first paragraph of the text (1.1) given in column TEXT-ID for each token of the sentence.
- The word *The* is the first word of the sentence with ID number 1. The original form of the word is *The* as given in column FORM and since the word is correctly spelled the same form is given in column NORM. The base form of the word *The* is *the* given in the column for LEMMA. The word is a determiner and is annotated as *DET* according to UD PoS tagset UPOS and as *DT* as defined by Penn Treebank tagset XPOS. The morphological analysis of the word is definite form w.r.t. species (DEF resp. Definite=Def) and its pronominal type is article. The word *The* has its head word *wind*, which is the third word with ID 3 in the sentence. *The* has determiner (*det*) as dependency relation (DEPREL) to its head word *wind*.
- The second word in the sentence is *cold*, with ID 2. Its original spelling FORM is *cold*, which is correctly spelled and has the same, normalised form in column NORM. The base form LEMMA is *cold*. It is an adjective marked as *ADJ* in UD (U-POS). Its head word is *wind* which is the third in the sentence and serves as an adjectival modifier (*amod*) to its head.
- Word number 3 is *wind* which is correctly spelled and its lemma is *wind*. It is a noun (NN marked by XPOS and NOUN as UPOS) and it is in singular form |Number=Sing given the UD tag. The noun *wind* has the following word, with ID=4 as its head word (*hitt*) and it is the subject NSUBJ in the sentence.
- The word number 4 *hitt* (FORM) has been misspelled and corrected to *hit* by SWEGRAM, which is shown in column NORM. The lemma is *hit* which is a verb (VERB resp. VB) with the morphological features in indicative (Mood=Ind), past tense (Tense=Past), finite form (VerbForm=Fin), and active (Voice=Act). Since the finite word of the sentence is the root of the sentence its function is ROOT.
- Word number 5 is *my* with the same base form and it is a pronoun (PRON resp. PRPS) in singular (Number=Sing), first person Person=1), possessive (Poss=Yes) form and

its subcategory type is personal pronoun (`PronType=Prs`). Its head word is the word with ID=6 (*cheek*) and serves as possessive modifier (`NMOD:POSS`) to its head noun.

- Word number 6 is *cheek* with the same base form and is a noun (`NOUN` resp. `NN`) in singular form (`Number=Sing`). Its head word is the finite verb *hitt* which is word number 4 and its function is direct object (`obj`).
- The sentence ends with the punctuation mark "." with `ID` number 7 and has its PoS tag (`PUNCT` resp. `MAD`) without any morphological features (-). Its head word is the finite verb, token number 4 and has punctuation (`PUNCT`) as function.

4.2 Annotation chain

Annotation is performed automatically through a chain of tools designed to annotate Swedish and English texts at both the word and sentence levels. By default, all tools in the chain are applied, except for normalization. If you wish to perform spelling correction, you need to enable the *Normalization* module. You can also choose to exclude certain modules from the tool chain if you prefer to work with specific analysis components. Detailed instructions on how to customize these settings are provided in Section 4.3.

4.2.1 Tokenization and sentence segmentation

Once the text has been formatted, it undergoes tokenization. Tokenization, a prerequisite for further annotation, divides the text into sentences and words while separating punctuation marks from words. This process splits the text into tokens (i.e., words and punctuation) and saves them in Column 3, as illustrated in 8. Each token is placed on a separate line, with new sentences preceded by an empty line. Abbreviations are kept as single words/tokens, and separated compound words are not corrected at this stage. For example, in the sentences *This is e.g. tokenized. The sentences are segmented.* the abbreviation *e.g.* appears on a separate line as a single token, as shown in Figure 9.

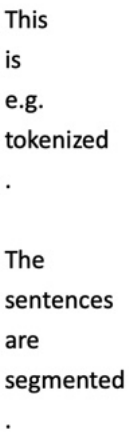


Figure 9: An example of a tokenized and segmented text.

Tokenization and sentence segmentation are based on the tokenization module developed for the Swedish Treebank (Cap et al., 2016), which is now a component of EFSELAB (Östling, 2016).

4.2.2 Normalisation, spelling and separated compound words

Misspelled words and incorrectly separated compound words can be corrected using the normalization module. In Figure 8, the original form of the word is displayed in the `FORM` column, while the corrected version appears in the `NORM` column. This means the original text is represented word by word in the `FORM` column, and the corrected sentence is displayed word by word in the `NORM` column.

For compound words that are incorrectly separated, a common misspelling type in Swedish, the elements are combined on a new row and indexed using the same numbers as the original words, separated by a hyphen. An example of this indexing and annotation for the compound word *inspirations källa* (English: source of inspiration) is shown in Figure 10. Word 2 (*inspirations*) and word 3 (*källa*) are not analyzed separately. Instead, the correct word *inspirationskälla*, now with the ID number 2-3 (combining the ID numbers of the original words), is shown in the `NORM` column and has been linguistically analyzed. In further statistical analysis, the separated compound word is excluded, and only the corrected form is used.

English translation	TEXT ID	ID	FORM	NORM	LEMMA	UPOS	UFEATS	HEAD	DEPREL
My	1.1	1	Min	Min	min	DET	Definite=Def|Gender=Com|Number=Sing|Poss=Yes	2-3	nmod:poss
	1.1	2-3		inspirationskälla	inspirationskälla	NOUN	Case=Nom|Definite=Ind|Gender=Com|Number=Sing	4	nsubj
inspiration	1.1	2	inspirations						
source	1.1	3	källa						
comes	1.1	4	kommer	kommer	komma	VERB	Mood=Ind|Tense=Pres|VerbForm=Fin|Voice=Act	0	root
from	1.1	5	från	från	från	ADP	_	6	case
text	1.1	6	texten	texten	text	NOUN	Case=Nom|Definite=Def|Gender=Com|Number=Sing	4	obl
.	1.1	7	.	.	.	PUNCT	_	4	punct

Figure 10: Annotation of the incorrectly spelled compound as two separate tokens.

Spelling correction is handled by a modified version of the Hist-Norm tool (Pettersson et al., 2013), originally developed for normalizing words in historical texts to modern spellings. A rule-based system is used to identify and correct separated compound words. Since the tool is still under development, it may not catch every misspelling or separated compound word. If a fully corrected version of the text is essential for your analysis, we recommend manually correcting the text before proceeding with linguistic analysis.

4.2.3 Lemmatisation

The base form, or lemma, of a word is typically its least inflected form and is generally the version you would find in a dictionary. The lemma is identified and displayed in the (`LEMMA`) column of the output, as shown in Figure 8. Lemmatization occurs alongside PoS tagging, using the EFSELAB tool (Östling, 2016).

4.2.4 Part-of-speech analysis

The PoS tagger is responsible for assigning Part-of-Speech (PoS) tags and morphological features to the text. For PoS analysis, three primary tagsets are referenced:

- *Universal Tagset*: A standardized set of PoS tags applicable across multiple languages as defined by the Universal Dependency framework (Nivre et al., 2016). The Universal PoS tags are defined in Table 2. The analysis is provided in the column named `UPOS`.
- *SUC Tagset*: Specifically designed for the Swedish language in the Stockholm-Umeå Corpus (Gustafson-Capková & Hartmann, 2006). The tagset is presented in Table 3 with examples in Swedish followed by their English translations in parentheses. The analysis is provided in the column named `XPOS`.

- *Penn Treebank Tagset*: The language specific tagset for English, the widely recognized Penn Treebank (Marcus et al., 1993) for the annotation of English is described in Table 11. The analysis is provided in the column named `XPOS`.

Tag	Explanation	Example
ADJ	Adjective	fine
ADP	Adposition (preposition)	on
ADV	Adverb	quick, very
AUX	Auxiliary verb	have
CCONJ	Conjunction	and
DET	Article/Determiner	a
INTJ	Interjection	yeah
NOUN	Noun	car
NUM	Numeral	two
PART	Particle	out
PRON	Pronoun	he
PROPN	Proper noun	Jenny
PUNCT	Punctuation mark	, .
SCONJ	Subordinating conjunction	that
SYM	Symbol	☺
VERB	Verb	in
X	Other	xbbe

Table 2: Universal PoS tags (`UPOS`).

Morphological analysis reproduces a set of features that describe the word's lexical and grammatical characteristics. Lexical features include subcategories of PoS, such as nouns/proper nouns/types of proper noun. Grammatical features describe the categorization of PoS for a given word form; for example, gender, numeral expression, case and species for nominal word classes (noun and pronoun) or case, species, comparative degree and numeral expression for adjectives. The type of morphological analysis varies between the tagsets.

Universal morphological features are stated in the form of Feature = Value Pair, where the feature indicates the morphological category (which may be in abbreviated form) and the value describes the actual features of the word. For example, the adjective *cold* is described as: `Case=Nom|Definite=Indef|Degree=Pos|Number=Sing`, meaning that the word *cold* is nominative in case (`Case=Nom`), indefinite in form (`Definite=Indef`), the comparative degree is positive (`Degree=Pos`), and it is singular in number (`Number=Sing`). Some of the most important morphological features in UD for English and Swedish are listed in Table 4. The morphological analyzis of the UD tagset is provided in the column `UFEATS`.

Unlike UD, SUC does not represent the feature itself but only the value for the morphological feature in question. SUC's morphological features are described in Table 5 and the analysis is provided in the column `XFEATS`.

The interested reader is referred to UD's description, the SUC 2.0 Manual (Gustafson-Capková & Hartmann, 2006) and the Penn Treebank documentation (Marcus et al., 1993) for further details. Annotation of PoS with morphological features is performed with the aid of EFSELAB (Östling, 2016).

Tag	Explanation	Example
AB	Adverb	inte (not)
DT	Determiner	ett (an)
HA	Interrogative/relative adverb	när (when)
HD	Interrogative/relative determiner	vilken (which)
HP	Interrogative/relative pronoun	som (as)
HS	Interrogative/relative possessive pronoun	vars (whose)
IE	Infinitive mark	att (to)
IN	Interjection	ja (yes)
JJ	Adjective	fin (pretty)
KN	Conjunction	och (and)
MAD	Major delimiter	. ? ! :
MID	Minor delimiter	, - ; / *
NN	Noun	bil (car)
PAD	Pairwise delimiter	([
PC	Participle	dansande (dancing)
PL	Particle	in (in)
PM	Proper noun	Jenny
PN	Pronoun	han (he)
PP	Preposition	på (on)
PS	Possessive pronoun	hennes (her)
RG	Cardinal number	två (two)
RO	Ordinal number	andra (second)
SN	Subjunction	att (to)
UO	Foreign word	nota bene
VB	Verb	fira (celebrate)

Table 3: PoS tags in SUC 2.0.

Feature	Value	Explanation	PoS
Case (Kasus)	Nom Acc Gen	Nominative Accusativ Genitive	ADJ, NOUN, PRON, PROPN
Definiteness (Definite)	Ind Def	Indefinite Definite	ADJ, DET, NOUN, PRON, PROPN
Gender (Genus)	Com Neut	Utrum Neutrum	ADJ, DET, NOUN, PRON, PROPN
Number (Numerus)	Sing Plur	Singular Plural	ADJ, DET, NOUN, PRON, PROPN
Possessiv (Possessive)	Yes	Possessive	DET
Degree (Komparationsgrad)	Pos Cmp Sup	Positive Comparative Superlative	ADJ, ADV
Mood (Modus)	Ind	Indicative	AUX, VERB
Tense (Tempus)	Pres Past	Present Past	AUX, VERB
VerbForm (Finithet)	Fin Inf	Finite Infinite	AUX, VERB
Voice (Diates)	Act Pass	Active Passive	AUX, VERB
Abbreviation (Förkortning)	Yes	Abbreviation	ADV

Table 4: Morphological features in UD for Swedish.

Feature	Value	Explanation	PoS
Gender	UTR NEU MAS	Utrum Neutrum Masculinum	DT, HD, HP, JJ, NN, PC, PN, PS, (RG, RO)
Number	SIN PLU	Singular Plural	DT, HD, HP, JJ, NN, PC, PN, PS, (RG, RO)
Definiteness	IND DEF	Indefinite Definite	DT, (HD, HP, HS), JJ, NN, PC, PN, (PS, RG, RO)
Case	NOM GEN	Nominative Genitive	JJ, NN, PC, PM, (RG, RO)
Tense	PRS PRT SUP INF	Present Past Supinum Infinite	VB
Voice	AKT SFO	Active Passive or deponens	VB
Mood	KON	Conjunctive	VB
Participle	PRS PRF	Present Perfect	PC
Grad	POS KOM SUV	Positive Comparative Superlative	(AB), JJ
Pronoun	SUB OBJ	Subject form Object form	PN
	SMS	Compound	All PoS

Table 5: Morphological features in SUC 2.0.

CC	Coordinating conj.	TO	infinitival *to*
CD	Cardinal number	UH	Interjection
DT	Determiner	VB	Verb, base form
EX	Existential there	VBD	Verb, past tense
FW	Foreign word	VBG	Verb, gerund/present pple
IN	Preposition	VBN	Verb, past participle
JJ	Adjective	VBP	Verb, non-3rd ps. sg. present
JJR	Adjective, comparative	VBZ	Verb, 3rd ps. sg. present
JJS	Adjective, superlative	WDT	Wh-determiner
LS	List item marker	WP	Wh-pronoun
MD	Modal	WP$	Possessive *wh*-pronoun
NN	Noun, singular or mass	WRB	Wh-adverb
NNS	Noun, plural	#	Pound sign
NNP	Proper noun, singular	$	Dollar sign
NNPS	Proper noun, plural	.	Sentence-final punctuation
PDT	Predeterminer	,	Comma
POS	Possessive ending	:	Colon, semi-colon
PRP	Personal pronoun	(	Left bracket character
PP$	Possessive pronoun	)	Right bracket character
RB	Adverb	"	Straight double quote
RBR	Adverb, comparative	`	Left open single quote
RBS	Adverb, superlative	``	Left open double quote
RP	Particle	'	Right close single quote
SYM	Symbol	''	Right close double quote

Figure 11: The Penn Treebank PoS tagset incl. morphological features.

4.2.5 Syntactic analysis

Syntactic analysis is grounded in the Universal Dependency framework[3] (Nivre et al., 2016). Words are linked in pairs, representing binary relationships where one word serves as the head and the other as its dependent. The verb acts as the central element of the sentence and is designated as the head word.

In contrast to other dependency formalisms that treat the finite verb (and any auxiliary verbs) as the head, UD Version 2 (UD2), which we employ, identifies the verb carrying the *semantic* information as the head. All other tokens, including words and punctuation marks, are connected to this semantic verb through directed arcs. In UD2, arcs are directed from dependents to their head words, illustrating the relationships between them.

This structure signifies the connection between a head word and its dependent(s), represented as linked arcs with the words functioning as nodes. The syntactic or grammatical functions are typically indicated on the links between each head-dependent relationship. Table 6 below presents the syntactic relationships along with links to their definitions.

The syntactic analysis in SWEGRAM's output is represented in two columns: `HEAD` and `DEPREL`. We illustrate the syntactic analysis for the sentence *The cold wind hit my cheek* in two formats: the CoNLL-U format, shown in Figure 12 and as a dependency graph in Figure 13. Each word in the sentence is part of a pair, consisting of one head word and one dependent, each with a specified syntactic function. Dependents point towards their head word, with their syntactic relationships clearly stated. Every sentence is based on a designated root node, numbered 0, which links all the words in the sentence. Each word is sequentially numbered according to its ID (in this example, 1-7). The head word of the sentence is the semantic verb

[3]Universal Dependencies, version 2 (UD2)

Name	Description	Name	Description
acl	clausal modifier of noun (adjectival clause)	expl	expletive
		fixed	fixed multiword expression
advcl	adverbial clause modifier	flat	flat multiword expression
advmod	adverbial modifier	goeswith	goes with
amod	adjective modifier	iobj	indirect object
appos	appositional modifier	list	list
aux	auxiliary verb	mark	infinitive marker
case	case marking	nmod	nominal modifier
cc	conjunction	nsubj	nominal subject
ccomp	clausal complement	nummod	numerical modifier
clf	classifier	obj	object
compound	compound	obl	oblique nominal
conj	conjunction	orphan	orphan
cop	copula	parataxis	parataxis
csubj	clausal subject	punct	punctuation
dep	unspecified dependency	reparandum	reparation
det	determiner	root	root
discourse	discourse element	vocative	vocative
dislocated	dislocated element	xcomp	open clausal complement

Table 6: Syntactic relations in UD2.

(in our example the word *hit*), which has the root as its head word. The verb *hit*, numbered 4, has several direct dependents: the subject of the clause *wind* (NSUBJ), the direct object *cheek* (OBJ), and the punctuation mark "." (PUNCT). Furthermore, these words also have their own dependents. The third word, *wind*, has two dependents: word 1 *The* and word 2 *cold*, which function as a determiner (DET) and an adjectival modifier (AMOD), respectively. The sixth word, *my*, depends on the following word, (*cheek*).

Analyzed sentence 21.06.2020

TEXT ID	ID	FORM	NORM	LEMMA	CPOS	UPOS	UFEATS	HEAD	DEPREL
1.1	1	The	The	the	DT	DET	Definite=Def|PronType=Art	3	det
1.1	2	cold	cold	cold	JJ	ADJ	Degree=Pos	3	amod
1.1	3	wind	wind	wind	NN	NOUN	Number=Sing	4	nsubj
1.1	4	hitt	hit	hit	VBD	VERB	Mood=Ind|Tense=Past|VerbForm=Fin	0	root
1.1	5	my	my	my	PRP$	PRON	Number=Sing|Person=1|Poss=Yes|PronType=	6	nmod:poss
1.1	6	cheek	cheek	cheek	NN	NOUN	Number=Sing	4	obj
1.1	7	.	.	.	.	PUNCT	_	4	punct

Figure 12: Annotation in CoNLL-U format.

It is important to highlight two cases that deviate from traditional linguistic analysis. First, in Universal Dependencies version 2 (UD2), prepositions are considered dependents of the nouns they belong to within the same prepositional phrase. Second, auxiliary verbs are treated as dependents, with the main verb carrying the semantic information and serving as the head word. In this structure, the auxiliary verb functions as a dependent.

This syntactic analysis is conducted using MaltParser (Nivre et al., 2007), which is integrated into EFSELAB (Östling, 2016). MaltParser is specifically trained on Universal Dependencies version 2 (UD2) for Swedish (Östling, 2016).

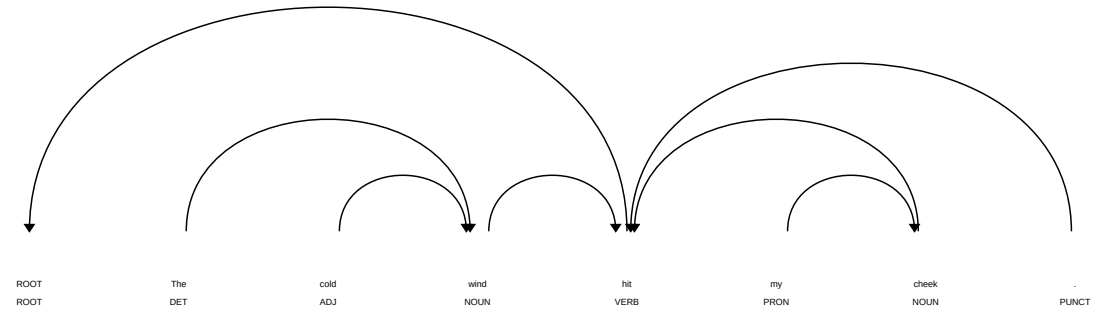


Figure 13: Example of syntactic analysis with dependency relations.

4.3 Correction of automatic analysis

Modularity has been a key factor in the development of this annotation tool. You can disable any of the modules, allowing you to exclude specific components of automatic annotation. We recommend doing this if you want to design your own text analysis or correct the proposed automatic analysis for further processing.

The components that implement automatic linguistic analysis are listed in the annotation tool: tokenization, normalization, and PoS tagging with syntactic analysis (parsing). These components are illustrated in Figure 6. As mentioned in Section 3.4, all components of the toolchain, except normalization, are activated by default. To deactivate a module, simply slide the button for that module to the left, as shown in Figure 6.

You can also use SWEGRAM's modules alongside your own modifications. Automatic normalization, including spellcheck and correction of separated compound words, can be performed manually before uploading your version for further analysis. Additionally, you can correct tokenization and sentence segmentation and then run the tagger and syntactic analyzer on the corrected, tokenized file. This allows you to upload your corrected annotations for automatic analysis by the subsequent components in the chain. By doing so, you can select the specific tools you need and achieve a more accurate linguistic analysis based on your corrections, ultimately contributing to better-annotated texts.

However, it is essential to deselect any tools you do not require while ensuring that your analysis is formatted correctly for the system. You *cannot* insert your own tagset; you must use the same tags as the system to ensure that the subsequent analysis components can be applied.

4.4 Storage of annotated texts

The annotated texts are stored in a database on the server for a limited time, with the system automatically deleting the texts after 7 days. Since no user registration is required, the system does not keep track of your annotated texts or any associated statistics. Therefore, we recommend saving your annotated texts and analyses on your local computer by downloading the annotated files and statistics, as described in Section 6.

5 Analysis

The second stage of SWEGRAM's functionality is the quantitative analysis of the annotated texts. SWEGRAM calculates statistics at various levels, which can be performed on a single

uploaded text, multiple texts within a single file, or across several uploaded files. Users can obtain general statistics on word, sentence, and paragraph length, as well as information on misspellings, PoS distribution, and readability metrics. Additionally, you can analyze individual texts using the visualization tool. Finally, the system allows you to filter your search to display statistics for selected texts among those you have uploaded. Learn more about statistics and filtering below.

5.1 Statistics and quantitative measurements

In the drop-down menu under *Statistics*, you can select from the headings *Linguistic features*, *Frequency*, and *Length* to receive a quantitative analysis of the selected text. The various functions are explained in more detail below.

5.1.1 Linguistic features

Linguistic features provide statistics generated at the text, paragraph, and sentence levels. Once you select the level of analysis, an overview of the statistics is displayed as a list that includes *General*, *Readability*, *Lexical*, *Morphological*, and *Syntactic features*. Please note that only tagged and parsed texts can yield lexical, morphological, or syntactic features.

The content of the text under investigation—whether it be a full text, a paragraph, or a sentence—is displayed in the *Content* section, as illustrated in Figure 14.

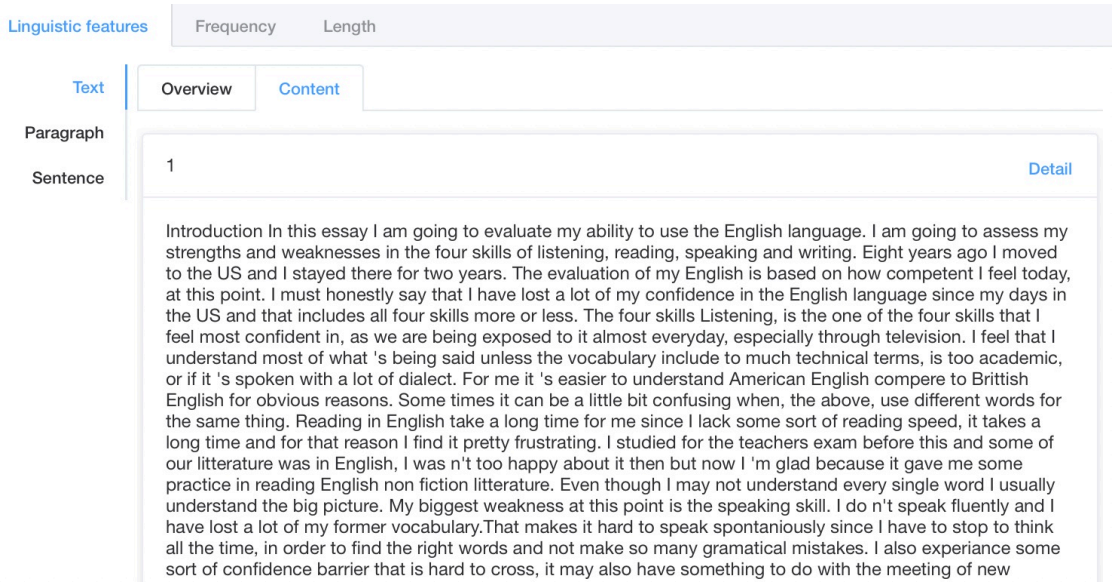
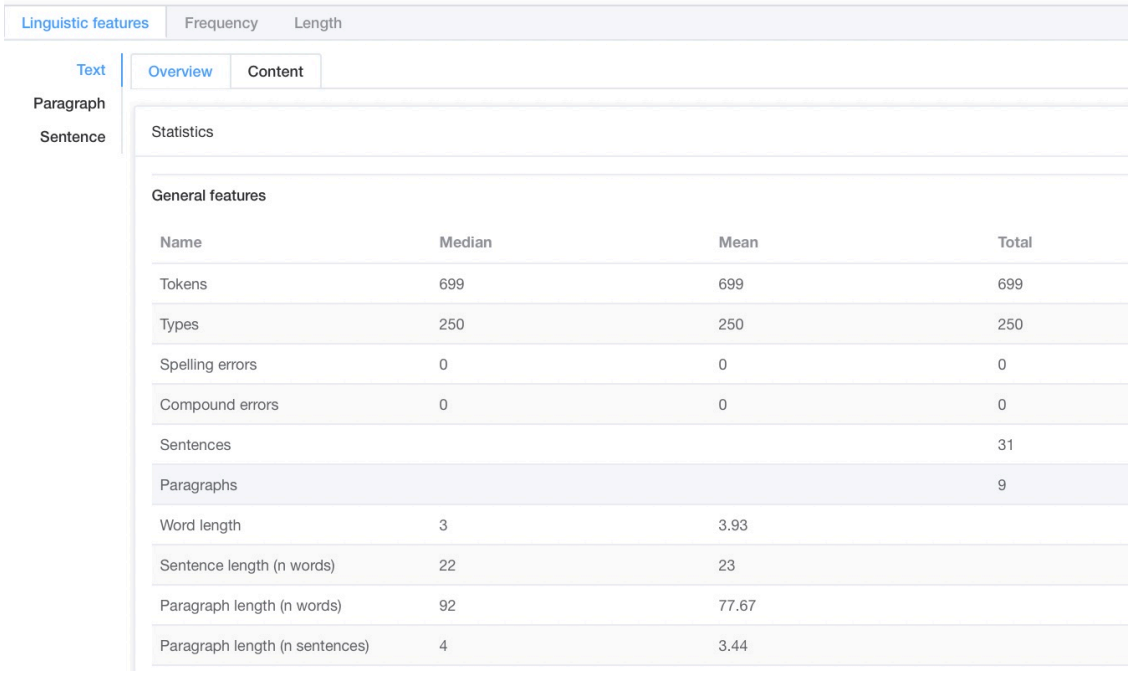


Figure 14: The visualization of the text in Content.

By clicking on *Detail* on the top right corner, the linguistic features are displayed in a popup window. The features for the specific content are computed immediately after the text has been uploaded to the system. The overview statistics represent the sum of the statistics from the specific linguistic levels. For example, when calculating the total number of tokens in the

selected texts, the system extracts the total number of tokens from each text and accumulates them as a feature value in the overview statistics.

Figure 15 illustrates how general statistics are presented, including information about the total number of tokens, types, sentences, paragraphs, misspellings, separated compound words, word length, sentence length, and paragraph length. Additionally, both the median and mean values are provided, measured in relation to the level at which the statistics are generated.



Name	Median	Mean	Total
Tokens	699	699	699
Types	250	250	250
Spelling errors	0	0	0
Compound errors	0	0	0
Sentences			31
Paragraphs			9
Word length	3	3.93	
Sentence length (n words)	22	23	
Paragraph length (n words)	92	77.67	
Paragraph length (n sentences)	4	3.44	

Figure 15: General features.

5.1.2 Readability features

SWEGRAM also calculates various metrics related to readability, word variation, and nominality. These metrics are categorized under *Readability features*, which include common readability measures for different languages. Since readability metrics are language-dependent, two distinct sets are provided: one for Swedish and another for English.

For Swedish, Lix, Ovix, type-token ratio (TTR), Simple nominal ratio, and Full nominal ratio are implemented. In contrast, the English metrics include two types of type-token ratio (Bilogarithm and Root), the Coleman-Liau Index, Flesch Reading Ease, Flesch Kincaid Grade level, Automated Readability Index, and SMOG, as illustrated in Figure 16. Each metric will be defined in the subsequent subsections.

5.1.2.1 Lix

Lix stands for läsbarhetsindex (English: readability index)and is calculated using a formula devised by Carl-Hugo Björnsson (1968). The Lix value of a text is determined based on the total number of sentences and the total number of words that are six letters or longer. The formula for calculating Lix is as follows:

Readability features

Name	Median	Mean	Total
Bilogarithm TTR	0.84	0.84	0.84
Root TTR	9.46	9.46	9.46
Coleman Liau Index	7.77	7.77	7.77
Flesch Reading Ease	61.14	61.14	61.14
Flesch Kincaid Grade level	9.91	9.91	9.91
Automated Readability Index	9	9	9
SMOG	11.14	11.14	11.14

Figure 16: Readability features for English.

$$\text{LIX} = \frac{\text{number of words}}{\text{number of sentences}} + \frac{\text{total long\_words} \times 100}{\text{total number of words}} \quad (1)$$

The Lix value serves as a simple metric to assess how the length of words and sentences may affect the readability of a text. The following list, derived from a textbook on writing style by (Melin & Lange, 2000), can be used to interpret the results:

Type of text	Lix	Words/sentences	Long words	Interpretation
Books for children and young people	27	12	15	Very easy
Fiction	33	15	18	Easy
Daily and weekly press	39	14	25	Intermediate
Factual literature	47	18	29	Difficult
Nonfiction	56	20	35	Very difficult

Table 7: Lix values, different genres (Melin & Lange, 2000).

5.1.2.2 Ovix

Ovix stands for *ordvariationsindex* (English: word variation index). This formula was originally developed by Tor G. Hultman as part of the research project Skrivsyntax, a comprehensive study of informational and student texts (Hultman & Westman, 1977). The metric measures the total number of lexemes (the minimal meaningful units of language) in relation to the total number of tokens in a text, designed to ensure that text length does not affect the result, unlike simpler metrics such as the type-token ratio (TTR) discussed above.

Various formulae for calculating Ovix exist in the literature. According to Hultman and Westman (1977, p. 264), Ovix is calculated as: $V = N \times (2 - N^k)$. where V stands for the number of unique words, N for the total number of words, K is a constant, and Ovix $= \frac{1}{k}$.

As per Hultman (1994, p. 62), Ovix can also be expressed as follows:

$$\text{OVIX} = \frac{1}{\frac{\ln\left(2-\frac{\ln(\text{total types})}{\ln(\text{total tokens})}\right)}{\ln(\text{total tokens})}} \tag{2}$$

Lix.se provides the calculation of Ovix according to a different formula, which is equivalent to the one above. This formula is also utilized in SWEGRAM:

$$\text{OVIX} = \frac{\ln(\text{total tokens})}{\ln\left(2-\frac{\ln(\text{total types})}{\ln(\text{total token})}\right)} \tag{3}$$

A higher Ovix value indicates greater word variation in relation to the length of the text (Hultman & Westman, 1977, p. 56). According to Hultman and Westman (1977, p. 60), Ovix should not be seen as a measure of the number of synonyms used in a text but rather as an indicator of the richness of viewpoints: "In all likelihood, our word variation metric measures the wealth of information, rather than synonym variation." However, there is no straightforward correlation between the Ovix value and text quality (Hultman, 1994, p. 63).

When comparing groups of texts, it is recommended to use a median value (Hultman & Westman, 1977, p. 56).

Many Swedish studies have utilized Ovix, including works by Nyström (2000), Östlund-Stjärnegårdh (2002), Kokkinakis and Magnusson (2011), and Nordenfors (2011). Nyström (2000, p. 192) explores the metric and concludes that "as a statistical value, Ovix probably has some kind of optimal level. Too high a value suggests a far too rapid subject progression, while too low a value may suggest a lack of substance."

5.1.2.3 Type-token ratio

The type-token ratio, abbreviated as TTR, is a simpler metric for assessing word variation and lexical diversity (Johansson, 2009). The Swedish terms *lex-/löp-kvot* are sometimes used interchangeably. TTR is calculated by comparing the number of unique words (types) to the total number of words (tokens) in a given text. The formula for calculating TTR is as follows:

$$\text{TTR} = \frac{\text{total number of unique tokens}}{\text{total number of tokens}} \tag{4}$$

A higher TTR indicates greater lexical diversity, meaning the text uses a wider variety of words, while a lower TTR suggests that the text repeats the same words more often, indicating less lexical diversity.

TTR can be influenced by text length; shorter texts tend to have a higher TTR due to fewer opportunities for word repetition, whereas longer texts typically exhibit a lower TTR. To mitigate the risk of results being affected by text length, bi-logarithm and square root TTR metrics are also implemented, as shown in the following equations:

$$f(\text{Bi-logarithm TTR}) = \frac{\text{In(Counts(type))}}{\text{In(Counts(token))}} \tag{5a}$$

$$f(\text{Square Root TTR}) = \frac{\text{Counts(type)}}{\sqrt{\text{Counts(token)}}} \tag{5b}$$

These transformations apply logarithmic and square root adjustments to the number of types and tokens, making the metrics more stable across texts of varying lengths.

5.1.2.4 Simple nominal ratio

The simple nominal ratio (Af Geijerstam, 2006, p. 108) provides an indication of a text's information density by highlighting the relationship between nouns (nn) and verbs (vb). The calculation of simple nominal ratio is show below:

$$\text{Simple nominal ratio} = \frac{\texttt{total number of nouns (NN)}}{\texttt{total number of verbs (VB)}} \tag{6}$$

A higher percentage of nouns compared to verbs indicates a more information-dense text, given that nouns are generally important to conveying facts.

5.1.2.5 Full nominal ratio

a more advanced measure of nominality. It emphasizes the relationship between nouns and noun-related parts of speech on one side, and verbs, adverbs, and pronouns on the other. Elements such as nouns (NN), prepositions (PP), and participles (PC)—both present and past—are associated with higher nominality and greater information density. In contrast, verbs (VB), adverbs (AB), and pronouns (PN) tend to dilute the informational content of a text. The formula for the full nominal ratio is presented below:

$$\text{Full nominal ratio} = \frac{\texttt{NN+PP+PC}}{\texttt{PN+AB+VB}} \tag{7}$$

Hultman and Westman (1977) demonstrate that written language has a significantly higher nominal ratio than spoken language. Consequently, the degree of nominality becomes a distinguishing feature, where a high proportion of nouns, prepositions, and participles characterizes written texts, while a lower nominal ratio indicates a style more akin to spoken language. Although the nominal ratio should not be viewed as a direct measurement of nominal phrases, it does provide insight into the degree of lexicality within those phrases. Research on writing development has shown that older writers are more likely to choose nouns as head words in their nominal phrases, whereas younger writers tend to favor pronouns (e.g., Scott (1988)).

5.1.2.6 Coleman Liau index

Coleman–Liau readability test (Coleman and Liau, 1975) was created to help the U.S. Office of Education adapt the readability of textbooks for the public school system. This index is based on the average number of characters per word and the average sentence length and it is calculated as shown below:

$$\text{CLI} = 5.88 \times \frac{\text{Count(letters)}}{\text{Count(words)}} - 29.6 \times \frac{\text{Count(sentences)}}{\text{Count(words)}} - 15.8 \tag{8}$$

The formula first computes the average number of characters per 100 words to estimate word complexity. Then, it calculates the average number of sentences per 100 words to assess sentence length and structural complexity.

The resulting number estimates the U.S. school grade level necessary to comprehend the text. The interpretation of the scores is presented in Table 8.

Score	School level	Comprehension
5	5th grade and below	Very easy to read
6	6th grade	Easy to read
7	7th grade	Quite easy to read
7-10	8th, 9th, and 10th grade	Conversational English
11-12	11th and 12th grade	Quite hard to read
13-16	College	Difficult to read
17+	Professional	Very hard to read

Table 8: Coleman Liau Index levels.

5.1.2.7 Flesch reading ease

The Flesch reading ease (Flesch, 1948) assesses sentence lengths and word complexity by calculating the average length of sentences in terms of the number of words, as well as the average number of syllables per word in the text.

$$\text{Flesch Reading Ease} = 206.835 - 1.015 \times \frac{\text{Count(words)}}{\text{Count(sentences)}} - 84.6 \times \frac{\text{Count(syllables)}}{\text{Count(words)}} \quad (9)$$

It provides a score between 0 and 100; the higher the reading score, the easier the text is to read. The interpretation of these scores is detailed in Table 9.

Score	Reading level
90-100	very easy to read, easily understood by an average 11-year-old student
80-90	easy to read
70-80	fairly easy to read
60-70	easily understood by 13- to 15-year-old students
50-60	fairly difficult to read
30-50	difficult to read, best understood by college graduates
0-30	very difficult to read, best understood by university graduates

Table 9: Flesch reading ease levels.

5.1.2.8 Flesch Kincaid grade level

Flesch-Kincaid grade level (Kincaid et al., 1975) is a variation of the Reading Ease formula with readjusted weights. The formula was derived three decades later (Kincaid et al., 1975) specifically to evaluate the readability of technical materials for military personnel. The measure assesses the approximate reading grade level of a text, based on average sentence length and word complexity.

$$\text{Flesch Kincaid Grade} = 0.39 \times \frac{\text{Count(words)}}{\text{Count(sentences)}} + 11.8 \times \frac{\text{Count(syllables)}}{\text{Count(words)}} - 15.59 \quad (10)$$

The scores correspond to US grade levels and ages, as detailed in Table 10.

Score	Reading level	School level	Age
0-3	Basic	Kindergarden	5-8
3-6	Basic	Elementary	8-11
6-9	Average	Middle school	11-14
9-12	Average	High school	14-17
12-15	Advanced	College	17-20
15-18	Advanced	Post-grad	20+

Table 10: Flesch Kincais Scores and US grade levels.

5.1.2.9 Automated readability index

The Automated Readability Index (ARI) (Kincaid and Delionbach, 1973) is derived from ratios that represent word difficulty in terms of the number of letters per word, and sentence difficulty as defined by the number of words per sentence. It takes into account the average number of characters per word and the average number of words per sentence. Characters include all letters, numbers, symbols, etc., excluding whitespace between characters.

$$\text{ARI} = 4.71 \times \frac{\text{Count(letters)}}{\text{Count(words)}} + 0.5 \times \frac{\text{Count(words)}}{\text{Count(sentences)}} - 21.43 \tag{11}$$

Table 11 summarizes the ARI scores along with their corresponding interpretations of reading level and grade.

ARI Score	Grade level	Reading level	Ages
1-5	Kindergarten	Extremely easy	5-6 y
1-5	1st grade	Extremely easy	6-7 y
6-7	2nd grade	Very easy	7-8 y
8-9	3rd grade	Very easy	8-9 y
10-11	4th grade	Easy	9-10 y
12-13	5th grade	Fairly easy	10-11 y
14-15	6th grade	Fairly aasy	11-12 y
16-17	7th grade	Average	12-13 y
18-19	8th grade	Average	13-14 y
20-21	9th grade	Slightly difficult	14-15 y
22-23	10th grade	Somewhat difficult	15-16 y
24-25	11th grade	Fairly difficult	16-17 y
26-27	12th grade	Difficult	17-18 y
28+	College	Very difficult	18-22 y

Table 11: ARI Scores and US grade levels.

5.1.2.10 SMOG

SMOG, or the Simple Measure of Gobbledygook (McLaughlin, 1969) is a readability measure that estimates the years of education required to understand a piece of writing. The SMOG formula evaluates a text based on the complexity of its sentences and words by considering the number of polysyllabic words (i.e., words with three or more syllables) and the total number of sentences. The SMOG formula is defined as follows:

$$\text{SMOG} = 1.0430 \times \left(\sqrt{\frac{\text{polysyllables} \times 30}{\text{Count(sentences)}}} + 3.1291 \right) \tag{12}$$

The interpretation of the SMOG scores is detailed in Table 12.

SMOG Score	Approx. Grade level (+1.5 grades)
1-6	5
7-12	6
13-20	7
21-30	8
31-42	9
43-56	10
57-72	11
73-90	12
91-110	13
111-132	14
133-56	15
157-182	16
183-210	17
211-240	18

Table 12: SMOG Scores and US grade levels.

5.1.3 Lexical features

To measure the lexical diversity and proficiency, we consult a Swedish KELLY-list[4] (Volodina & Kokkinakis, 2012) based on the project, KEywords for Language Learning for Young and adults alike (KELLY). The KELLY-list is a vocabulary sample that represents vocabulary usage, with each selected word or phrase classified into one of the six levels of the Common European Framework of Reference for Languages (Pilán & Volodina, 2016) (CEFR), listed in ascending order of difficulty: A1, A2, B1, B2, C1 and C2.

The Swedish KELLY-list contains 8,409 items generated from a corpus of web texts, illustrated in Table 13 with two example entries for each CEFR level. Each word entry includes its lemma, part of speech, relative word frequency measured in *word per million* (WPM), followed by the English translation. The distribution of lemmas in the text, categorized by their difficulty levels (A1, A2, B1, B2, C1, C2), is presented first, along with their median, mean, and total values. Following this, we identify *Difficult Words*, which refers to any tokens with a CEFR level equal to or above B1. Next, we provide the count of *Difficult NOUN&VERB*, referring specifically to nouns or verbs that have a CEFR level of B1 or higher. Finally, SWEGRAM provides the number of tokens in the text that are not included in the KELLY list.

5.1.4 Morphological features

The frequency of a part of speech or a morphological feature in relation to other parts of speech or morphological features is measured using relative frequencies through the incidence score (INCSC) (Graesser et al., 2004). The computation of INCSC is defined as follows:

[4]https://spraakbanken.gu.se/en/projects/kelly. Available on Sep 14, 2020.

CEFR	Lemma	PoS	WPM	Translation
A1	all	pron	2975,47	all
	för	adverb	421,08	for
A2	påstående	noun-ett	63,24	statement
	för	conj	44,36	therefor
B1	avgå	verb	23,46	resign
	stabilitet	noun-en	23,32	stability
B2	ödmjuk	adjective	11,86	humble
	i natt	adverb	11,85	tonight
C1	foster	noun-ett	7,06	fetus
	enastående	adjective	7,05	outstanding
C2	allergisk	adjective	3,40	allergic
	eskalera	verb	3,40	escalate

Table 13: Example entries from the Swedish KELLY-list.

$$\text{INCSC} = \frac{1000}{N_t} \times N_c \tag{13}$$

In this formula, N_c represents the number of tokens that belong to a specific category of interest, while N_t is the total number of tokens used for comparison. In most cases, N_c is a subset of N_t.

To illustrate the computation, consider the example where we want to determine the proportion of verbs in the present form among all verbs in a given text. We calculate INCSC by taking the ratio of the number of present tense verbs (N_c) to the total number of verbs (N_t), multiplied by 1000. If N_t is 0, the value of INCSC defaults to 1000.

SWEGRAM encompasses 30 morphological features divided into six subgroups based on their part of speech in the INCSC computation. These linguistic features have been shown to correlate with language development in both first and second language research (Pilán, 2018). We describe these features in terms of UD tags as follows.

5.1.4.1 VERBFORM

We measure different types of verb forms (N_c) in terms of modality, tense, and aspect, in relation to the total number of verbs (`VERB`), including auxiliary verbs (`AUX`). N_t refers to the tokens classified as either `VERB` or `AUX`: N_t = `VERB`+`AUX`. N_c calculated as follows:

- Modal verb: N_c = `AUX`
- Present participle: N_c = `Tense=Pres|VerbForm=Part`
- Past participle: N_c = `Tense=Past|VerbForm=Part`
- Verb in present form: N_c = `Tense=Pres|VerbForm=Fin`
- Verb in past form: N_c = `Tense=Past|VerbForm=Fin`
- Supine verbs: N_c = `VerbForm=Sup`
- S-verb: An S-verb refers to verb that ends with the character *s* due to passive verb construction, reciprocal actions, or deponent verbs in Swedish.

5.1.4.2 PoS - PoS

In this group, we examine the distribution of one part of speech in relation to another, focusing on three pairs: noun to verb (NOUN - VERB), pronoun to noun (PRON - NOUN) and pronoun to preposition (PRON - ADP).

5.1.4.3 SUBPoS – ALL

This feature encompasses subcategories of three parts of speech (SUBPoS) in relation to all tokens in the text (ALL), specifically: passive verb forms (S-VERB), third-person singular personal pronouns (PRON 3SG), and neuter nouns (NOUN NEU).

5.1.4.4 PoS – ALL

SWEGRAM calculates the proportion of each of the following parts of speech in relation to all tokens: adjective (ADJ), adverb (ADV), noun (NOUN), particle (PART), puncutation (PUNCT), subordinating conjunction (SCONJ), and verb (VERB).

5.1.4.5 PoS – Multiple PoS

SWEGRAM calculates the number of adjectives (ADJ), adverbs (ADV), nouns (NOUN), and verbs (VERB) in relation to the total of these four parts of speech.

5.1.4.6 Multiple PoS – Multiple PoS

SWEGRAM calculates the ratio of one group of parts of speech in relation to another. For instance, the relationship between lexical and function words can be assessed through specific groupings of parts of speech. Lexical parts of speech in UD include adjectives (ADJ), adverbs (ADV), interjections (INTJ), nouns (NOUN), proper nouns (PROPN), and verbs (VERB). In contrast, functional parts of speech consist of prepositions (ADP), auxiliaries (AUX), coordinating conjunctions (CCONJ), determiners (DET), numbers (NUM), particles (PART), pronouns (PRON), subordinating conjunctions (SCONJ), punctuations (PUNCT), symbols (SYM), and other (X). The distribution of these groups of parts of speech, given N_c and N_t, is detailed in Table 14.

Feature	**N_c**	**N_t**
Functional token	Functional PoS	ALL PoS
Lexical token	Lexical PoS	ALL PoS
Conjunction och subjunction	PoS = CCONJ+SCONJ	ALL PoS
Interrogative and relative	PoS = Int+Rel	ALL PoS
Lexical - functional token	Lexical PoS	Functional PoS
Nominal - verbal	PoS = NOUN+ADP+PC	PoS = PRON+ADV+VERB

Table 14: The specification of categories for N_c och N_t is based on specific morphological features. **ALL** refers to all parts of speech categories.

5.1.5 Syntactic features

SWEGRAM analyzes the syntactic structure and complexity of the text based on dependency analysis, as described in more detail in Section 4.2.5. We illustrate the extraction of syntactic features through graphical representations of a parsed sentence in Figure 17.

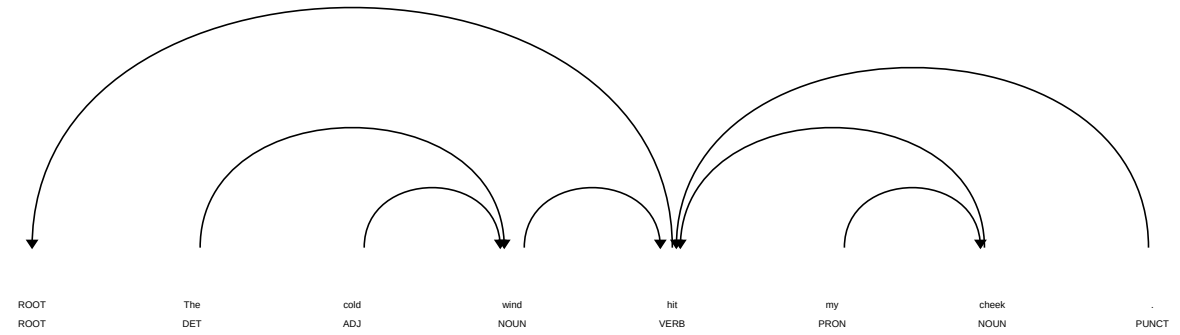


Figure 17: Example of syntactic analysis with dependency relations.

The syntactic structure of the sentence is represented through pairwise relations between tokens, known as head-dependent relations. Each relation is labeled to denote the syntactic relationship between the two tokens. As illustrated in Figure 17, each arc links a pair of words, with one acting as the head and the other as its dependent. The direction of the arc goes from the dependent to the head, with the arrow pointing to the head word from its dependent.

Every word in the sentence has exactly one head. A dummy node, referred to as `ROOT`, is added as the head of the sentence to ensure that all tokens are connected. The main verb, which carries the semantic meaning of the sentence, is the head of the sentence and has the `ROOT` node as its head. Arcs on the left side of their head are termed left arcs, while those on the right side are called right arcs.

We define the dependency arc length of a dependent with respect to a target head as the number of arcs through which the current dependent node reaches the target head node. In the example sentence above, the arc (the, cold, wind, my) is a right arc, while (hit, cheek, and .) is a right arc. Since `ROOT` is always positioned furthest to the left in the sentence, the arc that connects to `ROOT` is always considered a right arc.

5.1.5.1 Dependency arcs

To calculate syntactic complexity, five syntactic features are extracted based on statistics related to dependency arcs: dependence length, longest dependency length, dependency arcs longer than 5, ratio of right dependency arcs, and ratio of left dependency arcs. We illustrate these five features for the words in the same sentence *The cold wind hit my cheek.* as previously, shown in Figure 17.

- Dependence length: The sum of the count of dependency arcs from each token in the sentence to the `ROOT`.

  Example: (16) Dependency length is 16 since the number of arcs from the first token to the last (1-7) is 3, 3, 2, 1, 3, 2, and 2, respectively, summing up to 16.

- Longest dependency length: The longest dependency length to the `ROOT`.

  Example: (3) There are three tokens whose dependency length is 3, which is the longest dependency length in the whole sentence.

  - The→wind→ hit→ `ROOT`;

  - cold→ wind→ hit→ `ROOT`;
  - my→ cheek→ hit→ `ROOT`.

- Dependency arcs longer than 5: The count of tokens whose dependency length is longer than 5.

  Exempel: (0) There is no token whose dependency length is longer than 5.

- Ratio of right dependency arcs: The proportion between the count of right arcs and total arcs in the sentence.

  Example: (4/7) 4 out of 7 arcs in the example sentence is right arcs.

- Ratio of left dependency arcs: The proportion between the count of left arcs and total arcs in the sentence.

  Example: (3/7) 3 out of 7 arcs in the example sentence is left arcs.

5.1.5.2 Syntactic relations

Syntactic complexity is also measured in terms of six syntactic functions counted as N_c incidence scores. The syntactic relations examined include modifier variation (both pre-modifiers and post-modifiers), subordinate clauses, relative clauses, and prepositional complements.

- A modifier refers to any dependent whose dependency relation to its head is one of the following: adjectival modifier (`amod`), nominal modifier (`nmod`), appositional modifier (`appos`), numeric modifier (`nummod`), adverbial modifier (`advmod`), or discourse element (`discourse`). A modifier that occurs before its head is classified as a `pre-modifier` while a `post-modifier` indicates that the modifier appears after its head word.

  `Modifier Variation` refers to the total number of modifiers measured in relation to all syntactic functions.

- Subordinate refers to the dependency relation (`DEPREL`) annotated with one of the following labels: clausal subject (`csubj`), clausal complement (`ccomp`), open clausal complement (`xcomp`), adverbial clause modifier (`advcl`), clausal modifier of noun (`acl`), or relative adnominal clauses (`acl:relcl`). This applies to the relationship between the current token and any of the direct dependents of that token.

- Relative clause refers to any dependent whose relation to its head includes `PronType=Rel` indicating that the Pronominal Type is a relative pronoun, determiner, numeral, or adverb. For the feature of `Rel Clause INCSC`, N_c represents the number of tokens within the relative clauses.

- Prepositional complement refers to the dependent whose dependency relation with its head is annotated with `case`. For the feature of `PREP Comp INCSC`, N_c represents the number of prepositional complements and other dependents that are subordinated under the heads of the prepositional complements.

5.1.6 Frequencies

Under *Frequencies*, SWEGRAM displays the tokens—words and punctuation marks—along with their PoS tags of your choice (universal dependency tagset, as well as the language-specific tagsets: SUC tagset for Swedish and Penn Treebank Tagset for English). The display includes

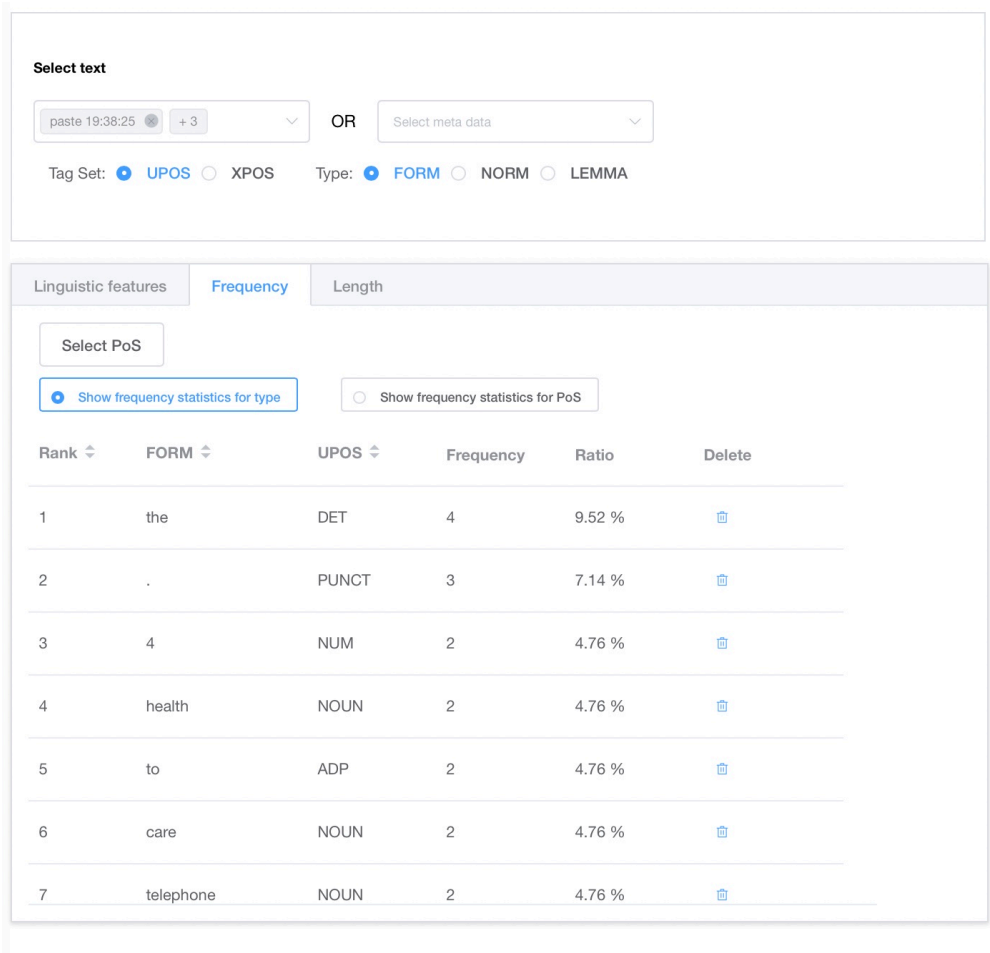


Figure 18: Frequency analysis.

the total number of occurrences and the ratio (percentage) in relation to the selected text(s). Figure 18 shows an example of what the beginning of a frequency list might look like.

The default setting for frequency analysis is the normalized word (NORM). However, if this is missing in the annotation due to the absence of normalization during the annotation process, the form words (FORM) will serve as the basis for the frequency analysis. The lemma can also be selected as an alternative in the upper row of the frequency window.

The frequency list can be ranked either in alphabetical order of the tokens or by part-of-speech categories.

In addition, frequency can be displayed separately for the PoS types, as shown in Figure 19. In this example, we find 35.71% nouns (`NOUN`) and 11.9% determiners (`DET`) and punctuation marks (`PUNCT`) among all PoS categories. If you wish to narrow your search, you can deselect unwanted PoS categories by unchecking the corresponding boxes under the Toggle column. If you only want to study the frequency of specific PoS types, it may be helpful to deselect all initially and then add your desired selections.

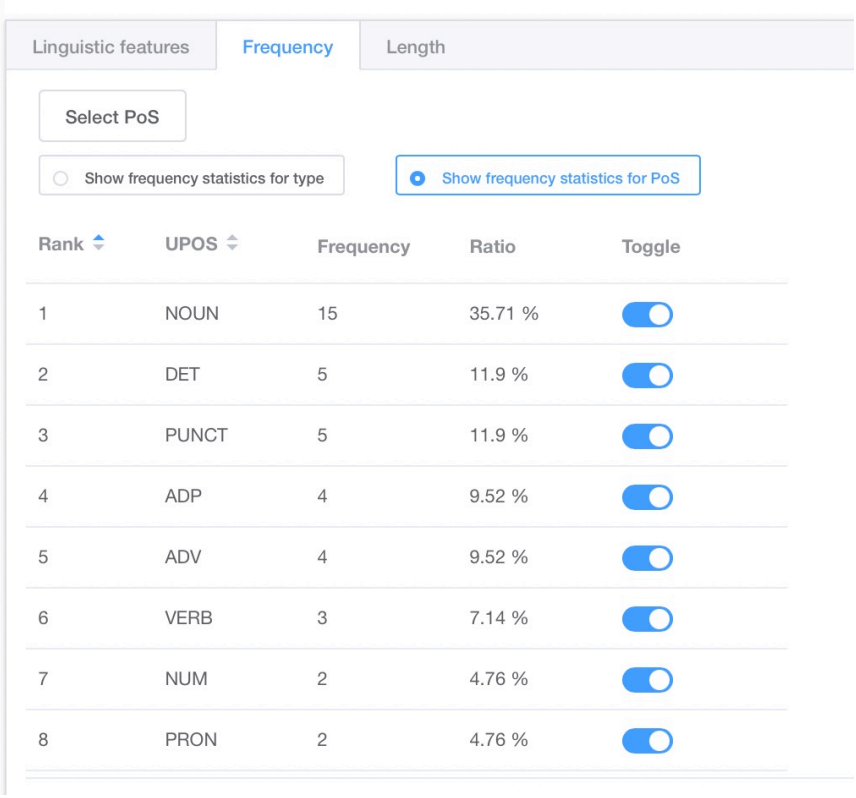
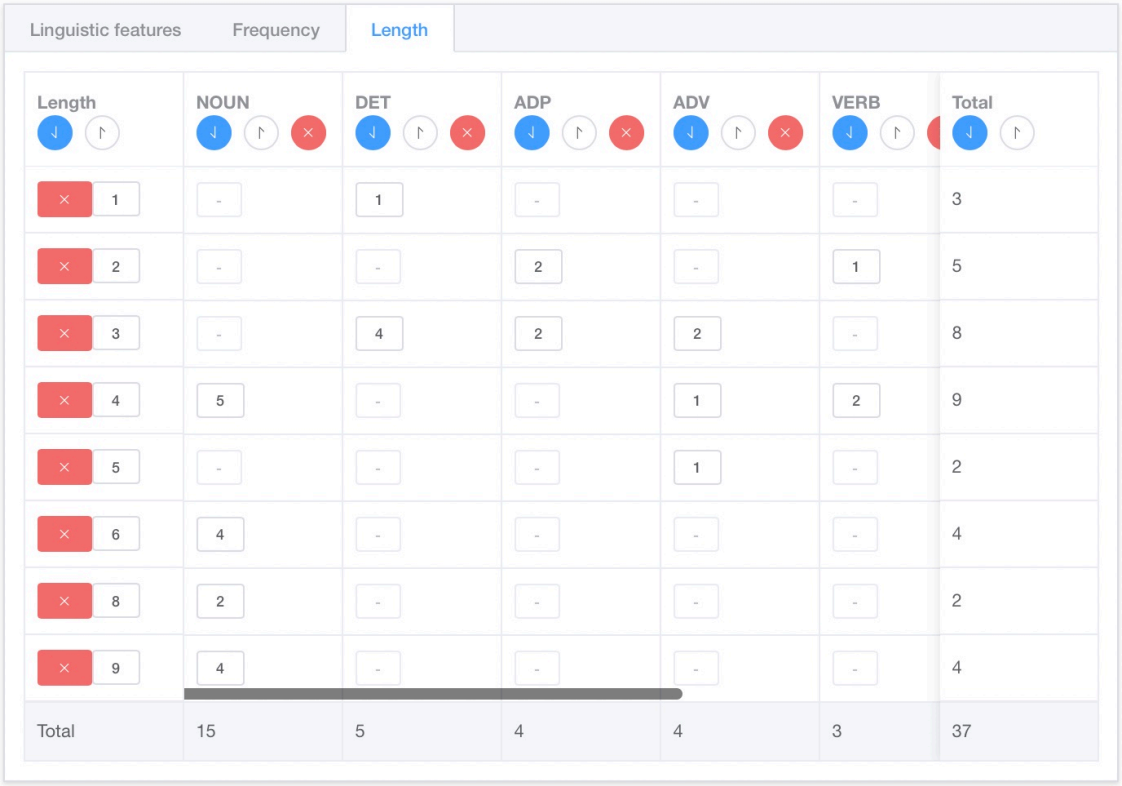


Figure 19: Frequency analysis of PoS categories.

5.1.7 Length

Sometimes, it is beneficial to study the length of various types of words (punctuation marks are excluded). When selecting the *Length* feature set, the number of tokens belonging to specific parts of speech (listed by columns) with character lengths ranging from 1 up to the length of the longest tokens in the text is visualized in the rows. Thus, the rows represent word lengths while the columns represent parts of speech. The cells in the table display the absolute frequency corresponding to each part of speech and length. These cells can be clicked to reveal the words behind the counts and how many times they occur in the selected texts. The list can also be sorted by increasing or decreasing frequency. Additionally, it is possible to remove any part of speech or word length from the table.

Figure 20 illustrates the statistics of the length features. In this particular example, there are five nouns with a character length of four, four nouns that are six characters long, two nouns that are eight characters long, and four nouns that are nine characters long. In the last row, we can see that there are a total of fifteen nouns. The last column lists the total number of words of each specific length.



Length	NOUN	DET	ADP	ADV	VERB	Total
1	-	1	-	-	-	3
2	-	-	2	-	1	5
3	-	4	2	2	-	8
4	5	-	-	1	2	9
5	-	-	-	1	-	2
6	4	-	-	-	-	4
8	2	-	-	-	-	2
9	4	-	-	-	-	4
Total	15	5	4	4	3	37

Figure 20: The analysis of character length sorted by PoS.

5.2 Visualising texts

Once the text has undergone the chosen annotation chain, it becomes available for visualization. By clicking the *Visualise* button in the upper row, you can access a visual representation of SWEGRAM's analysis of the selected text(s). All annotated texts are displayed and can be selected for visualization in the main panel.

There are two methods for selecting (or deselecting) annotated texts. These methods operate synchronously, meaning that a selection made in one method affects the other. The first method involves the text list, where all uploaded texts are listed. This approach is typically used for texts without metadata.

The second method utilizes the metadata associated with the texts. The text list includes both texts with metadata and those without it. Metadata is an optional feature for the text. This second method is only available for texts that contain metadata.

If you do not wish to include a text for analysis, you can either deselect or delete it. Deselecting the text will remove it from the analysis, while deletion permanently removes the text from the cache, making it inaccessible. Please note that deletion cannot be undone.

By selecting a specific text from the menu, the panel will display three components related to the chosen text:

- Text name

- Metadata (if available)
- The extracted sentences

Each sentence in the text is listed row by row. By clicking on a sentence, the parsed syntax tree will be displayed, as illustrated in Figure 21. To enlarge the syntax tree, simply click on it. The dependency tree is automatically generated in SVG format, reflecting the attributes of `FORM`, `UPOS`, `DEPREL`, and `HEAD`.

To view the morphological and syntactic analysis of specific tokens, click on the desired token, and a popup menu will appear, showing all features of that token. An underline will be displayed if the attribute value of a feature is not applicable.

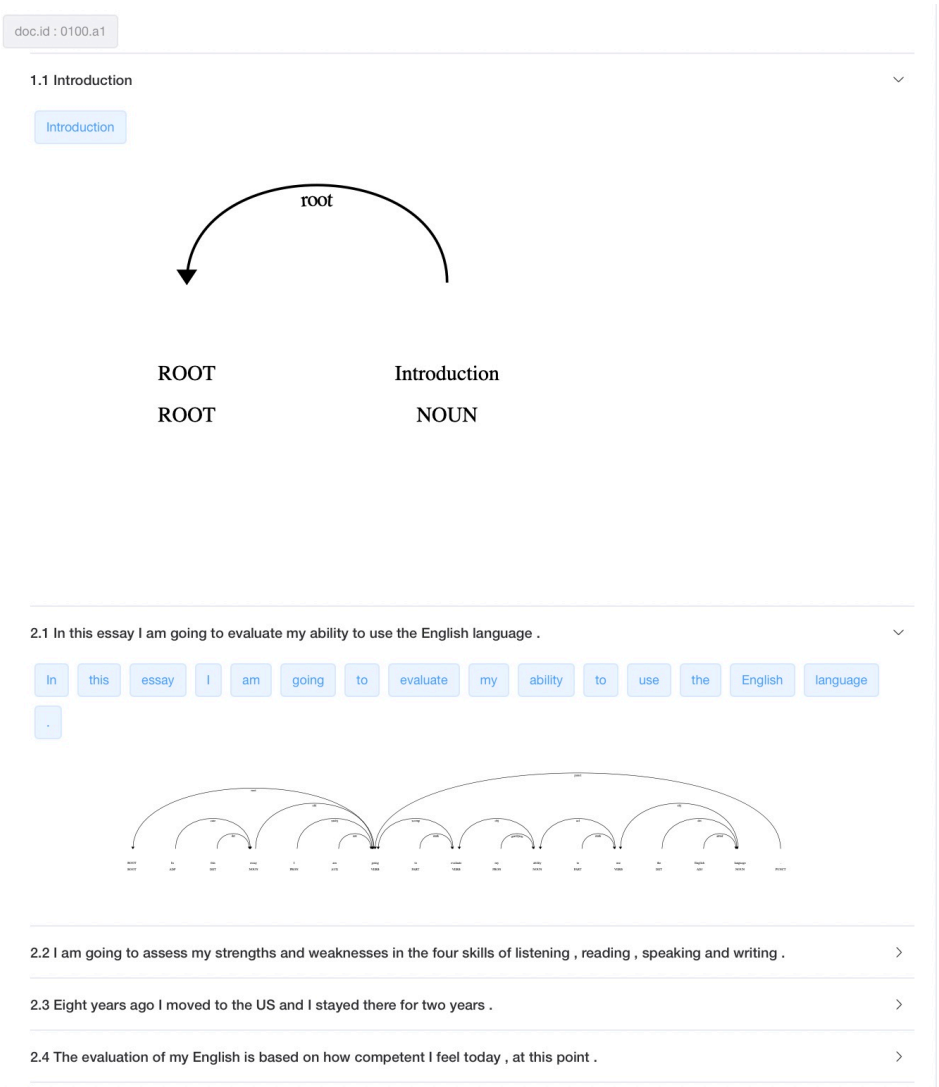


Figure 21: Visualization of the annotated text sentence by sentence.

Before visualization, you can use the menu to select individual tokens that you wish to search for in the text by typing the desired token (either a form word, a normalized word, or a lemma) into the search field. SWEGRAM highlights the chosen token in red, as illustrated in Figure 22.

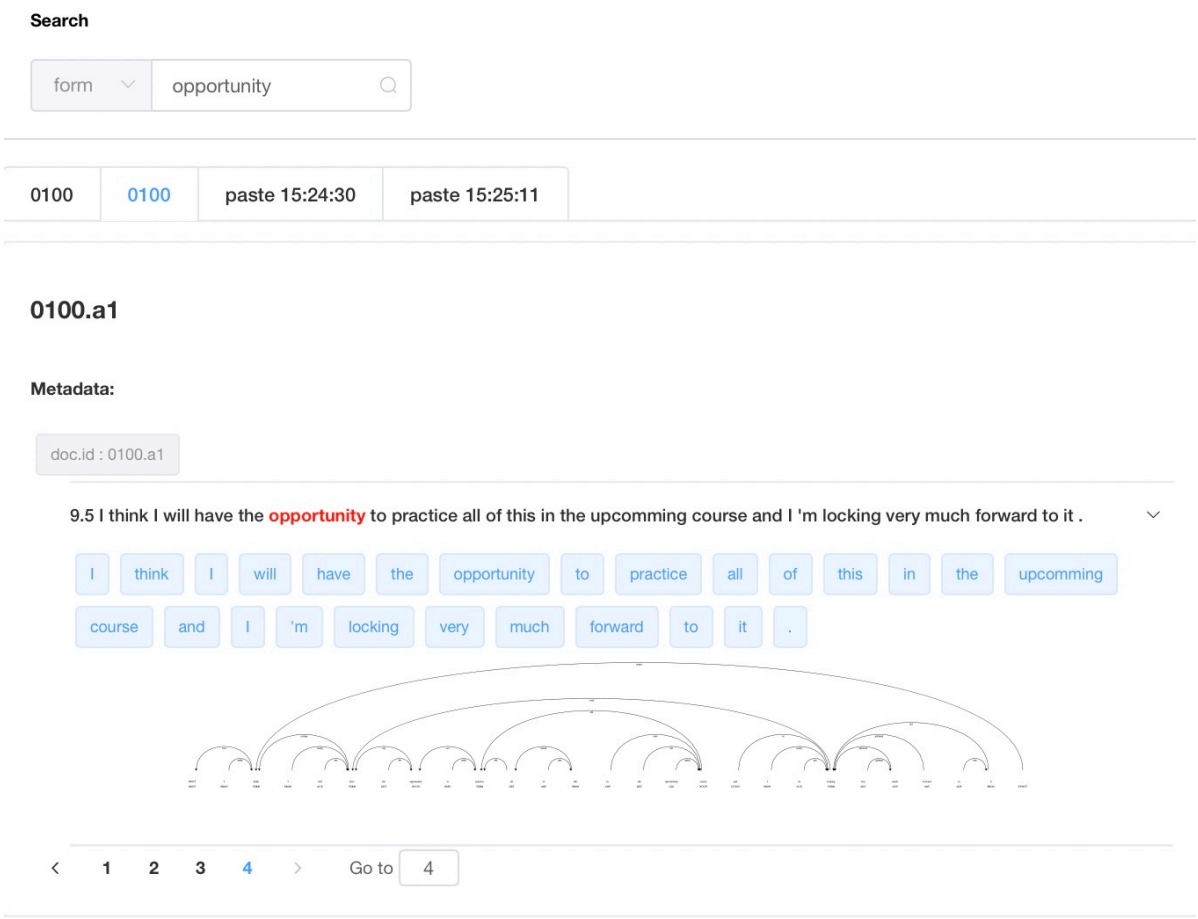


Figure 22: Visualization of the searched word *opportunity*.

You can also choose to search for specific parts of speech (`UPOS`), dependency relations (`HEAD`), or functions (`DEPREL`) in the same menu, located beneath the search field. The selected category will be highlighted in red for each associated token in the text.

The search input supports auto-completion, displaying all possible candidates based on the current search query. Additionally, the statistics and visualization pages are dynamically reloaded whenever there is a change in text selection.

To reduce the amount of data loaded simultaneously, not all sentences in the selected text are extracted in a single request. The default maximum number of items per page is set to 10. To navigate to a specific page, use the navigation bar at the bottom of the page, represented by $<$ and $>$.

6 Export

If you wish to save your analyzed text(s) and/or search results, you can download them in three formats: text (.txt), CSV(.csv), or Excel(.xlsx). You can then open these files in Microsoft Excel to create diagrams, graphs, or conduct further analysis of your choice. You have the option to download the annotated texts, the statistics, or both.

6.1 Export annotated texts

The selected texts can be merged into a single file for export. The exported file contains basic information, including the timestamp and working language. Before each text begins, the original file name, annotation settings, and any potential metadata line are inserted.

Figure 23 illustrates the layout of the *Export* menu when downloading the annotated text(s). Once you have selected the texts you want to download and chosen the output format, you can export the annotated text by clicking the download button.

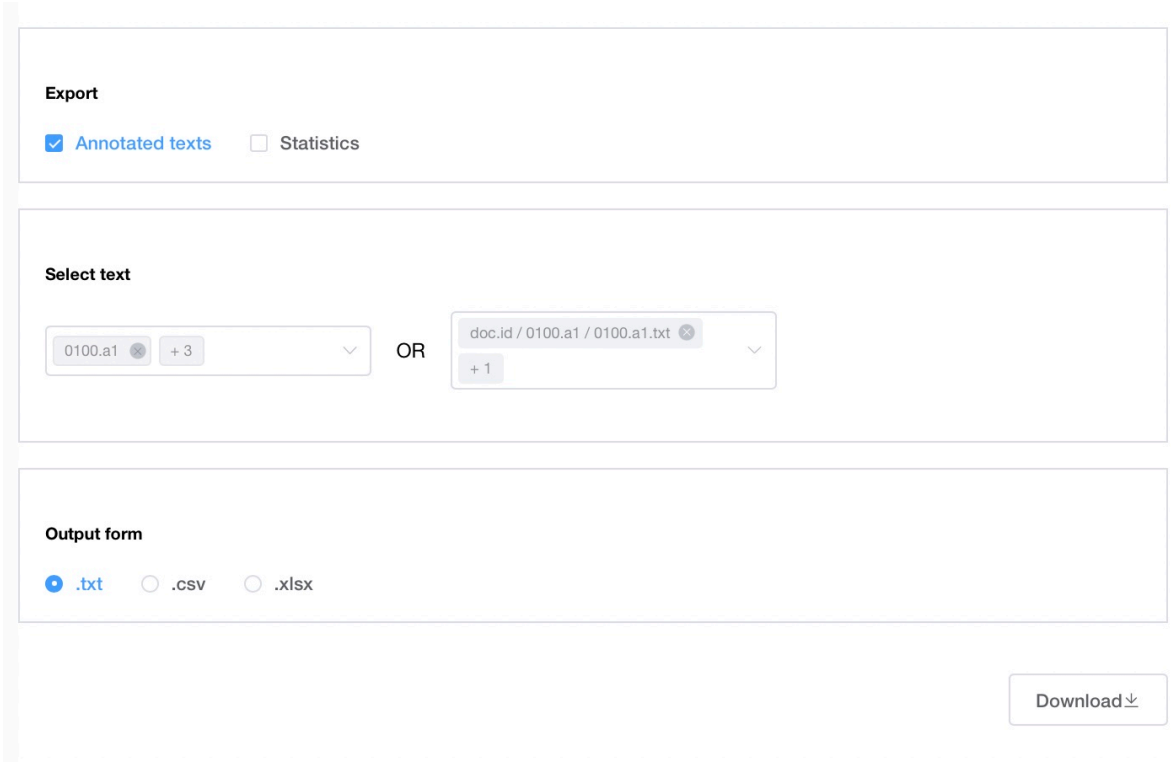


Figure 23: Export annotated text.

6.2 Export statistics

A collection of linguistic features can be exported as statistics based on five textual levels: word, sentence, paragraph, text, and multiple texts. The linguistic features are classified into five categories: general features, readability features, lexical features, morphological features, and syntactic features. The morphological and syntactic features are further divided into subcategories, which are presented as separate lines in the tables.

In some cases, when the selected texts are not tagged and parsed, only general and readability features are available; statistics for the other features cannot be calculated.

Figure 24 illustrates the download options for the statistics of annotated texts. Once you have checked the box for statistics in the upper row and selected the text(s) you wish to include, you can choose whether you want to download an overview or all analyses, including the fully annotated text. Additionally, you can specify whether you want the analysis at the text, paragraph, and/or sentence levels and select the features you are interested in. By default, the system includes all features, but you can remove any that you do not wish to include. Finally, you need to choose the format before clicking the Download button to export the data.

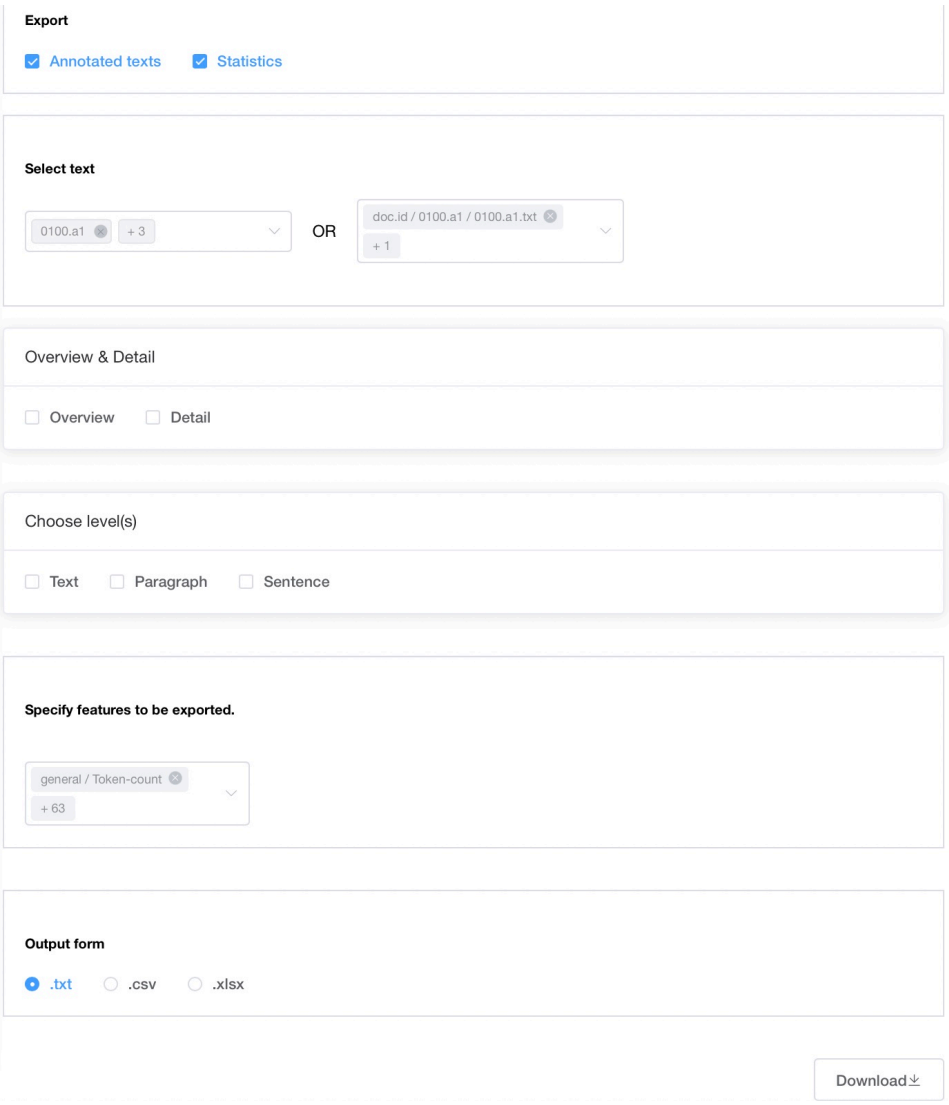


Figure 24: Export statistics.

7 Create Your Own Corpus

You can use SWEGRAM to upload one or more texts and create a linguistically annotated file, forming a corpus that you can then analyze using the analysis tool. You can build your corpus with any number of files, each containing one or more texts. These texts will be automatically annotated; they will be segmented into sentences and words, with words being PoS tagged and morphologically analyzed. Additionally, the base form will be provided, misspellings corrected, compound words separated, and each sentence syntactically analyzed. There is no limit to the number of files you can include in your corpus.

To take full advantage of the filtering capabilities during statistical analysis, you will also need to encode your text with metadata (see Section 3.2). Metadata can encompass any relevant information regarding your text; for example, details about the authors (identity, gender, age, domicile) and the texts themselves (text ID, year, genre, place of publication). You have the flexibility to choose which metadata and how much of it you wish to incorporate into your corpus. For illustration, let's assume you want to build a corpus containing 30 texts written by students. The class includes both girls and boys, and the texts may be categorized as either narrative or argumentative genres. Since these texts were gathered from an examination, they have also been graded: F, E, D, C, B, or A. Therefore, you will have metadata for your 30 texts, including the text ID (`TEXTID:1-N`), author's gender (`AUTHOR:M/F`), genre of each text (`GENRE:NARR/ARG`), and the corresponding grade (`GRADE:A-F`). If you lack a specific piece of metadata for any text, you can indicate this by specifying the feature and marking its value with an underscore "_". For example, if you do not know the gender of the writer, you can represent it as (`AUTHOR:_`), or you can remove the entire feature (`AUTHOR:F/M/_`).

The next step is to create one or more text files containing both the texts and their associated metadata. In this example, we will use Microsoft Word for the texts, although plain text files (.txt) can also be utilized. Since the corpus consists of only 30 texts, it is appropriate to compile them into a single document. While there is no technical limitation on the number of texts that can be included in one file (as long as the file does not exceed 10 MB), for larger corpora, it may be beneficial to divide the texts across multiple files for easier manual handling. An example of a tiny compiled corpus consisting of three texts written by three students is illustrated in Figure 25.

Once your texts are compiled and coded in the document, you will have created a corpus—a collection of texts ready for annotation. You can now annotate your corpus and utilize SWEGRAM's analysis tool to generate statistics and visualizations based on the data in your texts. You can also expand your corpus by uploading additional texts and document files for simultaneous analysis by SWEGRAM.

Please note that if your file is incorrectly formatted, the tool may be unable to perform the analysis, or the results may be inaccurate. In such cases, you will receive an error message and will need to correct the error in the file.

8 About the Tool

SWEGRAM is accessible in two formats: an online version available at the SWEGRAM portal (https://swegram.ling.su.se) and local installation options on your own PC via the GitHub project swegram-v2 (https://github.com/bmegyesi/swegram-v2). Depending on your intended use, we provide both a command-line interface and a web-based version, both powered by Docker containers, mirroring the functionality of the online version. Both approaches have been thoroughly tested in Linux and macOS environments. For detailed dependencies and installation instructions, please refer to the GitHub project homepage. The components of SWEGRAM are written in Python, while the interface is developed using HTML, CSS, and

<TEXT-ID:1;AUTHOR:M;GENRE:NARR;GRADE:E>
I like my family. My mom is old, my father is older and my sister is young. We live in a big house. I like my school. The food is not great.

<TEXT-ID:2;AUTHOR:F;GENRE:ARG;GRADE:C>
I like my family because they are important to me. My mom is older and gives good advice, while my dad is even older and helps us with important things. My younger sister is fun and brings energy to our home. We live in a big house where we can all be together. I enjoy my school too, even if the food isn't the best. What matters most is that I am learning.

<TEXT-ID:3;AUTHOR_;GENRE:NARR;GRADE:A>
In a lively neighborhood, I live in a big house filled with love. My family is my foundation. My mother, with her wise words, guides me, while my father, even older, shares invaluable life lessons. Then there's my younger sister, whose laughter fills our home with joy and reminds us to embrace curiosity.

School is also an important part of my life. I enjoy learning, even if the cafeteria food isn't the best. What truly matters is the knowledge I gain and the friendships I build along the way. Together, my family and education shape who I am, providing the support I need to grow and succeed. I am grateful for this journey filled with love and learning.

Figure 25: Example of metadata in a corpus.

JavaScript, with assistance from the Django framework. This tool is open-source and available to all interested parties.

8.1 License

SWEGRAM is licensed under *Creative Commons CC BY-SA*, which allows you to freely use, share, copy, edit, modify, and distribute the tool in various forms and formats for any purpose, including commercial use. Proper attribution to SWEGRAM is required for any use, dissemination, or processing of the tool. When utilizing SWEGRAM, please reference:

- Megyesi, B. and Ruan, R. (2024) *SWEGRAM: Guidelines to Annotation and Analysis of English and Swedish Texts.* Department of Linguistics, Stockholm University, Sweden.

Other publications of relevance:

- Megyesi, B., Palmér, A. & Näsman J. (2019) *SWEGRAM: Annotation and Analysis of Swedish Texts.* Department of Linguistics and Philology and Department of Scandinavian Languages, Uppsala University, Sweden.
- Näsman, J., Megyesi, B., & Palmér, A. (2017) SWEGRAM: A Web-based Tool for Automatic Annotation and Analysis of Swedish Texts. In *Proceedings of the 21st Nordic Conference on Computational Linguistics,* Nodalida 2017.

For the Creative Commons CC BY-SA license, please refer to the licensing terms at the following hyperlink: *https://creativecommons.org/licenses/by-sa/2.5/.* You are required to indicate whether any changes have been applied to the original tool and specify what those changes are, as part of good research practice. If you process, alter, or expand upon SWEGRAM, you must distribute your contributions under the same license as the original.

If you have any comments or encounter any issues with the tool, please reach out to us using the contact information provided on the SWEGRAM website.

8.2 News in version 2.0

SWEGRAM 2.0 has been updated to process English texts. We have also extended the linguistic features for both English and Swedish by incorporating standard analyses of second language acquisition. Additionally, the visualization of the morphosyntactic analysis has been improved to display the syntax tree for each sentence.

Furthermore, SWEGRAM now features a new backend, enabling faster and more efficient text processing. In this version, the analysis of relatively large texts—previously time-consuming—can now be displayed. This improvement is achieved by storing the annotated texts in a database, alleviating the burden of data loading. When visualizing texts, only the limited amount of data relevant to the selected text and pagination for sentences is extracted.

The current SWEGRAM online production is containerized and consists of three main components: the frontend user interfaces, built from a VUE project with an Nginx proxy; the backend API, powered by the FastAPI framework for text annotation and data processing; and the latest MySQL image, utilized as the database for saving data and interacting with the backend API. Each container operates independently, allowing for separate maintenance.

The SWEGRAM Project
Project manager: Beáta Megyesi, Department of Linguistics, Stockholm University
Project participants: Rex Ruan (2019-2024), Department of Linguistics, Stockholm University; Anne Palmér and Catrin Isaksson, Department of Scandinavian Languages, Uppsala University
Web development: Jesper Näsman (2015-2018) and Shifei Chen (2020-2022), Department of Linguistics and Philology, Uppsala University

---

References

Af Geijerstam, Å. (2006). *Att skriva i naturorienterande ämnen i skolan* [Doctoral dissertation, Institutionen för lingvistik och filologi].

Björnsson, C. H. (1968). *Läsbarhet*. Liber.

Cap, F., Adesam, Y., Ahrenberg, L., Borin, L., Bouma, G., Forsberg, M., Kann, V., Östling, R., Smith, A., Wirén, M., et al. (2016). Sword: Towards cutting-edge Swedish word processing. *SLTC 2016-The Sixth Swedish Language Technology Conference (SLTC) Umeå, Sweden, 17-18 November, 2016*.

Coleman, M., & Liau, T. L. (1975). A computer readability formula designed for machine scoring. *Journal of Applied Psychology*, *60*, 283–284. https://api.semanticscholar.org/CorpusID:144250124

Ejerhed, E., Källgren, G., Wennstedt, O., & Åström, M. (1992). *The linguistic annotation system of the stockholm-umeå corpus project*. University of Umeå, Department of General Linguistics.

Flesch, R. (1948). A new readability yardstick. *Journal of Applied Psychology*, *32(3)*(221–233).

Graesser, A. C., McNamara, D. S., Louwerse, M. M., & Cai, Z. (2004). Coh-metrix: Analysis of text on cohesion and language. *Behavior research methods, instruments, & computers*, *36*(2), 193–202.

Gustafson-Capková, S., & Hartmann, B. (2006). Manual of the stockholm umeå corpus version 2.0. *Unpublished Work*.

Hultman, T. G. (1994). Hur gick det med ovix? i språkbruk, grammatik och språkförändring. *En festskrift till Ulf Teleman. 13.1*, 55–64.

Hultman, T. G., & Westman, M. (1977). *Gymnasistsvenska*. Institutionen för nordiska språk, Univ.

Johansson, V. (2009). *Developmental aspects of text production in writing and speech* (Vol. 48). Department of Linguistics; Phonetics, Centre for Languages; Literature ...

Kincaid, P. J., & Delionbach, L. J. (1973). Validation of the automated readability index: A follow-up. *Human Factors*, *15(1)*, 17–20.

Kincaid, P. J., Fishburne, R. P. J., Rogers, R. L., & Chissom, B. S. (1975). Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel.

Kokkinakis, S. J., & Magnusson, U. (2011). Computer based quantitative methods applied to first and second language student writing. *Young Urban Swedish*, 105.

Marcus, M., Santorini, B., & Marcinkiewicz, M. A. (1993). Building a large annotated corpus of english: The penn treebank.

McLaughlin, H. G. (1969). Smog grading: A new readability formula. *Journal of Reading*, 639–646.

Megyesi, B., Palmér, A., & Näsman, J. (2019). *Swegram: Annotering och analys av svenska texter*. Department of Linguistics; Philology, Uppsala University, Sweden.

Melin, L., & Lange, S. (2000). *Att analysera text: Stilanalys med exempel*. Studentlitteratur.

Nivre, J., De Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajic, J., Manning, C. D., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., et al. (2016). Universal dependencies v1: A multilingual treebank collection. *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, 1659–1666.

Nivre, J., Hall, J., Nilsson, J., Chanev, A., Eryigit, G., Kübler, S., Marinov, S., & Marsi, E. (2007). Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, *13*(2), 95–135.

Nordenfors, M. (2011). *Skriftspråksutveckling under högstadiet*. Department of Swedish; Institutionen för svenska språket.

Nyström, C. (2000). *Gymnasisters skrivande: En studie av genre, textstruktur och sammanhang* [Doctoral dissertation, Acta Universitatis Upsaliensis].

Östling, R. (2016). Shallow learning for sequence tagging. *6th Swedish Language Technology Conference (SLTC16), Umeå, Sweden*.

Östlund-Stjärnegårdh, E. (2002). *Godkänd i svenska? bedömning och analys av gymnasieelevers texter*. Institutionen för nordiska språk,

Pettersson, E., Megyesi, B., & Nivre, J. (2013). Normalisation of historical text using context-sensitive weighted levenshtein distance and compound splitting. *Proceedings of the 19th Nordic conference of computational linguistics (Nodalida 2013)*, 163–179.

Pilán, I. (2018). *Automatic proficiency level prediction for intelligent computer-assisted language learning*. University of Gothenburg, Sweden.

Pilán, I., & Volodina, E. (2016). Classification of language proficiency levels in Swedish learners' texts. *Proceedings of Swedish language technology conference*.

Scott, C.-M. (1988). Spoken and written syntax. *Later language development: Ages nine through nineteen*, *1*, 49–55.

Volodina, E., & Kokkinakis, S. J. (2012). Introducing the Swedish kelly-list, a new lexical e-resource for Swedish. *LREC*, 1040–1046.

Appendix: Error messages at uploads

When uploading annotated texts to SWEGRAM, the following list of error types might occur:

1. METADATA ERROR: Unknown data structures for the detected metadata line; Duplicate labels in the same metadata line.
2. COLUMN ERROR: The annotation checker looks into data type errors. Each token shall stand on its own line and each column in the line shall stand for a specific sort of annotation of the certain token. A collection of error types detected given the column index are listed below.
   (a) COLUMN NUMBER ERROR: The number of columns shall always be 5 if the text is not normalized; Otherwise the total number of columns shall be 12 for annotated English text and 13 for annotated Swedish text.
   (b) COLUMN INDEX ERROR: Index for token is a positive integer; index for paragraph and sentence is two positive integers separated by a dot.
   (c) COLUMN FORM ERROR: The form column is to have a non-empty string.
   (d) COLUMN NORM ERROR: The norm column is to have a non-empty string. When normalization is enabled, the checker will compare form and norm. The form is considered as not being normalized iff norm is an underline while form is not.
   (e) COLUMN LEMMA ERROR: The lemma column is to have a non-empty string.
   (f) COLUMN UPoS ERROR: Unknown universal PoS tag.
   (g) COLUMN XPOS ERROR: Unknown X-PoS tag.
   (h) COLUMN FEATS ERROR: The current feature label is unknown; the current feature value is unknown to the given feature label; the current feature structure is not standard.
   (i) COLUMN UFEATS ERROR: The current feature is not included for Swedish. This error only occurs when working with Swedish texts.
   (j) COLUMN HEAD ERROR: The head is a non-negative integer.
   (k) COLUMN DEPREL ERROR: Unknown dependency relations.
   (l) COLUMN DEPS ERROR: The deps column is to have a non-empty string.
   (m) COLUMN MISC ERROR: The misc column is to have a non-empty string.
3. UD TREE ERROR: On syntactically parsed sentences, the checker extracts an array of heads for the whole sentence and checks if the sentence's heads generate a standard UD tree. Three criteria are consulted when inspecting the head array:
   (a) There is one and only one root.
   (b) Each node represented by the index of head has arcs directly or indirectly connected to the root node.
   (c) There is no detected cycle among nodes in which one node can be a head and dependent node to another node at the same time.