AAP 37 (1994): 181-195

FROM TEXT TO DICTIONARY: STEPS FOR A COMPUTERISED PROCESS

MADDALENA TOSCANO

INTRODUCTION

The aim of this study is to illustrate the state-of-the-art of technical tools which allow the user to build the lexicon of a Swahili text. Different kinds of statistical information can also be extracted from the text with the aid of tailor-made software.[1]

The basic operation in building the lexicon of a text is lemmatization, i.e. extracting the lemma from the forms contained in the text. Once the lemma list is ready it can be converted into a list of entries, to be filled according to selected criteria.

Tailor-made programs for automatic lemmatization of texts have been available in the main European languages since the 1960's (Bortolini et al. 1971:46-55). Automatic lemmatization of texts is usually dealt with according to the system adopted by the OCP (Oxford Concordance Program), which inserts special codes in the text to help the program identify words belonging to different categories, meaning that the more codes are inserted the more categories are recognised. An adaptation of this system to Kiswahili has been prepared (Christies 1988).

A noticeable improvement of this method is the use of a morphological parser. Such a tool, namely AINI (Schadeberg 1989), has recently become available for Kiswahili. AINI produces the lemma list of a text with the indication of the main morphological categories without inserting codes in the text.

Programs for dictionary building use a lemma list as input; information on each single lemma is then added by the operator.

The program illustrated here, i.e. the LEXXIKO function of DBT, has been adapted for Kiswahili by using as input the lemma list (plus the information concerning the frequency and the category) produced by AINI; interfacing with the text allows the operator to insert samples of the use of the lemma in the entry, besides other information (e.g. location of lemma).

---

[1] This work illustrates a development of my *Manuale per l'Analisi Morfologica Computerizzata di Testi Swahili*, (I.U.O., Naples, 1990/91). This previous work contained detailed instructions on how to obtain a lemma list and a context retrieval from a Swahili text. Here is an improvement of this process, i.e. how to import the lemma list and the context retrieval in the compilation of a dictionary. The first two steps, namely the copying of the text and the extracting of the lemma, are only briefly mentioned as they have been extensively treated in the above mentioned study. Particular attention is given to the techniques and the problems connected with dictionary building through the use of tailor-made software, namely the LEXXIKO function of DBT. Once again, I should like to express my thanks to Dr. Eugenio Picchi, who adapted the DBT program to my specific requirements.

It should thus be possible for Kiswahili to build the lexicon of a text by using the lemma list as input and by inserting, where needed, samples of its use taken directly from the text

I will first briefly illustrate the stages required to extract the dictionary from a text I will then illustrate the different steps and the software tools which are available at the moment Finally, I will comment on the state-of-the-art of this software

The text used for demonstration is 'Asiyesikia la mkuu huvunjika mguu' from *Hadithi za Bibi Maahira*, by Suleman Omar Said Baalawy (1969)

A counting on the text gives the following data:

	N	V	A	P	I	Tot
types	233	214	55	186	105	793
token	104	166	29	57	22	378
lemma	90	86	20	17	22	235

"Types" indicates the amount of words occurring in a text; "token" indicates the amount of different words in the text; "lemma" indicates the amount of bases (as they would be entered in a dictionary)

Owing to the fact that the aim of this study is not in the domain of statistics no further analysis of the kind will be performed on the text

THE TECHNICAL TOOLS

A few words about the technical tools are necessary Technical tools include hardware and software

The hardware required is a IBM or compatible, possible 486, with 4MB Ram, 40 MH; high speed and huge HD is very much recommended The MS DOS used is 6.0 version The software includes a generic word processing program, three tailor-made programs and a few utilities The generic word processor used is the Word Perfect 5.1 The tailor-made programs and the utilities are:

-AINI: used (basically) to compile the lemma list of the words contained in a text;

-DBT: used (basically) to build the dictionary structure and to retrieve the contexts of words

-Utilities:

-ORTHOGRAPHIC CORRECTOR: for Kiswahili;[2]

-DOLMENU: used to adapt the text to DBT;[3]

-UTIL1191: used to connect from DBT and AINI

---

[2] The ORTHOGRAPHIC CORRECTOR is being prepared (by the READ S.r.l. Software house - 23, Via Dalmazia - 39100 Bolzano - Italy tel 0471-919203) according to an algorithm which contains data an rules on Swahili morphology

[3] DOLMENU and UTIL1191 are by my husband and myself

A quick reference to the use and performances of both DBT and AINI follow,[4] the LEXXIKO function of DBT and the various utilities will be more extensively dealt with in the next sections.

DBT: getting the context(s) of forms

The DBT program can be used on any language which uses a Latin alphabet [5] It takes a text as input and produces the context(s) of the words as output. It can count the amount of types (different words which occur in the text) and of tokens (all the words occurring in the text). It lists the forms in alphabetical order and in decreasing order of frequency. It checks if a word is contained in a text; on request by the user it displays the context(s) in which the word requested appears. It also locates the form by indicating the title of the text, the line and page number. It can search for words according to an initial or final string of characters; this is a very useful device if the operator wishes to search a text for verbal forms or pronominal forms. It is also possible to search for two associated words within a context of a given size.

In order to allow the program to give this information the operator has to follow certain steps, both on the text and within the program itself (Toscano 1990/91:120f).

AINI: getting the lemma list of a text

AINI is a morphological parser for Swahili forms. It takes a Swahili text as input and produces a data base with one record for each word as output. Each word is analysed into a lemma and both lemma and word are assigned a morphological category; the analysis is then stored in the record. The user can now formulate queries to retrieve all the stored information.

The analysis concerns the words and the components of the words. The program recognizes each graphic unit as a word, i.e. a string of characters between two empty spaces. Not all characters are considered to form a word; punctuation marks and numerals are discarded; the apostrophe is considered as part of the word because in Kiswahili it does not divide words (like for ex. in Italian). Morphosyntactic and lexical units which are made up of more than one word are not considered; their components are counted as separate units. The program has special devices to deal with particular cases, like the occurrence in a text of non-standard forms and ambiguous forms (Schadeberg 1989:14-18).

---

[4] Instruction and samples of the use of these two programs are available in Toscano 1989 and 1990/91.

[5] DBT is by Dr. Eugenio Picchi of the I.L.C. (Istituto di Linguistica Computazionale, Via della Faggiola 32 - 56100 Pisa - Italy.

LEXXIKO function of DBT: compiling the dictionary

The LEXXIKO function of DBT takes a lemma list as input and gives a list of entries as output. Each lemma becomes an entry. The structure of the entry is organized by the operator, who can decide on:

- the number of fields;
- how to arrange the fields;
- which fields can be put in alphabetical order;
- which fields will appear on the screen (the remaining fields will be available on request).

Once the structure has been decided on, working on the entries starts. During this step the operator can:

- select the field which he wants to fill;
- modify the content of a field;
- insert, delete a field;
- duplicate, assign a new name to a field (to be chosen among those included in the structure).

Once the filling of the entries has been completed it is possible to print them, according to alphabetical order.

Utilities - communicating between various software

1) The ORTHOGRAPHIC CORRECTOR checks the text for orthographic mistakes

2) DOLMENU inserts into the texts the codes required by DBT.

3) UTIL1191 has been prepared to perform the following related steps:

- Code Transformer: converts the text prepared for DBT in a text for use in AINI. The text reported in pag. 10 is a result of such a transformation.
- Lemma lem Transformer: converts the list obtained from AINI (a sample is contained in pag. 12) into a list for use in DBT (see pag. 14).

Further improvements

A further improvement of the whole process is being considered at the moment, that is the possibility of inserting all the contexts of the forms contained in the text in MARIAMA, a relational data base which allows the user to search data organised at differeny levels [6]

---

[6] MARIAMA is by prof. R. Nicolai, GRILL - IDERIC - 63, Bd de la Madeleine, Bât. A - Université de Nice - Sophia Antipolis - 06000 Nice

DBT provides on request the contexts of each single forms The operator stores the significant contexts in the entry and discards those which are not of interest; there is always the possibility of recalling the contexts. Modifications can be operated on the single entries. To a certain extent the same modification can be operated on a selected set of entries, but searching on the entries is on the whole rather limited Thus the possibility of charging all the collected data in MARIAMA is being considered.

The main differences between DBT and MARIAMA lies in the following:

-MARIAMA is a relational data base which allows searching throughout entries in various combinations, but it gives no access to the text so that no context can be retrieved directly;

-DBT is a less powerful data base when compared to MARIAMA but it gives a direct access to the contexts

My next improvement will be: 1) find a quick way of getting, through DBT, all the contexts of all the lemma of a text and then charging them in MARIAMA; 2) set a structure and compile the dictionary by using MARIAMA.

FROM TEXT TO DICTIONARY: THE MAIN STAGES

The whole process can be divided into three main stages: 1) copying and preparing the text on disk; 2) obtaining the lemma list; 3) filling the entries of the dictionary.

1) Copying the text

The first stage, the recording of a text on disk, can be performed three ways: a) typing the text; b) using a scanner; c) copying it from another disk

The first option is the most common (the text used for demonstration is typed) though the longest one as well

The second option, i.e the use of a scanner requires special hardware and software equipment and a good quality print-out. The printing quality of most of the Kiswahili texts which one comes across is rather poor (characters not spaced out enough, contrast not homogeneous, etc.); consequently it is easy to misread and, furthermore, the mistakes are not always clearly marked. This requires an accurate check of the text, for which a Kiswahili orthographic corrector is of help.

As for the third option, this is a real opportunity in countries with high-tech printing processes (much printing, like newspapers, is done by computers). In this case the storing of a text on disk is one step of the printing Publishers can be persuaded to hand over their material once they are sure it will not be used in contrast with their interests.[7] It appears less practicable

[7] This agreement is effective in Italy between the I.L.C. (Istituto di Linguistica Computazionale) and some main publishing houses which deal with Italian literary texts

in countries where the publishing sector is frequently a rather individual and casual activity, especially when dealing with literary works, as it is, in fact, our case

The following is a partial reproduction of the first two pages of the text used here for demonstration

```
Hadithi ya Kwanza
ASIYESIKIA IA MKUU HUVUNJIKA MGUU

   BIBI MAAHIRA alianza kutoa hadithi hii kwa kufasiri fumbo
hili neno kwa neno kwanza, tena kwa kutoa undani wake,
ndipo akaendelea na hadithi yake. Alisema, "wajukuu wangu,
maana ya fumbo hii ni dhahiri, yaani asiyesikiliza na kufuata
maneno ya watu wazima basi miguu wake huvunjika.
   Mtoto huyo aliishi na babu yake, kama hivi ninavyoishi
nanyi, wajukuu zangu, hata akawa mkubwa. Babu yake huyo
hakuwa hodari wa kumfundisha mjukuu wake mambo mazuri
tokea alipokuwa mdogo, bali akimuachia kwenda mwendo
aliopenda. Kwa bahati mbaya mtoto huyo alikuwa akipenda
kutumia mapesa sana na akinunua vitu vingi visivyokuwa na
haja, kama vile matunda ya kutegea ndege, panda za kupigia
ndege, matunda machanga, muradi alikuwa na tabia ya kununua
ambacho alikuwa na haja nacho na asichokuwa na haja nacho.
Mtoto huyo aliendelea na tabia hii mbaya hata akawa analewa.
Alikuwa akifanya kazi ya kibarua na kupata kutwa shilingi nne,

1
lakini kwa tabia yake mbaya hii alikuwa habariki na mwisho hata
chakula na nguo zake za kuvaa vilimfanya taabu, yaani alikuwa
akivaa matambara kila wakati na akila chakula cha ovyo cha
kiasi cha kujaza tumbo lake tu akaishi.
   Iena yule mtoto aliingilia kuuza zile nguo na viatu vinginevyo
alivyonunua. Mwishoni alirudia kuvaa matambara yake.
   Yule aliyemuuzia kiwanja chake alianza kujenga nyumba
kwenye kiwanja hicho, basi, naye alikwenda kuomba kazi ya
kibarua, akapewa. Siku moja alikuwa akichukua mawe kwa
kichwa, jiwe moja kubwa likamponyoka na kumpiga mguuni
```

Once the text is recorded it needs special codes in order to signal titles and page number The utility DOLMENU can do most of the work Here is the converted text as it appears after the codes necessary for DBT have been inserted The code % identifies the sequence of characters which will be indicates as the reference The part of text contained between # and @ will not be indexed by DBT The page number is indicated with 0000 The code &A indicated that sequence of characters which contains an apostrophe are to be considered as one word and not as two words, as it would be the case with Italian

Here is how the same text appears after these adjustments have been performed:

```
%#ASIYESIKIA#
#Hadithi ya Kwanza@
#ASIYESIKIA IA MKUU HUVUNJIKA MGUU@
$0001$
&A

   BIBI MAAHIRA alianza kutoa hadithi hii kwa kufasiri fumbo
hili neno kwa neno kwanza, tena kwa kutoa undani wake,
ndipo akaendelea na hadithi yake. Alisema, "wajukuu wangu,
maana ya fumbo hii ni dhahiri, yaani asiyesikiliza na kufuata
maneno ya watu wazima basi miguu wake huvunjika.
   Mtoto huyo aliishi na babu yake, kama hivi ninavyoishi
nanyi, wajukuu zangu, hata akawa mkubwa. Babu yake huyo
hakuwa hodari wa kumfundisha mjukuu wake mambo mazuri
tokea alipokuwa mdogo, bali akimuachia kwenda mwendo
aliopenda. Kwa bahati mbaya mtoto huyo alikuwa akipenda
kutumia mapesa sana na akinunua vitu vingi visivyokuwa na
haja, kama vile matunda ya kutegea ndege, panda za kupigia
ndege, matunda machanga, muradi alikuwa na tabia ya kununua
```

```
ambacho alikuwa na haja nacho na asichokuwa na haja nacho.
Mtoto huyo aliendelea na tabia hii mbaya hata akawa analewa.
Alikuwa akifanya kazi ya kibarua na kupata kutwa shilingi nne,

$0002$
lakini kwa tabia yake mbaya hii alikuwa habariki na mwisho hata
chakula na nguo zake za kuvaa vilimfanya taabu, yaani alikuwa
akivaa matambara kila wakati na akila chakula cha ovyo cha
kiasi cha kujaza tumbo lake tu akaishi.
. . . . . .
. . . . . .
   Tena yule mtoto aliingilia kuuza zile nguo na viatu vinginevyo
alivyonunua. Mwishoni alirudia kuvaa matambara yake.
   Yule aliyemuuzia kiwanja chake alianza kujenga nyumba
kwenye kiwanja hicho, basi, naye alikwenda kuomba kazi ya
kibarua, akapewa. Siku moja alikuwa akichukua mawe kwa
kichwa, jiwe moja kubwa likamponyoka na kumpiga mguuni
```

The text can be used immediately in DBT to search for words and for their contexts. The possibility of retrieval of contexts will be used in the filling of the entry, where needed. The text with the codes for DBT is then processed by the utility Util1191 in order to convert the codes for DBT in codes suitable for for AINI.

2) Procuring the lemma list

Once the text is processed in AINI each word becomes a record; the lemma is extracted and the analysis of the word is performed. It is now possible to ask the program to produce the list of the lemma contained in the text, with indication of the frequency and of the grammatical category of each lemma. Procuring the lemma list is the central stage of the whole process and requires several different steps. Some of these steps are fairly quick to perform, others require more time, depending on three main factors: 1) how correctly the text has been typed; 2) the amount of non-standard forms in the text; 3) the amount of lemma already known by the parser.

Each word of the text is analysed and classified by AINI according to the following morphological categories and subcategories:

N Nominals - including nominals from class 1 to 15 (N1 .. N15)

V Verbals - three subcategories are recognized: V0- plurisyllabic verb stems ending in *-a*; V1- monosyllabic verb stems; V2- plurisyllabic verb stems ending in vowels other than *-a*. Extended verbs are considered as different lemma.

P Pronominals - including 9 subcategories (from P0 to P9).

A Adjectivals - including the non-variable (A0) and the variable (A1) adjectives

I Invariables - including non-variable forms (I0).

From AINI we obtain the lemma list with the indications of the morphological categories. The following is a sample of such a list as obtained from AINI:

Freq	Lemma	Gram
58	a	P0
1	achia	V0
1	aina	N9
41	ake	P2
3	ako	P2
1	amba	P4
4	ambia	V0
1	amka	V0
4	angu	P2
6	anza	V0
2	ao	P2
2	arobaini	N9
1	asubuhi	N9
1	baada	N9
3	baba	N9
14	babu	N9
1	badala	N9

It is very important to note that the lemma list of a text will include only the base[8] of the words contained in the text; affixes will not be listed If the user wishes to list the affixes in his dictionary he will have to prepare a special list which can be added to the main one or insert new entries for the morphemes when compiling the dictionary. By posing a number of very specific queries it is possible to get AINI to provide a list of the morphemes used in a text; but the process is rather long and repetitive.

3) Filling dictionary entries

The dictionary function of DBT will build a record for each lemma with the lemma as main entry and frequency and category as the next subsections; further subsections can be selected and inserted by the user. Samples of the usage of the lemma, with the reference to the text, can be obtained on request by the user.

As for the technical steps, some special files have to be created in order to allow the program to build the entries from the lemma list and to interact with the text(s).

Let us now see how to build the structure of an entry and how to fill the entries.

CONVERTING THE LEMMA LIST

The lemma list produced by AINI contains three columns: the first one refers to the frequency of the lemma in the text, the second contains the lemma, the third contains the abbreviations indicating the morphological category of each lemma. This list has to be partially modified for transferral to the dictionary program; the modifications are as follows:

- the first column is left as it is and it will be retrieved in the dictionary program;

- the second column contains the lemma; a dash will be added in the position of the morpheme(s);

---

[8] Invariables are listed as such. As for variables, Nominals, being independent, are indicated in the singular forms; all other categories, being dependent, are listed according to the base

- the third column is modified so that the morphemic abbreviation is converted into a grammatical abbreviation; only I0, i.e. the invariable, remains the same, because the grammatical function that an invariable can perform is rather unpredictable in Kiswahili.

On the whole, the conversion from morphological to grammatical abbreviation has a reasonable rate of accuracy. There will certainly be cases where the operator will have to make a few adjustments, as in the case of invariables and with other bases which can perform more than one function, but the saving of time is noteworthy. The following is a sample of the lemma list for use in the LEXXIKO function of DBT (converted by Util1191)

Freq.	Lemma	Gramm. Cat.
58	-a	pr
1	-achia	v
1	aina	n(-)
41	-ake	poss
3	-ako	poss
1	amba-	rel
3	-ambia	v
1	-amka	v
4	-angu	poss
6	-anza	v
2	-ao	poss
2	arobaini	n(-)
1	asubuhi	n(-)
1	baada	n(-)
3	baba	n(-)
14	babu	n(-)
1	badala	n(-)
2	bahati	n(-)
2	baisikeli	n(-)
1	-baki	v
2	bali	I0
1	-bariki	v

Now that we have an adapted lemma list let us proceed with the building of an entry

BUILDING THE STRUCTURE OF ENTRIES

An entry in a dictionary is usually arranged in subentries, according to the specific aim of the dictionary itself (Al-Kasimi 1977) Except in some very special cases, the first two subsections almost always contain the indication of the lemma and the grammatical category respectively Then follow other subsections dedicated to other grammatical usage and/or gloss of the entry Subsections with idiomatic and other special usages of the lemma, are often provided (Dubois 1981)

In this sample the entry will have the following subsections, the first three of which will be automatically filled, the fourth filled by the operator, the fifth and sixth ones automatically filled on request of the operator:

Name of field	Contents of field
Lemma	contains the lemma
Frqnc	contains the frequency number
GrCat	abbreviation of the gramm. category
Gloss	translation of the lemma
Cntxt	selected contexts in which the lemma is used
Refer	reference to the text

It is possible, however, to build a more complete structure. The structure built for this text is described in a special file prepared in order to make DBT build the required structure. The file is divided in two sections. The first section lists the 9 fields which compose a record, the last three of which cannot be indexed. The second section describes only the 6 fields which can be indexed.

```
09
001 Lemma 01  01
002 Frqnc 02  02
003 Grcat 03  03
004 Gloss 04  04
005 Sublm 05  05
006 Subgl 06  06
007 Varie 07  00
008 Cntxt 00  00
009 Refer 00  00
07
Lemma Lemma      *
Frqnc Frequency  *
Grcat Grammatical cat. *
Gloss Glossa     *
Sublm Sub-lemma  *
Subgl Sub-glossa  *
Varie Varied     *
```

As it is rather unlikely that all these fields will be needed for every entry, it is possible to ask the program to provide a minimum set of fields; here it has been decided to ask for the first four fields. The remaining ones will be added and/or duplicated and/or deleted for each single entry on request by the user.

FILLING ENTRIES

Now that we have the structure we can start to fill it.

Each lemma has now become a record, the first field of which contains the lemma; each lemma becomes an entry. The entry will have a minimum of the following fields: Lemma, Frequency, Grammatical category, Gloss, Context, Reference.

The first three fields, namely Lemma, Frqnc, Grcata, will be automatically filled by DBT during the first operation, that is when constructing the structure of the entries. The fourth has to be filled by the operator, the fifth and sixth fields will be automatically filled on request by the operator. This is a sample of how the entry **babu** appears after the first step, that is the construction of the entries operated by DBT.

```
Lemma babu
Frqnc 14
Grcat n(-/-)
Gloss
```

The gloss has to be written by the operator; he can also ask DBT to retrieve one or more contexts of the forms in which the lemma appears. This is how our entry appears after these further steps have been performed:

Lemma	babu
Frqnc	14
Grcat	n(-/-)
Gloss	grand-father; great-grand-father
Cntxt	Bibi Maahira alianza hadithi akasema, "Hapo zamani alikuwepo mtoto aliyelelewa na babu
yake	mzaa baba tokea alipokuwa mdogo mpaka akawa mkubwa. Baba wa mtoto huyo alikufa
ikabidi	aelewe na babu yake. -
Refer	ASIYESIKIA 15 - 0001 15[9]

It is important to stress that the retrieval of contexts is done through the searching of forms. When a lemma gives place to few forms, like it is the case with nominals, it is quick to check if the possible forms occur in the text. In case of searching of forms for a verbal lemma, than the list of forms is required. AINI can be asked to provide such a list. The following is a partial sample of the result produced by AINI to the quest to produce the list of forms of lemma -*ambia* and *-fanya*:

text	**pg.**	**line**	**form**	**analysis**
9999	4	11	alimwambia,	V0 A PST a mw - -
9999	4	22	akawaambia	V0 A SBS a wa - -
9999	4	32	aliwaambia	V0 A PST a wa - -
9999	1	22	kufanya	V0 A INF - - - -
9999	1	33	akifanya	V0 A SIT a - - -
9999	2	3	vilimfanya	V0 A PST vi m - -
9999	2	8	kufanya	V0 A INF - - - -
9999	4	8	kufanya	V0 A INF - - - -
9999	4	20	kukufanya	V0 A INF - ku - -

Here is how the entry *-ambia* appears when filled with the retrieved context of the form *akawaambia*:

Lemma	-ambia
Frqnc	3
Grcat	v
Glossa	say
Cntxt	Maahira akawaambia wajukuu wake, hii ndiyo hadithi ya "ASIYESIKIA la mkuu huvunjika
	mguu" -
Refer	ASIYESIKIA 86 - 0004 18

[9] The group of numbers refer to total line number, page number and page line number respectively

It is also possible to obtain the forms of a lemma by asking the DBT to provide the list of words which contain a set of characters The answer of DBT when asked to provide the words which contain **achi** is *akimuachia*. The entry *-achia* will thus be completed:

Lemma -achia
Frqnc 1
Grcat v
Gloss leave to, for; bequeath
Cntxt Babu yake huyo hakuwa hodari wa kumfundisha mjukuu wake mambo mazuri tokea alipokuwa mdogo, bali akimuachia kwenda mwendo aliopenda
Refer ASIYESIKIA 25 - 0001.25

In case the same lemma produces forms with different meanings the structure of the entry can be adapted to the needs, either by inserting another field, as it is the case when subentries are needed, or by duplication of fields, as it is the case with fields containing contexts and reference.

Here is how the same entry *-achia* appears when filled according to the contexts founded in the works of Mohamed Suleman Mohamed:

Lemma -achia
Frqnc 38
Grcat v
Gloss leave to, for
Cntxt Maliwazo aliyomwachia Aziza bado yakifanya kazi katika kichwa chake. Aziza akimpenda sana sana Fuad.
Refer NYOTA YA REHEMA chap 2 0020.1]
Subgl permit, allow
Cntxt "Nitakwachia ujaribu mtihani wako, na nitakwachia kwa ajili ya rafiki yako uliyekuja naye.
Refer Mji 0041.20]
Sublm -jiachia
Subgl drop, lie down; cfr -jiacha
Cntxt Rajabu alijiachia juu ya kiti cha pembea na kuhisi ameingia mtegoni.
Refer KICHEKO CHA USHINDI 0005 6

While the operator fills the entries, the DBT stores the information under the directory \LEXXIKO When the operation of filling the entries is over, the last stage, i. e. the printing, can start.

PRINTING THE ENTRIES

After the entries have been filled (either a group or the whole list) it is necessary to perform a few intermediate steps in order to adjust them for printing

The first step is performed in DBT by converting the file which contains the filled entries in a file suitable to be used in WP51; the remaining steps are performed in WP51. Here is how the file compiled by DBT appears when converted in an ASCII file readable in WP51; the sample referes to lemma *-a*:

$$A-a$$a $$B58$$b $$Cprep$$c $$Dof$$d$ $$Hkwa tabia yake mbaya hii alikuwa habariki na mwisho hata chakula na nguo zake za kuvaa vilimfanya taabu, yaani alikuwa akivaa matambara kila wakati na akila chakula ££Acha££a ovyo cha kiasi cha kujaza tumbo lake tu akaishi $$h $$IASIYESIKIA @35 - 0002 3]$$I $$Ccong$$c $$HYule mtoto alilia kwa sababu mali yake amekwisha kuipoteza na akitoka hospitali hajui ££Acha££a kutumia, na yule babu mtu alililia hilo hilo $$h $$IASIYESIKIA 73 - 0004 5$$i $$A-achia$$a $$B1$$b $$Cv$$c leave to, for; bequeath $$HBabu yake huyo hakuwa hodari wa kumfundisha mjukuu wake mambo mazuri tokea alipokuwa mdogo, bali ££Aakimuachia££a kwenda mwendo aliopenda $$IASIYESIKIA 25 - 0001 25$$i

The file is generated by DBT; it is then imported in WP51. Each record starts with the lemma, contained between the codes $$A and $$a; different fields are identified by different codes. What we have now in WP51 is a list of entries, with subentries, each one identified by different codes. By using the Macro function of WP51 codes are eliminated and, if necessary, fields are differentiated through the use of different fonts or style

Entries, and subentries, need to be separated and/or differentiate through the use of different fonts in order to facilitate the reading to the dictionary users. The steps performed in WP51 aim to obtain a 'cleaned' list of the fully compiled entries, which is in fact the final output of the whole process.

Here follows a sample of the final output obtained from the whole process.

-a 58 *prep* of [*kwa tabia yake mbaya hii alikuwa habariki na mwisho hata chakula na nguo zake za kuvaa vilimfanya taabu, yaani alikuwa akivaa matambara kila wakati na akila chakula cha ovyo cha kiasi cha kujaza tumbo lake tu akaishi.* - ASIYESIKIA @35 - 0002 3] *cong [Yule mtoto alilia kwa sababu mali yake amekwisha kuipoteza na akitoka hospitali hajui cha kutumia, na yule babu mtu alililia hilo hilo.* - ASIYESIKIA 73 - 0004 5]

-achia 1 *v* leave to, for; bequeath *[Babu yake huyo hakuwa hodari wa kumfundisha mjukuu wake mambo mazuri tokea alipokuwa mdogo, bali akimuachia kwenda mwendo aliopenda* - ASIYESIKIA 25 - 0001 25]

amba- 1 *rel pref* who, which *[muradi alikuwa na tabia ya kununua ambacho alikuwa na haja nacho na asichokuwa na haja nacho* - ASIYESIKIA @ 30 - 0001 30]

-ambia 3 *v* say *[Maahira akawaambia wajukuu wake, hii ndiyo hadithi ya "ASIYESIKIA la mkuu huvunjika mguu".* - ASIYESIKIA 86 - 0004 18]

-amka 1 *v* wake up *[basi alilewa sana hata asijifahamu, hapo rafiki zake waliiba ile fedha aliyokuwa nayo na aliamka asubuhi hana fedha wala hana rafiki.* - ASIYESIKIA 61 - 0002 29]

asubuhi 1 *n(-/-)* morning *[basi alilewa sana hata asijifahamu, hapo rafiki zake waliiba ile fedha aliyokuwa nayo na aliamka asubuhi hana fedha wala hana rafiki* - ASIYESIKIA 61 - 0002 29]

baada 1 *avv* after. **Baada ya** *prep* after *[Baada ya wiki moja fedhahiyo iibaki shilingi mia tano tu. Basi hapo ilimjia fikira akainunulie baisikeli.* - ASIYESIKIA 54 -0002 22]

baba 3 *n(-/-) (a-/wa-)* father; maternal uncle; ancestor *["Hapo zamani alikuwepo mtoto aliyelelewa na babu yake mzaa baba tokea alipokuwa mdogo mpaka akawa mkubwa. Baba wa mtoto huyo alikufa ikabidi aelewe na babu yake. Mtoto huyu alirithi kiwanja kizuri chenye thamani kilichokuwa katikati ya mji.* - ASIYESIKIA 16 - 0001 16] patron, protector, guardian *[Bibi Maahira alianza hadithi akasema, "Hapo zamani alikuwepo mtoto aliyelelewa na babu yake mzaa baba tokea alipokuwa mdogo mpaka akawa mkubwa.* - ASIYESIKIA 15 - 0001 15]

babu 14 *n(-/-) (a-/wa-)* grand father; great-grand father *[Bibi Maahira alianza hadithi akasema, "Hapo zamani alikuwepo mtoto aliyelelewa na babu yake mzaa baba tokea alipokuwa mdogo mpaka akawa mkubwa Baba wa mtoto huyo alikufa ikabidi aelewe na babu yake* - ASIYESIKIA 15 - 0001.15]

badala 1 *n(-/-)*; **badala ya** *prep* instead of *[Babu yake alimkataza kufanya hivyo kwani alijua ya kuwa hatanunua kitu cha maana badala ya kiwanja chake* - ASIYESIKIA 40 - 0002.8]

bali 2 *cong* but; on the contrary *[Babu yake huyo hakuwa hodari wa kumfundisha mjukuu wake mambo mazuri tokea alipokuwa mdogo, bali akimuachia kwenda mwendo aliopenda* - ASIYESIKIA 25 - 0001.25]

basi 11 *cong* well, and so, and then; *inter* very well! *[Alisema, "wajukuu wangu, maana ya fumbo hii ni dhahiri, yaani asiyesikiliza na kufuata maneno ya watu wazima basi miguu wake huvunjika* - ASIYESIKIA 9 - 0001.9] *[Mwishoni alirudia kuvaa matambara yake Yule aliyemuuzia kiwanja chake alianza kujenga nyumba kwenye kiwanja hicho, basi, naye alikwenda kuomba kazi ya kibarua, akapewa* - ASIYESIKIA 65 - 0002.33]

ABBREVIATIONS

N Nominal	P Pronominal	rel relative
V Verbal	V Verbal	emph emphatic
P Pronominal	adj adjective	cop copula
A Adjectival	n(*) noun	ng negative
I Invariable	pr preposition	dem demonstrative
A Adjectival	v verb	conj conjunction
I Invariable	poss possessive	
N Nominal	int interrogative	

BIBLIOGRAPHY

Al-Kasimi, A 1977 *Linguistics and Bilingual Dictionaries* Leiden: Brill

Bortolini, U , C Tagliavini, A Zampolli 1971 *Il Lessico di frequenza della lingua italiana contemporanea* Milano: Garzanti

Christies, J J 1988 "Computer selection of Swahili data (Abstract)." Department of Linguistics, University of Edinburgh

Dubois, J 1981 Models of the Dictionary: Evolution in Dictionary Design" *Applied Linguistics* 2,3:236-249

Hartmann, R R K ed 1983 *Lexicography: Principles and practice* London: Academic Press

Menard, Nathan 1983 *Mesure de la richesse lexicale* Genève: Slatkine; Paris: Champion

Prinsloo, D 1991. "Towards a computer compatible lexicography for Northern Sotho." 22nd Annual Conference on African Linguistics (ACCAL), Nairobi, July 1991

Schadeberg, T. 1989. *AINI - A Morphological parser for Kiswahili* Leiden: Department of African Linguistics, RUL,

Suleiman Omar Said Baalawy. 1969. "Asiyesikia la mkuu huvunjika mguu" *Hadithi za Bibi Mahira*. London: Evans Brothers Ltd

Toscano, M. 1989. "A few samples of computerised lexical analysis on some Swahili writings" 14th ALA Conference, Dakar, 20-23 March, 1989, University of Wisconsin-Madison, Madison.

Toscano, M. 1990/91. *Manuale per l'Analisi Morfologica Computerizzata di Testi Swahili* Napoli: Dipartimento di Studi e Ricerche su Africa e Paesi Arabi, I.U.O,