A Polish-to-English electronic dictionary designed for the purposes of MT.

Krzysztof Jassem, jassem@math.amu.edu.pl
Maciej Lison, lison@math.amu.edu.pl
Filip Graliński, filipos@venus.wmid.amu.edu.pl
Bogusław Rutkowski, brutal@venus.wmid.amu.edu.pl
Adam Mickiewicz University,
Faculty of Mathematics and Computer Science
ul. Matejki 48/49, 60-769 Poznań
Poland

Abstract

The aim of the research is to obtain a bilingual dictionary which could be consulted by an MT algorithm translating computer-oriented texts from Polish to English. A single entry of the dictionary consists of a Polish lexeme identifier (a canonical form of the lexeme and a code of its inflection), a list of its inflection forms and a list of "translation units". Each translation unit includes a list of constraints (syntactic, semantic, pragmatic) and a description of an English equivalent. The contents of the dictionary are based on the vocabulary of a text corpus. A corpus consisting of close to 200 000 words has been collected on the basis of Polish texts in computer-oriented magazines and manuals. The corpus has been processed in order to obtain three lists of lexemes occurring in the corpus: proper names, abbreviations and other lexemes. An interactive editor has been designed to assist a lexicographer in describing the entries of the dictionary. The output data of the editor are of the SGML-document type format. An application which converts the dictionary into a binary file has been worked out in order to optimise the access time. The procedures for binary storage of the dictionary are universal enough to be applied to NLP systems other than MT.

1. Preface

In [Jassem, 1997], a prototype MT Polish-English system was described. The system included an electronic dictionary whose format was bound up with the translation algorithm. The dictionary contained about 30000 inflected forms generated from 2000 canonical forms chosen randomly from various sources (mainly computational dictionaries and texts). The study reported here aims at creating an electronic Polish-English dictionary of universal character. The universality of the dictionary is understood in the following aspects:

1) The dictionary should enable machine translation of texts from various domains of computer science.
2) It should be possible to apply the dictionary to various types of MT systems.
3) The technology of binary storage of the dictionary may be used in electronic lexicons designed for various purposes.

The first of the above aims was achieved by means of the analysis of a corpus of Polish texts concerning computer science excerpted from a relatively wide range of sources. The results of the analysis are related in paragraph 2.

The second aim is intended to be achieved by means of the following solutions:

- the text format of the dictionary is consistent with the popular SGML (Standard Generalised Mark-up Language) document type,
- the encoded information in the key field, called *Complementation*, is described by means of a context-free grammar which makes it easy to parse.

The details on the format of the entry into the dictionary may be found in paragraph 3.

The third aim is an important ”side effect” of the research on binary storage of an electronic bilingual dictionary. A three-level structure: *lexeme-form-grapheme* of the dictionary has been worked out. Only the level of lexeme is relevant for MT. However, the three-level structure makes it possible to apply the technology to other NLP systems. This will be discussed in paragraph 4.

2. Analysis of text corpus

It seems that in order to design a dictionary for translating Polish texts from the domain of computer science, the analysis of a corpus of computational texts is indispensable. The alternative of basing the dictionary on existing computer-oriented ”paper” lexicons only does not meet the condition for the dictionary to contain all words and lexical phrases occurring in computer-oriented texts - including those which do not belong to the computational vocabulary.

2.1. Tagging the corpus

The corpus was based on various Polish texts concerning computer science and information technology, selected mostly from computer magazines accessible through the World Wide Web. All texts were converted into a coherent format by supplementing them with SGML tags. Some fragments of the source texts were tagged as ‘undesirable’. The ‘undesirable’ excerpts included:

a) longer fragments composed entirely or almost entirely of ‘non-lexical strings’ (a non-lexical string is meant here to be a sequence of characters which does not form a Polish word), e.g. a listing of a program or a message in English generated by an application (such fragments appear quite frequently in texts concerning information technology),

b) longer fragments (at least a few paragraphs) not dealing with computer science, e. g. a review of a computer game focused totally on the story of the game, disregarding technical aspects of the game.

These fragments were not taken into consideration in the next phases.

2.2. Incorporating morphological information within the corpus

The next phase consisted in processing the corpus word by word (a word is meant here to be an arbitrary sequence of Polish letters and hyphens included between two successive spaces) by means of a morphological analyser POLEX (details on POLEX may be found in [Vetulani et al, 1998]). The morphological analyser tagged each word with appropriate morphological information (the canonical form from which a given word is derived and the encoded information on inflectional features of the word). Words unrecognised by the analyser were tagged as ‘unidentified’. These were: some correct Polish words (mostly recent computer-related terminology), proper nouns, abbreviations and non-lexical strings.

Altogether, the corpus contained 198,486 words (identified as well as unidentified).

2.3. Creating the lexeme lists

The objective of the last phase was to create frequency lists of all the lexemes whose inflected forms were represented in the corpus. Three lists were created: proper nouns, abbreviations and other lexemes (the list of other lexemes will be further on referred to as the lexeme list). A fully automatic approach to the task would have failed on account of the following facts:

(1) some words were recognised by the morphological analyser as lexically ambiguous, e.g. the word *kopie* is either nominative plural of the noun *kopia* (Eng. *a copy*) or Present Tense, 3rd person, singular of the verb *kopać* (Eng. *to kick*),

(2) some of the words unidentified by the morphological analyser are correct inflected forms of Polish lexemes, e.g. *piksel* (Eng. *a pixel*), *pecet* (Eng. *a PC*), *bitmapa* (Eng. *a bitmap*). Such words ought to be included in the lexeme lists although it was not possible to distinguish them automatically from non-lexical strings,

(3) most proper names were not identified by the analyser.

Therefore human linguistic competence was indispensable in: choosing the correct lexeme for words recognised as ambiguous (1) and classifying unidentified words into one of the four categories: proper nouns, abbreviations, other lexemes and non-lexical strings (2, 3). The solution accepted her was a semi-automatic approach.

Amongst the words recognised by POLEX as ambiguous, a number had the feature of being an inflected form of two different lexemes, where one of the two lexemes is commonly used in Polish, while the other is extremely rare and probably never used in a computational text. Beneath, a few examples of such words are given.

Word	'obvious' lexical interpretations	probably inappropriate lexical interpretations
albo	the conjunction *albo* (Eng. *or*)	vocative, singular of the noun *alba* (Eng. *alba=a medieval Provençal song*)
bez	the preposition *bez* (Eng. *without*)	genitive, plural of the noun *beza* (Eng. *a meringue*)
można	the adverb *można* (Eng. *it is possible to*)	nominative, singular, feminine of the adjective *możny* (Eng. *puissant*)
musi	Present Tense, 3rd person, singular of the verb *musieć* (Eng. *must*)	nominative, plural, masculine of the adjective *muszy* (Eng. *fly-like*)

In such cases, in order to avoid time-consuming enquiring, it was assumed that the 'obvious' lexical interpretation was always correct.

2.4. The results

Finally, three frequency lists were produced:

- the list of all the proper nouns (485 different proper nouns),
- the list of all the abbreviations (677 different abbreviations).
- the list of other lexemes occurring in the corpus (9,634 different lexemes),

Beneath, the initial fragment of the third list is given for illustration.

Lexeme	English equivalent	number of occurrences	frequency
w	in	5939	3.3355%
być	to be	4631	2.6009%
i	and	3725	2.0920%
z	with	3544	1.9904%
na	on	3462	1.9443%
do	to	2783	1.5630%
się	oneself	2551	1.4327%
to	it	2056	1.1547%
nie	no	1962	1.1019%
program	program	1621	0.9103%
ten	this	1586	0.8907%

The corpus contained 12,346 non-lexical strings (6.22% of all words in the corpus).

3. Format of the entry

Section 3.1 aims at formalising the conditions of including a lexical phrase (a single word will be treated here as a special case of a phrase) into the dictionary. The format of the entry is described in section 3.2. Section 3.3 presents some examples of entries.

3.1. Dictionary phrase

In machine translation rule-based approach to transferring lexical phrases often fails. One of possible solutions to overcome this difficulty consists in binding the structure of lexical data in the dictionary to the grammar used by the syntax analysis of the source language in the translation algorithm. A dictionary word corresponds to a terminal symbol of the grammar and a dictionary phrase corresponds to a non-terminal symbol of the grammar. The MT algorithm assumes the preference of the 'dictionary transfer' of a phrase to the 'grammar transfer'. Using the notions of the theory of generative grammars the definitions of the lexical phrase, the open phrase and the dictionary phrase will be formed.

> *A sequence of terminal symbols derived from a non-terminal symbol in the given grammar is called a* ***lexical phrase****.*

In particular, a single word is a phrase if it can be derived from a non-terminal symbol in the given grammar.

It could be difficult to place into the dictionary all lexical phrases which should not be translated on the basis of grammar rules. For example, in order to translate the phrase 'wziąć *coś* pod uwagę' into the phrase 'to consider *something*', the algorithm would require the dictionary to store all phrases with *coś* substituted by a noun phrase (in accusative). Therefore, the dictionary should be able to contain phrases with non-terminal symbols.

> *A sequence of terminal and non-terminal symbols derived from a non-terminal symbol in the given grammar, containing at least one non-terminal symbol, is called an* ***open phrase****.*

It seems that the dictionary should not contain open phrases which have only non-terminal symbols. The dictionary should include only phrases which have lexical meaning, whereas an open phrase built exclusively of non-terminal symbols carries only grammatical meaning.

> *A sequence of terminal and non-terminal symbols derived from a non-terminal symbol in the given grammar, containing at least one terminal symbol, is called a* ***dictionary phrase****.*

It is assumed here that a dictionary phrase is the only structure allowed to be stored in an electronic dictionary. Let us notice that single words (provided that they are derivable from a non-terminal symbol, a condition which any sensible grammar should meet) as well as lexical phrases and open phrases including at least one terminal satisfy the above condition.

The dictionary designed for the purpose of MT should not include proper dictionary phrases (dictionary phrases which are not single words) amenable to rule-based transfer (rule-based transfer is left to the translation algorithm).

3.2. Textual format of the dictionary.

An entry into the dictionary is a lexeme of a Polish dictionary phrase. Homographic entries (entries of the same meaning represented by the same canonical form) are separated if and only if the graphical representations of any of their respective inflected forms differ. In other words: an entry is identified uniquely by the canonical form of the lexeme and the inflection paradigm. For example, the lexeme ‘mysz’ (a mouse) would be represented in the dictionary as one entry, although it has two different meanings (an animal, a computer device) because the inflection paradigms for both meanings are identical. However, the lexeme ‘organ’ (an organ) in the sense of a human organ (plural nominative: ‘organy’) would be separated from the lexeme ‘organ’ in the sense of an organisation (plural nominative: ‘organa’).

3.3. Graphical representation of an entry

Figure 1. presents the logical structure of the entry.

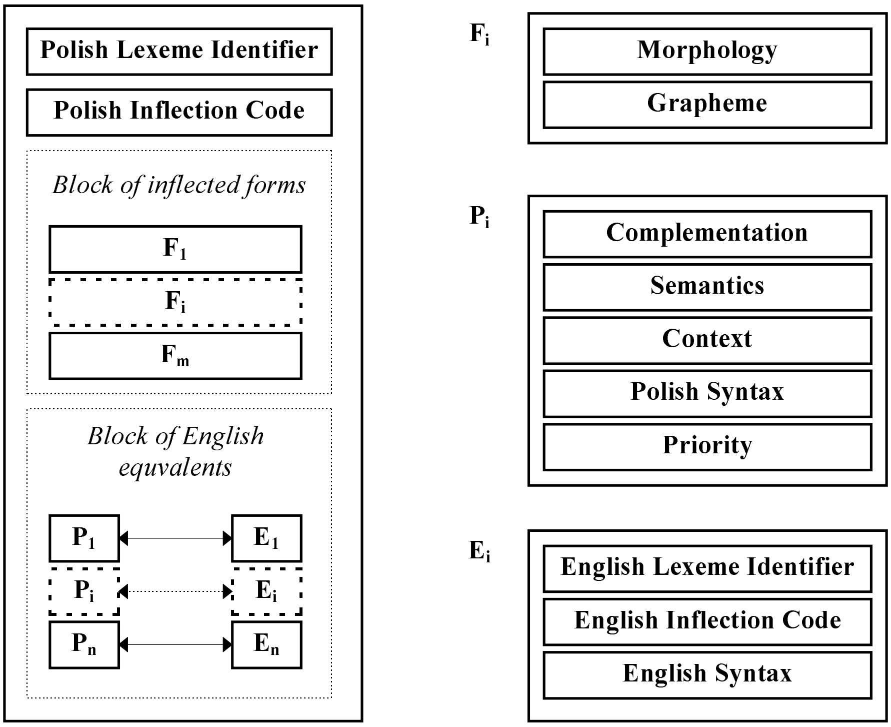


Figure 1. Graphical representation of the entry format.

The characteristics of all the fields (frames) in Figure 1. is given in Section 2.3.

3.4. The representation of the dictionary as an SGML document.

The SGML specification of the dictionary is shown in Figure 2.

```
<!-- POLENG / Polish-to-English Machine Translation / DTD -->

<!ENTITY     % doctype  "POLENG">

<!ELEMENT  POLENG           O  O  (Dictionary | D)>

<!ELEMENT  (Dictionary | D)  -   O  ((Lexeme | L)*)>
<!ATTLIST   (Dictionary | D)  name               CDATA     #IMPLIED
                              version            CDATA     #IMPLIED
                              authors            CDATA     #IMPLIED
```

```
                                        updated           CDATA    #IMPLIED>

<!ELEMENT  (Lexeme | L)         -  O  ((Form | F)*, (Translation | T)*)>
<!ATTLIST  (Lexeme | L)         id                ID       #REQUIRED
                                polishInflection  CDATA    #REQUIRED>

<!ELEMENT  (Form | F)           -  O  (#PCDATA)>
<!ATTLIST  (Form | F)           morphology        CDATA    #REQUIRED>

<!ELEMENT  (Translation | T)    -  O  (#PCDATA)>
<!ATTLIST  (Translation | T)    complementation   CDATA    #IMPLIED
                                semantics         CDATA    #IMPLIED
                                context           CDATA    #IMPLIED
                                polishSyntax      CDATA    #IMPLIED
                                priority          CDATA    #IMPLIED
                                englishInflection CDATA    #IMPLIED
                                englishSyntax     CDATA    #IMPLIED>
```

Figure 2. SGML specification of the dictionary

Further on here, if the logical structure of the dictionary is referred to, the notions: frame, field, field value will be used. If the dictionary will be considered as an SGML document, then the notions: element, attribute, element value, attribute value will be applied. For instance, the frame F_i in the logical structure corresponds to the element *Form* in the SGML document, the field *Morphology* of the frame F_i corresponds to the attribute *morphology* of the element *Form* (value of the field *Morphology* corresponds to value of the attribute *morphology*), and the value of the field *Grapheme* is represented by the value of the element *Form*.

3.5. Characteristics of the fields of the dictionary

3.5.1. Polish Lexeme Identifier

The value of this field is the canonical form of the lexeme. i.e. nominative singular (nominative plural for *pluralia tantum*) for nouns and noun phrases, infinitive for verbs and verbal phrases, masculine, nominative singular for adjectives, etc.

3.5.2. Polish Inflection Code

The value of the field is a code of a ”computer inflection paradigm”. The code must be consistent with the classification of inflection paradigms, prepared beforehand. The code enables automatic generation of inflected forms from the canonical forms without human interference. The first character(s) of the code define(s) what part of speech is represented by the canonical form of the lexeme.

3.5.3. Block of inflected forms

The block of inflected forms may consist of various number of elements. For non-flexional parts of speech the block consists of one element. Nouns (noun phrases) have mostly 14 elements corresponding to all cases of singular and plural. For adjectives the block consists of at most 30 elements including adverbs derived from adjectives (the phenomenon of syncretism is taken into account - the same forms corresponding to different values of case, gender and number are generated only once). Verbs may generate up to 58 inflected forms - all forms of active and passive participles as well as gerunds are generated.

Each element of the block (each frame F_i) consists of a graphemic representation of the form and encoded morphological information.

3.5.4. Block of English equivalents

The block of English equivalents consists of pairs of frames. Each pair characterises one English equivalent of the Polish word (phrase). Frame P_i specifies conditions under which the equivalent described in frame F_i should be chosen. Besides, frames P_i i E_i include other types of information relevant for the correct analysis of a Polish input as well as the correct synthesis of the English output. Beneath, all fields of the frames are listed.

Complementation

The field defines relations between the entry and its subordinate elements (modifiers) in Polish and English expressions. The following assumption is made:

The information stored in the field *Complementation* concerns only modifiers whose compositional transfer would yield incorrect result.

This approach is acceptable only for a bilingual dictionary and transfer approach to translation (as opposed to interlingua approach).

Example 1.

In the sentence *Tłumaczę teksty dla mojego szefa (I translate texts for my boss)* the complement ”dla mojego szefa” may be transferred compositionally, independently of the verb *tłumaczyć*. There will be no need to store the information of this kind of complement in the field *Complementation* for the verb *tłumaczyć.* However, in the sentence *Tłumaczę teksty na język angielski (I translate texts into English)* the complement *na język angielski* cannot be transferred independently of the verb because a translation algorithm would attempt to transfer the Polish PP into ”on English” or ”onto English” depending on which kind of transfer would be assumed as correct for the Polish PP consisting of a preposition „na” and the noun phrase in accusative (correct transfer of PPs is another problem - here it is solved by supplying the information on PPs in the description of prepositions). In order for an algorithm to transfer a phrase *tłumaczyć coś na coś* into *to translate something into something (a language)* it is necessary to insert the appropriate information in the field *Complementation* for the verb *tłumaczyć*.

Further on, the complements which are not amenable to compositional transfer will be called **requirements**.

The field *Complementation* may be non-empty for the following parts of speech (and the corresponding dictionary phrases):

- *verbs (see examples 1, 3),*
- *nouns (see example 2),*
- *adjectives (see example 4),*
- *prepositions (the value of the field denotes the case of a noun phrase with which the preposition forms a prepositional phrase).*

The description of requirements should fulfil the following postulates:

- the activity of inputting the description should not be time-consuming for a lexicographer,
- the description should be easily parsable.

In order to meet the first postulate the grammatical categories are encoded by the shortest possible sequences of characters. In order to realise the second postulate the description is consistent with a context-free grammar.

Below, the code of information stored in the field *Complementation* is expressed in the Backus-Naur notation.

Grammar of the language describing requirements

Non-terminal symbols are given in regular font, terminal symbols are bolded, symbols of the BN metalanguage and comments are italised.

Complementation *::=* ‘’

no requirements

Complementation *::=* Transfer

Transfer *::=* Source_Category**→**Target_Category

for each category of a Polish requirement a category of the corresponding English requirement is given

Transfer *::=* **[** Transfer **]**

brackets embrace (sets of) obligatory requirement(s)

Transfer ::= **(**Transfer **|** Transfer *{* **|** Transfer*}* **)**

disjunction of requirements

Transfer ::= **<** Transfer , Transfer *{*, Transfer*}* **>**

alternative of requirements in a definite order (the translation algorithm should check only for the given order of modifiers in a Polish expression)

Transfer ::= **{** Transfer , Transfer *{*, Transfer*}* **}**

alternative of requirements in a free order (the translation algorithm should check for an arbitrary order of requirements in a Polish expression. However, during the process of generation of the English output the requirements should be arranged in the order consistent with that in the description of the complementation).

Source_Category *::=*

IN | *infinitive, e.g. Ja chcę* ***spać****. (I want to sleep).*

AJ | *adjective, e.g. Stał się* ***nieznośny****. (He became intolerable).*

LC | *locative adverb, e.g. Zostaję* ***w domu*** *(I am staying at home).*

AV | *adverb (other), e.g.* ***Dobrze*** *wyglądasz. (You look good).*

TH | *relative clause starting with the "że" conjunction, e.g. Powiedział,* ***że przyjdzie.*** *(He said he would come).*

BY | *relative clause starting with a "by" conjunction, e.g. Chcę,* ***żebyś przyszedł****. (I want you to come).*

JK | *specific relative clause starting with "jak" conjunction, e.g. Słyszałem,* ***jak mówił****. (I could hear him talking).*

OB | *objective relative clause, e. g. Nie wiem,* ***kiedy przyjdę*** *(I don't know when I shall come).*

DS | *direct speech, e.g. „W porządku", powiedział Jan ("All right", said John).*

Source_Prep_Phrase |

Noun_Object

By convention, all Polish categories which have their „natural" equivalents in English are encoded in accordance with the English name (e.g. relative clause starting with the „że" conjunction is denoted in the same way as the English equivalent: relative clause starting with "that". However, two Polish relative clauses (denoted by „BY" and „JK") do not possess natural equivalents in English.

Source_Prep_Phrase *::=* Source_Prep Noun_Object

Polish Prepositional Phrase is composed of a preposition and a noun object, e.g. in a sentence: On jest dobry ***w strzelaniu*** *(He is good at shooting).*

Noun_Object *::=* Noun_Case

The case of a noun object is relevant in determining Polish complements, e.g. in a sentence: Powiedziałeś ***mi prawdę*** *(You told me the truth), the direct object is in dative, whereas the indirect object is in accusative.*

Noun_Object *::=* Noun_Case:Semantic_Value

Semantics of the noun modifier of a Polish word may be relevant for the determination of its proper English equivalent, e.g. the equivalent of "tłumaczyć coś" (non-human object) is "to explain something", whereas the equivalent of "tlumaczyć kogoś" (human object) is "to excuse someone".

Noun_Object *::=* Noun_Case**-GR**

By convention, Polish nouns denoting activities („equivalents" of English gerunds) are denoted by GR. For the sake of correct analysis of Polish input, the case of a gerund object is specified.

Noun_Case ::= **N** | **G** | **D** | **A** | **I** | **L**

Six cases are allowed for Polish noun objects.

Semantic_Value *::=* Semantic_Feature

Semantic_Value *::=* **-**Semantic_Feature

Semantic_Feature *::=* **Hum** | **Anim** | **Abstr**

The above classification of semantic features is unsophisticated but it proves quite efficient in the case of Polish-English translation.

Polish_Prep *::=* **od** | **do** | **z** | ...

Polish_Prep may be replaced by any Polish preposition or prepositional group.

Target_Category *::=*

TO | *"to" + infinitive, e.g. I want* ***to sleep****.*

IN | *bare infinitive, e.g. I must* ***go****.*

AJ | *adjective, e.g. You look* ***good****.*

LC | *locative adverb, e.g. I am staying* ***at home****.*

AV | *adverb, e.g. He is speaking* ***loudly****.*

RC | *relative clause without a conjunction, e.g. I think **he will come**.*

TH | *relative clause starting with the conjunction "that", e.g. I assume **that he will come**.*

OB | *objective relative clause, e.g. I know **when it happened**.*

DS | *direct speech, e.g. "**All right**", said John.*

ST | *"someone + to + infinitive", e.g. I want **you to come**. In the Polish sentence "Chcę, żebyś przyszedł" the object does not appear explicitly (the object is included in the conjunction "żebyś"). The transfer is encoded as BY→ST. By contrast, in the Polish sentence "Radzę ci, żebyś przyszedł" ("I advice you to come") the object (ci - Eng. you) is explicit and the transfer is encoded as <D→NP, BY→TO>.*

SI | *"someone + bare infinitive", e.g. He made **me cry**.*

GR | *gerund, e.g .I like **swimming**.*

NP | *noun phrase, e.g. I like **books**.*

Target_Prep **NP** *prepositional phrase, e.g.Give it **to me**.*

Target_Prep *::=* **from** | **to** | **with** | **...** *English prepositions*

Semantics

The field describes basic features of nouns (noun phrases) and subjects of activities for verbs (verb phrases). In case of verbs, the field is non-empty only if this kind of information may be relevant for choosing the appropriate English equivalent. This does not seem to be a frequent case. However, the verb *tłumaczyć się* is an example which shows the need for such a solution. If the subject of the verb is human, the English equivalent is *to excuse oneself (*or *to explain oneself).* For non-human subjects (e.g. in a sentence *Ta książka się dobrze tłumaczy - This book translates well*) the best equivalent of the Polish verb is *to translate*.

The admissible values of the field are expressions derived from the symbol *Semantic_value* in the grammar describing the language of the field *Complementation*.

Context

The value of this field is of the form *?Context_value* for polysemic entries for which the value of the field allows the determination of the proper English equivalent or +*Context_value* for characteristic entries which fix the context of the text. For example, in order to determine the correct equivalent for the Polish noun *hasło* consulting context seems necessary. There will exist (at least) two elements in the block of English equivalents in the dictionary: the *Context* field of the equivalent *entry* will have the value *?Lexicography*, whereas the *Context* field of the equivalent *password* will have the value *?Safety*. One of the words which seem to fix the context as *Lexicography* is a noun *słownik* (*a dictionary*) and therefore the value of the *Context* field of the noun *słownik* will be equal to +*Lexicography*.

Designing the full context hierarchy for all entries of the dictionary is an ambitious task and is not the authors' intention. We shall limit ourselves to choosing polysemic entries from the list destined to be included in the dictionary and defining a simple distribution of computational vocabulary into domains in such a way as to enable the separation of meanings for polysemic entries.

Polish Syntax

The field contains information on the type of a phrase (for phrasal entries) and/or reflexivity of verbs.

Priority

The field is non-empty only for the entries which have a few equivalents in English and the values of other fields do not make it possible to determine the most desirable one. The admissible values are natural numbers pointing out the priority of the equivalent (value ‘1’ points out the best equivalent).

English Inflection Code

This field encodes the inflection paradigm of the English equivalent. The code should be designed in such a way as to make it possible for an translation algorithm to work out inflected forms without the necessity of time-consuming consultation of any classification files. The proposal of such a code was described in [Jassem, 1997] and was called *a constructive code*.

English Syntax

This field is non-empty only for English verbs (verbal phrases). The field describes such features as: stativity, reflexivity and transitivity (for phrasal verbs). Let us notice that the value of the field is not independent of the field *Complementation*. For example, when the Polish verb *mysleć* is modified by a relative clause of the type *TH* (e.g. in a sentence *Myślę, że przyjdę - I think I will come*) the English equivalent is a stative form of the verb *to think* (continuous form is not admissible), whereas in other expressions the stativity of the equivalent is not demanded (e.g. in a sentence *Myślę o tobie - I am thinking of you*).

3.6. Examples of the entries into the dictionary

This section presents some examples of the entries into the dictionary in the format shown above.

Example 2.

```
<L  id ="praca" polishInflection="R414">
<F  morphology = "ŻMP">praca</F>
<F  morphology = "ŻDP">pracy</F>
...
<T  complementation="nad I:Abstr→on NP"
    semantics="Abstr"
    context="?Science"
    englishInflection="N00">research</T>
<T  complementation="(o L→on NP | z G→in NP)"
    semantics="-Anim"
    context="?Scient. article”
    englishInflection="N1"> paper</T>
<T  semantics="Abstr"
    context="?Application"
    englishInflection="N00">occupation</T>
<T  complementation="(jako N→as NP | G→of NP)"
    semantics="Abstr"
    context = ?Job
    englishInflection="N1">job</T>
<T  complementation=”{do GR→TO, dla G→for NP}) |
    ({(nad I-GR→of GR | przy L-GR→of GR | przy L→at NP),
    z I→with NP}”
    semantics="Abstr"
    englishInflection="N00">work</T>
<T  semantics = ”-Anim”
    englishInflection = ”N1”>work</T>
</L>
```

Suppose a translation algorithm aims at the correct transfer of the word ”praca” in a text. First, the algorithm should look for the modifiers listed in the *complementation* attribute. This gives the following pairs of equivalents (for better readability examples of complements are given):

Polish noun + Complementation	English Noun + Complementation
praca nad tłumaczeniem automatycznym	research on machine translation
praca o słowniku dwujęzycznym	a paper on a bilingual dictionary
praca z lingwistyki komputerowej	a paper in computational linguistics
praca jako tłumacz	a job as an interpreter
praca tłumacza	a job of an interpreter
praca do wykonania	work to do
praca dla ciebie do wykonania	work for you to do
praca nad (przy) obliczeniem (u) podatku	work of calculating the tax
praca przy samochodzie	work at a car
praca z pomocnikiem	work with an assistant

If in a given text the word „praca” is not modified according to any of the requirements in the dictionary, then the algorithm should try to fit the equivalent to the context of the text. If the context of the text fulfils none of the non-empty context conditions of the dictionary, the algorithm should choose the equivalent with the empty context condition (there should exist at least such an equivalent for each entry). In the case of the word ”praca”, the default ”no-context” equivalent is ”work”.

The value of the attribute *englishInflection* is equal to *N1* for regular nouns and *N00* for massive nouns.

Example 3.

To help the commentary on the entry, the elements of the English equivalents block are enumerated.

```
<L  id ="tłumaczyć" polishInflection="C58N">
<F  morphology = "OP1">tłumaczę</F>
<F  morphology = "OP2">tłumaczysz</F>
...
(1)
<T  complementation="(DS→DS | {A:Ab→NP, D:An→to NP} | {D:An→NP,
   [OB→OB]})
   englishInflection="V1">explain</T>
(2)
<T  complementation="{A: -Ab→NP, z G→>from, na I→into NP}"
    context="?Written translation"
    englishInflection="V1">translate</T>
(3)
<T  complementation="{A:-Ab→NP, z G→from, na I→into NP}"
    context="?Oral translation"
    englishInflection="V1">interpret</T>
(4)
<T  complementation="{[A:An→NP], (z G→for NP | za A→for NP),
    przed I:H→to NP}”
    englishInflection="V1">excuse</T>
(5)
<T  complementation="(DS→DS | {(z G→for NP | za A→for NP),
    (przed I:H→to NP | D:H→to NP)}"
    polishSyntax="refl"
    englishInflection="V1"
    englishSyntax="refl">excuse</T>
(6)
<T  complementation="{z G→from NP, na A→into NP}"
    semantics="-H"
    polishSyntax="refl"
    englishInflection="V1">translate</T>
```

There are 6 elements in the block of English equivalents for the Polish verb *tłumaczyć*. Below, English expressions corresponding to each of the elements are listed:

(1) - direct speech + to explain
- to explain something to someone
- to explain someone + objective relative clause

(2) - to translate something from something into something

(3) - to interpret something from something into something

In order to distinguish between 2) and 3) the determination of context is necessary. Note that the pattern *tłumaczyć coś* fits the above three equivalents. Again, the determination of context may be helpful in choosing the equivalent (if the context does not fit either 2) or 3) the first equivalent (*to explain*) should be taken by the translation algorithm as there is no condition on context in 1)).

(4) - to excuse someone for something to someone

(5) - direct speech + to excuse oneself
- to excuse oneself for something to someone

6) - to translate from something into something - in sentences with non-human subject (e.g. in the sentence *This text translates well from English into Polish*).

Example 4.

```
<L  id ="aktualny"
    polishInflection="P46">
<F  morphology = "MoMPR">aktualny</F>
<F  morphology = "MoMPR">aktualnego</F>
...
<T  priority="1"
    englishInflection="A0" >current</T>
<T  priority="2"
    englishInflection="A0">topical</T>
<T  priority="3"
    englishInflection="A0">up-to-date</T>
```

This example describes the Polish adjective *aktualny* which has a few English equivalents. The formalism did not allow a definition of the conditions for choosing the English equivalent. It was therefore necessary to give non-empty values to the attribute *priority*.

Example 5.

```
<L  id ="liczba całkowita"
```

```
    polishInflection="R414;b-bi P38">
<F  morphology = "ŻMP">liczba całkowita</F>
<F  morphology = "ŻDP">liczby całkowitej</F>
...
<T  semantics="Abstr"
    polishSyntax="attr_phr"
    englishInflection="N1">integer</T>
</L>
```

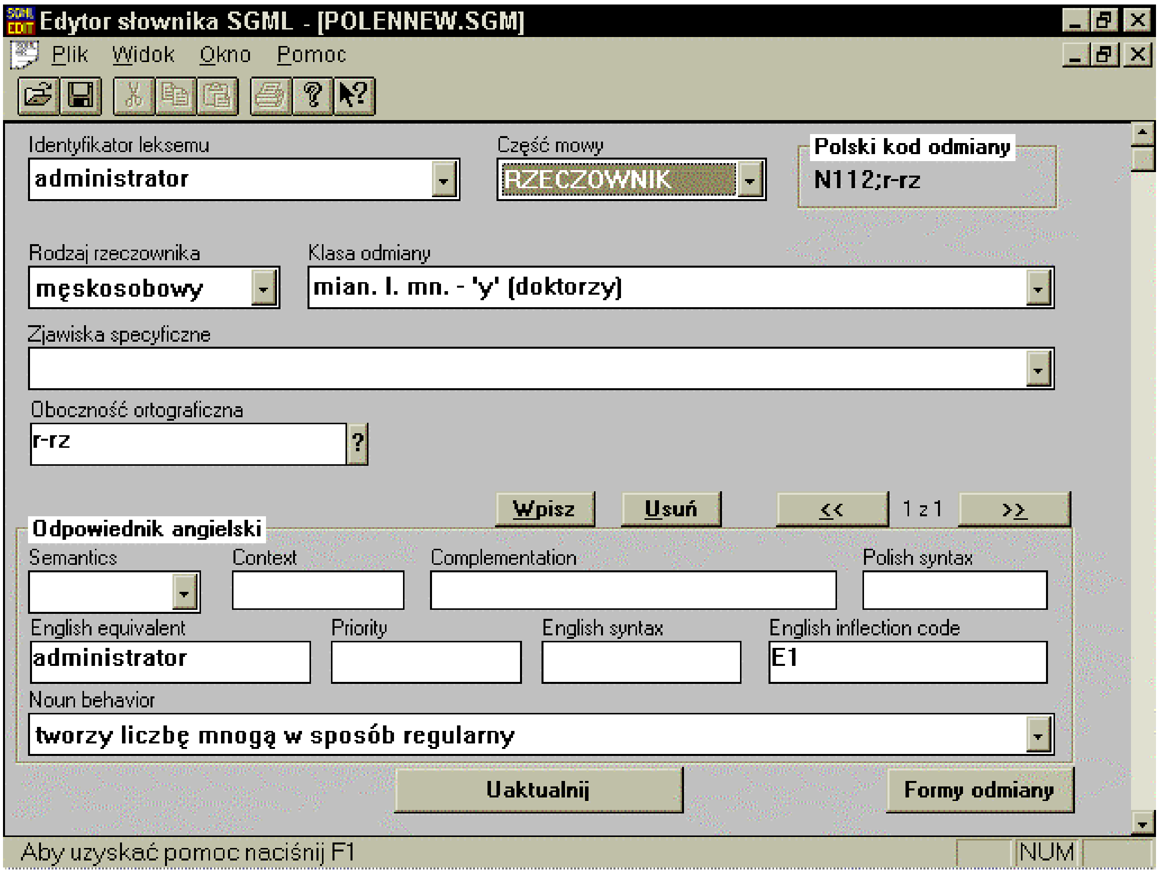
Above an attributive phrase *liczba całkowita* is described. Its English equivalent is a noun *integer*.

4. Dictionary software

The dictionary software consists of two modules: SGML Editor and procedures of binary storage. At present the modules are independent. The SGML Editor aids a lexicographer in creating the textual format (SGML document type) of the dictionary. The binary storage procedures are responsible for converting the textual format of the dictionary into a binary file as well as for the quick access. The procedures of on-line modification of the binary representation have been worked out. In near future it will be possible to link the two modules in such a way that any modification of the textual format (by means of the SGML Editor) will be mirrored by the corresponding modification of the binary representation.

4.1. SGML Editor

The editor enables the lookup and the modification of an electronic Polish-English dictionary stored in a SGML format by means of a graphical interface working under the MS-Windows. The main idea standing the editor may be formulated in the following way: ”let the machine do as much work as it is able to do and let it help the human to do the rest”. The details may be found in [Rutkowski, 1998]. Here, an editing window while inputting a noun is shown.



Edytor słownika SGML - [POLENNEW.SGM]
Plik
Widok
Okno
Pomoc
Wpisz
Usuń
<<
1 z 1
>>
Uaktualnij
Formy odmiany
Aby uzyskać pomoc naciśnij F1
NUM

Identyfikator leksemu: administrator

Część mowy: RZECZOWNIK

Polski kod odmiany: N112;r-rz

Rodzaj rzeczownika: męskosobowy

Klasa odmiany: mian. l. mn. - 'y' (doktorzy)

Zjawiska specyficzne:

Oboczność ortograficzna: r-rz

Odpowiednik angielski

Semantics:

Context:

Complementation:

Polish syntax:

English equivalent: administrator

Priority:

English syntax:

English inflection code: E1

Noun behavior: tworzy liczbę mnogą w sposób regularny

Figure 3. A window of the SGML Editor.

4.2. Binary representation of the dictionary

Storing the dictionary in a binary file aims at optimising the access time. The technology which is described in this section is universal, i.e., it could be applied to electronic dictionaries designed for purposes other than MT.

4.2.1. Structure of lexical data

The structure of lexical data of the dictionary is based on three basic notions: *grapheme*, *form* and *lexeme*.

A grapheme is a dictionary phrase considered as a sequence of characters. The grapheme is identified by the sequence of characters.

A form is a dictionary phrase with its morphological description. A single form is identified by the identifier of the lexeme it belongs to and the morphological description. A single form is represented by one or occasionally more graphemes.

A lexeme is a set of forms which have the same lexical meaning. In the dictionary designed for the purposes of MT a single entry is realised at the level of lexeme.

The possible links between the three levels of the structure are presented at the schema below:

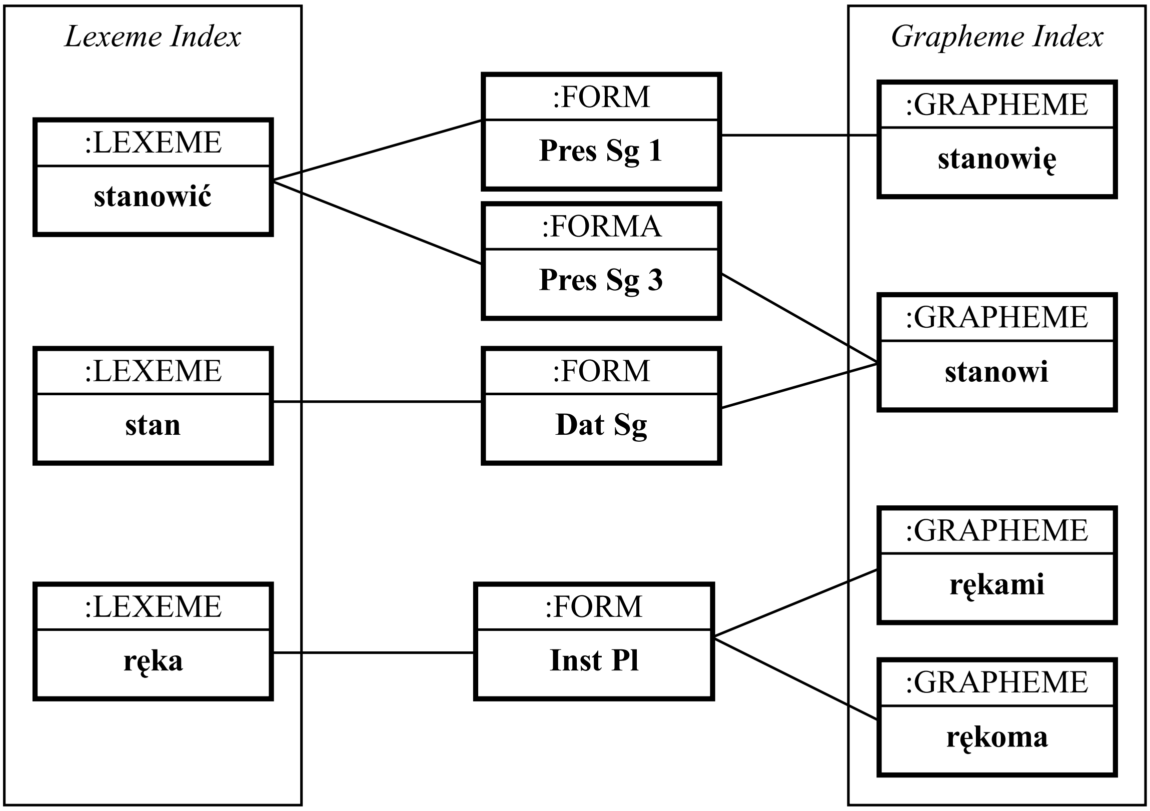


Figure 4. Structure of lexical data in the binary representation of the dictionary

The picture shows that one grapheme may be linked to forms of different lexemes. The grapheme 'stanowi' is the 3rd person, present tense form of the lexeme 'stanowić' (to determine) but it also represents dative singular form of the lexeme 'stan' (a state). One form may be represented by more than one alternative graphemes which is the case for the instrumental, plural form ('rękami', 'rękoma') of the lexeme 'ręka' (a hand).

Note that the structure of the dictionary is independent of its application. The information specific for the application may be attached to each of the levels: lexeme, grapheme and form. In the machine text translation only information on the level of the lexeme is attached: every lexeme has information about its English equivalents. In other applications information concerning the level of forms (such as: colloquial-official) or information concerning graphemes (such as: phonology in spoken output systems) may be relevant.

4.2.2. On-line modification

Experience in applying the dictionary to machine text translation algorithm has proved the need for frequent modification of the contents of the dictionary. In the previous version of the dictionary any minor modification required time-consuming recompilation of the dictionary [Jassem, Lison, Mączyński, 1996]. Therefore a study of designing a structure which would enable on-line edition of dictionary contents has been undertaken [Jassem, Lison, Mączyński, 1997]. The software designed for the dictionary enables insertion, modification and removal of lexemes, graphemes and forms.

4.2.3. Access time

To decrease the look-up time the index of graphemes was implemented as the finite-state automaton. If the grapheme is a single word it is searched for in the word automaton which has the set of Polish characters as its alphabet. If the grapheme is a proper dictionary phrase, it is searched for in the phrase automaton. In the previous version of the dictionary the phrases were represented by character strings. That approach proved ineffective (look-up time is directly proportional to the length of an entry, and the length of a phrase is 'on an average' a few times greater than that of a single word). In this version the process of seeking a phrase is executed 'word by word'. Each word and non-terminal symbol is assigned a unique identifier. A phrase is represented by a sequence of identifiers (the alphabet of the phrase automaton consists of word identifiers).

5. Interrelation between the dictionary and the translation algorithm

The project of a demonstrative translation program based on the dictionary has been worked out. The program will be based on the prototype system described in [Jassem, 1997], but it will use the lexical information available in the dictionary described in this paper. Below, the main points of the interrelation between the dictionary and the translation algorithm are listed:

- A notion of a dictionary phrase is bound to the source language grammar used by the algorithm.
- The dictionary includes only such phrases as would be translated incorrectly if their transfer were left to the algorithm.
- The format of the entry makes it possible for the algorithm to deterministically choose the best English equivalent of a Polish word (phrase) in an expression.
- The format of the information is ”parse-friendly”: the information may be easily converted to a format desired by the algorithm (i.e. Prolog clauses).
- The field *Complementation* enables the correct transfer of modifiers.
- The fields *Complementaton* and *Polish Syntax* enable the proper syntactical analysis of the Polish input.
- The fields *Complemetation* and *English Syntax* enable the correct syntactical generation of English output
- The fields *Semantics,* and *Context* enable simplified semantic and pragmatic analysis.
- The field *English Inflection* enable the correct morphological generation of English output.

6. Evaluation of the dictionary

The dictionary is based on a small corpus of texts (200 000 words). It will contain less than 10000 lexemes. These numbers are too small to claim that the dictionary will cover the vocabulary of any computer-oriented text. We hope that experience gathered while creating this lexicon will be helpful in creating more complete dictionaries designed for MT in the future.

REFERENCES

[Jassem, Lison, Mączyński, 1996] Jassem K., Lison M., Mączyński R. *The Implementation of a Bilingual Electronic Dictionary Designed for Computer Text Translation*) in: C.Basztura, K. Dobrogowska (eds): "Podstawowe problemy komputerowego tłumaczenia różnojęzycznego dialogu w czasie rzeczywistym", Poznań 1996.

[Jassem, Lison, Mączyński, 1997] Jassem K. Lison M. Maczynski R. *The implementation of a bilingual electronic dictionary amenable to on-line modification*, in: W. Jassem, C. Basztura (eds) "Speech and Language Technology”, vol. 1, Poznań 1997.

[Jassem, 1997], Jassem K. *POLENG - a Machine Translation System Based on an Electronic Dictionary*, in: Jassem W. Basztura C. (eds) ”Speech and Language Technology”, Vol. 1, Poznań 1997

[Rutkowski, 1998], Rutkowski B. *The editor of a Polish-English dictionary stored in the SGML format*, in: Jassem W. Basztura C. Jassem K. (eds) ”Speech and Language Technology”, Vol. 2, Poznań 1998

[Vetulani, et al., 1998], Vetulani Z. Walczak B. Obrębski T. Vetulani G. *Unambigous coding of the inflection of Poish nouns and its application in electronic dictionaries - format POLEX*, Wydawnictwo Naukowe UAM, Poznań, 1998