# Evaluation of terminology extraction tools

**TExt for TWIN**
**System Quirk**
**Xerox TermFinder**

**Britta Heidemann, Martin Volk**

**Universität Zürich**
**Institut für Informatik**
**Abteilung Computerlinguistik**

**12. Dezember 1999**

# Contents

### *Definition: Term*

A term is a word that has a certain meaning in a certain subject area which differs from the general meaning (in the subject area Computer Science, the following are terms: *windows, user, account, site, session* or verbs like *connect, disconnect*).
Furthermore, a term is a word which is used only in a certain subject area (*UNIX computer, SUID program*).

Further criteria: The specific meaning of the term is part of the shared knowledge of author and reader – if the reader does not know the term, text understanding is impossible. Therefore, Unix commands such as *ls, pwd* etc. have to be regarded as terms: *If you give the pathname to a directory**, ls** will list the directory but it will not change your working directory. The **pwd** command in the following example shows that.*

## *Description of the systems*

### TExt for TWIN

TWIN (Termbase for Windows):

TWIN is a "multi-user terminology database system developed under Microsoft Access 2.0", "replicated database with management features to administer virtual terminology networks".[1]

Term extraction tool (**TExt**) for TWIN:

"This tool will extract terminology from an input text file, compare it with a TWIN-termbase, and, depending on the selected output options, paste target language equivalents into the text or create a bilingual glossary, or a list of terms not found in the termbase. These features also work with multiword terms. Supported input language is English. Supported input format: Word for Windows and HTML. Supported editor: Word for Windows. Other features, as, for example, support of other input languages or editors may be implemented on a project basis."[2]

TExt Solution to frequent problems in term extraction[3]

Combination of Procedures:
- ⇨ run statistical text analysis on normalised input
- ⇨ run multi-level linguistic filter on statistical results

Analysis Steps
- ⇨ Text analysis and normalisation (tagging and lemmatisation)
- ⇨ Multiword Boundary detection
  - ⇨ Criteria for Multiword boundaries: Formal criteria (sentence/subsentence boundary); Linguistic criteria (e.g. multiword sequences have no modal verb form).
- ⇨ Statistical Analysis of Multiword candidates
- ⇨ Linguistic Filtering of Multiword candidates
  - ⇨ first level filter (legal multiword category sequences)
  - ⇨ second level filter (exclude stop words of a given class: adjectives, prepositions; exclude certain patterns)
- ⇨ Output Generation

Hardware: Pentium-PC, 16 MB RAM, "Fast Hard Disk"[4].

Price (TExt for TWIN): US$ 1800.

Company: triS System Solutions & Services GmbH & Co., 81929 München/ Deutschland

[1] http://www.tri-s.de/twinv11.html
[2] http://www.tri-s.de/tools.html
[3] Gr. Thurmair triS/GMS: Term Extraction from Texts.
[4] http://www.tri-s.de/twinv11.html

## System Quirk

"System Quirk is a flexible package of integrated tools for building and managing term bases".[5]

KonText (Knowledge on Text): "...is a text analysis program that allows you to select one or more text files from a corpus, and to view, edit, print or analyse the text file(s) by performing tasks such as Key Word in Kontext (KWIC), Weirdness, wordlists and indexes within user defined constraints."

Weirdness: "The task Weirdness creates a statistical approach to terminology extraction from text by comparing the relative frequencies of words that occur in specialist texts with their relative frequencies in general language corpora."

Ferret: "Ferret is a text analysis program that allows you to analyse a number of texts specifically for compound terms. It makes use of stoplists to create term boundaries and also allows you to create/remove lists of known terms for future analysis and also for use withing other applications such as KonText. Term lists are created by considering those compounds that appear between words in the stoplists."

ColloQuator: "A collocation statistics application. Provides statistical evidence for collocations. The full version of this application can be seeded with a number of tokens/terms and will return statistical information about terminess for compounds."

Company: School of EEITM, University of Surrey, Grossbritannien

Hardware: There are two versions of System Quirk available - the research (Unix) version and the PC version for Windows 95/98/NT (Minimum Pentium 166MHz processor, 10MB disk space, 32MB RAM)

Price of the PC version: £ 109 (Academic Institutions) resp. £ 219 (Commercial Institutions).

[5] http.//www.computing.surrey.ac.uk/ai/SystemQ/links2.html

## Xerox TermFinder

TermFinder, one part of the Xerox Terminology Suite, is supposed to help collecting the data then managed by the Term Organizer. Other components of the system are Term Checker (consistency check) and Xerox Multilingual Assistant (translation of words in context).

"This tool builds terminology databases from existing texts and documents. It extracts terms and provides a list that can be verified afterwards by a terminologist. It can be used with a single language or two, providing the equivalent translation and word in the target language for a source-language term. Using Xerox TermFinder with Xerox TermOrganizer results in large productivity gains and ensures accuracy and consistency in documents and aid in the creation of customised glossaries.
Xerox Term Finder doesn't have any direct key competitor. Only Xerox achieves bilingual linguistic multiword term extraction: no other terminology tool offers fully integrated source text analysis and automatic extraction tools to identify new terms. Tests at XRCE suggest Xerox's tool is eight time faster and cheaper than traditional manual methods of building terminology."[6]

Sentences of the source and target documents are aligned automatically and can be checked manually. The terminology data base is collected via monolingual or bilingual term extraction.

"Xerox Term Finder uses XeLDA, the unique Xerox linguistic engine. XeLDA is the result of twenty years of research in linguistics at the Palo Alto Research Center (PARC) and at the Xerox Research Centre Europe (XRCE). XeLDA uses very sophisticated computer entities called "transducers" that embed linguistic information in a very efficient, compact, language-independent way. XeLDA enables the Xerox TermFinder engine to quickly align sentences and find terms in Documents."

"Built on top of Open Database Connectivity (ODBC), Microsoft`s database independent layer, Xerox TermFinder is independent of any specific database. Xerox TermFinder supports SGML, HTML,XML, iso-8859-1 and RTF documents."

Hardware: Pentium-II-PC, 96MB RAM, 100 MB hard disk.

Price: ~ 20.000 $ (for the complete Xerox Multilingual Suite)

Company: Xerox Multilingual & Knowledge Management Services, Meylan/France.

[6] http://www.mkms.xerox.com

## *Evaluation*

### Procedure

1. Automatic extraction of terms from selected chapters of two UNIX books[7] by System Quirk (Unix Version), TExt/ Tris, TermFinder/ Xerox.

   The texts were HTML-texts. Ignoring the HTML tags they contained 17082 words. Each system automatically extracted one or two lists with items regarded as possible terms: Whereas Xerox TermFinder provided one list, System Quirk‘s results from the KonText and Ferret programs (see the system description above for details) were in the form of two lists with single items resp. multiwords. For the TExt system, we evaluated the list with items found in the TWIN-termbase (collected in a bilingual glossary) as well as the list with extracted items that were not found in this termbase.

   N.B.: We only evaluated the output (the extracted items) of the systems. We did not judge the user friendliness or the architecture of the three systems.

2. Classification of the extracted items according to a list of categories and patterns (see below for details on establishing these categories).

   Beginning with the extracted items Xerox TermFinder listed, we manually worked through the lists and annotated a category marker to each item in the list. Each item that we regarded as term was marked with a +. When finished with the lists of all three systems, a Perl-script was used to count the terms and the number of items in each category. In this report you find the tables with the results, furthermore you may want to have a look at the lists with all terms each system found in the appendices to this report.

3. Computation of “precision and recall”.

   Precision: How many of the automatically extracted items can be regarded as terms? The number of manually marked terms was counted with the help of a Perl-script. We then calculated the percentage of terms within the lists of extracted items each system provided.

   Recall: How many terms in the given texts have not been found by the system? In the source texts (the Unix book chapters) all those terms found by one or more of the three systems were automatically marked as terms. We then manually worked through the chapters and added markers to all other items that were terms in the context of this subject area. With a Perl-script the terms in the texts were compared to the list of terms each system had found.

[7] Jerry Peek, Tim O’Reilly & Mike Loukides: UNIX Power Tools. Second Edition, August 1997 (1120 Seiten). Daniel Gilly & the Staff of O’Reilly & Associates: UNIX in a Nutshell: System V Edition. Second Edition, June 1992 (444 Seiten).

## Categories and Patterns

1. Noun (N)
2. Noun + Noun (N2)
3. Noun + Noun + Noun (N3)
4. Nounphrase + Preposition (*of*) + Nounphrase (NON)
5. Adjectiv/Adverb (A)
6. Adjectiv (or Participle) + Noun (AN)
7. Adjectiv + Noun + Noun (AN2)
8. Verb (V)
9. Verb + Verb (V2)
10. Verb + Noun (VN)
11. Verb + Noun + Noun (VN2)
12. Abbreviations (K)
13. Abbreviation + Noun (KN)
14. Other Combinations of Words (C)
15. Unix Commands (B)
16. Command + Noun (BN)
17. Noun + Command (NB)
18. Keyboard Shortcuts (T)
19. Computercode (PC)
20. Webaddresses (W)
21. Filenames (D)
22. Names, Institutions (P)
23. Wrong segmentation (S)

The categories and patterns above were partly established in advance, partly during the evaluation. The categories containing word classes (nouns, adjectives and adverbs), had been defined beforehand, for example. Others - such as Unix commands, noun plus command, filenames etc. - became necessary in the evaluation process.

## Decisions in special cases

Homographs: A word was assigned to the class occuring most often in the text. In the list provided by Xerox TermFinder, for example, *answer* was classified as noun (occuring 3 times in the text) and not as verb (1 time), but *look* as verb (5 times) and not as noun (2 times).

Verbs/Commands: Whenever a verb might be a Unix command, it was classified as such (e.g. *find, open, undelete* etc.) – even if it occurred within the text as verb and not as command. The decision was determined by the fact that in this subject area the verb does have a special meaning.

Capital letters: In general, words beginning with capital letters were not marked as terms. An exception was made for proper nouns (names of systems and progams etc.): Both *emacs* **and** *Emacs* , for example, were classified as term.

Plural/Singular: If there was no lemmatization routine involved in the term extraction process, often both plural and singular were extracted. System Quirk, for example, counted 16 times the multiword term *relative pathname*, five times the plural multiword *relative pathnames* (another example: *window system* (5 times) and *window systems* (3 times)). In general, the singular form was counted as a term. Due to this, some words were not counted as terms since they occured only as plural form.

### *Results 1 – The extracted items*

Number of items extracted by each system

| | TExt | Quirk | Xerox |
|---|---|---|---|
| # of items | 2237* | 1843** | 2070 |

*TExt found 113 words in its termbase (which contained 4800 entries) – the system produced a glossary containing these words and their respective translations in a bilingual list. In addition, the system regarded 2124 unknown items as probable terms.
**Of which 329 items are multiwords.

Classification of all extracted items

| Category | TExt | Quirk | Xerox |
|---|---|---|---|
| a (adjective) | 112 | 167 | 21 |
| an (adj. + noun) | 185 | 61 | 328 |
| an2 (adj. + two nouns) | 47 | 6 | 60 |
| b (command) | 56 | 57 | 83 |
| bn* (command + noun) | 2 | 5 | 5 |
| c (other combination) | 268 | 50 | 137 |
| d (filename) | 60 | 70 | 108 |
| k (abbreviation) | 82 | 66 | 60 |
| kn* (abbrev. + noun) | 11 | 11 | 12 |
| n (noun) | 278 | 537 | 475 |
| n2 (two nouns) | 173 | 136 | 213 |
| n3 (three nouns) | 21 | 14 | 36 |
| nb (noun + command) | 11 | 1 | 9 |
| nk (noun + abbreviation) | 7 | 2 | 4 |
| non (prepositional phrase) | 467 | 0 | 70 |
| p (name) | 28 | 38 | 13 |
| pc (computercode) | 66 | 148 | 90 |
| s (segmentation error) | 219 | 54 | 209 |
| t (keyboard shortcut) | 7 | 2 | 8 |
| v (verb) | 84 | 403 | 45 |
| v2 (two verbs) | 0 | 0 | 3 |
| vb (verb + command) | 0 | 0 | 6 |
| vn (verb + noun) | 42 | 12 | 51 |
| vn2 (two verbs) | 4 | 2 | 17 |
| w (webaddress) | 2 | 0 | 3 |
| **extracted items** | **2237** | **1843**[8] | **2070** |

[8] number of evaluated terms (down to Frequency Ratio 1).

**Comments:**

Category “other combinations of words” (c): In the TExt system, a high number of verb phrases were classified as c, which is the reason for the gap between TExt and the two other systems in this category.

Category “prepositional phrases” (non): While Xerox` TermFinder extracted only those prepositional phrases containing the preposition *of*, TExt also recognized phrases containing *from, with, at* etc. as possible terms.

Category ”verb” (v): The words extracted by System Quirk contained a high number of participles, which were classified as verbs, due to missing context. As the lists extracted by the two other systems are concerned, those participles probably are part of the multiwords classified as an (adjective or participle plus noun).

Wrong segmentation (s): System Quirk produced comparably less segmentation errors than the two other systems. Both Xerox TermFinder and TExt seem to have problems with HTML tags within the texts: Often two words were directly attached to each other in the list of extracted items that were in the text separated by an HTML tag – obviously those tags should be replaced by a space to ensure proper segmentation and tokenization of the text.

## Further Observations

### Xerox TermFinder

Lemmatization sometimes produced unwelcome results. For instance, the system mistakingly performed the following reductions:
*operating system => operate system*
or
*stopped jobs => stop jobs*

TermFinder delivered a high number of multiword combinations that contained general purpose adjectives such as *different, good, last, many, new, old, same, simple, special* plus noun (phrase):

*simple alias*
*simple command*
*simple command line*
*simple design*
*simple method*
*simple pipeline*
*simple type error*

*old command*
*old content*
*old directory entry*
*old version*

Such combinations are most likely no terms. They should be sorted out with the help of a stop list.

**TExt for TWIN**

The system often produced wrong word forms due to lemmatization of adverbial participles:
*line-oriented => line-orient*
*berkeley-based (mechanism) => berkeley-base*
(This is not true in all instances: While Xerox' TermFinder counted nine times the phrase *delete file*, TExt counted eleven times *deleted file*.)

For the TExt system, category *c* (other combinations of words) contains more verb phrases than the same category for the other two systems. For example:
*performing non-sequential output*
*plan to use into external variables*
*print character after character*

As with the Xerox system, there were a high number of multiword combinations that contained adjectives such as *common, different, new, old, own, simple* plus noun. Again, these are no likely terms:

*common beginner*
*common control character*
*common login name*
*common program*
*common type*

*new application*
*new directory*
*new entry*
*new file*
*new group*
*new name*

The TExt system extracted a higher number of prepositional phrases than the other systems containing *from, with, at* etc., e.g.:

*new directory with its absolute pathname*
*password for this group*
*pathname to the play directory*
*private shell file in each project*

There seems to be no lemmatization routine involved in the collection of terms for the glossary list: Evidence for this assumption is the high number of different verb forms and plural nouns in this list. (Plural nouns were not marked as terms!)
Furthermore, the translations found in the glossary often were wrong:

*PC => pC – parallel C*

Sometimes homonyms were translated as belonging to the "wrong" (unusual) word class in this subject area:

*links => verknüpfen*
*loop => schleifen*

Both words were manually classified as nouns by the evaluator.

**System Quirk**

The single word list was sorted according to the so-called “frequency ratio”, which compares the frequency of a word within a general corpus with its frequency of occurrence in a special subject area. Those words that have a high “frequency ratio” belong to the specialized vocabulary. Words with a low frequency ratio are not supposed to be terms (except for those words with a special meaning parallel to the general one such as *window, system, network*), therefore we cut the list of possible terms below frequency ratio = 1. That means: Only 1517 of 2278 extracted single word items have been classified, since we did not assume to find any terms within those with a frequency ratio less than 1.

The list contained a number of words that could not be determined with respect to our categories (e.g. conjunctions and prepositions such as *via, below, if...*), so they were assigned to category c (other combinations of words). Pronouns were classified as n (Noun). The high number of participles were counted as verbs, therefore System Quirk has a high score in the category “verb” with respect to the two other systems. (There, those particples often occurred in noun phrases and therefore were marked as adjective (category an)).

The list with multiword combinations contained no verb or prepositional phrases, but only noun phrases. This is probably due to the usage of stop lists.

## Results 2 – Classified terms

Number of terms according to category

| **category** | **TExt** | **Quirk** | **Xerox** |
|---|---|---|---|
| a (adjective) | 6 | 7 | 1 |
| an (adj. + noun) | 23 | 9 | 28 |
| an2 (adj. + two nouns) | 4 | 0 | 5 |
| b (command) | 21 | 31 | 35 |
| d (filename) | 1 | 4 | 4 |
| k (abbreviation) | 12 | 17 | 15 |
| kn* (abbreviation + noun) | 6 | 2 | 9 |
| n (noun) | 68 | 116 | 125 |
| n2 (two nouns) | 100 | 55 | 116 |
| n3 (three nouns) | 8 | 8 | 12 |
| nb (noun + command) | 4 | 1 | 5 |
| nk (noun + abbreviation) | 3 | 1 | 1 |
| p (names) | 0 | 2 | 0 |
| pc (computercode) | 0 | 3 | 0 |
| non (prepositional phrase) | 3 | 0 | 0 |
| t (keyboard shortcut) | 7 | 1 | 7 |
| v (verb) | 13 | 37 | 3 |
| vn (verb + noun) | 1 | 3 | 6 |
| vn2 (verb + two nouns) | 0 | 0 | 2 |
| **# of all terms** | **284** | **297** | **375** |
| # of extracted items | 2237 | 1843[9] | 2070 |
| **percentage of terms within the list of extracted items** | **12,69%** | **16,11%** | **18,1%** |

The numbers in the table above show the distribution of categories within all manually classified terms of the extracted items.
For example, Xerox TermFinder extracted 2070 items. Out of these 375 were regarded as terms by the evaluator. That means that only 18,1% of all extracted items were terms. The numbers in the above table show the distribution of the terms over the categories. It can be seen that mostly nouns and noun phrases were regarded as terms.
The percentages at the bottom of the table show the so-called “precision” of each system in term extraction: How many of the automatically extracted items in the list are terms?

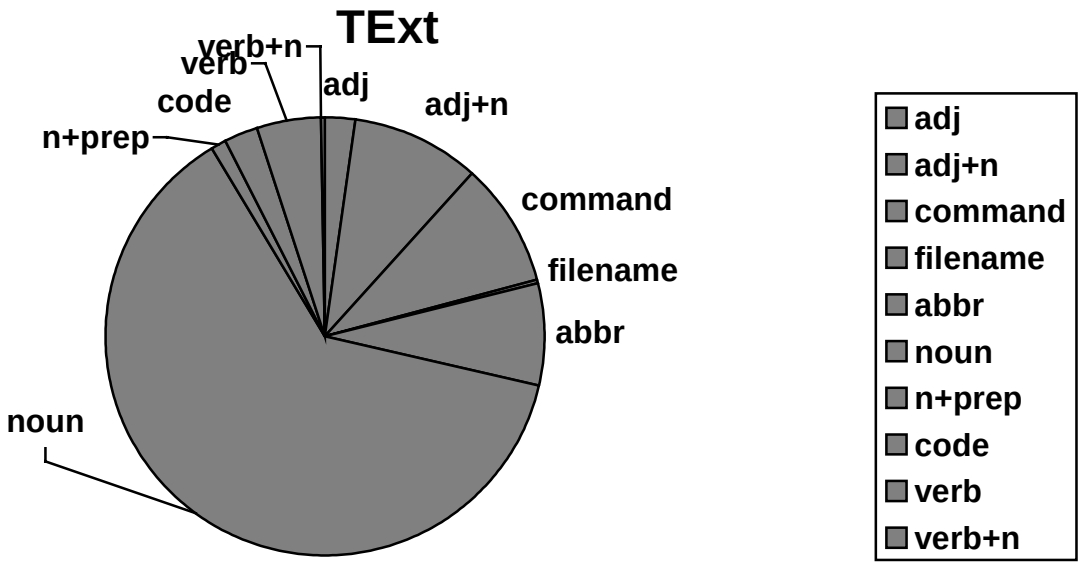
[9] number of extracted terms (down to Frequency Ratio 1).

Percentage of terms according to category

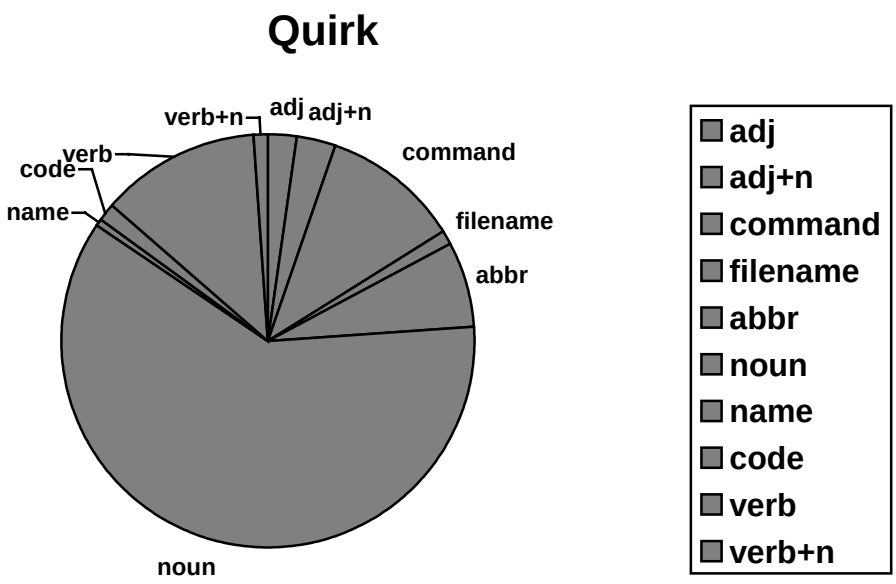
NB: The category s is not shown since it does not contain any terms.
NB2: All numbers are percentages.

| | TExt | Quirk | Xerox |
|---|---|---|---|
| a (adjective) | 2,11 | 2,35 | 0,27 |
| an/an2 (adj. + noun) | 9,51 | 3,03 | 8,80 |
| b/bn/nb/vb (command + combination) | 8,80 | 10,77 | 10,67 |
| d (filename) | 0,35 | 1,01 | 1,07 |
| k/kn/nk (abbreviation + combination) | 7,39 | 6,73 | 6,77 |
| n/n2/n3 (nouns) | 61,97 | 60,27 | 67,47 |
| non (prepositional phrase) | 1,05 | 0 | 0 |
| p (names) | 0 | 0,67 | 0 |
| pc/t/w (computer code, keyboard shortcut, webaddress) | 2,46 | 1,34 | 1,86 |
| v (verb) | 4,58 | 12,46 | 0,80 |
| vn/vn2 (verb+ noun) | 0,35 | 1,01 | 2,13 |

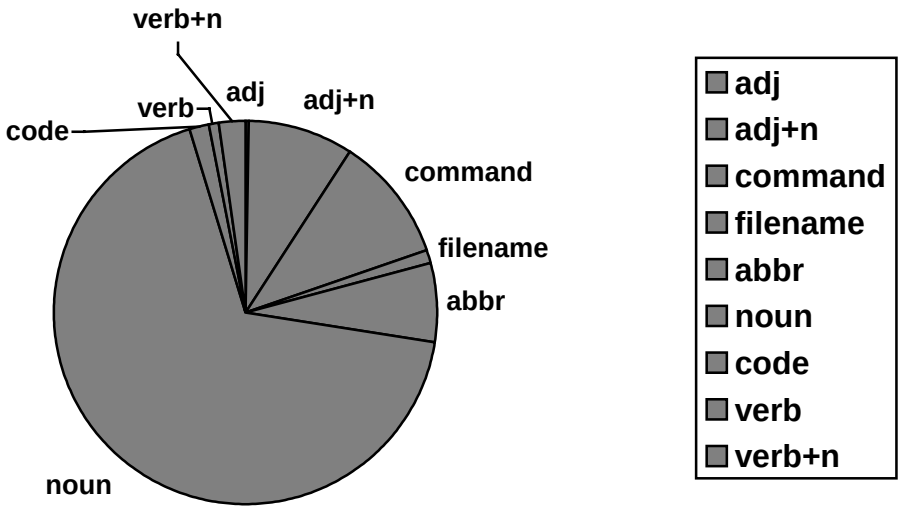
The numbers in the table above show the distribution of categories within all manually classified terms in percentages. We have grouped similar categories in order to get a clear picture over general tendencies. For example, for Xerox TermFinder we have noted that 375 extracted items were regarded as terms. Out of these 375 terms 67,47% were nouns or noun sequences, 10,67% were command names or combinations of commands and nouns or verbs, 8,80% were adjective noun combinations and so on. On the following pages, you can see those numbers in a diagram for each system with further explanations on the program‘s performance.




In the list of extracted terms provided by the TExt system, Unix commands or command plus noun combinations make up for 9,25% of all terms, followed by 8,37% adjectives and adjectives plus nouns.




In addition to the nouns the terms provided by System Quirk consist to a high percentage of verbs (12,46%), followed by commands (plus nouns) (10,77%), and abbreviations (6,73%).




The terms extracted by Xerox TermFinder consist of a high amount of nouns – and additionally of commands plus nouns (10,67%), adjectives plus nouns (8,80%), and abbreviations plus nouns (6,77%).

## Percentage of terms within each category

The table below shows how many items in each category within the list of extracted items have been considered as terms by the evaluator. (I.e., how many of them have been marked with a +.) All numbers are percentages. For example, 37,13% of all nouns (n/n2/n3) that have been extracted by the TExt system were classified as terms, but only 5,35% of all adjectives.

NB: The categories s and p are not shown since they do not contain any terms.

| | **TExt** | **Quirk** | **Xerox** |
|---|---|---|---|
| a (adjective) | 5,35 | 4,19 | 4,76 |
| an/an2 (adj. + noun) | 11,58 | 13,43 | 8,52 |
| b/bn/nb/vb (command + combination) | 36,76 | 72,73 | 38,83 |
| d (filename) | 1,67 | 4,35 | 3,70 |
| k/kn/nk (abbreviations + combination) | 21,00 | 25,32 | 32,89 |
| n/n2/n3 (noun) | 37,13 | 26,06 | 34,94 |
| non (prepositional phrase) | 0,64 | 0 | 0 |
| p (names) | 0 | 5,26 | 0 |
| pc/t/w (computer code, keyboard shortcut, webaddress) | 9,33 | 2,67 | 6,93 |
| v (verb) | 15,47 | 9,18 | 6,67 |
| vn/vn2 (verb+ noun) | 2,17 | 21,43 | 11,76 |

**Comments**
Generally speaking, the percentage of terms is highest among expressions consisting of one or more nouns or combinations of commands or abbreviations and nouns (categories b, k, n). Noun phrases plus prepositional phrases (non) are obviously no reliable pool for terms – even in the high number of multiwords in the category “prepositional phrase” provided by the TExt system only 0,64% were terms.

### *Results 3 - Recall*

The so-called “recall” answers the question of how many terms have not been found by the system in the given source text.

The numbers in the table below were collected via the following procedure: In the source texts (the Unix book chapters) all thoses terms found by one or more of the three systems were automatically marked with <t> at the beginning and </> at the end of each word or phrase. We then manually worked through the chapters and added markers to all other items that were terms in the context of this subject area. With a Perl-script the list of all terms in the texts was compared to the list of terms each system had found.

| | **TExt** | **Quirk** | **Xerox** |
|---|---|---|---|
| # of all terms in the texts (types) | | 586 | |
| # of found terms (types) | 284 | 297 | 375 |
| # of terms (types) not found | 302 | 289 | 211 |
| recall in % | 48,46% | 50,68% | 63,99% |

The total number of terms in the texts was 586 – if all 586 terms had been found by one of the systems, its recall would have been 100%. Neither of the systems reached this number. As can be seen in the table above, Xerox TermFinder by far reached the best recall score with 63,99%. Quirk and TExt reached comparable scores of 50,68% resp. 48,46%.

The list of terms not found by the TExt system contained multi- and single-word expressions of all categories, whereas System Quirk had problems finding multiword expressions. Often parts of the multiwords could be found in the list of extracted words (e.g. the abbreviation *ID*), but not the whole expression (e.g. *user ID*).

The contrary is true for Xerox` TermFinder: The list of not found terms contained many single-word expressions, most often verbs (such as *connect, quit, print, execute* etc). This is due to the fact that the system does not regard/count verbs as terms.

For more detailed information, please compare the lists of all terms in the source text and terms found by the three systems in the appendices.

## *Conclusions*

### Precision and Recall

As can be seen in the table above, Xerox TermFinder has reached the highest percentages of precision (number of terms in the extracted list) and recall (number of terms found with respect to possible number), followed by System Quirk and TExt.

| | **TExt** | **Quirk** | **Xerox** |
|---|---|---|---|
| **Precision** | **12,69%** | **16,11%** | **18,11%** |
| **Recall** | **48,46%** | **50,68%** | **63,99%** |

It should be noted that an evaluation of term extraction tools neccessarily suffers from a range of subjectivity factors. First and foremost, it is not clearcut what counts as a term. Second it is difficult how to value morphological variations (plural forms) or spelling variations (upper case vs. lower case spelling). Therefore the figures should not be taken as absolute values but rather as indications of comparative merits. The figures indicate that the three evaluated systems fare rather similar with regard to precision. The Xerox TermFinder has slight advantages since it focuses on noun phrases which account for the majority of phrases (at least in the subject area of Computer Science). All systems produce more than 80% noise in automatic term extraction. We believe this could be reduced by half by subject specific filters that use statistical and linguistic clues. Surprisingly no system reached more than 70% recall. Too many specially structured terms were missed by all of them.

### Comparison of extraction methods

| | **TExt** | **Quirk** | **Xerox** |
|---|---|---|---|
| Use of a data base during the evaluation procedure | yes (=> creation of a glossary) | no | no |
| Lemmatization | yes (but not in the glossary) | no | yes |
| Frequency of extracted items | yes | yes | yes |
| Ranking of extracted items | no | yes | no |
| Verbphrases as terms | yes | no | ? |
| Prepositional phrases as terms | yes | no | yes (but only those with *of*) |
| Context examples | yes | no | yes |
| Bilingual term extraction | no | no | yes |

## Which system for which task?

Xerox TermFinder got the best scores for precision and recall and is the only system that extracts bilingual term lists from bilingual corpora. And still lacks some important features: It does not “rank” the extracted words with respect to their “terminess”. If the user is especially interested in verb or prepositional phrases, he might be disappointed, since TermFinder only extracts special types of those phrases.

The advantage of System Quirk is its “ranking” feature – the user can rely on the fact that the extracted items with a high frequency ratio have a high “terminess”. On the other hand, System Quirk does not extract verb or prepositional phrases at all and does not provide any contextual examples for the items extracted.

Besides the use of a data base during extraction (and the collection of known terms in a bilingual glossary), the TExt system offers special features that make the system attractive even if the user considers the scores for precision and recall the system received. For example, the system extracts verb phrases and prepositional phrases including those with prepositions like *from, with, at* etc., uses a number of lemmatization features during term extraction and gives context examples for the extracted terms.

**Appendix 1**
**List of all terms**

.c
.cshrc
.exrc
.login
.o
.profile
.txt
ASCII
AT&T-based system
BSD
Berkeley UNIX
Berkeley group mechanism
Berkeley-based mechanism
Berkeley-based version
Bourne-type shell
C
C shell
C shell manual page
CD-ROM
CTRL
CTRL-C
CTRL-D
CTRL-H
CTRL-Q
CTRL-S
CTRL-U
Command-Line argument
DOS
DOS utility
Emacs
Emacs backup file
File Access Permission
Filesystem Permission
Filesystem permission bit
GNU version
I/O
ID
INFINITY 6
Linux
MS-DOS
Macintosh
Multitasking
NFS
PC
RFS
SGID
SGID bit
SGID file
SGID program
SUID
SUID bit
SUID file
SUID program
Shell Prompt
Solaris
UID
UNI
UNIX
UNIX Account
UNIX Environment
UNIX account
UNIX book
UNIX command
UNIX computer
UNIX computer system
UNIX directory
UNIX directory tree
UNIX environment
UNIX filesystem
UNIX filesystem tree
UNIX kernel
UNIX operating system
UNIX power tool
UNIX program
UNIX session
UNIX shell
UNIX shell prompt
UNIX system
UNIX user
UNIX version
UNIX wildcarding
VMS
VMS system
Vi
[BACKSPACE]
[CTRL-H]
[DELETE]
absolute pathname
access
access bit
access file
access mode
access permission
access privilege
access time
accidental file deletion
account
account name
account use
administrative tool
administrator
alias
align
amount of storage
application
argument
ascii
asterisk
authorized system user
automatic setup
backup
backup file
backup tape
bash
berkeley
berkeley group mechanism
berkeley mechanism
berkeley unix
binary
binary character
bit

block
bourne shell
browse
bug
byte
c
cancel a command
capability
capability value
cat
cd
character
chgrp
child directory
chmod
chown
clear
code
command
command date
command exit
command file
command interpreter
command line
command logout
command name
command ps -aux
command tool
command-line argument
compile
compiles
compound command
computer
computer network
computer system
computing
connect
control character
control code
copy file
cron
crontab
crontabs
csh
cshrc
ctrl
current directory
current working directory
cursor
daemon
data
data block
data file
database
default
default group
default path
default rm
default search path
default system
default umask
default value
delete
deleted file
digit
dir
directory
directory entry
directory level
directory location
directory name
directory permission
directory tree
directory user
disconnect
disk
disk block
disk drive
disk partition
display
display format
ditroff
dos
drive
driver
dump
edit
editor
electronic mail
emacs
emacs editor
email
end
enter
entomb
entry
environment
environment variable
erase
erase character
error message
escape
escape sequence
executable
executable file
execute
execute access
execute permission
executes
exit
expression
expunge
external variable
family tree
figure
file
file access
file access permission
file bar
file content
file datum
file datum block
file erasure
file group
file name
file owner

**Appendix 2**
**List of all terms found by the 3 systems (The number indicates by how many of the systems a term was found)**

1 .c
1 .cshrc
1 .exrc
1 .login
1 .o
1 .profile
1 .txt
1 ASCII
1 AT&T-based system
2 BSD
1 Berkeley UNIX
2 Berkeley-based mechanism
2 Bourne-type shell
2 C
1 CD-ROM
2 CTRL
2 CTRL-C
2 CTRL-D
2 CTRL-H
2 CTRL-Q
2 CTRL-S
2 CTRL-U
1 Command-Line argument
1 DOS
2 DOS utility
2 Emacs
1 File Access Permission
1 Filesystem permission bit
1 GNU version
2 I/O
2 ID
1 INFINITY 6
2 Linux
1 MS-DOS
2 Macintosh
1 Multitasking
2 NFS
2 PC
2 RFS
2 SGID
1 SGID bit
2 SGID file
2 SGID program
2 SUID
1 SUID bit
2 SUID file
2 SUID program
1 Solaris
1 UID
2 UNIX
1 UNIX Account
1 UNIX Environment
1 UNIX account
1 UNIX book
2 UNIX command
2 UNIX computer
2 UNIX computer system
2 UNIX directory
2 UNIX directory tree
2 UNIX environment
2 UNIX filesystem
2 UNIX kernel
1 UNIX power tool
1 UNIX program
2 UNIX session
1 UNIX shell
1 UNIX shell prompt
2 UNIX system
2 UNIX user
2 UNIX version
1 UNIX wildcarding
1 VMS
2 VMS system
1 Vi
3 absolute pathname
2 access
1 access bit
1 access file
3 access mode
1 access privilege
1 access time
2 accidental file deletion
2 account
2 account name
2 account use
3 administrator
3 alias
1 align
1 amount of storage
1 application
1 argument
1 ascii
3 asterisk
1 authorized system user
1 automatic setup
3 backup
1 backup file
3 backup tape
2 bash
2 berkeley
2 berkeley group mechanism
2 berkeley mechanism
1 berkeley unix
2 binary
2 bit
1 block
1 bourne shell
3 browse
3 bug
2 byte
2 c
2 capability
1 capability value
2 cat
1 cd
3 character
3 chgrp
2 child directory
2 chmod

2 chown
1 clear
2 code
2 command
1 command date
2 command exit
2 command file
2 command interpreter
2 command line
2 command logout
2 command name
2 command ps -aux
2 command tool
1 command-line argument
1 compile
1 compiles
2 compound command
2 computer
3 computer network
3 computer system
1 computing
1 connect
2 control character
2 control code
1 copy file
2 cron
2 crontab
1 crontabs
2 csh
1 cshrc
1 ctrl
2 current directory
1 current working directory
3 cursor
3 daemon
1 data
1 data block
2 data file
2 database
2 default
3 default group
3 default path
2 default rm
2 default search path
1 default system
2 default umask
3 default value
2 delete
2 deleted file
2 digit
3 dir
3 directory
3 directory entry
2 directory level
3 directory location
2 directory name
1 directory permission
3 directory tree
2 directory user
2 disconnect
2 disk
2 disk drive
2 disk partition
2 display
2 display format
1 ditroff
1 dos
2 drive
1 driver
2 dump
2 edit
2 editor
1 electronic mail
2 emacs
1 emacs editor
3 email
1 end
1 enter
2 entomb
2 entry
2 environment
1 environment variable
2 erase
1 erase character
3 error message
1 escape
1 escape sequence
1 executable
3 executable file
1 execute
1 execute access
1 executes
3 exit
2 expression
2 expunge
1 external variable
3 family tree
2 figure
2 file
2 file access
1 file access permission
2 file bar
1 file content
1 file datum
1 file datum block
1 file erasure
1 file group
1 file name
1 file owner
1 file ownership
2 file permission
1 file protection
2 file recovery system
1 file removal
1 file server
1 file space
2 file type
1 file user
1 file-deletion
2 file-deletion protection
3 filename
1 filename extension
3 filesystem
2 filesystem check
3 filesystem layout
1 filesystem root

```
3 trashcan
3 tree
1 tree structure
1 troff
2 tset
1 uid
3 umask
1 unalias
3 undelete
1 unix
1 unix computer
1 unix computer system
1 unix directory
1 unix directory tree
1 unix environment
1 unix filesystem
1 unix filesystem tree
1 unix kernel
1 unix program
1 unix programming environment
1 unix shell
1 unix system
2 unlink
2 user
2 user ID
2 user ID number
1 user directory
1 user entry
1 user interface
3 username
2 utility
2 utility program
2 uucp
2 variable
1 vi.
2 video display
2 virtual
1 virtual memory
1 vms
2 wildcard
1 wildcard match
2 wildcarded name
3 window
3 window system
2 working directory
2 workstation
1 world-write access
1 world-write permission
3 writable
1 write
2 x
1 y
```

**Appendix 3**
**All terms found by TExt**

ASCII
BSD
Berkeley UNIX
Berkeley-based mechanism
Bourne-type shell
C
CTRL
CTRL-C
CTRL-D
CTRL-H
CTRL-Q
CTRL-S
CTRL-U
DOS utility
Emacs
File Access Permission
GNU version
I/O
ID
Linux
Macintosh
Multitasking
NFS
PC
RFS
SGID
SGID file
SGID program
SUID
SUID file
SUID program
Solaris
UID
UNIX
UNIX Account
UNIX book
UNIX command
UNIX computer
UNIX computer system
UNIX directory
UNIX directory tree
UNIX environment
UNIX filesystem
UNIX kernel
UNIX session
UNIX shell prompt
UNIX system
UNIX user
UNIX version
VMS
VMS system
Vi
absolute pathname
access mode
accidental file deletion
account name
account use
administrator
alias
amount of storage
asterisk
authorized system user
backup
backup file
backup tape
bash
binary
browse
bug
byte
cat
character
chgrp
child directory
chmod
clear
command exit
command file
command interpreter
command line
command logout
command name
command ps -aux
command tool
command-line argument
compound command
computer network
computer system
control character
control code
cron
crontab
current directory
cursor
daemon
data block
data file
default group
default path
default rm
default search path
default value
delete
deleted file
dir
directory
directory entry
directory level
directory location
directory name
directory tree
directory user
disconnect
disk drive
disk partition
display format
edit
email
entomb
environment variable
erase
error message
escape

executable file
exit
expression
expunge
family tree
file bar
file permission
file recovery system
file space
file type
file-deletion
file-deletion protection
filename
filesystem
filesystem check
filesystem layout
filesystem tree
folder
foo
freeze
generic command
group ID
group guest
group ownership
hostname
initialization
inode
inode number
insert
interpreter
interrupt character
kermit
kernel
linkname
listing
local computer
login
login file
login name
login shell
login time
logout
loop
mail
mailbox
mailbox file
manpage
manual page
menu
mkdir
mode bit
modem
multi-task
multi-tasking operating system
multiuser
newgrp
nroff
octal digit
octal number
od
operating system
override protection
parent directory
passwd
passwd file
password
pathname
performance
permission in a directory
personal computer
program file
program name
program output
programming language
project directory
prompt
pwd
quit
readable
recompile
recover
recoverable
relative pathname
remote computer
remote system
remote-login
rename
rlogin
rm
root
root directory
root password
screen editor
screen-oriented program
scroll
search path
session
set group ID
set user ID
setup file
shell
shell manual page
shell prompt
shell setup file
shell start
shell-script
shelltool
shortcut
shutdown
single-user system
source code
source file
standard input
stopped job
subdirectory
subroutine
subshell
superuser
superuser password
symbolic link
system administrator
system call
system command
system message
system program
system to system

tar
telnet
termcap
terminal
terminal description
terminal emulation
terminal emulation program
terminal number
terminal screen
terminal type
text editor
text file
textfile
touch
trashcan
tree
tset
umask
undelete
unlink
user ID
user ID number
user interface
username
utility program
video display
virtual
virtual memory
wildcard match
wildcarded name
window
window system
working directory
workstation
writable

**Appendix 4**
**All terms found by Quirk**

absolute pathname
access
access mode
account
administrator
alias
align
ascii
asterisk
automatic setup
backup
backup tape
bash
berkeley
berkeley group mechanism
berkeley mechanism
berkeley unix
binary
bit
bourne shell
browse
bug
c
capability
cat
cd
character
chgrp
chmod
chown
code
command
compile
compiles
computer
computer network
computer system
computing
connect
cron
crontab
csh
cshrc
ctrl
cursor
daemon
data
data file
database
default
default group
default path
default system
default umask
default value
delete
deleted file
digit
dir
directory
directory entry
directory location
directory tree
disconnect
disk
display
ditroff
dos
drive
dump
edit
editor
emacs
emacs editor
email
enter
entomb
entry
environment
erase
erase character
error message
executable
executable file
execute
execute access
executes
exit
expunge
family tree
figure
file
file access
file erasure
file ownership
file recovery system
file removal
filename
filesystem
filesystem layout
filesystem root
find
flag
folder
foo
freeze
fsck
gnu
group
group access
group id
group ownership
hardware
helv
home directory
host
hostname
html
id
initialization
inode
input

telnet
termcap
terminal
terminate
terminfo
text editor
text file
textfile
th
tool
touch
trashcan
tree
tree structure
troff
tset
uid
umask
unalias
undelete
unix
unix computer
unix computer system
unix directory
unix directory tree
unix environment
unix filesystem
unix filesystem tree
unix kernel
unix program
unix programming environment
unix shell
unix system
unlink
user
username
utility
uucp
variable
virtual
vms
wildcard
window
window system
workstation
writable
write
x

**Appendix 5**
**All terms found by Xerox**

.c
.cshrc
.exrc
.login
.o
.profile
.txt
AT&T-based system
BSD
Berkeley-based mechanism
Bourne-type shell
C
CD-ROM
CTRL
CTRL-C
CTRL-D
CTRL-H
CTRL-Q
CTRL-S
CTRL-U
Command-Line argument
DOS
DOS utility
Emacs
Filesystem permission bit
I/O
ID
Linux
MS-DOS
Macintosh
NFS
PC
RFS
SGID
SGID bit
SGID file
SGID program
SUID
SUID bit
SUID file
SUID program
UNIX
UNIX Environment
UNIX account
UNIX command
UNIX computer
UNIX computer system
UNIX directory
UNIX directory tree
UNIX environment
UNIX filesystem
UNIX kernel
UNIX power tool
UNIX program
UNIX session
UNIX shell
UNIX system
UNIX user
UNIX version
UNIX wildcarding
VMS system
absolute pathname
access
access bit
access file
access mode
access privilege
access time
accidental file deletion
account
account name
account use
administrator
alias
application
argument
asterisk
backup
backup tape
berkeley
berkeley group mechanism
berkeley mechanism
bit
block
browse
bug
byte
c
capability
capability value
character
chgrp
child directory
chown
code
command
command date
command exit
command file
command interpreter
command line
command logout
command name
command ps -aux
command tool
compound command
computer
computer network
computer system
control character
control code
copy file
crontabs
csh
current directory
current working directory
cursor
daemon
database
default
default group
default path
default rm

parent
parent directory
passwd file
password
path
pathname
performance
permission
permission character
permission control
personal computer
pipe
pipeline
power tool
print work directory
printer
process
program
program file
program name
program output
project directory
protection
ps
pwd
r
range
read access
read permission
relative pathname
remote computer
remote computer file
remote login
remote system
remote-login program
rlogin
rm
root
root directory
root password
screen
screen editor
screen-oriented program
script
scroll
search path
security
sequential output
server
session
set group ID
set user ID
setenv
setup file
sh
shell
shell manual page
shell script
shell setup file
shell start
shelltool
single-user system
site
source code
standard input
standard output
startup file
sticky bit
storage
string
subdirectory
subshell
superuser
superuser password
switch
symbolic link
system
system administrator
system command
system file
system message
system path
system program
tar
tcsh
telnet
termcap
terminal
terminal database
terminal description
terminal emulation program
terminal number
terminal screen
terminal type
terminfo
text editor
text file
textfile
tool
touch
trashcan
tree
umask
undelete
user
user ID
user ID number
user directory
user entry
username
utility
utility program
uucp
variable
vi.
video display
wildcard
wildcarded name
window
window system
working directory
world-write access
world-write permission
writable
x
y

Bilingual terminology extraction plays an important role in the application of bilingual dictionary compilation, bilingual Ontology construction, machine translation and cross-language information retrieval etc. This paper addresses the issues of monolingual terminology extraction and bilingual term alignment based on multi-level termhood. Design/methodology/approach: A method based on multi-level termhood is proposed. The new method computes the termhood of the terminology candidate as well as the sentence that includes the terminology by the comparison of the corpus. â€¢ List of terms extracted from ST Extraction â€¢ List of terms to validate (accept or reject). â€¢ List is added to a termbase Translation â€¢ List is translated and additional data added. 1100. Monolingual extraction &. validation. Importing & looking for equivalents. â€¢ List approved by a person in charge of terminology â€¢ When the client has requested there is an addtional Approval step for client approval. 11.Â  EXTRACT-ALIGN 1ST step: monolingual terminology extraction in both languages. 2nd step: cross-linguistic matching using word-alignment or co-occurrence statistics to find equivalents. Commercial systems in this approach.Â  - Concordace programs: WordSmith Tools, (terminology extraction from TMX). AntConc (free), â€¦ 3. Evaluation. Bate under evaluation. Sketch Engine. 19. Documentary Search Tools. Term-Extraction Tools. Terminology Research Tools. Data-Recording Tools. Electronic-Publishing Tools.Â  Evaluation of Terms and their Relationships. 28. Language Management and Terminology Harmonization 30.Â  Principal Tools for Terminology Work. Any terminological activity, from term identification through product delivery, can be performed manually. However, computerization allows unprecedented improvements in productivity, quality and accessibility.