

Recurrent Patterns in Technical Documentation

Magnus Merkel

NLPLAB
Dept of Computer and Information Science
Linköping University
S-581 83 Linköping
Sweden
Email: magme@ida.liu.se

Abstract

This paper addresses some of the problems involved in the production and translation of technical documentation. The techniques and methods developed within Natural Language Processing in general and Machine Translation in particular have still a long way to go before we can see any commercial products that would be general enough to automatically translate unrestricted text. Instead of merely aiming for the perfect MT system, we should also focus on how to make use of existing and simple techniques and the capacity of today's hardware to make the production of technical documentation faster, better and cheaper. Even a twenty per cent gain in efficiency compared to manual translation is considerable compared by any industry standard.

In this paper I describe a tool that pre-processes the source text and gives various kind of information that forms decision support whether translation tools should be applied at all. Examples from analyses show that up to 43 per cent of a text could be repetitious and that this should be utilised before the translator starts translating. If we consider both repetitions within one document as well as repeated patterns across documents, there is evidence in the corpus that 55 per cent of the text in one document can be regarded as recurring. The tool has been run on several real handbook texts from major computer software companies and a summary of the results is presented.

1. Introduction

The techniques and methods developed within NLP in general and MT in particular have still a long way to go before we can see any commercial products that would be general enough to automatically translate unrestricted text. Very little of NLP technology is used in the translation industry today and translation of documentation for various technical products is becoming somewhat of a bottleneck, both for time reasons and for economical reasons, in, for example, the software industry. Here localised products need to be out on the market as quickly as possible after the original, otherwise the company in question risks losing substantial sales and market shares.

In this paper I describe a tool that pre-processes the source text and gives various kind of information that forms decision support whether translation tools should be applied at all. The output is a table with all recurring sentences and all recurring phrases together with frequency and reference positions. It also gives percentage figures for how much of the total text that is made up by recurring sentences and translatable phrases. I show examples from analyses made that up to 43 per cent of a text could be repetitious. This should be utilised before the translator starts translating. If we consider both repetitions within one document as well as repeated patterns across documents, there is evidence in the corpus that 55 per cent of the text in one document can be regarded as recurring. The tool has been run on several real handbook texts from major computer software companies and a summary of the results is presented.

In my view, MT has focused too much on the large-scale systems and theoretically interesting problems that arise in translation and too little on using well-established techniques and the processing power of today's hardware to make the translation process more efficient. It is sometimes mentioned that MT systems will cut the time and costs to a fraction of manual translation. But, in any industry a cut in costs by 10 or 20 per cent would be regarded as a substantial improvement; this seems not to be the case for researchers and developers of MT systems. However, there is a recent trend in MT where researchers study parallel texts in multiple languages and where the object is to align sentences which have the same translation (Klavans & Tsoukemann, 1990, Brown et al, 1991, and Gale & Church, 1991). There are also related approaches. The first one is based on Translation Memories (TMMT) where the system retrieves and stores pairs of source phrases, sentences or texts and their translation from a textual databases (cf. Linguistic Industry Monitor 1992). The second one is called Example-Based Machine Translation (EBMT), where the translation memory approach is complemented with heuristics to generalise on existing translation pairs and translate by making analogies from the existing corpus (Sumita and Iida, 1991). The advantages of the two latter approaches compared to the more traditional Rule-Based MT (RBMT) are that the problems of updating large-scale rule-bases are avoided. In RBMT it is difficult to anticipate the effect of adding new rules. Also, the RBMT approach is more time-consuming and it is difficult to make use of domain-specific or situational information in the translation process.

It should be mentioned that this kind of tool would also be of use not only for translation purposes, but also as a means for a company that would like to streamline and standardise their (source) documentation with the aid of sentence and phrase libraries.

1.1. Problems

Like any industrial activity the production of technical documentation involves the following critical areas:

- Time
- Cost
- Quality

Due to the large amount of text that TD usually consist of, a great deal of planning and effort have to go into the production of almost any kind of technical documentation, be it a user's guide or a service manual. TD is often bulky. In this study we have looked at handbooks of around 800 pages, but for other domains than computer software, volumes of several thousand pages is not unusual. For an aeroplane, it is often said that the documentation weighs more than the plane itself.

Large volumes of TD cannot be produced by a single writer, if you want to have it out on the market at the same time as the product. A well-oiled team of technical writers/translators that can work together is an absolute requirement, but when there are several people involved there will be co-ordination and consistency problems. They must use a consistent style, use the same terminology, etc. Apart from their linguistic expertise, they have to have considerable knowledge about the product and the field they are writing about. Large corporations have their own technical writers, but for many companies contracting a specialist company is the usual way of handling the production of TD.

Solutions to this are under investigations in the NLP community, but the results have not yet been rewarding for industry.

1.2. A simple and fast approach

Instead of waiting until advanced MT and writers systems are available for the mass market we suggest that simpler techniques that would pay off immediately should be used. This means to utilise the features of technical documentation as a text type and well-established techniques within NLP. Below some of these features and strategies are listed:

- TD include repetitions on various levels (word, phrase and sentence).
- Repetitions should be exploited both in the writing and the translation process. All instances of the same type can in principal be translated in one step.
- Previously written and translated material should be reused in new versions. (This should be viewed in contrast to the prevailing situation where most new versions are written from scratch each time.)

2. Measuring recurrency

Depending on what level of abstraction texts are viewed, there will always be different degrees of recurrency. On the string level, words, phrases and even exact sentences recur. Moving up one level would mean to use strings in conjunction with variables. The pattern "Choose the X button." where X is a variable for button names, would describe string instances such as "Choose the OK button.", "Choose the Cancel button.", "Choose the Define button.", etc. Another step up the abstraction ladder would be to specify context-free grammar rules, such as "V(imp) Det Nmod N" where it is specified that an imperative verb form is followed by a determiner, an modifier, and a noun. This rule would be more general in the sense that it would correctly describe sentences like "Choose the Save command." and "Remove the document header.". The higher we go on the abstraction ladder the more we need to specify about the lexical entries and the syntactic construction types in order to characterise the text in an adequate way.

In this study we start by looking at the simplest level, i.e. the string level. Here we identify two separate types: sentences and phrases.

2.1. Sentence level

On the string level a sentence is characterised by means of punctuation. In the analyses presented later, the definition is taken to be a technical one, i.e., a technical sentence is a string of words that ends with a period plus space (". "), exclamation mark (!), question mark (?), carriage return/paragraph mark (¶) or tab mark. Apart from ordinary sentences, this definition also covers normal conventions for writing headings and tab separated text in tables.

If a sentence recurs fifteen times in a manual, there is a likelihood that it has been reused from other similar sections and that it should be translated in a consistent manner.

2.2. Combination of sentence and phrase level

With simple string matching techniques there will be overlapping information. As a recurring phrase is likely to be found in recurring sentences, there will be a certain redundancy. But if the strategy in this work is to look for maximal strings this problem is partly avoided.

2.3. Phrase level

To single out linguistically valid phrases is more difficult without linguistic information. Ideally it should be possible to single out stereotypical complex noun phrases, prepositional phrases and the like. Here we adopt the approach to search for maximum strings in an unrestricted way. Any recurring string pattern will be considered as a potential phrase. A few heuristics can be applied for a language like English, such as filtering out phrases ending with *the*, *a* and *an*, and phrases that only contain one bracket ("(" or ")") or one quotation mark. There is no way of vouching for the possibility of translation of a phrase if it is generated by this simple string matching technique. Therefore it is necessary to revise the automatically derived phrase list manually. Furthermore, as languages differ considerably in phraseology, the phrase list of the source language would have to be fine-tuned for each language. With more linguistic knowledge, this is possible, but when processing on string level a manual check is necessary.

2.4. Word level

In corpus-based linguistics a great deal of attention has been paid to frequency analysis of word types, mostly graphic words. Here we use this well-established technique to characterise the text type in the corpus in relation to the Lancaster-Oslo-Bergen corpus (Hofland & Johansson, 1982). The LOB corpus is identical in size to the corpus under study here, 1,000,000 words). As in the LOB corpus, the notion of graphic word is used, i.e., a sequence of alphanumeric characters surrounded by spaces. The emphasis is here put on the differences between the two corpora in frequency ranking and the distribution of function and content words.

3. The Text Analysis Tool¹

The implemented Text Analysis Tool (TAT) runs in a Sun UNIX environment. Below are some data about the implementation.

3.1. Specifications

Programming Language:	C
Hardware:	Sun Sparc Station (UNIX environment). Runs on PCs (DOS environment) for smaller data.
Input:	Text files (specified by name or by directory), tables with information from previous analysis.
Maximum text capacity:	Unknown. Handles several MB of ASCII text.
Restrictions:	Works only on unformatted text (ASCII or ANSI format).
Speed (example from 90,000 word texts):	Sentence analysis - 10 seconds.
	Word frequency: 10 seconds.
	Phrase analysis: 1 hour.

Fig. 1 Specification of Text Analysis Program

Output: A table which can be configured in a number of ways:

- strings with frequency, file names and positions in files
- strings with frequency only
- output sorted alphabetically
- output sorted by frequency
- output sorted by length of analysed string.

¹ Bernt Nilsson has done the implementation of the tool.

3.2. Other features

1. The intersection of two texts (on sentence or phrase level) is found simply by taking the intersection between two analysis tables. This means that we don't have to reanalyse a handbook if we want to compare it with a different one. This is also much faster than analysis from scratch.
2. The analysis tables for sentences and phrases can be used to automatically calculate how much of the text that the recurring sentences and phrases cover. This figure is a percentage of the text, based on the words of the total text.
3. Typographical "errors" like double spacing is handled by flagging occurrences of this within phrases. That is, two phrases which are identical except for spacing are considered to be equal, but they are both listed in the output file and can then be given the same translation. This is to make it possible to find the original text in the source document in the automatic translation process.
4. It is possible to set the minimum length of phrases to look for. The data above are processed with a setting of 3 words in a phrase, but it's easy to change this. The lower this setting is, the more processing power is needed. Also, if the tool is looking for two-word phrases it will also produce a lot of useless information.
5. The tool can handle different languages. The texts described in this paper are all English, but we have run the tool on both French and Swedish texts successfully.

4. Analysis of handbooks

4.1. Texts

The text corpus (referred to as the TD² corpus) consists of computer program manuals from two different companies. The texts vary in subject matter between programming environments, word processing programs and spreadsheet programs. There are six different manuals in the TD corpus, five from one company (here called Company A) and one from another (Company B). Two interesting features of the material from Company A are that two of the manuals describe word processing programs, but for different platforms (Macintosh and Windows) and there are two versions of the same spreadsheet program (here called version 1 and 2). These features give us the possibility of not only measuring internal recurrency (within the same document) but also to compare two similar products (in the case of the word processing programs) and two versions of the same product. The size of the TD corpus is just over one million words taken together (1,065,477 to be more exact), which makes it easier to compare it to the one million word Lancaster-Oslo-Bergen corpus. The comparison between the two corpora is made in section 4.4.

As the TD texts have all been used as source texts for manual translation into several languages, this study indicates how well a full Translation Memory MT system would perform given that sentences and phrases would have been stored in a multi-lingual database.

We use the following labels for the different documents in the TD corpus:

² TD is an uninspired acronym for Technical Documentation

Label	Description	Manual Type	Source
PLE	Programming Language Environment	User's Guide	Company A
CWP	Character-based Word Processor	Reference Guide	Company B
MWP	GUI-based Word Processor for Macintosh	User's Guide	Company A
WWP	GUI-based Word Processor for Windows	User's Guide	Company A
SS1	Spreadsheet version 1	User's Guide	Company A
SS2	Spreadsheet version 2	User's Guide	Company A

Fig. 2 Description of the texts in the corpus

4.2. Methods

In this section the difference measures of recurrency is described and how it has been used in the corpus analysis.

4.2.1. Measuring internal recurrency

Internal recurrency is taken as a measurement *within* one document. On the sentence level all instances of one sentence type are listed. On the phrase level the same applies for all instances of a certain phrase. The percentage figures given are measurements of how much of the total text in words that the recurring patterns cover. For sentences a frequency of two or higher is recorded and for phrases there must be a frequency of four or higher. When it comes to phrases, the figure given is calculated on a manually revised phrase list. In this study the phrase lists have been revised with a Swedish bias. This means that phrases that cannot have a one-to-one translation between Swedish and English have been removed from the original list.

The internal recurrency figures for a given text is useful as a measurement for how repetitive a document is. The figures should be taken as a guideline as for whether it is worth the while to apply translation tools or systematic search-and-replace functions based on a translated sentence and phrase glossary.

Below is an example of the top elements of the internal sentence and phrase lists (before and after manual revision) for WWP.

```

choose the ok button.      211
result 152
example147
- or - 82
- or - 82
you can display a new result by pressing the update field key (f9) or
by choosing print merge from the file menu.30
do this30
example:      26
from the tools menu, choose options (alt, o, o). 22
do one of the following: 21
if the field is in a header or footer, you can update the field once
by choosing the print command from the file menu.18
...

```

Fig. 3 Sentence list for WWP (most frequent technical sentences)

The third and fourth item in the above sentence list reveals a possible editorial mistake: The sequence of dashes including an "or" is either written with em-dashes (long) or with en-dashes (short). The list above only shows the frequency for each sentence. When calculating the coverage of a sentence list, there is also information about in which files and in what character position each instance is found.

```

you want to 932
menu, choose 498
the insertion point405
the ok button332
choose the ok331
choose the ok button      329
if you want 288
the file menu257
for more information      241
table of contents 230
from the file menu 220
you can use 212
if you want to      207
in your document 189
where you want 184
position the insertion point 182
the format menu 176
...

```

Fig. 4 Unrevised phrase list for WWP

In fig. 4 the raw output of phrases from the analysis tool is shown. The list shows that some of the phrases are not to be considered to be translatable units. The first string (*you want to*) is not considered to be as translatable as the phrases *if you want* (example 7) and *if you want to* (example 13). The second example makes no sense and the fourth one is not as complete as the fifth one (*choose the ok button*), etc. The same list revised manually with a bias towards a Swedish translation would look like the following:

the insertion point	405	
the ok button	332	
choose the ok button		329
if you want	288	
the file menu	257	
for more information		241
table of contents	230	
from the file menu	220	
you can use	212	
in your document	189	
position the insertion point		182
the format menu	176	
...		

Fig 5 Revised phrase list for WWP

The above phrase list shows the type of lists that have been used in this study when figures are given on internal and external recurrency of phrases.

4.2.2. Measuring external recurrency

External recurrency is taken as a measurement across two or several documents. Two independent documents which previously have been analysed into sentences and phrases are cross-analysed (or rather the output tables are). The result is a list of all sentences that occur in both documents (at least one occurrence in each document) and a list of all phrases that occur at least four times in each document.

The external recurrency figures actually tells you what two documents have in common on sentence and phrase level. When two related documents are being created or translated, one could function as the source document and thereby save the writer/translator a lot of work. "Related documents" could mean for example documents for a word processor on two different platforms or a new generation document compared to an old version of the same product.

In the study there are two examples of where we have measured external recurrency: 1) MWP and WWP (two word processing programs for different platforms), and 2) SS1 and SS2 (two versions of the same spreadsheet program).

4.3. Results on sentence and phrase levels

The first two studies made were on CWP and PLE, from two different companies. The documents are comparable in the sense that they have roughly the same size (just under 90,000 words). But there are more differences than similarities. CWP is a word processing program and PLE is a programming environment system. CWP has a character-based interface whereas PLE uses a graphical interface. Furthermore, CWP is written as a Reference Guide (where most of the contents are listed by keywords or functions) and PLE is a User's Guide (which has a more thematic structure). Some of these differences are visible in the analysis below. CWP is by far a much more stereotypical document than PLE (12 % compared to 3.5 % sentence recurrency, 26 % compared to 11 % phrase recurrency).

	CWP	PLE
Total no of words	89,000	87,000
No of word types	2,928	4,599
Sentence recurrency	12 %	3.5 %
Phrase recurrency	26 %	11 %
Sent & Phrase recurrency	35 %	13.5 %

Fig. 6 Internal recurrency for CWP and PLE

The next two documents are two word processors from company A. They are both built on a graphical interface, but for two different platforms (Windows and Macintosh). Here the recurrency figures are higher than for PLE, but not as high as for CWP. But they could still be considered worth exploiting.

Internal Recurrency	WWP	MWP
Total no of words	218,271	168,767
Different word types	5,497	4,459
Sentence Recurrency	10 % (766 sents)	6 % (442 sents)
Phrase Recurrency	25 % (1,903 phrases)	23 % (1,452 sents)
Sentence & Phrase Recurrency	31 %	27 %

Fig. 7 Internal recurrency for WWP and MWP

As WWP and MWP could be regarded as related products, we could take a look at what is shared between the two. Here we find that 2,170 sentences and 573 phrases are identical in both documents. Comparing WWP to MWP yields a combined external sentence and phrase degree of 22 % (see the left table below). This indicates that if WWP was translated after MWP a maximum of 22 per cent would already have been translated in the first document and that this could have been exploited in the second translation.

If the production or translation order was reversed the figures would be as is shown in the table to the right. The higher percentage figures here are explained by the fact that MWP is a shorter document than WWP.

External Recurrency in WWP relative to MWP	
Shared sentences	2,170
External Sentence Recurrency	11 %
Shared phrases	573
External Phrase Recurrency	13 %
External Sentence + Phrase Recurrency	22 %

Fig. 8 External recurrency in WWP relative to MWP

External Recurrency in MWP relative to WWP	
Shared sentences	2,170
External Sentence Recurrency	14 %
Shared phrases	573
External Phrase Recurrency	14 %
External Sentence + Phrase Recurrency	26 %

Fig. 9 External recurrency in MWP relative to WWP

Even more interesting is to look at the figures if we combine the results from the internal and external recurrency tables. Here the results show that there is up to 38 per cent and 37 per cent recurrency respectively for WWP and MWP.

Combination Internal/External Recurrency	WWP	MWP
Internal + External Sentences	20 %	20 %
Internal Sentences + External Sentences + External Phrases	30 %	30 %
Internal Sentences + External Sentences + Internal Phrases	38 %	37 %

Fig. 10 Combination of Internal and External Recurrency for WWP and MWP

The last two documents are the two versions of the spreadsheet program. Here the figures are considerably higher than in the previous examples. The combined internal recurrency figures are 43 per cent for SS1 and 39 per cent for SS2.

Internal Recurrency	SS1	SS2
Total no of words	254,350	248,089
Different word types	5,362	5,691
Sentence Recurrency	25 % (2,290 sents)	15 % (1,822 sents)
Phrase Recurrency	29 % (1,824 phrases)	31 % (1,911)
Sentence & Phrase Recurrency	43 %	39 %

Fig. 11 Internal Recurrency for SS1 and SS2

Comparing the two spreadsheet documents is even more rewarding. Here there are 3,677 shared sentences and 620 phrases.

External Recurrency in SS1 relative to SS2	
Shared sentences	3,677
External Sentence Recurrency	20 %
Shared phrases	620
External Phrase Recurrency	14 %
External Sentence + Phrase Recurrency	31 %

Fig. 12 External Recurrency in SS1 relative to SS2

External Recurrency in SS2 relative to SS1	
Shared sentences	3,677
External Sentence Recurrency	20 %
Shared phrases	620
External Phrase Recurrency	15 %
External Sentence + Phrase Recurrency	31 %

Fig. 13 External recurrency in SS2 relative to SS1

Combining the internal and external figures yield the table below, where the highest figures reach 55 per cent and 52 per cent respectively.

Combination Internal/External Recurrency	SS1	SS2
Internal + External Sentences	41 %	33 %
Internal Sentences + External Sentences + External Phrases	48 %	42 %
Internal Sentences + External Sentences + Internal Phrases	55 %	52 %

Fig. 14 Combination of Internal and External Recurrency in SS1 and SS2

4.4. Results on word level

On the word level we compared the texts under study with the LOB-corpus (see section 2.3). There is a word frequency list for each of the texts and also a global frequency list of the whole corpus. By comparing the global frequency list with the equivalent list for the LOB-corpus some interesting observations about the text type can be made. The fact that the TD corpus is much narrower in subject matter is best illustrated by the fact that the total number of graphic word types in LOB is around 50,000, but only 13,586 in the TD corpus.

However, by looking at the actual frequency data a great deal of information about the texts can be uncovered. Below the 25 most frequent words in both the corpora are contrasted.

Rank	Word type	Freq.
1.	the	98972
2.	to	35361
3.	you	29782
4.	a	29270
5.	in	24700
6.	and	19821
7.	of	18082
8.	or	14518
9.	for	13149
10.	can	9564
11.	is	9288
12.	on	9244
13.	if	8768
14.	<i>document</i>	8236
15.	<i>text</i>	7242
16.	<i>box</i>	7226
17.	<i>choose</i>	7127
18.	that	6883
19.	with	6757
20.	<i>select</i>	6681
21.	<i>command</i>	6605
22.	want	6501
23.	<i>menu</i>	6317
24.	<i><company name></i>	6270
25.	from	6223

Fig. 15 The top 25 graphic words in the TD corpus rank list.

Rank	Word type	Freq.
1.	the	68315
2.	of	35716
3.	and	27856
4.	to	26760
5.	a	22744
6.	in	21108
7.	that	11188
8.	is	10978
9.	was	10499
10.	it	10010
11.	for	9299
12.	he	8776
13.	as	7337
14.	with	7197
15.	be	7186
16.	on	7027
17.	i	6696
18.	his	6266
19.	at	6043
20.	by	5796
21.	had	5391
22.	this	5287
23.	not	5142
24.	but	4956
25.	from	4686

Fig. 16 The top 25 graphic words in the LOB corpus.

A brief glance at these rank lists suggests the TD corpus is more homogeneous than the LOB corpus. There are eight content words among the top 25 words in the TD corpus, whereas the LOB corpus only contains function words.

There are also indications that the TD corpus holds texts where a reader is addressed directly (cf. the high frequency of *you*) and where the personal pronouns *he* and *she* may not be as dominant as in "general" English. Another way of viewing the word frequencies would be to view the highest ranked words in LOB with how these words are ranked in the TD corpus. This gives us the following table:

Word	LOB	TD
the	1	1
of	2	7
and	3	6
to	4	2
a	5	4
in	6	5
that	7	18
is	8	11
was	9	449
it	10	43
for	11	9
he	12	4206
as	13	27
with	14	19
be	15	58
on	16	12
i	17	756
his	18	5957
at	19	76
by	20	38
had	21	1913
this	22	32
not	23	56
but	24	206
from	25	25

Fig. 17 The top 25 words in LOB compared to their ranking in TD

This table gives us more information about the differences of the two corpora and further strengthens the assumptions made earlier. Compare for example the frequencies of the personal pronouns *he*, *I* and *his*, and the past tense auxiliaries *was* and *had*. In the appendix at the back of this paper, the above list is reproduced together with the rankings in each document of the TD corpus. There we also see that each handbook closely resembles the overall pattern found in the whole TD corpus.

Lastly, and a trivial remark about the rank lists, the first 25 word types constitute 34.2 per cent of the LOB corpus and 42.3 per cent of the TD corpus respectively, which is another strong indication of the more homogeneous character of the TD corpus.

5. Usability and application areas

The process of pre-processing documentation in the manner described in previous sections have various practical purposes. Below a number of these purposes are listed:

- Compiling sentence dictionaries.
- Compiling phrase dictionaries.
- Measuring the suitability for applying (semi-)automatic translation. Recurrency degrees and other measurements could give information on which translation tool that would be most suitable to automate (parts of) the translation process.
- Comparing different kinds of documentation within the same product (User's Guide, Reference Guide, Getting Started, Help files, etc.).
- Comparing the same type of documentation across products.
- Inserting translations (from the sentence and phrase glossaries) into the formatted document before manual translation). In the simplest case this can be done by macros or by integrating the sentence and phrase data base in a Translation Memory-based MT system.
- Feedback to the technical writers. Should they reuse phrases and sentences that they already have written in a more consistent way? Can they write in a more constrained way and thereby making translation easier? Can vocabulary and the sentence patterns be constrained further without losing readability? What's the trade-off?

5.1. To make things more general...

The patterns discussed in this paper are merely recurring strings. But it would be possible and indeed desirable to generalise on level of descriptions. Below are a few levels of descriptions that can be realised to make the representation more compact and general. It would also be possible find out higher-level patterns in a given text.

1. Strings with word and phrase variables.
2. Strings with syntactically/semantically restricted variables.
3. Grammars. Specially designed grammars for certain subtypes of technical documentation, such as direct instructions, examples, cross-references, etc.
4. Pattern representations (of type 2 and 3) which are compatible with SGML and works with such parsers.

Item (1) means that we replace words and phrases with variables of different types. In (2) we could have variables which would be tied to a certain class of words or phrases. The last two items would involve a more traditional NLP approach, like context-free grammars.

5.2. In the long perspective...

Measuring and recording recurrency on string level as presented in this paper can be seen as the first building stone of a much more elaborate set of techniques and tools that would address the problem of making production of technical documentation faster, cheaper and of higher quality than it is today. But we need a firmer empirical foundation to base such methods and techniques on. It is necessary to study large text corpora of various technical documentation, preferably in both the source and target languages, to find out more about the text type. This should also involve comparative studies of how technical documentation differs from other text types such as legal texts, newspaper articles and fiction. (We have looked at some of these text types and made preliminary observations that the recurrency degree in, for example, fiction is, as might be expected, a mere fraction compared to the data presented in this paper. But this needs further investigation on a much larger scale.)

One interesting path of study would be to measure the "likeness" of for examples chapters in a handbook. What chapters in a set of documentation could be grouped together to maximise the efficiency gains? Here we would need to recover where the recurrencies on different levels are and by doing so it would be possible to leave out chunks (perhaps whole sections or chapters) where the recurrency degree is low or non-existent. By cutting off "dead meat" in this manner, the efficiency gains would be much higher. This would give the translators a real tool for deciding where automatic tools would pay off better and also assist in assigning teams of translators to linguistically homogenous texts.

It would also be of interest to actually verify the results of this study in real translation situations and to compare different translation techniques in practice. This could involve case studies of professional translators working with different types of tools on different kinds of texts. The trade-off between the actual effort to have a document translated (including translation of sentence and phrase dictionaries or customising grammars and lexicon in a RBMT system). To make a case for new techniques involves more than just stating that a certain approach is more efficient in theory; we need to verify the efficiency gains in practice and in particular show for what type of efficiency gain we are talking about. As mentioned before, the characteristics of a given text will dictate what tools should be used to maximise productivity and quality of the end result.

References

Brown P. F., Lai, J.C. & Mercer, R:L. (1991), "Aligning Sentences in Parallel Corpora", in *Proceedings from ACL-91*, pp 169-176, Berkeley.

Gale W. A. & Church K. W: (1991), "A program for aligning sentences in bilingual corpora", in *Proceedings from ACL-91*, pp 177-184, Berkeley.

Hofland K. & Johansson, S. (1982), *Word Frequencies in British and American English*. Bergen.

Klavans J. & Tsoukemann E. (1990), "The BICORD System", in *Proceedings from COLING-90*, pp 174-179, Helsinki.

Language Industry Monitor (1992), Issue No. 8, March-April 1992. Ed. Brace C. & Joscelyne A. Amsterdam.

Sumita E. and Iida H. (1991), "Experiments and Prospects of Example-Based Machine Translation", *ACL-91*, pp 185-192, Berkeley.

Appendix

The 25 most frequent graphic words in the LOB corpus compared to ranking in the TD corpus - a global comparison and rankings in each document of TD.

Word	LOB	TD
the	1	1
of	2	7
and	3	6
to	4	2
a	5	4
in	6	5
that	7	18
is	8	11
was	9	449
it	10	43
for	11	9
he	12	4206
as	13	27
with	14	19
be	15	58
on	16	12
i	17	756
his	18	5957
at	19	76
by	20	38
had	21	1913
this	22	32
not	23	56
but	24	206
from	25	25

PLE	CWP	MWP	WWP	SS1	SS2
1	1	1	1	1	1
4	8	6	7	7	8
6	6	7	6	6	6
2	2	2	2	2	2
3	4	4	4	4	3
5	3	5	5	5	5
11	60	27	25	26	19
7	17	19	16	17	16
213	450	549	495	423	460
21	114	42	39	49	48
9	14	11	9	9	9
2198	n/a	n/a	4524	n/a	3235
18	56	26	27	29	33
19	21	23	23	27	28
37	51	90	76	57	65
29	11	16	20	10	13
742	1132	951	333	1079	1662
n/a	2560	n/a	3125	n/a	n/a
40	68	66	71	118	115
31	73	48	35	38	40
1805	n/a	1665	1800	1391	2099
17	36	30	30	44	50
46	46	59	52	62	63
74	280	242	224	264	269
36	65	15	17	52	24