Project P923-PF

Multilingual WEB sites: Best practice, guidelines and architectures

Deliverable 1

Guidelines for building multilingual Web sites

Volume 3 of 5: Annex B

Overview of Language Processing Tools and Techniques

Suggested readers:

This document is primarily aimed at anyone who is involved in the process of designing, building or managing WEB sites. It is of immediate relevance to those involved with multilingual WEB sites, but it nevertheless, provides information which will allow monolingual WEB site designers to design sites that are economically upgraded to multilingual sites.

EDIN 0009-0923

Project P923

For full publication

September 2000

EURESCOM PARTICIPANTS in Project P923-PF are:

- Koninklijke KPN N.V.
- France Télécom
- British Telecommunications plc
- Telecom Italia S.p.A.
- Portugal Telecom S.A.

This document has been approved by EURESCOM Board of Governors for distribution to all EURESCOM Shareholders.

Executive Summary

The language processing tools and techniques that could potentially reduce the cost of maintaining a multilingual WEB site are discussed. The "typical" localisation process for any Information Technology product is described along with general guidelines for localisation. A variety of language processing technologies is discussed, including machine translation, summarisation, and text generation. Under certain conditions, (e.g. where language is restricted in structure or coverage) such technology can provide a useful way to reduce the cost of providing information in several languages, or at least, make it easier for users to understand information that is not presented in their first language.

List of Authors

Stephen Appleby (BT)

Nuno Beires (PT)

Malek Boualem (FT)

Louis Boves (KPN/University of Nijmegen)

Maurizio Codogno (IT)

Els den Os (KPN)

Marta Pombo Prol (BT)

Jérôme Vinesse (FT)

Contents

1 The "typical" localisation process

The typical localisation process is described here by way of a reference. There will no doubt be certain similarities between the normal localisation process and that which needs to be applied to multilingual WEB sites. This section also provides a comprehensive description of those areas that typically need to be addressed so that we can have more confidence that the architecture and guidelines that are to be produced by this project do not overlook anything.

The three main stages of a localisation project are planning, translation and after translation (Esselink 1998). Planning is essential to ensure that things run smoothly throughout the project's lifecycle. Translation represents the core of localisation, where real translation and adaptation to other languages take place, the process after translation involves a lot of both linguistic reviewing and functional testing and also dealing with updates. Each of them has its own importance but also a seamless combination of these three stages will contribute to a successful project.

1.1 Stage 1:Planning

One of the most decisive factors when undertaking a localisation project is to be able to anticipate possible problems and try to find solutions to prevent them before they appear. In other words, proper *planning*. It is essential for the localisation agency to keep a continuous contact with the client, both before the project starts and during its development. It is indeed advisable that a first contact with the software engineers takes place before the project gets started, so both parties will get a better understanding of each other's roles.

When meeting the software developers that are writing or have already written or designed a software product both parties must try to exchange as much information as possible in order to make the process run smoothly. In the first case the localisation manager can suggest some useful ways of managing files and file content. In case the software has already been written, s/he can obtain some useful information by finding out about how material was organised when it was first designed.

Project management is certainly a key element in the whole process, and the project manager must make sure that s/he understands the customer's requirements. It is fundamental to establish beforehand which languages the product is going to be translated into, which locales, size of the project etc. Then, depending on the client's experience in localisation projects the project manager can suggest best ways of handling it.

1.1.1 Parties involved in the process

Project manager

Software developers

Translators

Software developers in the target country (linguistic review)

Localisation vendor

Localisation Engineer

Proof-reader

DTP staff

The project manager is responsible for co-ordinating the different parts, contact the developers, find the translators, and in general supervise the project from beginning to end.

Translators also play an essential role in the successful completion of the project. They should be provided with the right hardware and software version to be able to run the program in its original form. They also need to have access to all the documentation and online help provided with the product, so they can use it as a reference and also to learn about the product. The project manager has to make sure that translators understand what the project requirements are, if there is a particular order in which the material should be translated, style guides, if there is an existent glossary of terms that they should use, etc. Translators should always work into their mother tongue. It is also advisable to find technical translators if the project requires a certain level of expertise in a particular technical area.

Software developers are those who design and develop the original product. Software developers in the target country will review the translated product to check if it is suitable for commercialisation from a linguistic cultural and technical point of view. The localisation vendor is the company hired for localisation. The Localisation engineer is the person responsible for all the technical aspects related to the localisation process, like compiling the translated product and testing, s/he also assists with any other technicalities involved. A proof-reader is responsible for checking and testing the translated material from a linguistic point of view. Desktop publishers are responsible for formatting and layout of both printed and online material.

1.1.2 Organising the material

The project manager at this stage would have to assess the amount of material to be localised.

S/he should identify how many files need translation, i.e. which files contain language dependent material. Then it should be established which text should be translated in those files. In order to facilitate this stage, it is useful to contact the software developers prior to starting off the project in order to advise them to keep text files and source code separately. When translating web sites, the customer must specify if the whole site is going to be translated or just some sections relevant to the target market/s.

It is also useful to find out whether there is an existing glossary of terms or any previous translations

1.1.3 Into which languages?

This is also part of the planning stage, and basically is about establishing into which languages the product is going to be translated. This affects both directory structure complexity and is needed by the project manager for finding the translators, assessing the volume of work to be done and providing a quote to the client.

1.1.4 Project management tools

There are some tools available in the market that can assist at this stage of localisation. Some of the most well known ones like WebBudget, Multilizer, Euro-Dollar and TransWeb Express.

WebBudget. This is a tool that helps project managers, translators and localisation engineers to assess all the translatable text in an HTML project and also provide the client with a quote. WebBudget counts all the translatable text in an HTML file including translatable tags. It produces a report containing all the translatable text classified according to categories. There is a downloadable free version http://www.webbudget.com

Corel Catalyst. This is an integrated localisation tool that provides utilities for all the key parties involved in the localisation process. It includes several utilities specifically designed to provide solutions for project management tasks. Project managers can use Corel Catalyst for word counts, statistics generation and project status monitoring

http://www.corel.com/

TransWeb express. Mainly for HTML localisation projects, it also includes a project management utility. It counts words, links, and graphic references and can even provide with an estimate of resources needed for a particular project.

http://www.berlitz.ie/twe/default.htm

1.2 Stage 2: Translation

1.2.1 Software Translation

Software files are normally distributed in two formats. They can be text-only resource files that can be translated using a normal word processor. These files need to be compiled into program files after translation has been completed. The localisation manager should make sure that translators know exactly which text is translatable in those files. Alternatively an automatic or manual filter can be run on those files in order to separate translatable text from pure programming code.

Translation can also be done directly on program files using a resource editor. These files do not need re-compiling after translation. The translator can see how the translated text would look in the actual interface, s/he can thus adjust dialog box and menu sizes at the same time that translation takes place.

1.2.2 Help files

In a typical windows help file there are three main files that contain translatable text. These are files with the extension .cnt, .hpj and .rtf. . There is one .cnt file in a help project and it contains all the elements included in the table of contents as it appears on the screen. An .hpj file contains all the information needed for the help file to be compiled, not all the contents of this file need to be translated, only certain sections like the title and text that appears in some buttons. .RTF is the format in which most of the information contained in the help file is stored.

1.2.3 HTML

Html files can be translated in several ways. Text can be translated either by using a text editor having direct access to the HTML source or using a standard HTML editor. The second option is rather more secure than the first one as tags are less likely to be altered during translation and at the same time, all the formatting elements are still present and visible. This helps translators compare files and ensure that source and target files look alike

A third option is to use a standard Computer Assisted Translation (CAT) tool. These tools include filters that allow the translator to translate the text without altering any Html tags. They also store translations for future re-use.

1.2.4 Tools for Text Translation

There are several tools for text translation in the market. Most of them offer integrated utilities like filtering file formats and translation memory. Filters are useful for several reasons. On the one hand translators do not have to deal with any formatting or mark-up elements in a file so they just concentrate on plain text. But also it prevents any formatting from being altered or corrupted. Translation memory is widely used nowadays and is recognised as a very practical utility. It can save important amounts of both money and effort to the translator, the customer and the localisation vendor at the same time that it ensures terminology and translation consistency. Due to the fact that repetition is very common in technical fields and also in frequently updated Web pages or help files, translation memory comes as a very handy and efficient option. A translation memory utility takes a certain source text and it stores it together with its corresponding human translation. Before a new translation starts, the translation tool will scan the text and find exact or fuzzy matches for the new text, suggesting previously stored translations to the translator. The translator can usually interact with the system and choose whether to accept them or not, but it is recommended to stick to an existent terminology in order to avoid extensive updating across files.

Some of the most well known products in the market are TRADOS's Translator's Workbench http://www.trados.com/workbench/, Déjà vu http://www.atril.com/, and IBM's Translation manager http://www.software.ibm.com/ad/translat/

Machine Translation can be used as part of the translation process as an extra aid for the translator. It can become a useful tool when combined with some other CAT tools such as translation memory. The translator can enter some parts of the text in a MT system and obtain a draft translation for it. S/he can either accept it as it is or make the necessary changes, then that translation can be directly stored on the TM and be used on future translations. There are several ways to improve the quality of MT in certain environments. When dealing with a customisable system such as Globalink http://www.lhs.com/ the user can adapt it to best suit the needs for translation of a particular type of text or a certain vocabulary and thus maximise the system's capabilities

Some other tools:

Multilizer http://www.multilizer.com/lm/index.html

It offers utilities such as:

- Text filtering. It extracts those words and lines that need translation from the program source

- Language dependent information is easily maintained by means of its Language Manager. This utility also includes as default some translations for frequently used software elements.
- Translation memory. It allows the user to store translations for future re-use
- In allows information sharing among several people working on the same project
- There are two versions. The standard one includes 18 European languages, the advanced one adds 6 Eastern languages to those 18.

OpenTag (http://www.opentag.org/otspecs.htm) is not a specific tool but a highly recommended protocol for its use in an i18n context. It is suitable for its use in combination with a TM.

" Translation Memory (TM) is a leveraging solution based on the storage and retrieval of previously localised material. TM contains both content (text) and mark-up (formatting), and can provide statistics on the re-usability of material across product releases or between project components." [ILE]

"There are many factors to consider in deciding whether to use TM. For example, what are the volume and frequency of updates for your localised materials? What is the expected life cycle of your localised product, and what is its relationship to other products you may want to localise?" [ILE]

OpenTag has also been proposed as the standard to be used for Translation Memory eXchange (TMX, http://www.lisa.org/tmx/index.htm). TMX is a standard format for the exchange of translation memory data files between different TM tools.

1.2.5 Character expansion

English characters:	Additional space required:
1-10	200-300%
11-20	100%
21-30	80%
31-50	60%
51-70	40%
70+	30%

In German the Edit menu is called Bearbeiten, up from 4 characters to 10, or an expansion of 250%. Undo is Ongedaan maken in Dutch, up from 4 characters to 14. Printer setup in French is Configuration de l'imprimante...

1.2.6 Non-textual elements: Dates and numbers

Apart from pure text translation, localisation often involves some other aspects that need to be taken into account in order to adapt a certain piece of software or a web site to a new locale. Some of the most common elements that need some kind of change are dates, time and numbers.

Number representation. A transformation is needed. For example, the internal number could be 12345.67 and the external representation could be 2,345.67 or 12.345,67.

Dates (including time). A transformation is needed. For example, the internal representation could be 19951231 and the external representation could be

December 31st 1995, or 31-12-1995.

1.2.7 Hot Keys

Hot keys are combinations of ALT or CTRL key plus a character and they perform the same function as clicking with a mouse. Hot keys must be kept consistent throughout a software application and at the same time they must be unique for a particular function. Sometimes there already exist some typical hotkeys used for common commands in a certain language. In case there are some standards hotkeys those should be preferred, as users of that language would already be familiar with common ALT+character combinations.

1.2.8 Cultural issues

Icons have different levels of acceptance depending on the culture. A native from the target country should check for any offensive symbols.

Changes on UI structure. For instance Arabic languages need a different UI layout. The file menu should be on the right, as opposed to the Western layout

Colours. They should be changed and adapted to the target culture. Pastel colours are more appropriate for Asian countries, whereas more bright colours are suitable for Latin American countries

Culture specific examples should be avoided as these pose great difficulties for translation or to find equivalent examples in other languages/cultures. Religious references are also problematic and assuming a certain religion should be avoided.

1.2.9 Currency

Software adaptations for supporting single European currency

Euro-dollar. This site lists several tools that will help with the Euro conversion. At the moment and up to the year 2002 dual currency must be allowed for European countries

http://www.euro-dollar.com/

1.3 Stage 3: After Translation

1.3.1 Testing

Testing needs to be done at different levels.

1.3.2 Linguistic testing

It is normally carried out by the translator him/herself. The first priority of this type of testing is to check whether all the visual information has been properly translated into the target language. A more detailed testing involves checking that all accented characters are properly displayed, text in dialog boxes, buttons and menus, hot keys.

Menus, dialog boxes and buttons must be big enough to accommodate all the translated text. Hot keys must be unique, i.e. different characters must be used for different functions, and at the same time they need to be consistent throughout the project and agree with any already existing terminology.

1.3.3 Technical or functional testing

Functionality testing often means installing and running a software program in the localised languages to check it works and it does what it was meant to do in the original version. In case of HTML files this test involves checking that all the links are updated and working. When translation is carried directly in the source files, it is common to find broken links in the case of HTML files or bugs in software programs.

1.3.4 Tools for testing

This tool is designed specifically for HTML comparison and QA, not useful for software testing

HtmlQA http://www.tcraft.com/

HtmlQA is a quality assurance tool used to check the functional identity of two html files or groups of files

"Our tools' development team has written two of the industry's standard QA tools, HelpQA, for cross-language verification of RTF-based Winhelp systems and HtmlQA, for verification of Microsoft's HtmlHelp and more generic Html-based information systems. In addition to these, we have developed HtmlCAT, a full-blown Html editing environment with built-in project management, preview, translation memory. We are also shortly going to be releasing LocSmith, a complete software localisation environment for RC file. Finally, if you're involved in user-interface testing, you should check out ToolProof which fully automates all of the most time-consuming aspects of running interface acceptance testing."

1.3.5 Formatting

DTP and format checking. All translated files need to be Desktop Published after translation. Sometimes translators carry out DTP work themselves, but most agencies hire DTP staff specifically for this job. Sometimes formatting elements get altered during translation resulting in divergent layout between languages. DTP operators will ensure that files look alike, that margins, tables, graphics, etc are in place. This stage is particularly important for Technical manual translation specially when documentation needs to be in printed format. In some cases a format checking is carried out after DTP work is done. This consists on manually comparing printed original and translated documents in order to spot any differences that DTP may have missed.

1.3.6 Updates

In the software industry is very common to have updates and new version several times pre year. Being able to re-use previous translations will save a lot of money and effort. This is yet another good reason for the use of translation memory tools. The translation memory has kept all the existing translation on its database, so when new material comes in for translation, the TM scans it and finds all the exact or fuzzy matches. Translators and/or project managers can thus have a quick appreciation of how much text needs to be translated from scratch.

1.4 Links

Software Localisation Interest Group SLIG

http://lrc.csis.ul.ie/SLIG/SLIGmainFR.html

Localisation industry Standard Association

http://www.lisa.org/

The Localisation Institute

http://www.localization-institute.org/

International Language Engineering

http://www.ile.com

W3C Localisation/Internationalisation,

http://www.w3.org/International/Overview.html

2 Integration of external localization processes

This section is intended to discuss the effect that the use of external agencies to provide localization services will have on the architecture of multilingual web sites. However this section can not be fully developed at this stage of the project since the architecture of the web sites has not been clearly defined yet. This discussion will understandably be continued again along next tasks related to the architecture specification and the demonstrator service creation. Moreover it is important to mention that defining a multilingual web site architecture is influenced by the rapid evolution of web technologies and capabilities and the emergence of internet applications as electronic business, etc. To that end we will discuss the localization tasks that may be done internally and the kinds of services that can be provided by external agencies. Of course depending on the tasks that will be done internally and externally the architecture of the web site can be defined differently.

2.1 Localisation process

A complete and useful description of the typical localization process is given in a previous section *("The typical localization process").* A localization process takes a web site and adds features and elements that match the target culture. The localization process involves translation of menus, messages, help scripts, manuals, on-line tutorials, user input/output interfaces, etc. In case the web site has already been created, it is necessary to find out about how material was organized when it was first designed. The localization process can also be integrated from the first design steps of the web site. In this case the localization process can include useful ways of managing data, files and file contents, like keeping translatable parts independent from the rest of the web site, thus influencing considerably the web site architecture.

2.2 External services of localization

Most of the Localization tasks are professional and may be entrusted to specialists at least for the initiation of the localization. Updates and adaptation tasks can later be assigned to non-specialist people. Localization involves different parties from both the client and the external agency (see earlier text on localisation issues). It also involves different aspects of technicality: technical aspects related to software tools that may be

able to manage multilingual information, pragmatic aspects related to knowledge of languages and cultures and linguistic aspects related to text translation. Technical aspects include specific multilingual input/output devices (keyboard, terminal, printer, etc.), specific multilingual software applications (text editor, HTML editor, emailer, etc.), specific multilingual information encoding (language encoding, character sets encoding, etc.), specific features such as fonts, typography, etc. Pragmatic aspects include knowledge of languages and cultures, knowledge of country tags, marks, etc. Linguistic aspects include competence in translation, resolution of ambiguities, etc.

A large collaboration between the client and the external agency is necessary to carry out an efficient localization process. The agency needs precise information about the parts of the web site to be localized. Ongoing communication between the client and the localisation agency are required. Indeed both the client and the agency can not achieve the work without mutual consultation.

3 Overview of MT techniques

Translation of text is likely to be the most expensive aspect of creating and maintaining a multilingual WEB site. There is therefore a strong incentive to use an automatic method of translation. However, machine translation cannot (and will not for the foreseeable future) produce translations which approach the quality of those produced by humans. Each method of translation has its strong and weak points. If these are borne in mind, then there are places where machine translation might be useful.

This section presents an overview of general aspects related to machine translation with a description of different techniques: bilingual, transfer, interlingual and corpus-based techniques, including translation memory, statistical and example-based models.

Machine translation (MT) is the application of computers to the task of translating texts from one natural language to another. One of the very earliest pursuits in computer science, MT has proved to be an elusive goal, but today a number of systems are available which produce output which, if not perfect, is of sufficient quality to be useful for certain specific applications, usually in the domain of technical documentation. In addition, translation software packages which are designed primarily to assist the human translator in the production of translations are enjoying increasing popularity within professional translation organisations. Comprehending the enormous complexity of translating human language and the inherent limitations of the current generation of translation programs is essential to understanding MT today (European Association for Machine Translation, http://www.eamt.org/).

MT systems are designed according to one of the following parameters: coverage and reliability (Carl 1999). An MT system can either be designed to reproduce for a small language segment i.e. a sub-language or a controlled language with high fidelity and precision or it may be designed to perform informative, general purpose translations. In the former case, the system will have high reliability, whereas in the latter case, its coverage will be high. However, both properties are, to a certain extent, mutually exclusive.

- Coverage refers to the extent to which a great variety of source language texts can successfully be translated into the target language. At minimum, a successful translation could be described as informative in when it allows a user to understand more or less the content of the source text.
- Reliability refers to the extent to which an MT system approaches an “ideal” translation (of a restricted domain) for a given purpose or for a given user. A reliable translation is user-oriented and correct with respect to text type, terminological preferences, personal style, etc.

3.1 Difficulties with translation

3.1.1 Ambiguity

There are various kinds of ambiguity. The most obvious is word sense ambiguity. Take any common word, look it up in a good dictionary and you will find several meanings for it. The more common and apparently insignificant the word, the more likely it is to have a long entry in the dictionary. Words like "of" or "be" will usually

have many senses listed. For a good overview of the difficulties with Machine Translation see Arnold and Balkan (1994).

Word sense ambiguity causes two problems to the designer of a machine translation system; one is how to represent the fact that the same word may have several meanings, the other is how to select the appropriate meaning.

Simply listing the possible senses of a word is one way to represent the different meanings of a word. However, it is not clear how many entries a word should have in order to characterise all its possible senses. For example, if we are considering the dictionary entry for a word such as "bank", we might decide that there are the following senses;

1. A financial institution
2. The side of a river
3. To place one's financial business with a financial institution
4. To rotate (as of an aeroplane)
5. Various colloquial phrases ("to bank on something")

Remembering that our aim is to translate, we find that when we want to translate the word "bank" into French, it would be useful to know whether it was a high-street bank or a merchant bank (caisse d'épargne, banque). In this case, we could consider sense (1) of bank to be further subdivided. So for translation, whether a word appears to be ambiguous or not in the source language could depend upon the granularity of the distinctions that are made in the target language.

Word sense ambiguity can become more subtle if we allow the distinction say, between the financial institution and the building containing the financial institution. For example, if I say "my bank refused to extend my overdraft", I would be referring to the institution (or perhaps more correctly, to someone speaking on behalf of the institution). If I said ,"A bolt of lightening hit the bank", I would be referring to the building. Ambiguity between closely related word senses is called "polysemy" (Pustejovsky 1998).

Another kind of ambiguity is "structural ambiguity". Structural ambiguity is where, even if the precise meaning of each word in a phrase is known, it is not possible to tell what the structure of the sentence is, and so you cannot interpret it properly. For example, the two sentences have two different syntactic structures;

I went to the park with swings

I went to the park with my brother

In the first case, "the park" is associated with "swings", in the second case either "I" or "went" is associated with "my brother".

When languages are very similar in structure, say English and French, it may be possible to translate a text even if its syntactic structure cannot be resolved. In general though, syntactic structure will be important for choosing the right translation.

3.1.2 Semantic spaces

Another difficulty with translation is that different languages associate words with different concepts (see Nagao 1989); that is, their partitioning of reality may be different. This means that considerable paraphrasing will need to be carried out to the

able to produce a text which conveys roughly the same meaning as the original. Such paraphrasing could require extensive background knowledge (Nirenburg 1992).

Related to the problem of semantic spaces is that of missing vocabulary. Consider translating the phrase "how quickly?" from English into French. French has no equivalent for the word "how" in this context. To find a translation, it is necessary to paraphrase "how quickly" as "à quelle vitesse?" (which is literally "at what speed"). To be able to carry out such paraphrasing in a general way, the translation system will need to know that there is a relationship between the concepts associated with "quick(ly)" and "speed". This could mean that the translation system would have to store numerous relations between concepts in order to have sufficient knowledge to paraphrase where directly corresponding words are not available in a particular language.

3.1.3 Other aspects of translation

Suppose we wish to compare two texts in two different languages to say whether they represent accurate translations of one another. We might compare them in terms of their truth conditions. That is, in a purely logical sense, are the statements made in the texts true and false in exactly the same situations? For example, the statement "Romeo loves Juliet" is true in exactly those situations in which "Juliet is loved by Romeo". So a translation which switches from active to passive mood would be accurate in a truth conditional sense.

Language is a communication medium, and as such to decide what a good translation is, we need to go beyond truth. The theme of "Romeo loves Juliet" is Romeo, since this is a statement about Romeo. However, the theme of "Juliet is loved by Romeo" is Juliet, since this is a statement about Juliet. The communicative value of these two statements is therefore not the same (Halliday 1994). A translation which is sensitive to these communicative aspects of language will clearly be better than one that is not.

Normally, theme is marked in English using word order. In other languages, such as Japanese, theme is marked using a particle. To translate with any kind of accuracy between English and Japanese, it will be necessary to identify the communicative function of a phrase as well as its logical content.

There are still more subtle aspects to communication, such as choosing the right register and adjusting the rhythm of a particular piece of language. When adding subtitles to a film for example, it is necessary to choose the translations in such a way as to try to keep the flow of the subtitles in step with the images.

3.1.4 Referential ambiguity

This kind of ambiguity is related to referential structures (anaphora). For example in the sentence: *Mark took his car*, it is not obvious whether Mark took his own car or another one.

3.2 Generations of machine translation systems

Machine translation systems are classified into 3 generations:

3.2.1 Systems of the first generation

The main feature of these systems is the fact that different translation programs are designed for each couple of languages (bilingual translation). Moreover there is no

separation between the programs and the linguistic data. They are based on linear (non-arborescent) data structures and do not use real computational linguistic methods such as regular languages or syntagmatic grammars. The *Systran* ancestor (developed at Georgetown University) is one of these systems.

3.2.2 Systems of the second generation

The main feature of these systems is that the translation process is developed into three different stages: analysis phase, transfer phase and generation phase. The analysis process transforms the source text into a source structural description which is transformed into a target structural description at the transfer phase and then to a target text at the generation phase. The systems of the second generation separate linguistic data (lexicons and grammars) from the processing programs. But these systems are not powerful at the semantic level. The first versions of the *Ariane* system (developed at GETA[1] Grenoble and used in the current UNL project) are classified in this generation of machine translation systems.

3.2.3 Systems of the third generation

The main feature of these systems is the ability to understand the meaning of a text before its translation. Due to the complexity of the natural language, these systems are generally dedicated to specialised and controlled languages. They are based on artificial intelligence techniques (expert systems and linguistic knowledge bases) to represent the semantic information of the texts. There exists no real functional system belonging to the third generation of MT systems.

3.3 Translation categories

Translation is categorised into four types where a computer and a man can collaborate:

3.3.1 Machine aided human translation (MAHT)

The MAHT translation consists of using a word processing software completed by electronic dictionaries, which can be improved during the translation. Translations are human-made.

3.3.2 Human aided machine translation (HAMT)

This category of translation requires a human assistance before and after the automatic translation (pre-edition of the source text and post edition of the target text). The Canadian *Meteo* system is classified into this category of translation.

3.3.3 Interactive translation (IT)

In this category of translation, the system translates with an interactive human assistance. For each ambiguity problem during the translation process, the system asks for a human disambiguation. *Alps* is one of the interactive translation systems.

[1] Groupe d'Etude pour la Traduction Automatique, IMAG, Université Joseph Fourier, Grenoble, France.

3.3.4 Machine translation (MT)

Theoretically the machine translation aims to completely avoid the human assistance to the system. Nowadays, no one of the existing machine translation systems can be qualified as being a MT system.

3.4 Machine translation techniques

In this section we introduce a descriptive presentation of the different machine translation techniques. These techniques are based on different models: bilingual, transfer, interlingual and corpus-based model which includes the memory-based, statistical-based and example-based models.

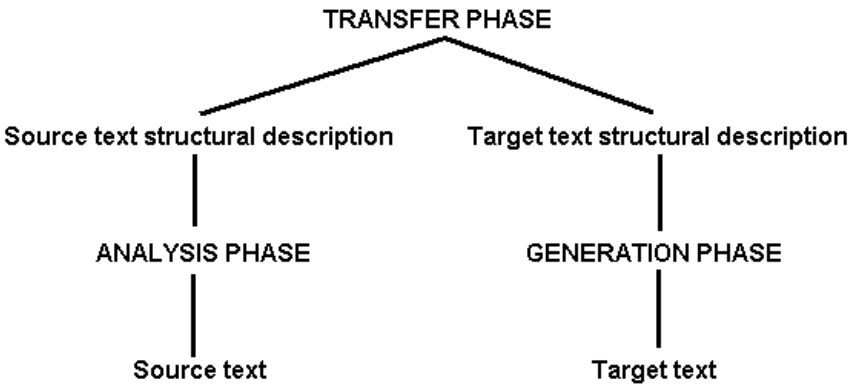
3.4.1 Bilingual-based machine translation

A bilingual machine translation system is dedicated only to a pair of languages and can not be adapted to other languages. Indeed the translation process is built according to specific characteristics of the two languages. A source text in one language is analysed to be specifically generated to another language. The transfer phase is minimised to bijective lexical and syntactical relations. It is understandable that the programs may be dependant on the language pairs making difficult their adaptation to new languages. The *Systran* system is a collection of bilingual sub-systems dedicated to different language pairs.

3.4.2 Transfer-based machine translation

The transfer translation model is built on three modules:

- Analysis module that transforms the source text into a source structural description.
- Transfer module that transforms the source structural description into a target one.
- Generation module that transforms the target structural description into a target text.

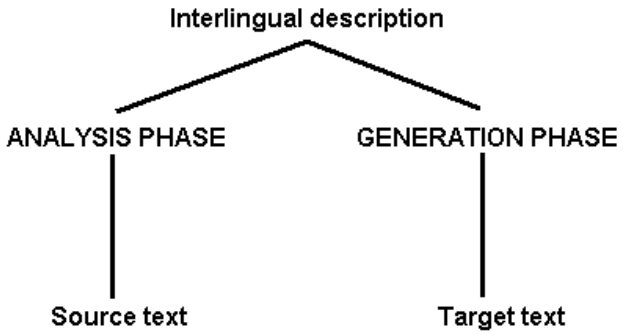


Eurotra European project system was built in the principle of the transfer model. It aimed to translate from/to the nine European languages. For each language, analysis, generation and transfer modules according to other languages have been developed..

3.4.3 Interlingual-based machine translation

The interlingual translation model is built on two main modules:

- Analysis module that transforms the source text into an interlingual description.
- Generation module that transforms the interlingual description into a target text.



The Ariane system developed by GETA research group is representative of this concept (pivot language) that is being used for the current UNL project.

3.4.4 Memory-based machine translation

Machine translation based on the “translation memory” is a corpus-based approach. It is dedicated to professionals or experts in the translation services. The system does not really analyses the source text to translate but just reuses possible translations previously stored by the professional translator. For the parts of text that have not been previously translated, a terminology (dictionary) support is used to help the expert to translate them. This “new” translation concept offers a computer-assisted translation that automates repetitive tasks, freeing the professional translator to attend to the finer points of translation that require the judgement of an expert.

The *IBM TranslationManager* (http://www.software.ibm.com/ad/translat/tm/<@ibmtm>) is one of the systems based on that concept. As an example, the following figure (Schmidt 1998) shows the translation environment of IBM TranslationManager with the translation editor, the window for the translation memory proposals and the window for terminology support.

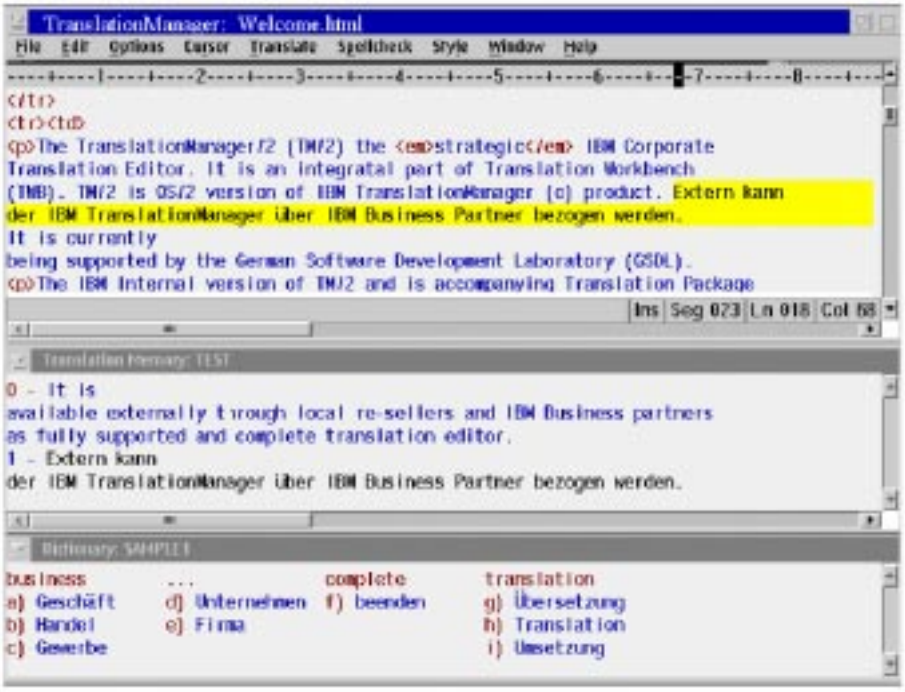


Figure extracted from the IBM web site
http://www.software.ibm.com/ad/translat/tm/tama/epaper.htm

Trados Translator's Workbench is a Translation Memory database which uses different translation editors/translation front ends for different file formats (http://www.trados.com/workbench).

3.4.5 Statistical-based machine translation

Statistical-based machine translation is a corpus-based approach. Statistical concepts are among the first techniques for machine translation. They was proposed by Warren Weaver in the early 1949's but that theories foundered on the rocky reality of the limited computer resources of the day. In the late 1980's IBM researchers felt that the increase in computer power made reasonable a new look at the applicability of statistical techniques to translation. Statistical machine translation was re-introduced by the *Candide* group at the *IBM Watson Research Center* (Berger 1994). The principle of this translation concept is that the computer inspects large collection of translated data and, from the collection, "learns" how to translate. However the statistical techniques are no longer encouraged in the machine translation domain.

3.4.6 Example-based machine translation

Example-based machine translation allows to rich systems. Translation examples are stored as feature annotated and sometimes structured representations. Translation templates are generated which contain (weighted) connections in those positions where the source language and the target language equivalences are strong. In the translation phase, a multi-layered mapping from the source language into the target language takes place. Sentences are more finely decomposed into phrases and linguistic constituents e.g. NPs, PPs, subject, object, etc. The example-based approach

can make use of morphological knowledge and relies on word stems as a basis for translation. Translation templates are generalised from aligned sentences by substituting differences in sentence pairs with variables and leaving the identical sub-strings unsubstituted. An iterative application of this method generates translation examples and translation templates which serve as the basis for an example-based MT system.

4 Current MT systems

In 1996 the European Commission charged Equipe Consortium Ltd with the task to make a survey of existing MT products and services. According to their Web site

Equipe claim to be the only international consultancy specialising in the analysis, forecasting and tracking of markets for language technology (LT) in the communications, Internet/intranet, data warehousing and business applications sectors.

Although the study is almost four years old, and the field is changing rapidly, most of the conclusions are still valid:

1. Only a few products are plausible for handling EC translation loads
2. A few "niche" products can handle new pairs (Finnish, Danish, EE and Asian languages)
3. Development of new pairs needed by the EC will probably come from established vendors
4. The competition form of evaluation is acceptable to vendors
5. The most established services are currently provided by product vendors, but these are rudimentary
6. No commercial service organisation currently exists which is remotely comparable to the Commission's SdT
7. It is likely, however, that such organisations will emerge in the next couple of years.

Our own survey of the market for MT products has confirmed the existence of a very large number of niche products. Many of these products feature a small number of language pairs. Moreover, the domain in which the products can work are very narrow. Many products are effectively electronic versions of the conventional phrase books or letter template books that were used by secretaries in the era of electrical typewriters. Because these niche products are very unlikely to be of wide use in the construction and maintenance of multi-lingual web sites, we will not include them in this survey.

Contrary to what one might have expected the number of companies that offer professional MT systems that are, at least in potential, able to handle a large number of languages and many or broad domains has only diminished since the Equipe Report. A number of companies that were present on the market in 1996 have disappeared, mostly because of mergers and acquisitions. Especially Lernout & Hauspie has been active in acquiring companies in the field of translation, be it human or machine. As a consequence, it seems that only two sources of professional MT services remain, viz. L&H and Systran. In addition to these companies there is a number of independent companies which offer multi-lingual document processing services. One of these companies is LANT, that in its turn incorporated much of the technology and staff that originally worked at Sietech and GMS. Although it may not be very relevant for individual customers, it is very likely that LANT uses the MT technology inherited from GMS to make their multi-lingual document processing more efficient.

In a discussion with L&H in August 1999 L&H explained that the acquisitions of the last couple of years have left them with a large number of tools, resources and modules that will eventually grow into powerful MT systems that handle multiple

language pairs. For the immediate future, however, the tools and resources are not simple to combine, if only because of their independent origins.

L&H explained that they want to avoid the impression that broad domain, high quality, fully automatic MT is feasible at the moment. On the contrary, L&H aims at offering document processing services to international companies. In doing so, they use existing and expressly developed resources and modules to support human translators. L&H strongly advises to leave translation of important documents (like company descriptions directly accessible from the home page) not just to humans, but even to native speakers of the target language who are knowledgeable in the domain. It seems that this opinion is shared by virtually all serious companies who are active in the field of multi-lingual document processing.

According to L&H most of their translation business is in the form of consultancy for large multi-national companies. For each company customised terminologies are constructed, that form the basis for all kinds of computer aided translation. Such aids can range from on-line dictionaries and terminologies, through translation memories to fully automatic MT with human post-editing.

L&H recommend to limit fully automatic MT to on-line and on-demand translation of pages, under the assumption that the person who demands translation knows the subject domain well enough to be able to interpret the result. L&H is working on a product called iTranslator, that will be offered as a tool that can do on-demand translation between a large number of languages. For the time being, iTranslator is targeted at Intranet systems in companies which use L&H as their major provider of multi-lingual document processing services. In addition, L&H offers PowerTranslator, a product that can also be deployed for translating web pages, e-mails, etc. PowerTranslator is strongly focused on English as the source or target language.

It is to be expected that most of the older ‘trademarks’ like Globalink will disappear, as L&H proceeds to integrate the technology that came with its acquisitions. Globalink still has its own url, but that refers directly to PowerTranslator (and another consumer product SimplyTranslating marketed by L&H).

The product range of Systran is comparable to that of L&H. Here too, there is a strong focus on English as source and target language. Contrary to L&H Systran does not seem to provide their own translation services. Instead, they team up with other professional companies, like Berlitz and Trados, to provide such services. Systran also provides the translation engine for AltaVista’s BabelFish service.

The number of lines dedicated to L&H compared to the number dedicated to Systran should not be construed as support for the opinion that the former is more powerful, more promising, delivering higher quality, etc. than the latter. No comparison of the quality of the products and services provided by the two companies is intended or implied.

It is unlikely that another company with a product range comparable to Systran and L&H will appear on the market in the near future. The only exception might be Japanese or US companies backed by very large corporations. However, we have not seen signs of such newcomers.

4.1 Machine Assisted Translation

As was already said above, this document does not intend to give an overview of the extremely large number of niche marker tools for Machine Assisted Translation (MAT). Companies and products on this market come and go at a very high speed.

Product categories that are of potential interest for those cases where translation and localization is mainly performed by human experts include translation memories and term bank builders. These classes are closely related: they help to avoid repeated time and effort for the translation of the same sentence, clause or 'terms'. In addition to saving time, the use of these tools also improves consistency. Perhaps this spin-off (improved consistency) is the most important contribution of these tools.

For reasons which are easy to explain the MT research community has not devoted much attention to the intregration of the translation software into commercial word processing software. Yet, this lack of integration, which effectively forced translators to learn a new editor for each translation system (and most of the time these editors were clumsy and came with very limited sets of features) has been perhaps the major cause for the lack of success of early MT systems. Modern MAT tools are tightly integrated in the popular word processors. However, to be really useful in the translation and localization of Web pages it is essential that the tools are able to correctly handle mark-up information, on all levels. In the evaluation of MAT tools for use in translation and localization of Web pages the way in which they handle mark-up information is decisive.

Recently, an increasing number of MAT tools for Web applications have started to appear. It is very important that the MT system (probably some component that does a preliminary analysis of the document) detects all mark-up in a document to be translated, and subsequently handles the mark-up text properly. What proper handling is, may be difficult to predict. There are several classes of mark-up text that should never be touched (e.g., those that determine character size, heading level, etc.). Other classes of mark-up text may or may not need translation, or adaptation, for that matter. For instance, if the mark-up text is a url, then the proper handling depends on the question whether this url must be maintained, or rather must be updated to point to the language specific address. Not all MT applications may handle mark-up in the same way. And correct handling is all but trivial. Handling mark-up may prove to be a feature that determines the usability of an MT system for the development and maintenance of multi-lingual web sites to an extent that is comparable to translation quality for running text.

Problems with handling mark-up text may easily proliferate, for instance if the marked-up phrase in the source language is discontinuous, but translates into a continuous phrase in a target language. The reverse situation is probably even more difficult: if a continuous phrase in the source language trans;ates into a discontinuous phrase in the target language.

For localization of Web sites translation tools should be able to detect text elements that need special treatment. For instance, North American Web sites are likely to give temperatures in degrees Fahrenheit, whereas most European countries are used to degrees centigrade. Another example of this sort is the conversion between inches/feet and metric measures (remember the Mars vehicle disaster). One might also want to see measures for cloths and shoes converted between countries. Special attention is needed to currencies. When localizing a Web site that contains prices one might want to have two figures, one in a currency that the reader is accustomed to, and in addition the price in the currency of the country where the price must be paid. For EURESCOM related travel: one might want to know not only the price of a taxi ride in Porto in Euros, but also in Escudos, if only to be able to decide how much cash one must have handy on arrival at the airport. We are not aware of any MAT product that provides support in this area.

Text embedded in graphics is extremely difficult to detect. We have not yet seen a single MT application that does not explicitly say that such texts are excluded from translation attempts.

The cause of the problem is simple and clear: text is only recognized as long as it consists of ASCII characters. In the graphics it usually does not; there the text is rendered as pixels or bit patterns. Of course, this failure to detect text must be taken into account when designing multi-lingual web sites that rely on (partial) Machine Translation.

4.2 Controlled Languages

During our visit to L&H Geert Adriaens stated that the use of controlled language may not have a substantial effect on the quality of the output of an MT system. At first, this statement is quite surprising, Yet, it may very well be true, at least under the conditions that the professional translation services of L&H are used to work in. Geert Adriaens explained that L&H custom builds document handling systems for each large customer. In doing so, they start with the construction of a terminology base. Most probably, this is what determines the quality of any MT system that is subsequently built by putting together whatever syntactic analysis, transfer and generation components that are available for the source and target languages. Once the basis is in place, the fact that use of controlled language prevents overly complex sentences from being written probably makes only marginal additional contributions to the output quality of an MT system. It may very well be that this is just another example of a situation where seemingly simple technology (the terminology) solves so large a part of the problem that any additional piece of sophisticated linguistic intelligence fails to be significant, because the number of sentences where the linguistic intelligence is decisive is way too small.

Geert Adriaens also said that the concept of 'controlled language' is far from unambiguous. It also comprises guidelines like "introduce new concepts slowly', whatever that may mean in actual practice. This may mean that most text that were carefully written with readability in mind qualify as written in a 'controlled language'.

However, this medal also has a flipside: if an organisation does not have the tradition of writing texts according to strict guidelines that should maximize readability, and if that organisation cannot afford the very substantial burden of building a comprehensive terminology, then the introduction of 'controlled language' (which implies a large degree of system in the use of terms) should help to improve the quality of MT.

In summary: it is not possible to make very broad and general statements about the relation between controlled language and MT quality in terms of a simple cause-effect scheme. However, it is clear that there is a relation, and it is possible to understand and explain it, provided that one knows the relevant facts and details.

For our own work in BabelWeb, where it is probably not possible to build a comprehensive terminology, the use of some kind of constrained language for the texts that must be processed by a MT system will probably help to improve the output quality. More even, it may very well be the case that avoiding specific words or constructions that are known to cause problems for a specific MT system will make for a substantial improvement of the quality of the MT output.

This is all very trivial, perhaps, Yet, we have asked the question about controlled languages, because we did not follow the line of reasoning explained above. Moreover, not too long ago I saw a similar question appear on the MT-list.

5 Improving MT for WEB sites

The aim of this project is to discuss the various technologies that are available to help create and manage multilingual WEB sites. Clearly, automatic, or semi-automatic machine translation would help reduce the cost of maintaining a site in several languages. However, as was made clear earlier, machine translation technology is not yet at the point where it can be used to produce high-quality translations. The aim of this section is to discuss what can be done to help a machine translation achieve a higher level of quality.

5.1 Customised MT systems

Machine translation systems which are intended for use by professional translators (as opposed to simple desktop systems) usually allow the user to configure of various aspects of the translation process. The most obvious of these is to be able to add new words to the lexicon. However, we may in addition wish to alter the grammar rules, the sortal restrictions and the transfer rules (if there are any).

One example where considerable effort has been expended to configure a machine translation system is in the European Commission (EC). The EC employs around 1500 full time translators (human) and therefore has a considerable incentive to improve their efficiency. The EC uses the "Systran" translation system (mentioned earlier in this document). The EC has bough a developer license for Systran and has extended the vocabulary and lexical rules to such an extent that it bears little resemblance to the system that they originally purchased.

The latest version of Globalink's (now Lernout and Hauspie's) Power Translation is a transfer type translation system and allows a considerable amount of configuration by the end users. As well as extending the vocabulary, the user can alter the grammar rules, change the various parameters that are in the grammar rules (such as restrictions on which semantic markers can combine), create new semantic markers and alter the rules used in the various stages of translation.

Configuration of a system to translate the various words and syntactic structures used in a particular domain correctly can improve the quality of translation significantly. However, it is surprising just what variety of language is used even in specialist areas. When CompuServe first provided fully automatic translation of their discussion forums, they though that translation quality could be improved by tailoring the translation system to the subject being discussed in the forum that it was translating. Unfortunately, they found that there was such a wide variety of language being used that the system performance was not improved.

5.2 Controlled languages

Some companies (e.g. Caterpillar, Perkins and many others) employ so-called controlled languages for producing manuals (Newton 1992, Bernth 1998). These are formally prescribed subsets of (usually) English which are less ambiguous and more consistent than normal language. The reduced ambiguity not only makes the documents clearer to non-native speakers, but also reduces the time and cost the translation process.

Controlled language help to eliminate both the word-sense ambiguity problem and the structural ambiguity problem. For example, in a formal language we might always

have to use the word "right" might only have the meaning of opposite to left, and the word "correct" might have to be used for the other sense of "right".

Controlled languages also attempt to reduce structural ambiguity by prescribing a relation between linear word order and syntactic structure. For example, the phrases,

"The girl on the swing in the park"

and

"The girl in the park on the swing"

have very similar semantics, and yet the order in which the prepositional phrases attach is different. In a controlled language, one order of attachment only would be allowed, so the second of these two forms could only mean that it was the park that was on the swing and not the girl.

Controlled languages are usually supported by a controlled language editor. The editor will indicate phrases that are allowed in the controlled language and will verify the intended sense of phrases that might be ambiguous.

Because controlled languages reduce the number of word sense ambiguities and use a smaller range of grammatical structures than an uncontrolled language, it is possible to tune machine translation systems to take this into account. Certain word senses could simply be removed from the lexicon (such as "right" meaning "correct") and grammatical structures could be interpreted in a much more reliable way. See Adriens 1995 or Almqvist 1996 for descriptions of systems that have used controlled languages to improve Machine Translation.

In practice, there are a few problems with using controlled languages.

To be able to consider the use of a controlled language, you need to have control of the authoring process. Many WEB sites pass on information provided by third parties. In such cases it would be impractical to persuade all the information providers to use a controlled language. The only alternative would be to re-edit the text after it was received.

Another disadvantage is that a considerable amount of interaction could be necessary to produce a text in a controlled language which conveys the authors meaning. Controlled language editors give various amounts of feedback to the author to help in this process. Some will detect errors and make suggestions for re-wording, others will just say that a particular phrase was not acceptable and give very little indication as to why.

5.3 Word sense annotation

One possible compromise between using controlled language and allowing free text, might be to tag ambiguous words with their intended sense. This is not a conventional technique but it could be a simple way of reducing word sense ambiguities. To achieve this some kind of text annotation tool would be required which had access to the translation systems lexicon. The tool would look up each word in turn in the lexicon, present the user with a list of descriptions of the possible meanings of the words, and then add the appropriate annotation to the text.

The translation system would need to be modified so that it could understand the annotations that were added, and also, some gloss text would have to be added to every ambiguous lexical entry.

5.4 Interlingual representation

Suppose a language could be created which was unambiguous and yet could represent the content of a text written in a normal human language. Such a language is called an interlingua (as discussed earlier). If we could author text to be presented on a WEB page in such a language, then translation would be a (relatively) simple matter of rendering the interlingua as text in the appropriate language.

The Distributed Language Translation (DLT) project [72, 73] was a multilingual machine translation (MT) project, that ran from 1984 to 1990. DLT used a fully developed Interlingua (IL), viz., a slightly modified version of Esperanto.

The project was proposed by BSO (Buro voor Systeemontwikkeling), a Dutch software firm. Funding of $10 million came from BSO and the Dutch government. A prototype program, designed to translate texts on airplane maintenance in a simplified English, was completed in 1988. DLT was discontinued because of the failure to find an industrial partner willing to invest in further development.
The most difficult problem in translating by computer is the indeterminacy of meaning in natural languages. DLT approached this problem by considering the context associated with any word being translated and by using a knowledge bank of relationships between words.

Ambiguity in natural language was handled in two ways. First, a knowledge bank is consulted. If there is no precise match in the knowledge bank with any of the candidate translations, an algorithm known as SWESIL (Semantic Word Expert in the IL) calculates the semantic proximity of the natural language words in the knowledge bank to the Esperanto candidate words. In DLT, several different methods of computing the most appropriate meaning were tried.

If the automatic analysis failed to disambiguate the input sentence, the writer could be asked to reformulate his/her thoughts, in order to remove the ambiguity. This function is naturally combined with the automatic check of adherence to the rules of controlled language.

Interlingual translation systems are very unusual in commercial systems. One exception is the Fujitsu "Atlas" system which claims to use an interlingual representation.

Usually, the terms "interlingual translation" and "knowledge based translation" are seen as synonymous, but this need not be the case. There can exist shallow interlingual systems which carry out little more by way of analysis than the surface analysis used by transfer systems.

The two best known research systems that use an interlingual approach to translation are the "Kant" system being developed at Carnegie Mellon University, and the "Mikrokosmos" system being developed at New Mexico State University. The development histories of these two systems are intertwined.

Of most interest to this project is the "Kant" system, since it makes use of a controlled language to improve translation quality. It is proposed that the author be involved in a pre-processing phase in which words or other text (numbers, dates, names etc.) can be annotated to reduce ambiguity.

One way to generate text in multiple languages is to translation text templates by hand and have a text generation system fill in the gaps from data that may be changing frequently or is otherwise uneconomic to have translated fully by humans. This is a

kind of interlingual system in the sense that the data from which the text is being generated is language-independent.

5.5 Translation memory

One of the characteristics of information on the WEB is that it changes very quickly. If each new page added to the a WEB site had to be translated from the beginning the ongoing cost of maintaining a multilingual WEB site would be very high.

Translation Memory has been described elsewhere in this document. Translation Memory is particularly advantageous where,

- There is a high degree of repetition
- Consistency is required in the control of terms

When information on a WEB site is updated, it is very likely that only a percentage of the text will have changed. This means that it would be highly inefficient to have the whole of the text translated without regard for the original translation. A Translation Memory can be used to automatically translate those elements of text that have already been translated. Where the Translation Memory only finds a partial (fuzzy) match to previously translated text, it can show this to the translator, highlighting the parts of text that are different. This is useful for maintaining a consistent terminology throughout a WEB site.

It is likely that different translators will be used at different times to translate the WEB site. If they are working without a Translation Memory, it is likely that their translations of similar phrases and terms will differ from one another. When using a Translation Memory, because the translator is instantly prompted with the previous translations used, it is much easier to maintain consistency, both across different pages on the same site, and across different translators.

As discussed earlier, Translation Memories often come with various filters which can understand text in different formats, and usually they work as an integral part of common document editors (such as MS Word). Terminology Management utilities are also available, which allow the (semi-)automatic creation of terms that have been used in the translation process.

The data in the Translation Memory and the associated termbank is a valuable resource that is created (or extended) by the translator. This resource can either be owned by the client or by the translator. If translations are being carried out in-house, then this is not an issue. However, if the client is using freelance, or agency translators then there may be various practical problems. For example, the client needs to ensure that data is stored in a format that they can access and that can be shared with other translators if necessary. To aid this there is a standard for translation memory exchange called TMX which is supported by most translation memory systems.

5.6 Integration of translation memory and machine translation

Typically, the various pieces of software that can help a translator (Translation Memory, term management, machine translation software, filters) are integrated into a "Translator's Workbench". When attempting to find a translation for a phase, a Translator's Workbench will normally first look in the Translation Memory to match a previously translated phrase (with a precision that is set by the user). If a suitable example is not found, the Workbench can pass the phrase to a machine translation

system. In general, previously translated phrases are more likely to produce an accurate translation that a machine translation system.

Most professional translators do not like to use machine translation systems. This is because the systems are so poor that it is often easier to translate the original text by hand, than attempt to re-word the machine output.

It is conceivable that a different way of combining Translation Memory and Machine Translation could result in improved translations at a reasonable cost.

Suppose we have a general purpose machine translation system and we wish somehow to use the translation memory to improve the quality of its output. The most obvious thing we can do, is where there is a 100% match between some previously translated phrase and the current text, we simply use the previous translation and ignore the machine translation system (just like with a translator's workbench). However, 100% matches are likely to be rare, except for that text which remains the same between updates of the WEB site. The problem occurs when there is only a partial match. The translation memory has no algorithm for working out what modifications ought to be made to the target text in the example to correspond to those parts in the source text that do not match. Instead, all partial matches could be given to the machine translation system. The machine translation system will have a number of possible translations (because of the various ambiguities mentioned earlier) and ordinarily does not know which one to pick. However, comparison of the translation memory with each of the possible translations from the machine translation system will yield some that match better than others. The best of these matches could be picked as the "best" translation.

Systems such as the Globalink Power Translator already have an interactive mode, where the possible output texts can be viewed and the appropriate one selected.

Another approach would be to attempt to use the translation memory to improve the translation quality. One way this could be done is to use the translation memory to provide estimates of say, collocation frequency, which in turn could be used to adjust the rules in the machine translation system.

Neither of these methods has been tried (according to the author's knowledge) and so their likelihood of producing useful translations is not known.

Another, perhaps more obvious, way of combining a Translation Memory with Machine Translation is to use an Example-Based Machine Translation system. How successful this is depends on the kind of examples required by the system. As described earlier, some systems require examples that are annotated structurally. It is unlikely to be feasible to carry this out as part of the normal translation process. However, some proposed Example-Based systems only require aligned sentences pairs. This is exactly what is produced by the Translation Memory.

6 Language Generation

6.1 Introduction

The field of natural language generation (NLG) is concerned with the ways computer programs can be made to produce high-quality natural language text from computer-internal representations of information [55]. For the BabelWeb project this definition contains at least two terms that may be problematic. First 'high-quality' is difficult to define; probably, the meaning of the term differs depending on the type of output that must be generated: short captions for figures and tables most likely pose other requirements than descriptions consisting of several paragraphs. Second, the term 'text' suggests that the focus of the research is on multi-paragraph output, whereas texts that are suitable for display on a home page or similar type of web page (or utterances to be produced in spoken language interaction) may require completely different optimality criteria and cost functions.

Text generation is an essential part of many NL applications. For instance, Machine Translation and Summarisation are not possible without some kind of natural language generation. However, in this section we focus on software and tools for generation proper, i.e., modules that can operate with a high degree of independence. For a component of an MT system to qualify as an NLG module it would be necessary that some other module can provide the proper input.

The simplest text or language generation technique involves the use of 'canned' text. In the most primitive form ready-made strings are printed without any change. This approach is successfully used in several operational spoken dialogue systems. Slightly more advanced is the use of so-called 'templates', string patterns that contain empty slots where other strings must be filled in. This is mostly done in a straightforward rule based way. This type of generation is used in several applications, for instance in the automatic generation of fairly standard letters. Linguistic notions do not play a crucial role in this simple technique. Some systems can handle agreement and /or conjunction, but not in a theoretically sound way. This way of language generation may be used in limited domains. The advantages are that the technique is very simple and fast (real time operation). However, this advantage is obtained at the cost of output sentences which are very simple, with no or very little variation. Furthermore, the use of canned sentences and patterns is application specific and the result is not re-usable in other applications. For some applications, these limitations may not be problematic. Alongside the application oriented template techniques principled linguistic approaches to NLG have been developed. These approaches cannot be implemented without some form of reference to a domain, if only because the semantics and the vocabulary must be specified. Although principled linguistic approaches may be less domain dependent in theory, in actual practice the enormous difficulties in developing semantic models, vocabularies and grammars may cause problems in porting between domains of the same order of magnitude as with template based systems.

6.2 Aim of this section

In this document we intend to give a reasoned summary of the state-of-the-art in the field of Natural Language Generation. In doing so, we will always try to keep an open eye for the requirements of the BabelWeb project.

It is explicitly not our goal to provide a comprehensive overview of all R&D activities in the field that have been published over the last couple of decades. In our opinion, such an overview would contribute little or nothing to an understanding of the potential use of NLG software for BabelWeb. Moreover, several good overviews are already available, even if none of them can claim to be comprehensive.

NLG is perhaps the NL component for which it is most clear that a large degree of domain and application dependence is inevitable, given the present state of development and the expected development for the medium term future. The information in this document should convince the reader that for some applications seemingly simple template based techniques are fully adequate, whereas other applications might need the full power of linguistically inspired approaches, and probably even more. Consequently, a detailed analysis of specific approaches is only warranted after BabelWeb has completed the functional specification of the web site(s) that the project wants to design, build and investigate. This approach will remain valid until a fundamental breakthrough in the field is obtained.

6.3 Architectures and terminology

The more scientifically oriented approaches to text and language generation are mostly based on some kind of linguistic theory. All approaches that have received some attention in the community appear to distinguish two or three major stages of generation: single-sentence generation (also called "*realization*" or "*tactical generation*") and multi-sentence generation (also called "*sentence planning*" or "*micro planning*") and content selection (also called "*text planning*" or "*macro planning*"). Sometimes content selection and multi-sentence generation are also collectively referred to as "*strategic generation*". It is not completely clear whether the use of specific terms corresponds with specific theoretical or philosophical convictions. However, for the goals of this summary it is probably not necessary to go into these details.

The need for and the function of a strategic generation component depends very much on the application. If an NLG system is used as part of an MT application, some kind of sentence level processing may often be adequate (for instance, one might be able to copy the same anaphora and cataphora devices in the source and target language sentences). If, on the other hand, the NLG system is part of an application that makes complex non-linguistic information available in linguistic form (e.g. weather forecasts derived from data about wind speeds, barometer pressures, humidity, etc.; or route descriptions from geographical data) the strategic component that is responsible for the selection of the information and the specification of the format in which the information is to be presented is absolutely essential.

Most published full-fledged generation systems (i.e., those systems which comprise both strategic and tactic generation) have more or less the same linear (or pipeline) architecture, in which the following three components (that reflect the terminology introduced above) may be distinguished:

- *Content or text planning* - the information which must be expressed is selected and ordered. Determination of the text structure is often done through schemata or rhetorical relations.
- *Sentence planning* – in this intermediate stage, information is chunked into sentences and the underlying sentence structure is determined. In many cases, this includes lexical choice. Sometimes no separate sentence planning stage is

distinguished, in which case these tasks are dealt with in the linguistic realization module.

- *Linguistic realization* – deep sentence structures are converted to grammatically correct surface forms. Word order, choice of function words, realization of negation, etc. are handled here. Proper morphological forms are chosen.

In practical implementations the modules may be cast in the form of some monolithic piece of software; alternatively, a much more modular structure may have been chosen, in which each of the three major modules consist of several clearly separable sub-modules. Moreover, applications that do not need a powerful strategic generation component may consist of only a single module that takes care of linguistic realization. At this moment there seems to be no general agreement on the most suitable modular structure and architecture of a full-fledged NLG system. However, few systems seem to support architectures that allow for recursive actions of (sub-)modules. Thus, linear control (or pipeline) architectures appear to be adequate.

Despite the absence of recursive control structures, full-fledged NLG systems are generally very cpu- and memory hungry, to the extent that real-time operation becomes problematic; moreover, the use of these techniques requires specialized language engineering knowledge of experts in the field.

In some cases, it may be necessary to combine speed/efficiency with high quality text output. Then, hybrid systems may be used that combine canned text and linguistic generation. Canned text is used for linguistic structures that are difficult to generate. Obviously, their reliance on canned phrases/sentences implies that hybrid techniques can only be used in fairly limited domains.

A survey of publishes NLG systems that comprise both strategic and tactical generation makes the impression that in all cases only normal ‘body text’ is generated. For some applications, especially in the realm of multi-lingual web building, it should be very advantageous if the generation system could make a difference between captions and body text, and produce output enriched by mark-up information. This is somewhat reminiscent of the situation in MT, where automatic detection and proper automatic treatment of mark-up information in HTML/XML and other web documents is of utmost importance. It is our impression that NLG lags behind MT in this respect (except, of course, the NLG modules in products like Systran and iTranslator which are designed to correctly re-generate mark-up information).

6.4 Literature and web sites on NLG

A survey of the literature and the most relevant web sites suggests that NLG is an active research topic, but that relatively few commercial products are available. This may very well be connected to the fact that most applications that need some kind of NLG effectively cover limited domains, while at the same time a large proportion of the generation activities is tightly coupled to more comprehensive applications (like MT), where generation is treated as one component in an otherwise tightly integrated system. Therefore, straightforward approaches that mainly rely on the concatenation of phrases with slots to fill are appropriate in most applications. Using fixed phrases combined with some form of slot filling is probably most efficiently done with application specific software.

As in parsing, where shallow parsers are distinguished from deep parsers, at least part of the research community appears to distinguish shallow from deep generation. Here ‘shallow’ seems to refer to systems that use canned phrases and phrases with slots to fill.

Searching the Internet yielded several 'tutorials' on NLG. A quick scan suggests that most of these presentations are very similar.

There is a full chapter devoted to Language Generation, edited by Hans Uszkoreit, in the Survey of the State-of-the-Art in Language Technology, commissioned by the EC in 1995. This chapter gives a good introduction to the field. It is one of the texts that proposes the three major components of an NLG system mentioned above. Paiva (1998) provides a good overview of a large number of research systems. The list of references also contains several recent books on the topic of NLG.

It may or may not be of interest that the comprehensive report on Multilingual Information Management: Current Levels and Future Abilities, commissioned by the US National Science Foundation, the European Commission's Language Engineering Office and the US Defence Advanced Research Projects Agency of April 1999, does not mention NLG as a topic. A quick scan of the text did not even produce the term in one of the section or subsection titles. However, the report does cover topics like multi-lingual information retrieval, extraction and summarisation, and, of course, machine translation.

To get a more or less complete picture of what is currently done in the world of language generation three www sites are very interesting:

- http://www.itri.brighton.ac.uk/projects/rags

  This site belongs to the RAGS (Reference Architecture for Generation Systems) project, a joint project between the Information Technology Research Institute of the university of Brighton and the department of Artificial Intelligence of the University of Edinburgh. This project has two main goals: provide a generic framework for NLG systems and to establish standards for assessing NLG systems. The site provides a list with publications of which the one titled "Survey of Applied Natural Generation systems" is very interesting (see below).

- http://www.dynamicmultimedia.com.au/siggen/

  This is the site of the ACL Special Interest Group on Generation. It contains all kinds of information related to Natural Language Generation. On the site one may also find software and commercial systems. Unfortunately, this web site does not seem to be maintained at a high level of quality.

- http://www.dfki.de/fluids/General_Overview.html

  The FLUIDS project (Telematics Engineering, TE 2006) was/is devoted to building interfaces for decision support systems. One of the major tasks is to explain the system status to the user. To this end natural language generation is used (among other presentation techniques). FLUID does not seem to develop proprietary NLG technology; rather, it maintains an overview of NLG systems developed and marketed by other projects and companies. It appears that the FLUIDS project has ended, and it is questionable whether the web site is kept up to date.

An extensive search of the WWW and consultation with the experts on NLG in the Netherlands did not result in additional web sites that are of high interest.

From a close review of the three sites, one cannot but get the impression that NLG is very much an academic enterprise. Worse even, the major groups active in the field seem to be reluctant to refer to each others work. However, DFKI pas participated in the latest RAGS workshop, held in Edinburgh, 12-12 November 1999. Apparently, the RAGS project is trying to establish some kind of commonly agreed architecture for

future research and development in the field of NLG. In its latest proposals the RAGS project focuses on a quite general architecture that should be able to support a wide range of existing and future NLG approaches. It remains to be seen, however, to what extent the RAGS proposal will be taken up by other R&D teams. Also, because of its intended generality the RAGS architecture may prove to be not suitable for commercial implementations.

It is also evident that very few commercial products in the field of NLG are currently available.

Quite a large proportion of the research prototypes that we found are 'multi-lingual', in the sense that they can generate output in two or three languages, one of which is almost invariably English.

6.5 Commercial Products

For the BabelWeb project, we are not so much interested in very elaborate, scientific language generation systems, that need lots of CPU-power and cannot work in (or at least close to) real-time. Moreover, for all practical purposes it would be advantageous if we could use commercial "off-the-shelf" products.

The web site of the RAGS project does not refer to commercial products in the field of NLG.

The SIGGEN web site lists two commercial systems related to text generation, viz.

- Concordance: (http://www.rjcw.freeserve.co.uk/) advertised as a sophisticated text analysis software tool for making concordances, wordlists, and Web Concordances. The product supports many different Western languages. It can turn a concordance into HTML. A fully functional version is available for download with a time limited license.

  Actually, it is not quite clear to us what "Concordance" has to do with text generation. Its usefulness for BabelWeb is certainly not evident.

- Project Reporter (http://www.cogentex.com/products/reporter/) generates dynamic web-based project status reports from files created with Microsoft Project or other compatible project management software. Reports feature hyperlinked textual descriptions of project elements, as well as coordinated multimodal display with an interactive Gantt chart applet. The tool is implemented in Java.

  Obviously, use of this product is tightly coupled to the use of another commercial software product. This severely limits its usefulness for the BabelWeb project. The product is marketed by the company CoGenTex, Inc., that also advertises a number of 'systems' and 'development tools', many of which are also mentioned in the paper by Daniel S. Paiva. All systems generate text in well defined, limited domains.

The SIGGEN web site, unfortunately, does not make the impression that it is very well maintained, even if we have not been able to find other web sites that list more commercial products for stand-alone NLG. However, this site does contain an up-to-date list of books on NLG.

The FLUIDS web site refers to the AlethGen toolbox, that is also described by Paiva. GSI-Erli, the company that marketed AlethGen, has changed its name to LexiQuest (http://www.lexiquest.com). The LexiQuest web site does no longer mention NLG

products. Apparently, the toolbox is no longer available as a product. LexiQuest seems to focus on (multi-lingual) information retrieval.

FLUIDS also refers to the EFFEDI system, supposedly marketed by Daimler-Benz. However, the most recent reference dates back to 1996. Moreover, the major application area for EFFEDI is spoken dialogue systems. This may render the system less suitable for the purpose of the BabelWeb project.

Through the FLUIDS site we found the company Artificial Life http://www.artificial-life.com/default1.asp, that advertises several products that must include some kind of NLG. One of these products is Alife-Messenger, that automatically interprets and replies to e-mails (but the small print says 'planned for future release'). Another product (also 'planned for future release') that must use NLG is Alife-PersonalTutor, a natural language based tutoring program developed to automatically adapt the difficulty and content of lessons to the skill level of the student. The student's skill level is dynamically analyzed using the natural language conversations between the bot and the student user.

DFKI maintains 'The Natural Language Software Registry' http://www.dfki.de/lt/registry/generation-over.html, an annotated list of software tools for NL applications. Under the category 'Generation' 16 software packages are listed, but only a small part of these tools can be considered as NLG tools proper. The systems mentioned here are (almost) all related to research carried out by DFKI, or by related german research groups. A quick scan of the annotation shows that licenses are typically available for research purposes only. Also, extension of the systems is virtually always dependent on collaboration with the author/owner of the software. Yet, the general impression is that this is representative for the state-of-the-art in the field.

The DFKI site describes the work done at DFKI in some detail. For BabelWeb the most interesting results is probably the TG/2 Practical generation of Natural Language text system, by Stephan Busemann. It is a good example of a shallow system, that combines phrase concatenation with slot filling. TG/2 has been used successfully in two applications, viz. the project TEMSI aimed at building a Transnational Environmental Management Support & Information System, and the project COSMA, on Automated Appointment Scheduling by E-Mail. Both are examples of applications that have limited domains.

In conclusion, it is fair to say that presently there are hardly any commercial products for NLG that can be easily and quickly integrated into newly designed NL applications, like the multi-lingual web sites that the BabelWeb project is aiming at. R&D groups may be able to lay their hands on NLG components that were originally developed as part of other applications. Tuning and adaptation of these components to make them suitable for the purpose of BabelWeb will require intensive programming efforts, and can only be done by expert NL engineers. Most likely, adaptation will also require a fair amount of domain expertise. A rough estimate of the resources needed to adapt an existing NLG component to the needs of BabelWeb is on the order of three to five person months. This may explain why the generation component in many existing applications, especially in limited domains, has been tailor made.

6.6 Different types of input

In the paper by Paiva three different classes of NLG systems are distinguished, based on the source and type of input.

1. Database or knowledge base, automatically generated

2. From a real user, through symbolic authoring
3. Hand-crafted by the system developers, using some knowledge representation language

For BabelWeb all three classes of systems are potentially relevant. Of course, one would want to avoid hand crafting and authoring in some symbolic language, but fully automatic processing of database information may not always be possible (or most efficient).

In any case, it seems to be clear that for a project like BabelWeb the first two components in the pipeline (viz. 'Content Determination' and 'Sentence Planning') are the most specific, if not the most important.

6.7 Output of typical NLG systems

All systems discussed by Paiva aim at generating paragraph level 'body text'. No system is mentioned that tries to add mark-up information in such a way that the resulting text is ready for display on a web page. Thus, the existing expertise and experience in NLG is mainly related to the generation of coherent plain text. As a result, BabelWeb might be able to re-use knowledge and expertise to generate narrative descriptions of, for instance, concerts and festival programmes, or restaurants, but it may not be equally easy to find existing expertise to automatically generate headings with the attendant mark-up information. Neither is there any known existing NLG system that can be used to decide what information to display in what field of the graphical page layout. This is even true for the "Caption Generation" system, discussed by Paiva, that produces paragraph length descriptions of what can be seen in a graph.

6.8 Conclusions

From a (necessarily somewhat superficial) analysis of the literature we get the impression that most of the 'intelligence' in the existing NLG systems is in the strategic part of the generation. This is probably also the part that is most tightly linked to the application (domain). As soon as the coarse structure of the sentences to be generated is available, it is possible to combine deep and shallow generation techniques, probably taking advantage of domain limitations. For BabelWeb this implies that –at least at present- there are no commercial tools that can be used to support on-line generation of text, neither isolated sentences, nor paragraphs. The few commercial and research tools that are available will require substantial tuning ands adaptation to the domain(s) covered by a web site. Most of the tuning will be connected to the modules that transform database information or other non-linguistic information from a web site into a formal representation that can be processed by a full-fledged NLG module. Alternatively, hand crafting of isolated phrases and sentences that can be simply and straightforwardly linked to specific database or web information items will be needed. It remains to be seen under what conditions the use of off-the-shelf tools is more effective and efficient than tailor made software development. The eventual decision in BabelWeb, but most likely also in all other multi-lingual web projects, will depend on the details of the application. Therefore, a firm decision as to what is the best approach to generation can only be made after the functionality and the architecture of the web application have been specified.

7 Multilingual/Cross-Linguistic Information Retrieval

7.1 Introduction

In this text we slightly survey the major techniques for information retrieval. In the first part, we provide an overview of the traditional techniques (full text scanning, inversion, signature files and clustering). In the second part we discuss attempts to include semantic information (natural language processing, latent semantic indexing and neural networks).

7.2 Traditional methods for text retrieval

7.2.1 Full text scanning

The most straightforward way of locating the documents that contain a certain search string (term) is to search all documents for the specified string or sequence of characters. The obvious algorithm is as follows:

- Compare the characters of the search string against the corresponding characters of the document.
- If a mismatch occurs, shift the search string by one position to the right and continue until either the string is found or the end of the document is reached.

Although simple to implement, this algorithm is too slow. If *m* is the length of the search string and *n* is the length of the document (in characters), then it needs up to *m*n* comparisons. Knuth, Morris and Pratt proposed an algorithm that needs *m+n* comparisons. Their main idea is to shift the search string by more than one character to the right, whenever a mismatch is predictable. The method needs some pre processing of the search string, to detect recurring sequences of letters. The fastest known algorithm was proposed by Boyer and Moore [45]. Their idea is to perform character comparisons from right to left; if a mismatch occurs, the search string may be shifted up to *m* positions to the right. The number of comparisons is *n+m* in the worst case and usually it is much less: for a random English pattern of length *m=5*, the algorithm typically inspects *i/4* characters of the document (where *i* is the starting position of the match). Again, it requires some pre processing of the search string.

7.2.2 Signature Files

The signature file approach has attracted much interest. In this method, each document yields a bit string ('signature'), using hashing on its words and superimposed coding. The resulting document signatures are stored sequentially in a separate file (signature file); which is much smaller than the original file, and can be searched much faster. The main disadvantage of this method is the response time when the file is large. The advantages are the simplicity of its implementation, the efficiency in handling insertions, the ability to handle queries on parts of words, ability to support a growing file, and tolerance of typing and spelling errors.

7.2.3 Inversion

This method is followed by almost all the commercial systems. Each document can be represented by a list of keywords, which describe the contents of the document for

retrieval purposes. Fast retrieval can be achieved if we invert on those keywords. The keywords are stored, eg., alphabetically, in the 'index file'; for each keyword we maintain a list of pointers to the qualifying documents in the 'postings file'. More sophisticated methods can be used to organise the index file, such as: B-trees, TRIEs, hashing or variations and combinations of these. The disadvantages of this method are: the storage overhead (which can reach up to 300% of the original file size), the cost of updating and reorganizing the index, if the environment is dynamic, and the cost of merging the lists, if they are too long or too many. The advantages are that it is relatively easy to implement, it is fast, and it supports synonyms easily (e.g., the synonyms can be organised as a threaded list within the dictionary). For these reasons, the inversion method has been adopted in most of the commercial systems.

7.2.4 Vector Model and Clustering

The basic idea in clustering is that similar documents are grouped together to form clusters. The underlying reason is the so-called cluster hypothesis: closely associated documents tend to be relevant to the same requests. Grouping similar documents accelerates the searching. Clustering has attracted much attention in information retrieval and library science as well as in pattern recognition. Note that clustering can be applied to terms, instead of documents. Thus, terms can be grouped and form classes of co-occurring terms. Co-occurring terms are usually relevant to each other and are sometimes synonyms. This grouping of terms is useful in automatic thesaurus construction and in dimensionality reduction.

7.3 Merging natural language processing and information retrieval

The traditional information retrieval techniques use only a small amount of the information associated with a document as the basis for relevance decisions. Despite this inherent limitation, they often achieve acceptable precision because the full text of a document contains a significant amount of redundancy. Recent methods try to capture more information about each document, to achieve better performance. Some of these methods are :

methods using parsing, syntactic information and natural language processing in general,

Latent Semantic Indexing method,

methods using neural networks and specifically spreading activation models.

7.3.1 Natural language processing techniques

Natural language processing techniques seek to enhance performance by matching the semantic content of queries with the semantic content of documents. Natural language techniques have been applied with some success on some corpora. Although it has often been claimed that deeper semantic interpretation of texts and/or queries will be required before information retrieval can reach its full potential, a significant performance improvement from automated semantic analysis techniques has yet to be demonstrated. The boundary between natural language processing and shallower information retrieval techniques is not as sharp as it might first appear, however. The commonly used stoplists, for example, are intended to remove words with low semantic content. Use of phrases as indexing terms is another example of integration of a simple natural language processing technique with more traditional information retrieval methods. The benefit of using phrases as terms is that phrases carry greater

semantic content, but the risk is that the greater specificity of a phrase can reduce the performance of ranking or matching algorithms which depend on generality.

The first step in a more complete natural language processing information retrieval system would likely be automatic syntactic analysis. Considerable advances have been made in recent years in syntactic modelling of natural language, and efficient parsers with a broad domain have recently become available. Semantic analysis is less well understood, but progress is being made with a syntax-directed semantic technique called Lexical Compositional Semantics. Deeper semantic interpretation appears to require extensive knowledge engineering, limiting the breadth of systems which depend on natural language processing.

7.3.2 Latent Semantic Indexing

Latent Semantic Indexing (LSI) is a vector space information retrieval method. From the complete collection of documents a term-document matrix is formed in which each entry consists of an integer representing the number of occurrences of a specific term in a specific document. The singular value decomposition of this matrix is then computed and small singular values are eliminated. The resulting singular vector and singular value matrices are used to map term frequency vectors for documents and queries into a subspace in which semantic relationships from the term-document matrix are preserved while term usage variations are suppressed. Documents can then be ranked in order of decreasing similarity to a query by using normalised inner products on these vectors to compute the cosine similarity measure.

7.3.3 Neural Networks

The main idea in this class of methods is to use the spreading activation methods. The usual technique is to construct a thesaurus, either manually or automatically, and then create one node in a hidden layer to correspond to each concept in the thesaurus. It is not clear yet how to extract the maximum benefits from this method. A lot of on-going research is exactly concentrating on this issue.

8 Text Summarisation

The rapid growth of the Web and on-line information services, which have made available vast amounts of textual resources, creates the need of developing text-processing technologies to access and to handle multi-source, multi-modal and multi-lingual information efficiently. One of this technologies is Automatic Text Summarisation.

Automatic Text Summarisation is the process of condensing text information from one or more sources to present the most relevant information content to the user, based on task and user requirements. So, it covers the processes of source text analysis, the extraction of the essential information content and the generation of the summary information.

Notice that, even so the summary's input is text, this text may represent source(s) in some other media such as image, audio, or video.

The main references to elaborate this contribution are references [47] and [48].

8.1 Types of Summaries

The uses of Text Summarisation vary with different needs and applications. The amount of compression (ratio of summary length to source length) or the "most relevant content" depends on the intended use.

In order to develop consistent procedures to create summaries automatically, responding to this different needs and applications, it is necessary to identify and to take into account the "context factors": input (source form, subject type and source unit), purpose (audience and function) and output (material, format and style), as stated by Sparck Jones in [49].

To provide a better understanding of the context factors, a typology of summaries can be made according a set of parameters and the aimed features (see [47][48]):

1. Source

The summary can be based on a *single source* or based on a *set of documents*.

2. Form and Granularity

An *Extract* is a list of fragments of the original and the granularity can be keywords, paragraphs, sentences, phrases. An *Abstract* is a concise and coherent description of the original, which covers the full scope of its content.

3. Intent

An *Indicative* summary provides the topics addressed in the source, an *Informative* summary reflects the concepts of the original, even may process the content and an *Evaluative* summary locate the content or opinions of the text within the context of other texts treating similar topics (offer a critique of the source [49]).

4. Focus

The *Generic* summary cover all the important content of the source and provides the authors view, while the *Query-oriented* is user-focused, covering only part of the source to reflect the user's interest and expertise.

5. Audience

The *Just-the-News* summary provides just the newest facts, assuming the reader is familiar with the topic and the *Background* summary assumes that the knowledge of reader is poor, so it teaches about the topic.

Each type of summary (or the combination of types) has different features, need different methods and techniques to be create and must be evaluated according to different criteria.

8.2 Evolution

The earliest work in development of automatic text summarisation was carried out in the 50s with the Luhn's auto-extracts [40]. Over the next two decades there has been little research effort in this area. However, the rapid growth of the Web and the on-line information services, and the advances in natural language processing, have stimulated a new interest in automatic summarisation.

In the last years, besides the research activities developed by the academic community and official institutions, the industry brought into market several summarisation systems. Most of those systems, commercial or not, perform simple extraction of the most relevant sentences or paragraphs from the source document [47], applying different methods and techniques.

Many of those systems are based in Luhn's assumptions (positional and frequency methods, see next section) in association with other shallow, statistical or symbolic techniques, in order to identify and to extract salient source content and to produce summaries, by the concatenation of the sentences or paragraphs. This extracts may be a list of keywords or a list of single sentences that indicate the major content of the source. This approach is more extraction rather than summarisation but, although the result is not necessarily coherent, the reader can have the most important topics of the source content.

More recent studies are concerned with interpretation of the source (local sentence analysis and their integration into an overall source meaning) and generation of a coherent and synthetic summary by the fusion of the major concepts of the source. Those are complex tasks since they require semantic analysis, discourse processing and inferential interpretation (using domain/world knowledge, i.e., knowledge not explicitly in the text in order to be able to decide how to fuse selected concepts into a more general concept).

Nowadays there are too a strong research effort to extend summarisation techniques, namely:

- Multi-document summarisation, identifying and synthesising similar elements across related text from a set of related documents [51] or analysing how the perception of an event changes over time, using multiple points of view over the same event/series of events [52].

- Summarisation of multimedia sources and mixed media such as charts and tables or knowledge databases, applying extraction technologies to select the key information and then fusing and integrated this information to produce a coherent (multimedia) summary [53][54][74][75].
- Multilingual summarisation, developing engines that employ language-neutral methods or simplified language-specific methods to work across languages, and linking summarisation engines to translation engines [55][56][65]

Although the research efforts that have been done and a few trial systems developed, the deliver of summaries in a coherent and structured text form, with the appropriate scale, depth and orientation seems to be a arduous task and there is still much work to be performed! And of course, multi-lingual, multi-source and multi-modal considerations just increase the difficulty.

8.3 Methods and Techniques

As stated by Sparck Jones in [49] *a summary is a reductive transformation of source text to summary text through content reduction by selection and/or generalisation on what is important in the source.*

The two dominant paradigms in Computational Linguistic - the symbolic and the statistical approaches – are used to perform this transformation process [57].

The symbolic approach is based on knowledge-intensive NLP techniques such as morphological, syntactic and semantic analysis, discourse analysis and text planning. It is performed a deep analysis of the text and its meaning, to create a symbolic representation of the contents, which is then used by inference required for summarisation. The inference process makes use of domain knowledge in the form of scripts, rules and other similar knowledge structures.

The statistical approach is primarily concerned with how to construct systems (or rules for systems) that perform the necessary transformations. It operate at lexical level, using shallow methods that do not involve the understanding of the texts, and traditional techniques which include vector spaces, collecting alternatives and then ranking them using various metrics, counting frequencies, and measuring information content in various ways. This approach needs very large data sets, like lexicons and corpora, organised in terms of many variables.

The symbolic approach is considered superior in the quality area but is somewhat limited since it is knowledge domain dependent; on the other hand, statistics-based methods are generally considered more robust and they are applicable in broad domains, but seem limited by their inability to account for a number of linguistic phenomena like synonymy, polysemy, anaphora, metonymy and context sensitivity [57].

Nowadays the research efforts are concerned with the combination of the two approaches, trying the combination of the robustness and broad domain of the statistical methods with the higher quality of the symbolic ones.

Some of the methods and techniques that may be used, independently or combined, to select and extract the most important content from a document(s) and to build summaries may include [48]:

- Position and frequency methods: This methods was initially proposed by Luhn. The basic assumption is that the frequency of word occurrence and its relative position within a sentence, provide a useful measurement for determining the significance of word/sentences: important sentences occurs at the begin and/or end of text; words in titles and headings are relevant to summarisation; important sentences contain words that occurs frequently. Scores are assigned to sentences according this and the best-scoring sentences are presented in the summary. See [58].
- Cue phases method: important sentences contain indicator phases (e.g. significantly, in conclusion, etc.) and, based on corpora, a system can be training to detect this phases automatically [59].
- Lexical Cohesion methods: This methods are based in the internal text structure, that allows different parts of a text to function as a whole. This lexical cohesion arises from semantic relationships between words. The more relevant sentences in a text are the highest connected entities in this semantic structure. The connection between this entities can be achieved by different techniques:

  - word co-occurrence – use word similarity (repetitions, synonyms) measures (IR-based) to establish links between the paragraphs [60];

  - local salience and grammatical relations – the important phrasal expressions are given by a combination of grammatical, syntactic and contextual parameters [61];

  - co-reference – the more important sentences are traversed by co-reference chains (noun/event identity, part-whole relations) detected between query (if query-based) and document, title and document and sentences within document;

  - lexical chains – the lexical cohesion occurs between pairs of words and over sequences of related words. Using lexical databases to determine the lexical relations it is possible to create strong chains. The most important sentences are traversed by strong chains [62];

  - connectedness : the text structure is represented in terms of cohesion relations (proper name, anaphora, reiteration, synonymy and hypernymy) and coherence. The text is mapped in a graph, whose nodes represent word instances and links represent adjacency, grammatical, co-reference and lexical similarity relations. The salience of words and sentences is calculated applying statistical metrics. The most important are those sentences with high score [63].
- Discourse structure method: This method is based in the Rhetorical Structure Theory and the central idea is the notion of rhetorical relation, witch is a relation between two text spans called Nucleus and Satellite; this rhetorical relation can be assembled into rhetorical structure trees. A rhetorical parser is used to build this discourse representation structure and the centrality of the textual units (nucleus) [76].
- IE-like method: pre-define a template, extract the relevant information and the important document entities, e.g., person names, place names, company names, organisations, numeric data and temporal data, fill it and generate the summary as the template's content.

8.4 Relationship with others areas

Text Summarisation is directly related with Information Extraction. At least their final purpose is quite similar – find and deliver the portion of text and information that is relevant to the user's needs. The differences between IE and TS lie mainly in the techniques used to identify the relevant information and in the ways that information is delivered to the user.

For Information Extraction the criteria for relevance is pre-defined by the user in the form of a template – "I know what I want!". The Text Summarisation does not start with a predefined set of criteria specified as a template, but at a higher granularity, i.e., expressed in keywords or even whole paragraphs. The user can specify dynamically what he/she is interested in – "What is this?".

Both Information Extraction and Text Summarisation are closely related with other language processing technologies and applications as Information Retrieval (IR) and Machine Translation (MT). Together they compose an effective information analysis environment to access and to manage multilingual and multimedia information.

This is quite clear for the web-based information services. The Information Retrieval systems processes a user's query, search a collection of documents and return the most relevant. By applying Information Extraction and Text Summarising technologies on the documents it is possible to present to the user the set of the relevant documents with a coherent and concise description of each one. Furthermore, the summary could be in a different language from the source, in association with Machine Translation technologies.

As an example, we can refer two research projects/systems that uses a multilingual analysis environment to access and manage web information: the "Mulinex" [55] project, which is an EC projects, and the "C*ST*RD" system developed by The Natural Language Processing group at the Information Sciences Institute of the University of Southern California (USC/ISI) [64]', included in the DARPA Tipster program [65].

MULINEX is a multilingual Internet search engine that supports selective information access, navigation and browsing in a multilingual environment. The system integrates language identification, categorisation, retrieval, summarisation and translation.

The "C*ST*RD" project aims to construct an information analysis environment that helps the English-speaking user manage multilingual and multimedia information. It contains modules that perform multilingual document retrieval, clustering, text summarisation and translation into English.

8.5 Commercial Systems

In this section is present a short list of summarisation products that are available commercially.

All systems produced Extracts only.

8.5.1 Extractor (NCR / IIT)

Extractor is a multilingual summarising tool. It takes a text document (ASCII or HTLM) as input and generates a list of keywords and keyphrases as output. The

keyphrases provide a “mini summary” of the text. The user can set the length of the summary.

The summariser tool uses a set of ranking strategies, statistical-based, on word and on sentence level to calculate the relevancy of a sentence to a document. The highest ranking sentences are extracted to create the document summary. The length of the summary can be set either by the number of sentences or by the percentage of the document’s length.

The Extractor is available for several platforms and work with English, French, German and Japanese.

http://extractor.iit.nrc.ca/

8.5.2 MS Word AutoSummarize (Microsoft)

The AutoSummarize is a feature of Microsoft's Word package. The document is analysed statistically and linguistically, using word frequency, position-based and title-based techniques, to identify and to extract the most important sentences.

The identified sentences can be highlighted or separated from the remainder of the text, providing a summary. The user can specify a target percentage of the text for AutoSummarize to mark as important.

http://www.slate.com/features/cogitoautosum/cogitoautosum.asp

8.5.3 ProSum (British Telecom)

ProSum, short for PROfile-based SUMmarisation, use statistical techniques based on the co-occurrences of word stems, the length of sentences and their position in the original text to calculate the importance of a sentence in the context of the overall text in which it occurs. The most important sentences are then used to construct the summary or, alternatively, these sentences can be presented highlighted in the source text.

It is available either as an on-line service via the Internet, or as a Microsoft Word add-in. The on-line Internet version can also be used to summarise Web pages.

http://transend.labs.bt.com/prosum/word/index.htlm

8.5.4 LinguisticX – Inxight Summary Server (Xerox Company)

Inxight extracts key sentences from documents and creates a general summary of the document. If associated to a search engine query can generate too a key sentence

The tool examines the content of a document in real-time to identify the document’s key phrases and extract sentences to form an indicative summary, either by highlighting excerpts within a document or creating a bulleted list of the document’s key phrases.

Summarisation features use morphological analysis, name tagging and co-reference resolution. They used a machine learning technique to determine the optimal combination of these features in combination with statistical information from the corpus to identify the best sentences to include in the summary.

http://www.inxight.com/products/enterprise/summserv.htlm

8.5.5 ConText (Oracle Corporation)

ConText integrates full text management with the Oracle database. It is a natural language processor which uses a knowledge base consisting of over 1,000,000 words and phrases, related to more than 200,000 concepts, associated with approximately 2,000 thematic categories.

Based on user's queries ConText search information in the texts related and, using the knowledge base, isolates the significant concepts in the document. Then, through a process of comparison and abstraction in relation to the knowledge base, ConText discovers and ranks the themes for each paragraph and for the document as a whole. The text reduction function uses this output to produce either general or theme-specific summaries of the document. The text classification feature automatically categorises documents according to the main themes of the documents.

http://oracle.com.ar:1000/products/context

8.5.6 WebSumm (MITRE)

WebSumm is a scalable, multi-document text summarisation using to enhance Web searches by filtering and summarising text. It uses statistical and symbolic language processing to text extraction and retrieval, and to the generation of the summaries.

WebSumm generates an index of prominent words, phrases, and proper names from the initial texts returned by Web search engines. The user can then filter the search results by selecting terms from this index. The filtered results can then be summarised and compared, allowing users to quickly find answers to their queries.

http://www-i.mitre.org/pubs/edge/july_97/first.htlm

9 Other Linguistic Resources/Utilities

EuroWordnet and other multilingual dictionaries/thesauri are of interest for both translation and cross-linguistic information retrieval. In information retrieval, such resources could be used both for translating keywords from one language to another, and for disambiguating search and indexing terms.

As part of a translation system, lexical relational databases (such as EuroWordnet) do not contain enough syntactic or semantic information to be used as a translation lexicon. However, the information present in EuroWordnet could make a useful addition to an existing translation system to improve translation quality. The EDR and Acquilex databases provide an increased amount of syntactic information and they also provide information on the semantic argument structure which is vital for translation.

9.1 EuroWordnet

"The goal of EuroWordnet is to build a multilingual lexical database with wordnets for several European languages. , which are structured along the same lines as the Princeton WordNet".

WordNet is a lexical-relational database. That is, it represents relations between lexical concepts. The lexical concepts themselves are grouped into "synsets" which have no internal structure. Each synset is associated with one or more words and some gloss text explaining the meaning of the synset. Syntactic information is contained using a coarse part of speech classification (5 categories are used). Verbs however, are associated with frames which give alternations that each verb can take part in.

EuroWordnet is structured as a collection of individual monolingual wordnets that are related to each other by means of a language-independent list of synsets (primarily taken from the Princeton WordNet). The language-independent list of synsets is called the "Interlingual Index". Each synset in each language is linked to the node in the interlingual Index that most closely represents its meaning. The Interlingual Index is relatively unstructured, however 1024 Interlingual Index synsets are organised under a "Top Ontology". The Top Ontology consists of 63 fundamental semantic distinctions (which have been agreed by the EuroWordnet team to be language-independent). Each of the classified synsets in the Interlingual Index may be assigned to one or more nodes in the Top Ontology.

It is not straightforward to organise a distributed effort like this. The team proceeded by allowing each partner to develop a fragment of a language-specific wordnet, relate the synsets back to the Princeton WordNet synsets and then compare the coverage. From this a common set of synsets is determined (the "Base Concepts"), and each partner modifies their wordnets to ensure that they cover the agreed common synsets. This cycle is repeated to build the full EuroWordnet.

One technique (mentioned earlier) for producing text quickly and economically in different languages is to have templates that are translated by hand and completed by words that are generated automatically from data. Although EuroWordnet does not provide enough syntactic information for normal translation, it could provide a suitable lexicon from which words could be selected to complete the hand-translated templates.

9.2 Japanese EDR project

The Japanese Electronic Dictionary Research Institute (http://www.iijnet.or.jp/edr/) has several parallel research initiatives all aimed at improving the ability of computers to understand human language. In order to accomplish this, the institute has begun the development of several complementary, full-sized dictionaries: and English word dictionary, a Japanese word dictionary, a concept dictionary and a bilingual dictionary as well as associated English and Japanese interfaces. With these, they hope to make significant inroads into the Natural Language Interface and Machine Translation fields. Construction of the dictionaries ran from 1986 to 1994.

The EDR concept dictionary is a network of concept labels and relations between them, including simple concepts like

<bird>,

and relations between simple concepts, like

<bird> - a kind of - <food>,

as well as compounds to represent phrasal and sentential level concepts which are recursively defined and infinitely extensible. One section of the concept dictionary provides compound concepts defined by two or more concept entries consisting of head concepts and a relations. For example, the concept "Sumo wrestlers drink much liquor", is represented by,

[<to drink> -agent- <wrestler of Japanese wrestling>

<to drink> -object - <liquor>

<to drink> - quantity- <a large amount>]

The EDR bilingual dictionary is a list of triples whose components are the source and target words and an "interlingual correspondence label" which is either "equivalent relation", "synonymous relation", "subset relation" or "superset relation". The interlingual correspondence labels are intended to tell the generation component of a translation system that some additional explanations needs to be added to the translation.

9.3 Acquilex (I and II)

Acquilex is an Esprit initiative which is aims to produce lexical resources that will serve many purposes. The long-term goal of the project is stated as "the development of a multilingual knowledge base containing the most general and domain-independent aspects of

lexical knowledge represented in a fashion which makes it maximally reusable". The intention is to use existing Machine Readable Dictionaries and combine them into a unified framework using an overarching conceptual structure. One of the key features of Acquilex is that individual word senses are linked to descriptions in a common conceptual/semantic structure. The intention for the first phase (Acquilex I) is to extract information in a semi-automated manner from monolingual MRDs for English, Italian and Dutch and bilingual MRDs for English-Italian and English-Dutch.

The second phase of the project (Acquilex II) was to extend the work of Acquilex I but will in addition use corpora as a source of some the information required.

The second phase of the project (Acquilex II) finished in September 1995.

One of the problems of assigning senses to words is that of polysemy (described earlier). The problem is that there is no way to choose a particular granularity for word senses when creating a lexicon resource. Acquilex takes this into account and uses some of Pustejovsky's ideas for creating a generative lexicon. In Acquilex two cases are distinguished; first, where a word is underspecified in the lexicon and adopts a more specific meaning in a particular context, and secondly, where the word is considered to have distinct but related meanings.

10 References

[1] 'Altavista WEB site' http://www.altavista.com/

[2] 'Systran WEB site' http://www.systranmt.com/

[3] Yang J. and Lange E. D. 'SYSTRAN on AltaVista: A User Study on Real-Time Machine Translation on the Internet' pp275-285 in 'Machine Translation and the Information Soup' eds. Farwell D. and Gerber L., Springer, Berlin (1998)

[4] 'Yahoo WEB site' http://www.yahoo.com

[5] 'Logos WEB site' http://www.logos.com

[6] 'Euromarketing WEB site' http://www.euromktg.com/

[7] Esselink B. 'A Practical Guide to Software Localization', Benjamins, Amsterdam (1989)

[8] 'WebBudget download page' http://www.webbudget.com/

[9] 'Corel WEB site' http://www.corel.com/

[10] 'Berlitz WEB site' http://www.berlitz.ie/twe/default.htm

[11] 'The Trados Workbench' http://www.berlitz.ie/twe/default.htm

[12] 'The DéjàVu WEB site' http://www.atril.com/

[13] 'IBM TranslationManager WEB page' http://www.software.ibm.com/ad/translat/tm/

[14] 'Lernout and Hauspie WEB site' http://www.lhs.com/

[15] 'The Multilizer WEB page' http://www.multilizer.com/lm/index.html

[16] 'OpenTag format Specifications' http://www.opentag.org/otspecs.htm

[17] 'TMX: Translation Memory Exchange Format Specification' http://www.lisa.org/tmx/index.html

[18] 'The Eurodollar WEB site' http://www.euro-dollar.com/

[19] 'Translation Craft WEB site' http://www.tcraft.com/

[20] 'The Software Localisation Interest Group WEB site' http://lrc.csis.ul.ie/SLIG/SLIGmainFR.html

[21] 'The Language Industry Standards Association (LISA) WEB site' http://www.lisa.org/

[22] 'The Localization Institute WEB site' http://www.localization-institute.org/

[23] 'International Language Engineering WEB site' http://www.localization-institute.org/

[24] 'The World Wide WEB consortium WEB site' http://www.w3.org/International/Overview.html

[25] 'European Association for Machine Translation' http://www.eamt.org/

[26] Carl M.'Towards a Model of Competence for Corpus-Based Machine Translation' Internal report: Human Language Technology Center, Hong Kong University of Science and Technology (1999)

[27] Arnold D., Balkan L., Humphreys R. L., Meijer S. and Sadler L. 'Machine Translation: An introductory guide' Blackwell, Oxford (1994)

[28] Pustejovsky J. 'The Generative Lexicon' MIT Press, Cambridge (1996)

[29] Nagao M. 'Machine Translation - How far can it go?' Oxford University Press, Oxford (1989)

[30] Nirenburg S., Carbonell J., Tomita M. and Goodman K. 'Machine Translation: A Knowledge-Based Approach' Morgan Kaufmann, San Mateo (1992)

[31] Halliday M. A. K. 'Machine Translation: An Introduction to Functional Grammar' Arnold, London (1994)

[32] Schmidt G. 'Overview of Current IBM Translation Technology' in Terminology in Advanced Microcomputer Applications: Tools for Multilingual Communication, 15-16 January (1998)

[33] Berger A., Brown P., Pietra S. d., Pietra V. d., Gillett J., Lafferty J., Mercer R., Printz H. and Ures L. 'Candide System for Machine Translation' in Human Language Technology: Proceedings of the ARPA Workshop on Speech and Natural Language (1994)

[34] 'A Reference Architecture for Generation Systems' http://www.itri.brighton.ac.uk/projects/rags

[35] 'Association for Computational Linguistics: Special Interest Group on Text Generation' http://www.dynamicmultimedia.com.au/siggen/

[36] 'Future Lines of User Interface Decision Support' http://www.dfki.de/fluids/General_Overview.html

[37] 'The Concordance Text Analysis System' http://www.rjcw.freeserve.co.uk/

[38] 'Project Reporter Status Report System' http://www.cogentex.com/products/reporter/

[39] 'Artificial Life WEB site' http://www.artificial-life.com/default1.asp

[40] 'The Natural Language Software Registry' http://www.dfki.de/lt/registry/generation-over.html

[41] Newton J. 'The Perkins Experience' pp46-57 in 'Computers in Translation' eds. Newton J., Routeledge, London (1992)

[42] Bernth A. 'Easy English: Addressing Structural Ambiguity' in 'Machine Translation and the Information Soup' pp164-173, eds. Farwell D. and Gerber L., Springer, Berlin (1998)

[43] Adriaens G. 'Simplified English Grammar and Style Correction in an MT Framework: The LRE SECC Project', pp (1995)

[44] Almqvist I. and Hein A. S. g. 'Defining ScaniaSwedish - a Controlled Language for Truck Maintenance' in First International Workshop on Controlled Language Applications (1996)

[45] Boyer R. S. and Moore J. S. 'A Fast String Searching Algorithm' CACM, 20, 10, pp762-772 (1977)

[46] 'Japan Electronic Dictionary Research Institute, Ltd.' http://www.iijnet.or.jp/edr/

[47] Grishman R., Hobbs J., Hovy E., Sanfilippo A. and Wilks Y. 'Cross-lingual Information Extraction and Automated Text Summarization', in 'Multilingual Information Management:Current Levels and Future Abilities. A report Commissioned by the US National Science Foundation and also delivered to the European Commission's Language Engineering Office and the US Defense Advanced Research Projects Agency' eds. Hovy E., http://www.cs.cmu.edu/~ref/mlim/chapter3.html

[48] Hovy E. and Marcu D. 'Automated Text Summarisation' pp in 'COLING/ACL '98' eds. , (1998) http://www.isi.edu/natural-language/people/hovy.html

[49] Jones K. S. 'Automatic summarising: factors and directions', in 'Advances in Automated Text Summarization' eds. Mani I. and Maybury M., MIT Press, (1998)

[50] Luhn H. P. 'The Automatic Creation of Literature Abstracts' IBM Journal of Research and Development (1958)

[51] McKeown K. R. M. et al. 'Towards Multidocument Summarisation by Reformulation: progress and prospects ' (1999)

[52] McKeown K. R. and Radev D. R. 'Generating summaries of Multiple News Articles' in Proceedings of the eighteenth Annual International ACM SIGIR Conference on Research and Development in IR (1995)

[53] Chen, F., et al., 'Mixed-Media Access' http://www.xrce.xerox.com/

[54] McKeown K. R., Pan S., Shaw J., Jordan D. A. and Allen B. A.'Language generation for multimedia healthcare briefings' in Proceedings of the ACL Conference on Applied Natural Language Processing (1997)

[55] Erbach G. and Uszkoreit H.'MULINEX - Multilingual Indexing, Navigation and Editing Extensions for the World Wide Web' D 0.19 Final Report (1997)

[56] 'MINDS project WEB site (New Mexico State University)' http://crl.nmsu.edu/research/projects/minds/goals.htlm

[57] Chanod J. P., Hobbs J., Hovy E., Jelinek F. and Rajman M. 'Methods and Techniques of Processing' pp in 'Multilingual Information Management:Current Levels and Future Abilities. A report Commissioned by the US National Science Foundation and also delivered to the European Commission's Language Engineering Office and the US Defense Advanced Research Projects Agency' eds. Ide N. (1999)

[58] Lin C. Y. and Hovy E. 'Identifying topics by position' in Proceedings of the ACL Conference on Applied Natural Language Processing (1997)

[59] Teufel S. and Moens M.'Sentence Extraction as a Classification Task' in Proceedings of ACL/EACL Workshop on Intelligent Scalable Text Summarization (1997)

[60] Abracos J. and Lopes G. P.'Statistical methods for retrieving most significant paragraphs in newspaper articles' in Proceedings of ACL/EACL Workshop on Intelligent Scalable Text Summarization (1997)

[61] Boguraev B. and Kennedy C. 'Salience-based content caracterisation' in 'Proceedings of ACL/EACL Workshop on Intelligent Scalable Text Summarization' (1997)

[62] Barzilay R. and Elhadad M.'Using lexical chains for text summarisation' in In Proceedings of ACL/EACL Workshopm on Intelligent Scalable Text Summarization (1997)

[63] Mani I., Bloedorn E. and Gates B.'Using cohesion and coherence models for text summarization' in In AAAI 98 Spring Symposium on Intelligent Text Summarization (1998)

[64] 'Natural Language Group at USC/ISI ' http://www.isi.edu/natural-language/nlp-at-isi.html

[65] 'The Tipster WEB site' http://www.itl.nist.gov/div894/894.02/related_projects/tipster/

[66] 'Extractor WEB site' http://extractor.iit.nrc.ca/

[67] 'MS Autosummarize WEB site' http://www.slate.com/features/cogitoautosum/cogitoautosum.asp

[68] 'ProSumm WEB site' http://transend.labs.bt.com/prosum/word/index.htlm

[69] 'LinguiticX WEB site' http://www.inxight.com/products/enterprise/summserv.htlm

[70] 'Oracle Context WEB site' http://oracle.com.ar:1000/products/context

[71] 'MITRE WEBSumm site' http://www-i.mitre.org/pubs/edge/july_97/first.htlm

[72] Victor Sadler, "Machine Translation Project Reaches Watershed", Language Problems and Language Planning , 15 (1991), 78-81.

[73] Dan Maxwell, Klaus Schubert, Toon Witkam (eds.).(1988) "New directions in machine translation : conference proceedings, Budapest August 18-19, 1988. Dordrecht, Foris

[74] Liwei He, Elizabeth Sanocki, Anoop Gupta, Jonathan Grudin, 'Auto-Summarization of Audio-Video Presentations'. Microsoft Research. (1999) http://www.research.microsoft.com/research/nlp/

[75] Maybury, M. and Merlino, A., "Multimedia Summaries of Broadcast News". In International Conference on Intelligent Information Systems. (1997) http://www-i.mitre.org/resources/centers/it/g062/bnn/mmpapers.html

[76] Marcu, D., 'From discourse structures to text summaries', In Proceedings of the ACL/EACL Workshop on Intelligent Scalable Text Summarization. (1998)