

# Utilizing Evaluation in Networked Machine Translation\*

Jörg Schütz

IAI

Martin Luther Straße 14

D-66111 Saarbrücken

Germany

joerg@iai.uni-sb.de

**Abstract.** In this paper we develop the prerequisites of a processing strategy to be employed in a networked machine translation environment. Networked Machine Translation (NMT) is a new MT paradigm based on the recent and emerging developments in the field of information technology (IT), World Wide Web (WWW) technology and language technology (LT), in particular networked computing. This new MT paradigm aims at combining the different technologies in an integrated way to achieve a better translation throughput in terms of translation quality and translation processing speed. The basic methodology of NMT is the deployment of cooperating and communicating software agents which make use of local and distributed information resources (different types of MT systems, language resources, user parameter, evaluation and verification parameters, and so forth) that are based on agreed standards (especially information exchange formats). The language resources of this new kind of MT software are intended to be shareable with other network-based language technology applications and services such as multilingual search engines, text retrieval, information brokering and telematics services. NMT can be deployed in intranet applications (corporate networks based on Internet and World Wide Web technologies) and is thus open to general translation services to the public users through the Internet.

## 1 Introduction

In order to thrive in the next millennium industries will have to overcome the confusing revolutions introduced by business and technological areas. On the one hand, the business revolution is characterized by increased competitiveness, the pressure to reduce costs, increased productivity and an increase of responsiveness; on the other hand, the technological revolution concerns the step-wise deployment of distributed systems, PCs, client/server technology, graphical user interfaces (GUI) and open systems. Currently, significant reengineering efforts are underway to taming the complexity of both business processes and computer information systems. The information systems that are needed for tomorrow's global markets must be far more robust, intelligent and user-centered than the data processing systems of today.

With the increasing global competition of trade and industry, multilinguality is an invaluable asset of future information systems, particularly in combination with the ubiquitous information networks (or information highways) that form the digital foundation of

---

\* The R&D work reported herein is partly funded by the Commission of the European Union (EC) within the Telematics Applications Programme of the Fourth Framework Programme in the MULTIDOC Concerted Action (LE3-4230). The views and conclusions contained in this paper are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of the EC and the industrial companies involved in MULTIDOC. Additional information on MULTIDOC can be found at URL <http://www.iai.uni-sb.de/MULTIDOC>.

the evolving knowledge society. Past MT experiences, in particular the various evaluation and validation projects, have shown that current translation approaches are not flexible enough to solve the diversity of the multilinguality problems of today's and particularly tomorrow's global business and communication situations.

The NMT paradigm defines MT as a specialized information system application that is able to contribute to solving the problem of multilingual information management, as well as to overcoming the existing language and cultural barriers of the world-wide network community.

The remainder of this paper is organized as follows: In the second section we briefly introduce NMT in general, and we contrast it with the current MT employment in industry. We will extend this scenario with the concept of intelligent software agents which results in the definition of an intelligent translation assistant with (semi-) automatic selection and evaluation capabilities. In the third section we report on how to define different selection and evaluation strategies of an NMT environment. The fourth section outlines on how we can benefit from these developments yet to come into operation in an industrial multilingual information management applications. The paper closes with some prospects of ongoing research and development activities.

## **2 Networked Machine Translation**

### **2.1 Networked Computing enables NMT**

Today, MT systems with acceptable translation results are still very costly and mostly available for high-end workstation platforms. The existing PC-based systems do not accomplish comparable translation results and they are only available for a limited number of language pairs; we particularly start getting into trouble when our application focuses on Asian and Arabic languages. However, similar problems exist when we look for systems dealing with French as source language. This situation is clearly demonstrated on the Internet: multilinguality simply does not exist; the English language is the cyber-lingua of today's information highways.

Obviously, the idea for an MT service available as an Intranet/Internet navigator plugin is fascinating and is becoming more and more feasible with the emerging Networked Computing (NC) technologies. Some MT vendors have already started to offer MT services via the Internet, such as the off-line translation of HTML documents (e.g. Systran and MTSU) or form/email-based translation services (e.g. CAT2). However, real on-line translation services and other interactive WWW services are not yet available. Today, the Web is only mildly interactive with just hyperlinks to take the user from location to location.

This situation has changed a bit with the introduction of forms, for example, users can specify search queries or parameters and texts for off-line services. A form consists of two parts: the form itself, which is rendered in the browser, and a Common Gateway Interface (CGI) script <or program located on the server. This script processes the user's input mainly to validate the correctness of the user data (e.g. on-line subscription). This form mechanism is also employed by most of the above mentioned Internet translation services. However, the server executes only one program (CGI script). Since the interaction is immediately direct over the network, this approach causes a higher load of the network. Incorrect and incomplete input can only be perceived on the server side, and the possibility to implement appropriate user interfaces is very limited. This situation led to the idea to execute some tasks on the client side similar to client/server programming. For this the Web browser must be able to run a program which has been implemented by the service provider and which is accessible via the offered HTML page.

Thus, the next step to interactivity is the use of so-called *browser plug-ins*. The employment of plug-ins also reduces the network traffic, which today is still a major problem. However, with the introduction of high-speed networks based on the Asynchronous Transfer Mode (ATM) and mobile communication capabilities via satellites, this will no longer be a real bottleneck, especially because the costs are going down in this technical area. Plug-ins can be installed by running a setup program supplied with the plug-in. Plug-ins reside on the local hard drive and are detected by the browser when it starts up. When the browser encounters data handled by a plug-in (either embedded in a HTML page or in a separate file), it loads the appropriate plug-in and gives access to all or a part of a window. The plug-in remains active until the associated page or file is closed. Currently the programming language Java is mostly used for this application, because it is platform independent. A Java program is compiled into Java bytecode, which can be embedded in a HTML page. A Java enabled browser on the client side loads this bytecode over the network and executes the code. This sort of program is called *Applet*. Other programming languages or scripting languages used for plug-in development are for example SafeTcl/SafeTk, Phantom and TeleScript.

Such a plug-in program can also use additional resources needed by the application and located somewhere on the Intranet/Internet. Thus, plug-ins are the ideal mechanism for handling the trade-off between local and distributed resources and the overall network infrastructure. Today's plug-ins can be seen as the first operational examples of NC which commonly is defined as: *Remote servers and clients cooperate over Intranets and/or the Internet to fulfill a certain task* (cf. [Schütz, 1996]).

In a NC application the user on the client end will connect to a Web site, but instead of asking for a file (the usual way today) the user will request a session and will receive client code. Once this code is loaded, the client computer and the (network) service server will cooperate, exchange data and communicate. While standard Web applications are based on listening, NC applications are based on conversation. On the one hand, NC will extend the classical client/server applications which are local, prearranged and limited to a set group of users, because it will have global reach and a potentially massive scale, and on the other hand, it will also enhance Web applications, which handle static documents, by allowing clients and servers to carry on rich discussions based on Web technology, they will use LANs, the Internet, Web browsers and Web-enabled servers.

## 2.2 Intelligent Software Agents

As outlined in the previous section, the interactivity of current network-based applications and services is limited, i.e. mildly interactive. Current Intranet/Internet applications provide static Web documents, and the user can use, on the one hand, presentation services provided by Web browsers, and on the other hand, search services as maintained by AltaVista, Lycos, Yahoo, HotBot, and so forth.

With the evolution of NC a new type of software is emerging which will operate on behalf of a user or another program. Thus, it is called intelligent software agent. In the Intranet/Internet scenario of tomorrow the intelligent software agent acts as the personal navigator of a user. Therefore, we talk about either a personal net assistant or a softbot (software robot).

This softbot cooperates with other software agents for the fulfillment of a specific task by the technical means of NC. The cooperating agents constitute the intelligent network infrastructure, which is the knowledge layer above the information entities, e.g. multimedia Web documents, of an Intranet and the Internet.

A softbot of a user is able to execute different tasks such as

- Information filtering according to user defined parameters, e.g. technical watch in terms of selective dissemination of information and competitive intelligence of large companies.
- Information condensing, e.g. for the storage in digital libraries and in multimedial databases.
- Information brokering, e.g. for on-line services and emergency applications.
- Telematics assistance, e.g. in telecooperation situations with working cooperations across time zones including the synchronous exchange of working materials and the asynchronous development.
- Translation services, e.g. machine translation and on-line speech translation for teleconferencing.
- Data analysis, e.g. automatic validation and evaluation of service results.
- Knowledge acquisition, e.g. distance service operations, and distance learning and training.

In order to do this the softbot must have information about the available resources on the Intranet/Internet, either on the basis of local knowledge or through the communication and cooperation with other software agents located on the intelligent infrastructure layer, and information about his master (user), for example, predefined and learned usage patterns, a user model or a specific consumer/customer profile.

Such an intelligent infrastructure service scenario is shown in Figure 1. In this scenario the personal softbot gets an order from the user for whom it is navigating, and according to the user's order and its knowledge about the user, the softbot negotiates with the software agent of a selected supplier who offers the appropriate information, goods or services the user is looking for.

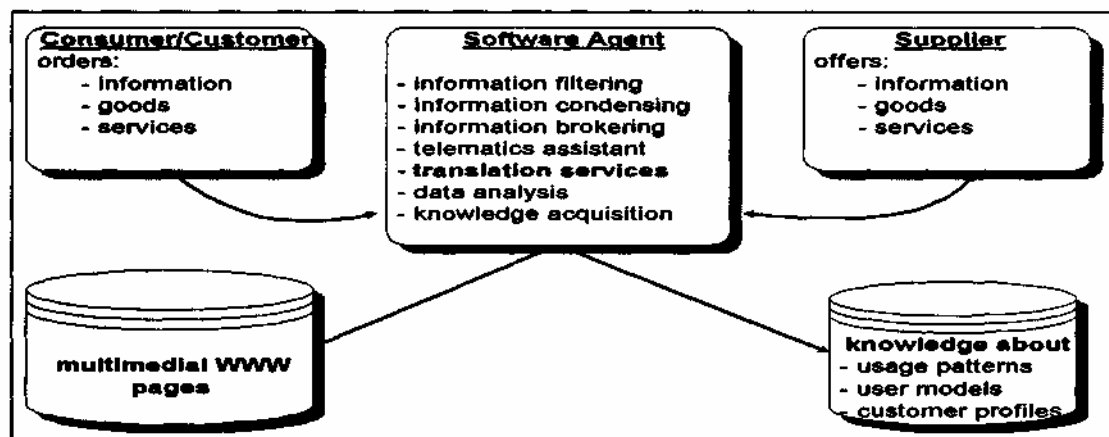


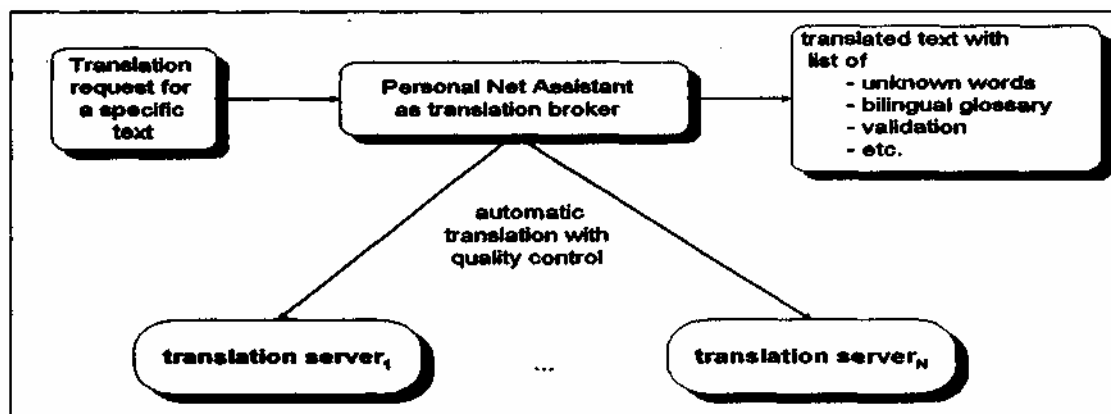
Figure 1: Intelligent infrastructure services

### 2.3 Intelligent Translation Assistant

Up to now we have described the technical basis of current and future network-based services, and we have demonstrated how the traditional MT system approaches could be an integrated service within this scenario. The feasibility of the integration in today's network infrastructure is also proven by the recent efforts of some of the international MT system vendors.

However, the concept of NMT aims at the design and specification of an intelligent translation agent, the translation broker, that on the one hand is able to identify and select translation services on the network infrastructure (intranet as well as the Internet) for a specific application domain, and on the other hand provides a validation and evaluation of the translation results, which will be used in future decision making processes (MT service

selection). In addition the MT service agent can inform the personal MT brokers about new features and enhancements of their services according to the principles of NC, i.e. communication and cooperation. This global scenario is shown in Figure 2.



**Figure 2: Intelligent translation assistant**

In [Schütz, 1996] NMT is defined as *"Remote translation servers and translation brokers (clients) cooperate over Intranets and the Internet to fulfill a translation task"*, and this definition implies the relationship between NMT and NC on the basis of intelligent software agents. Actually, in this scenario a translation broker will request a translation session from a MT server, which will provide a basic client code for a specified translation task. Once this client code is loaded, the translation broker and the MT server will cooperate, e.g. specific requirements for the translation result, exchange information, e.g. local lexical resources, and communicate, e.g. for the identification of additional resources. In the next section we will discuss this cooperation in more detail.

### 3 Selection, Negotiation and Evaluation in NMT

The NMT paradigm allows the use of different processing modules for the fulfillment of a translation task based on user specifications and knowledge derived from the evaluation of the obtained results. In an actual application (cf. the industrial MULTIDOC scenario in the next section) this knowledge contributes, on the one hand, to the evolution of the language technology employment through continuous verification, control and rearrangement (configuration) of modules to be activated, and on the other hand, to continuous improvements, growing functionality and better performance on the LT provider side.

In our NMT environment the Personal Translation Assistant (user softbot) is responsible for the overall management of a translation task on the user side. The management task is executed in cooperation with the softbots of a MT vendor or of a LT service provider. Thus, the user softbot tasks include:

- Selection of MT services according to user defined criteria such as costs, quality and processing time.
- Negotiation with the MT/LT provider softbot about how to access additional language resources, the setting of parameters for the optimal use of the service and the exchange format to be used between the software agents.
- Verification and evaluation of the translation results.
- Presentation of the result (MT output and evaluation).

In the following we describe each of the tasks in more detail.

### **3.1 Selection Strategies**

The user softbot already maintains knowledge about the application scenario for which it has to select appropriate resources, modules and tools, as well as about the available network resources and systems. This knowledge consists of static information and dynamic information (selection parameter). For example, new resources in a network and new functionalities of a system will be updated on a regular basis according to the principles of NC (push paradigm in Web applications).

Based on this knowledge the user softbot creates a task plan for a specific user request which includes possible processing steps as well as possible resources for the fulfillment of these steps. According to set user parameters and the actual state of the network the user softbot decides on the plan execution (this can also be done interactively with the user).

### **3.2 Negotiation Strategies**

According to the task plan the user softbot initiates the dialogue with possible provider softbots. Depending on the negotiation between the softbots the task plan may be revised or an exchange format has to be generated for the sharing of additional resources. The strategies employed for this task are directly those employed in NC.

### **3.3 Evaluation Strategies**

The selection of an appropriate evaluation strategy is of crucial importance in an NMT application scenario because it shall contribute to the establishment of an appropriate environment for translation tasks on behalf of a specific customer (user). Because of this task orientation traditional approaches of MT evaluation are not suited; they view the translation task as an absolute event and do not take into account the situatedness of the task. In our scenario MT is an integrated part of a larger end-to-end LT system (cf. the MULTIDOC application). Currently, we distinguish four cases which can be used in an automatic evaluation procedure for translation tasks aiming at information scanning (word-to-word translation or phrase-based translation), raw translation (existing MT systems with additional resources), and high-quality translation (preprocessing modules combined with existing MT systems with additional resources):

1. Translation memory (TM) output with a 100 % match is obviously a criterion for a high-quality translation result because most TM databases are created with documents checked by human translators.
2. If the MT system makes use of a terminology provided by the user then the translation result should be consistent according to this terminology. Additional checking processes may verify this again. The result of this application will be at least a raw translation.
3. If the MT system makes use of a user defined controlled language (CL) or the system offers the definition of a CL, including the support of unapproved rules and vocabulary, and the input has been checked for CL correctness then the result should be consistent according to the CL rules, i.e. high-quality translation.
4. All other cases will only apply to information scanning.

This evaluation strategy takes also into account traditional criteria such as the suitability of the source document for MT, the complexity of the language used for the document, the availability of the vocabulary (terminology) of the document and possible stylistic requirements for the source and target document, as well as the analysis and translation capabilities of a given translation system.

In addition, in our MT environment all relevant information may be stored on many servers (across networks and across countries in a company), mission critical decisions require access to the most recent information, a variety of formats has to be supported

(document formats), and multilingual documents may be located in databases or repositories, not just on the file system.

## 4 NMT Application Scenario

This section gives a short overview of the first results obtained in a case study within the MULTIDOC project that aims at designing and building an integrated architecture for multilingual document processing and production in the automotive industry. The MULTIDOC intelligent technical information management system shall bring together leading SGML editing tools with sophisticated language technology and repository software, to support the process of authoring, translating and managing shared corporate information in a supportive, structured environment, and to enable this information to be published in a variety of formats. The MULTIDOC system shall also provide full support for translated text in all formats, radically reducing the cost of managing multilingual translations.

### 4.1 Business Requirements

The MULTIDOC partners belong to different European car and truck manufacturers, in particular to the departments which are responsible for the production and publishing of technical information, aimed at helping technicians to maintain and repair units and their constituent components. These departments are dedicated to the production of all the various types of technical information used in dealers workshops, as well as the customer-oriented information provided with a car when it is delivered.

The information provided at dealerships includes not only traditional hard copy, but information delivered electronically by the companies computer systems. The content of the electronic and paper version of a particular information set may be similar, but the output format is very different, and the means by which information is accessed and navigated differs radically.

In most companies information has historically been authored, translated and managed on a variety of different, incompatible platforms. There were/are different publishing methods, and different authoring approaches, for substantially the same information; a lot of the effort expended in such a situation is redundant. Since all of the MULTIDOC partners information is published in different languages (up to 23 languages), there are major translation overheads incurred by a revision to any existing publications, as well as upon release of a new publication.

Most cars are designed with re-use of existing components in mind, it makes sense that the information relating to those components, and the major units they make up, should be prime for re-use also. This would help to relieve the authoring workload, at the same time as reducing or even eliminating necessary duplication; it would also help to reduce translation costs if, in the event of changes to a piece of information, the computer system could identify exactly what changes had taken place, and constrain the translator to re-translate only those elements.

In the automotive industry the question was: *how do we ensure that the information we author is re-usable, rather than adding to the burden of managing a pool of increasingly technical information?* The answer was prompted by the imminent SAE J2008 legislation, which affects all automotive manufacturer who sell into the US market. J2008 prescribes the use of SGML (Standard Generalized Markup Language) as the data tagging protocol of electronic service-related information, where that information is pertinent to the maintenance of emission-related components of a car. Therefore most/all European car manufacturers have developed an SGML authoring capability within their companies. In addition, efforts are under way to devising an object-oriented approach to information management, which penetrates deeper than the document, to address individual information units. These information units, SGML elements which are tracked throughout their

lives, can be stored as discrete objects and shared many times between output publications and other information-related activities.

Whereas the evolution of this approach is open for addressing a corporate information warehousing strategy, which will enable many users within the enterprise to share a common corporate information pool, there is obviously a gap between the envisaged information technology being deployed (e.g. thin client/server models) and the effective integration of language technology to support the information management system by linguistic means, which, on the one hand, ensures better intelligent information access and update, and on the hand, significantly improves the authoring process in terms of consistency, accuracy and translatability.

#### **4.2 Language Technology Deployment**

Up to now only standalone solutions have been elaborated such as multilingual terminology databases, monolingual syntax and style checkers and computer-aided translation utilities (translation memories and fully automatic translation), but the aim is to include linguistic technology (or linguistic intelligence) in the overall workflow of information processing and production to eliminate the existing overheads in multilingual technical information management. In this context NMT is considered as an open architecture approach which could satisfy some of the language-related requirements of MULTIDOC, which are SGML-compatibility, platform-independency and openness for distributed linguistic applications and language resources, extensibility of language resources (included the use of different resources), and low linguistic maintenance.

At the moment, the results of our investigation are promising in that the NMT concept perfectly fits with the trend of employing *lean production environments* in both manufacturing and information management areas.

### **5 Conclusions and Prospects**

In this paper we have introduced the prerequisites of a processing strategy within an NMT environment. NMT is based on the recent and emerging developments in the fields of information technology, Web technology and language technology. This new MT paradigm aims at combining the different technologies in an integrated way to achieve better translation throughput. This concerns the translation quality by negotiating and evaluating the deployed MT resources (MT systems, components of MT systems, lingware modules, user and quality requirements), and the translation processing speed by making use of distributed process modules. Technically this is facilitated by means of cooperating and communicating network-based intelligent software agents, which make use of local and distributed language resources that are based on vendor-independent standards. The language resources of this new kind of MT software are intended to be shareable with other Intranet/Internet based language technology applications and services, such as multilingual search engines, text retrieval, information brokering and telematics services.

The presented approach is applicable in an industrial multilingual information management environment, where it enables the increase of document quality and the decrease of the lead time of the production process together with an overall reduction of the production costs.

#### **References**

- [SAP, 1996] SAP. *OTELO - EU Project LE-2703*. SAP AG, Walldorf, Germany, 1996.
- [Schütz, 1996] Schütz, Jörg 1996. Network-based Machine Translation Services. In *Proceedings of EAMT Machine Translation Workshop 1996*, Vienna, Austria, pp. 145-157.