1.3: Survey of Machine Translation Evaluation

Distribution: Public

EuroMatrix
Statistical and Hybrid Machine Translation
Between All European Languages
IST 034291 Deliverable 1.3

Project funded by the European Community
under the Sixth Framework Programme for
Research and Technological Development.

Information Society
Technologies

Project ref no.	IST-034291
Project acronym	EuroMatrix
Project full title	Statistical and Hybrid Machine Translation Between All European Languages
Instrument	STREP
Thematic Priority	Information Society Technologies
Start date / duration	01 September 2006 / 30 Months

Distribution	Public
Contractual date of delivery	n.a.
Actual date of delivery	December 7, 2007
Deliverable number	1.3
Deliverable title	Survey of Machine Translation Evaluation
Type	
Status & version	
Number of pages	80
Contributing WP(s)	WP1
WP / Task responsible	Cameron Shaw Fordyce
Other contributors	
Author(s)	
EC project officer	Xavier Gros
Keywords	

The partners in EuroMatrix are:
Saarland University (USAAR)
University of Edinburgh (UEDIN)
Charles University (CUNI-MFF)
CELCT
GROUP Technologies
MorphoLogic

For copies of reports, updates on project activities and other EuroMatrix-related information, contact:
The EuroMatrix Project Co-ordinator
Prof. Hans Uszkoreit
Universität des Saarlandes, Computerlinguistik
Postfach 15 11 50
66041 Saarbrücken, Germany
uszkoreit@coli.uni-sb.de

Phone +49 (681) 302-4115- Fax +49 (681) 302-4700

Copies of reports and other material can also be accessed via the project's homepage: http://www.euromatrix.net/

Contents

List of Figures

Chapter 1

Introduction

This survey, while presenting a general overview of the evaluation methods for Machine Translation (MT) systems, was conceived as a publicly available resource within the framework of Euromatrix, a European research project which aims at increasing the quality of automatic translation of technical, social, legal and political documents by developing an MT system prototype able to deal with any possible pair of official EU languages.
This project also promotes the investigation of novel combinations of statistical techniques and linguistic knowledge sources as well as hybrid machine translation architectures in order to combine the accuracy of rule-based techniques with the adaptability of data-driven approaches. For most translation directions that involve languages of the new and near-term prospective member states, the project aims at providing baseline MT functionality for the first time.

Through the Euromatrix portal (www.euromatrix.net) different resources are made available for public use, such as development and dev-test sets, language models trained for each language, an open source decoder for phrase-based Statistical MT (SMT) called Moses ([Koehn and Monz, 2006]), a training script to build models for Moses and sentence-aligned training corpora for most European languages. Annually the project partners organize an evaluation campaign in which the performances of several MT systems are compared in order to assess what the state of the art in MT is. In these campaigns, two different approaches are used to evaluate systems results: human and automatic MT evaluation. The automatic evaluation uses the translation of the test-set provided by the EuroMatrix team, as a reference.

Machine Translation Evaluation (MTE) is therefore a central issue among the activities of the Euromatrix project, and it is not an easy task. The choice of property to evaluate, how to evaluate it, and what context to use in the evaluation are problematic issues. Evaluating machine translation (MT) is important for everyone involved: researchers need to know if their theories make a difference, commercial developers want to impress customers and users have to decide which system to employ. Given the richness of the literature, and the complexity of the enterprise, there is a need for an overall perspective, something that helps the potential evaluator approach the problem in a more informed way, and that might help pave the way towards an eventual theory of MT evaluation [Hovy et al., 2002].

Evaluation is difficult because the ultimate criteria is translation quality, which can, itself, be difficult to judge. Evaluating the quality of a translation is an extremely subjective task, and disagreements about evaluation methodology are widespread. Nevertheless, evaluation is essential, and research on evaluation methodology has played an important role from

the earliest days of MT [Miller and Beebe-Center, 1958] to the present. A principle-based approach to MT evaluation takes into account the fact that no unique evaluation scheme is acceptable for all evaluation purposes [Hovy, 2002].

Most lay-users of machine translation systems usually evaluate the quality of the resulting translation by asking the system to translate the output sentence back into the source language. This process is called round-trip translation and involves the following steps:

- ask the system to translate a text from the language A to the language B;
- take the resulting translation, and ask the system to translate it back into language A.

It is a commonly held opinion that a good system should return the original text. Nevertheless it is evident that it is not a good way to evaluate the performance. The following two cases prove this.

Case 1: The system produced a good translation from language A to B, but a bad one from B to A.

- Lang. A - Select this link to look our home page.
- Lang. B - Selezioni questo collegamento per guardare la Home Page.
- Lang. A - Selections this connection in order to watch our Home Page.

Case 2: The system produced a literal but meaningless translation of an idiomatic phrase. However, the translation of the resulting text from language B back into language A reproduces the original text.

- Lang. A - Tit for tat.
- Lang. B - Melharuco para o tat.
- Lang. A - Tit for tat.

These problems occur because the round-trip translation uses two machine translation systems, and so one cannot identify where the error occurs. Moreover even a pair of good translators would not be expected to complete a perfect RTT. They may return a perfectly correct translation which is not, however, word-for-word identical to the source text.

Three kinds of evaluations are defined based on their appropriateness for achieving different goals
[Hirschman and Thompson, 1996]:

Adequacy Evaluation: The goal of this kind of evaluation is to determine the fitness of a system for a given purpose. That is:

1. does it do what is required?;
2. how well?;
3. at what cost?

Adequacy evaluations are focused on the users and their needs.

Diagnostic Evaluation: The goal of this kind of evaluation is to examine the system's performance profile with respect to a range of possible inputs. To this extent, diagnostic evaluations require the creation of a test suite that enumerates all the elementary linguistic phenomena addressed in the input domain. Once the test suite is constructed, comparisons of two generations of the same system can be done automatically. Diagnostic evaluations are typically done by or for system developers.

Performance Evaluation: The goal of this kind of evaluation is to measure the system's performance in one or more specific areas. Performance evaluations are typically done to assist developers and researchers.

Another evaluation type distinction is Glass-box versus Black-box.

In black-box MT evaluation, the evaluator has access only to the input and output of the system under evaluation. Black-box evaluation tends to focus on evaluating the translation quality of the output. Essentially it is an attempt to measure the acceptability of the translation to the users. To produce the most objective measure possible, a standard test-suite of input /output pairs is established for judging whether the system is performing "correctly" and whether it will be cost-effective. Another difficulty in applying the black-box evaluation approach is the number of dimensions to which MT developers must limit their systems. These systems can be thought of as shells that are customized to apply to a particular domain, language pair, or type of text. Due to this quality, some systems may need to be customized for the chosen ranges in order to perform comparative evaluations[Jordan, 1991].

In Glass-box evaluation, the evaluator also has access to the various workings of the system and can thus assess each sub-part of the system. Component-based evaluation and detailed error analysis are also important types of evaluation [Nyberg et al., 1994]. The glass-box approach attempts to evaluate the system's internal processing strategies to measure how well the system does something. This type of evaluation should include a determination of the system's linguistic coverage, and an examination of the linguistic theories used to handle the linguistic phenomena. The examination of the linguistic theories used should include how closely these theories were followed during the implementation, and noting what modifications had to be made to the theories. The performance of the system's various modules should be examined and the evaluation of each of these modules should be treated as individual black-box evaluations.

Because of the complexity of natural languages, manual evaluation of MT is the most reliable evaluation. Nonetheless it has some disadvantages that move researchers to use automatic evaluation metrics. First of all human evaluation is very expensive: it is time consuming and needs the effort of many people to complete the task. Another important problem is that two persons may evaluate the same translation in different ways, and it could happen that the same person gives different opinions on the same translation at different times.

Both these problems are not relevant if algorithms are used for the evaluation. The expense is considerably reduced as it only requires an algorithm to be executed: it can be used on a different translation (reusable) and it would give the exact same judgements if the same texts are used (reproducible). Yet automatic MT evaluation is not as reliable as a human one.

Since during the history of MT several human and automatic evaluation metrics have been proposed and tried this survey provides an introductory overview to MT evaluation,

starting with a discussion of some issues concerning the MT task through the analysis of some common approaches to automatic translation (Chapter 2), in order to focus afterwards on human (Chapter 3) and automatic translation (Chapter 4).

Chapter 2

Machine Translation Overview

MT of Natural Language (NL) is a very difficult task. It can be perceived as the simple substitution of words in one natural language for words in another. Yet it is not so simple because of the complexity of natural languages: many words have various meanings and so they can be translated in different ways. Also, the sentences might be ambiguous and have various meanings. The relationship between linguistic entities is often vague; grammatical relations can vary depending on the languages, and translating sentences from languages having different relations means reformulating the sentence. Besides, problems due to the associated world knowledge may be encountered and these are usually difficult to solve.

From a linguistic point of view, we have to consider various types of dependencies:

- morphologic;
- syntactic;
- semantic;
- pragmatic dependencies.

"Machine translation of natural languages, commonly known as MT, has multiple personalities [Nirenburg and Wilks, 2000]. First of all, it is a venerable scientific enterprise, a component of the larger area of studies concerned with the studies of human language understanding capacity. Indeed, computer modelling of thought processes, memory and knowledge is an important component of certain areas of linguistics, philosophy, psychology, neuroscience, and the field of artificial intelligence (AI) within computer science. MT promises the practitioners of these sciences empirical results that could be used for corroboration or refutation of a variety of hypotheses and theories. But MT is also a technological challenge of the first order. It offers an opportunity for software designers and engineers in constructing very complex and large-scale non-numerical systems and for computational linguists, an opportunity to test their understanding of the syntax and semantics of a variety of languages by encoding this vast, though rarely comprehensive, knowledge into a form suitable for processing by computer programs" [Nirenburg and Wilks, 2000].

Typically, three different types of MT systems are distinguished according to the level of analysis that is performed. Figure 2.1 [Och, 2000] gives the standard visualization of the three approaches:

1. Direct Approach;

2. Transfer Approach;

3. Interlingua Approach.

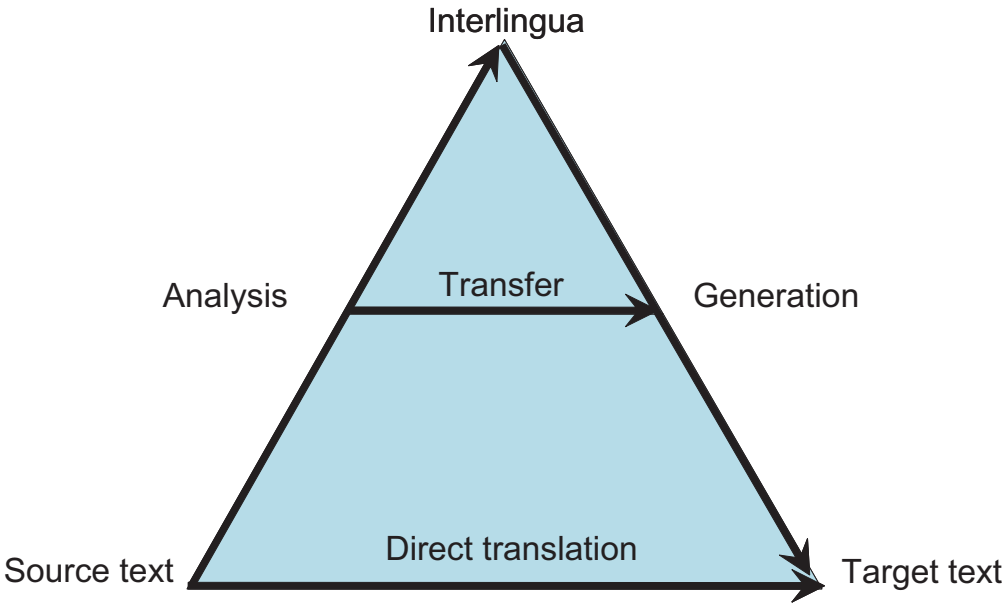

Figure 2.1: Different levels of analysis in an MT system

2.1 Direct Approach

Historically, the 'direct approach' was the first developed; it is adopted by most MT systems where a word-for-word translation from the source language to the target language is performed. An example of the direct approach method is the GAT (Georgetown Automatic Translation) system. Direct systems are limited to the minimum work necessary to effect the translation; for example, disambiguation is performed only to the extent necessary for translation into that one target language, irrespective of what might be required for another language [Slocum, 1985].

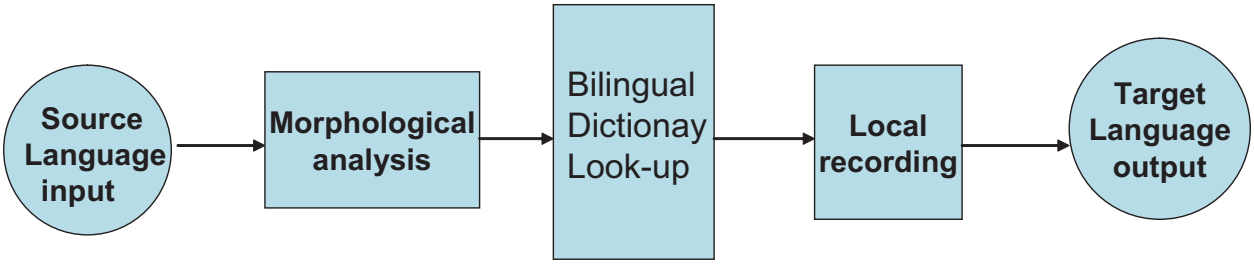

Figure 2.2: Direct MT System

Begun in 1952, and supported by the U.S. government, Georgetown's GAT system became operational in 1964 with its delivery to the Atomic Energy Commission at Oak Ridge

National Laboratory, and to Europe's corresponding research facility EURATOM in Ispra, Italy. Both systems were used for many years to translate Russian physics texts into "English". The output quality was quite poor compared with human translations, but for the intended purpose of quickly scanning documents to determine their content and interest, the GAT system was nevertheless superior to the only alternatives: slow and more expensive human translation or, worse, no translation at all. GAT was not replaced at EURATOM until 1976; at ORNL, it seems to have been used until at least 1979 [Slocum, 1985].

The GAT strategy was direct and local: simple word-for-word replacement, followed by a limited amount of transposition of words to result in something vaguely resembling English.

The Georgetown MT project was terminated in the mid-60s.

2.1.1 A Brief History of SYSTRAN

Peter Toma, Ph.D., a linguist researcher for MT, began his work in 1957 at the California Institute of Technology. Later, Dr. Toma became involved in the pioneering work in Russian to English MT at Georgetown University, the largest MT project in the US of that time. In 1968, Dr. Toma established a company in La Jolla, California, USA, with a product called SYSTRAN; an acronym for System Translation. Soon thereafter, the company was contracted to develop Russian to English MT for the US Air Force [Slocum, 1985] [1].

The first SYSTRAN system was tested in early 1969 at Wright-Patterson Air Force Base in Dayton, Ohio, USA. Since 1970, the system has continued to provide translations for the US Air Force's Foreign Technology Division.

In 1975, Dr. Toma demonstrated a prototype of English to French MT to representatives of the Commission of the European Communities (CEC), which resulted in a contract to develop MT systems for various European language pairs. The CEC uses more than 12 SYSTRAN MT systems for the translation of its internal documents.

In 1981, SYSTRAN initiated the development of the Japanese to English and English to Japanese MT systems. Under new European influence, the first World SYSTRAN Conference, organized by the CEC, was held in Luxembourg, shortly thereafter. This conference, the first dedicated to one single MT system, brought together all the principal SYSTRAN users from around the world. SYSTRAN introduced the utility "Customer Specific Dictionaries", (referred to as CSDs), in 1989. CSDs are dictionaries created by users with their own specific terminology.

As the market for MT begins to show traces of maturity in 2002, more and more corporations realize that the implementation of a customized MT solution can be a great benefit to the company and can help them to remain competitive in today's multilingual marketplace.

Today, 36 SYSTRAN MT language pairs are commercially available.

Here is a list of the source and target languages SYSTRAN works with. Many of the pairs are to or from English or French:

Russian into English (1968), English into Russian (1973), English source (1975) for the European Commission, Arabic, Chinese, Danish, Dutch, French, German, Greek, Hindi, Italian, Japanese, Korean, Norwegian, Serbo-Croatian, Spanish, Swedish, Persian, Polish, Portuguese, Ukrainian, Urdu.

[1] http://www.systranet.com/systran/net

2.2 Transfer Approach

In the transfer approach, the translation process is decomposed into three steps [Och, 2000]:

- analysis;
- transfer;
- generation.

According to [Slocum, 1985] "the transfer approach is characteristic of a system in which the underlying representation of the "meaning" of a grammatical unit (e.g., sentence) differs depending on the language from which it was derived or into which it is to be generated; this implies the existence of a third translation stage which maps one language-specific meaning representation into another: this stage is called Transfer. Thus, the overall transfer translation process is Analysis followed by Transfer and then Synthesis."

"In the analysis step, the input sentence is analyzed syntactically and semantically producing an abstract representation of the source sentence. In the transfer step, this representation is transferred into a corresponding representation in the target language. In the generation step, the target language sequence is produced. An example of a transfer approach is the TAUM system." [Och, 2000]

In 1965 the University of Montreal established the TAUM project with Canadian government (Department of Environment) funding. This was probably the first MT project designed strictly around the transfer approach [Slocum, 1985].

After an initial period of more-or-less open-ended research, the Canadian government began adopting specific goals for the TAUM system. A chance remark by a bored translator in the Canadian Meteorological Center (CMC) had led to a spin-off project: TAUM-METEO.

TAUM was commissioned in 1975 to produce an operational English-French MT system for weather forecasts. A prototype was demonstrated in 1976, and by 1977, METEO was installed for production translation [Thouin, 1982].

2.3 Interlingua Approach

The interlingua approach to machine translation (MT) is characterized by the following two stages [Nirenburgq et al., 1985]:

- translation of the source text into an intermediate representation, an artificial language (interlingua) which is designed to capture the various types of meaning of the source text;
- translation from the interlingua into the target text.

According to [Hiroshi, 1993] "there are two major merits of the interlingua approach in developing machine translation systems. The first is that the interlingua approach can localize the development of machine translation system. It is impossible to develop a machine translation system for a language by collecting people who have no knowledge of the language.

The rules to analyze and generate a language and the dictionaries must be developed by trained native speakers of the language. The interlingua interface completely separates analysis and generation, enabling the development of analysis and generation systems for one language to proceed independently from those of other languages. Developers of these systems need only know the interlingua and the language being analyzed or generated.

The second merit of the interlingua is the common use of knowledge for machine translation. World knowledge is needed in semantic analysis, which is essential for high quality machine translation. Knowledge described in interlingua may be used by the analysis systems for each language."

To understand the sentence written in the natural language both humans and computers must know the meaning of words and their usage. If this knowledge is described using interlingua, it can be utilized commonly in analysis systems of different languages. This makes for a more efficient use of the translating knowledge developed for each language (i.e. knowledge acquired for translating the interlingua and a specific natural language can be reutilised irrespective of the target language).

In the interlingua approach, a very fine-grained analysis produces a completely language independent representation of the input sentence. This representation is used to produce the target language sentence. An often claimed advantage of the interlingua approach is that developing translation systems between all pairs of a set of $n >> 1$ languages is more efficient. There are only n components which need to be translated into the interlingua and n components are needed to translate from it. In a transfer approach or a direct translation approach, the development of $n(n-1)$ components for each pair of languages is needed [Och, 2000].

An example of an interlingua approach is the CETA system.

In 1961 a project to translate Russian into French was started at Grenoble University in France. Grenoble began the CETA project with a clear linguistic theory [Slocum, 1985].

The theoretical basis of CETA was interlingua (implying a language-independent, neutral meaning representation) at the grammatical level, but transfer (implying a mapping from one language-specific meaning representation to another) at the lexical [dictionary] level. The state of the art in computer science still being primitive, Grenoble was essentially forced to adopt IBM assembly language as the software basis of CETA [Hutchins, 1978]. The CETA system was under development for ten years; during 1967–71 it was used to translate 400,000 words of Russian mathematics and physics texts into French. The CETA workers learned that it is critically important in an operational system to retain surface clues about how to formulate the translation (Indo-European languages, for example, have many structural similarities, not to mention cognates, that one can take advantage of), and to have "fail-soft" measures designed into the system (i.e., the ability of failing with a backup; although it does not correct the problem, it avoids blocking).

An interlingua does not allow this easily, if at all, but the transfer approach does. A change in hardware (thus software) in 1971 prompted the abandonment of the CETA system, immediately followed by the creation of a new project/system called GETA, based entirely on a fail-soft transfer design [Slocum, 1985].

2.4 Rule-based vs empirical approach

Most current classification methods according to the core technology and the processing of text for translation used, are distinguished in rule-based and empirical approaches [Och, 2000].

In the rule-based approach, human experts specify a set of rules, aimed at describing the translation process. Typically, this is very expensive as linguistic experts need to work on it.

Using an empirical approach, the knowledge sources to develop an MT system are computed automatically by analyzing example translations. A major advantage of empirical approaches to MT is that MT systems for new language pairs and domains can be developed very quickly, provided that sufficient training data is available. Figure 2.3 [Och, 2000] shows the architecture of an empirical MT system. As in [Och, 2000] "in a fully-fledged empirical approach, the starting point is a parallel training corpus that consists of translation examples, which were produced by human translators. In the training phase, the necessary knowledge sources are computed automatically. The search or decision process has to achieve an optimal combination of the knowledge sources to perform an optimal translation. An empirical approach might pursue a direct or a transfer approach.

In the empirical approaches, two kinds of systems can be distinguished:

1. example-based MT;

2. statistical MT.

In example-based MT, a translation of a new sentence is performed by analyzing similar translation examples previously seen. In statistical MT, the translation examples are used to train a statistical translation model. The decision rule used to decide for the actual translation is derived from statistical decision theoretic considerations."

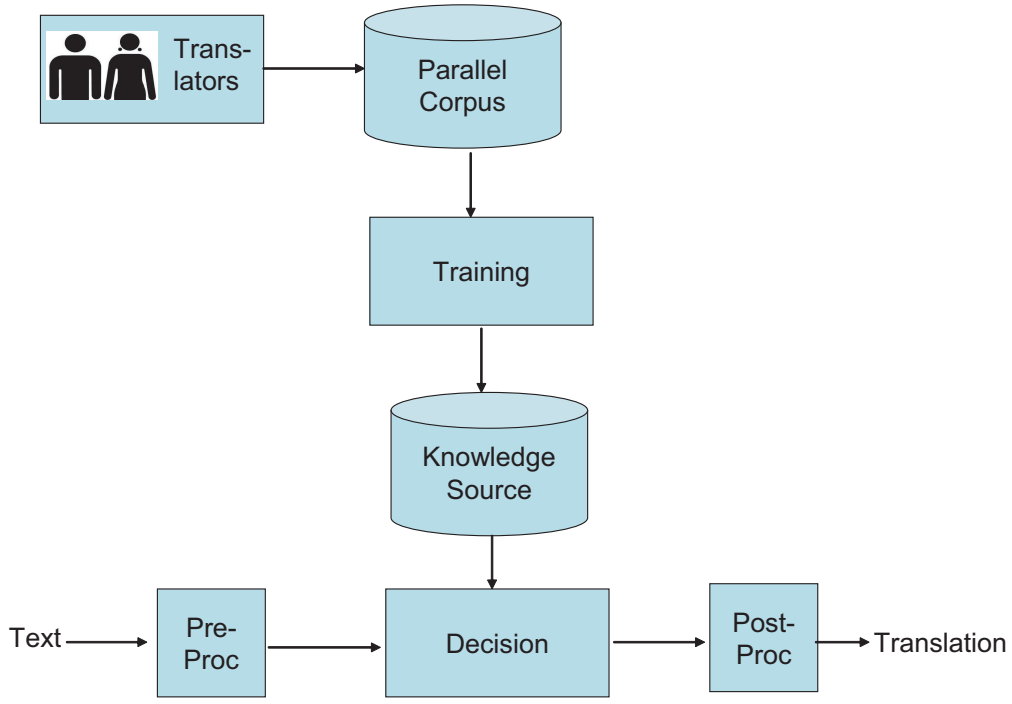

Figure 2.3: Architecture of an empirical MT system

2.4.1 Rule-based Machine Translation Systems

Rule-Based Machine Translation (RBMT) is a major approach in MT research. RBMT needs linguistic knowledge to create appropriate rules of translation. RBMT paradigm is associated with systems that rely on different linguistic levels of rules for translation between the source and the target language. The prototypical example is Rosetta [Rosetta, 1994], an interlingual system which divides translation rules in 2 categories [J.Dorr et al., 1998]:

- M-rules, which are "meaning-preserving rules", map between syntactic trees to underlying meaning structures;
- S-rules, which are "non-meaningful rules", map lexical items to syntactic trees;

Most of the currently available commercial machine translation systems are rule-based translation systems, in which rules play a central role, mainly because it is difficult to gather data that exhaustively cover diverse language phenomena. In rule-based MT systems, translation is based on formalized linguistic knowledge, represented in dictionaries and grammars. This type of approach, however, makes it difficult to port a particular system to other domains, or to upgrade the system to accommodate new expressions. With the increased availability of substantial bilingual corpora in the 1980s, corpus-based machine translation (MT) technologies, such as example-based MT and statistical MT, were proposed to cope with the limitations of the rule-based systems that had formerly been the dominant paradigm.

A rule-based approach is characterized by several features,

- It has a strict sense of well-formedness in mind;
- Grammatical errors are explicitly prohibited;
- Most rules are based on existing linguistic theories that are linguistically interesting.

When the required knowledge does not appear in any literature, ad hoc heuristic rules are commonly used. The major advantage of a rule-based system is that existing linguistic knowledge can be incorporated into the system directly.

In RBMT, as its name suggests, there are linguistic "rules" which describe and determine how to interpret the internal structure of the words ("morphology", e.g. the '-s' of 'cakes' indicates plural), as well as how the words combine to form sentences ("syntax", e.g. article + noun is correct, but article+verb is not: in this way the category of 'book' in 'the book' is identified). Of course, the rules are generally more complex than that. Other rules may relate to more subtle aspects. For example, the translation of a word like 'for' can depend on its function in the sentence. This is the traditional approach to MT in which input text is first analyzed into a more or less abstract representation. This representation of the SL(source-language) text is then manipulated so as to be more appropriate for the TL(target-language), from which the TL text is then generated. This manipulation, usually called "transfer", is also rule-based, as is the "generation" process. These "representations" are often in the form of a tree structure, very familiar to linguists, which show explicitly the relationships between the parts of the sentence, though other representations are also possible. Rules for TL generation deal only with TL phenomena such as word-order, agreement, mutation and so on.

But where do these rules come from? This is perhaps the biggest draw-back to the rule-based approach to MT, because someone has to write them! Typically this is the job of computational linguists who study the language(s) concerned and try to write computational grammars that correctly analyze and generate grammatical structures. This task is facilitated by the existence of various rule-writing "formalisms" which have been developed over the years, with associated software so that the grammars can be tested on computer. In some cases there are even sophisticated software tools which make the job easier, by allowing the grammar writers to check their grammars for consistency and redundancy, and to visualize with graphic interfaces how the rules work.

By far the biggest "overhead" in RBMT is the dictionary. A system's dictionary is somewhat different from what is expected of a dictionary for human use. For MT we need to know the grammatical properties of a word, some of which may be obvious to human users. RBMT is a well-understood paradigm, and can be safely predicted the likelihood of success depending on various factors. The most successful commercial MT systems to date are all rule-based, though many reflect a long period of development and investment. [Somers, 2004].

A rule-based system is an effective way to implement a machine translation system because of its extensibility and maintainability [Kaji, 1988]. However, it is disadvantageous in processing efficiently. In a rule-based machine translation system the grammar consists of a lot of rewriting rules. While the translation is carried out by repeating pattern matching and transformation of graph structures, most rules fail in pattern matching. It is to be desired that pattern matching of the unfruitful rules should be avoided.

The logical relationships between rules are pre-analyzed and a set of antecedent actions, which are prerequisites for the condition of the rule being satisfied, is determined for each rule.

During execution time, a rule is activated only when one of the antecedent actions is carried out. The probability of a rule being activated is reduced to close to the occurrence probability of its relevant linguistic phenomenon. As most rules relate to linguistic phenomena that rarely occur, the processing efficiency is drastically improved.

In rule-based machine translation, a grammar is comprised of a lot of rewriting rules [Boitet, 1982] [Nakamura, 1984]. Translation is carried out by repeating pattern matching and transformation of tree or graph structures that represent the syntax or semantics of a sentence. A great part of the processing time is spent in pattern matching which mostly results in failure. The key to improve the processing efficiency is to avoid the pattern matching that results in failure. A number of methods such as the Rete pattern match algorithm [Forgy, 1982] have been developed to improve the processing efficiency of rule-based systems. However, peculiarities in machine translation systems make it difficult to apply the whole of an existing method. The general idea of existing methods is to restructure the set of rules in a network such as a cause-effect graph or a discriminant network, and maintain the state of the object in the network. The following are distinguishing features of a machine translation system [Kaji, 1988].

First, the object data is a graph structure, and the state of the object must be handled as a collection of states of respective sub-graphs which are created dynamically by applying rules. Therefore, maintaining the state of the object in a network causes a large amount of overheads.

Secondly rules are applied in a controlled manner, so that a linguistically insignificant result is avoided. The computational control of rules to improve the processing efficiency must be superimposed on the linguistic control of rules.

Most of the machine translation software on the market today is rule-based. These systems consist of:

- a process of analyzing input sentences (morphological, syntactic and/or semantic analysis);
- a process of generating sentences as a result of a series of structural conversions based on an internal structure or some interlingua.

The steps of each process are controlled by the dictionary and the rules. As the accuracy of translation by the system is the product of the accuracies of each process, it is necessary to enlarge the magnitude and to upgrade the precision of existing dictionaries and rules for each step and this is extremely labour intensive.

Further, in-depth analysis enables the use of long-distance relationships and related information yet they tend to lose the collocations relations between words. In addition, most text produced by rule-based methods are incohesive. This is for two reasons:

- the rules needed to increase cohesion are not yet fully understood
- those that are understood often rely on a full semantic and pragmatic analysis of the text, which is rarely available. [Satoshi Shirai and Takahashi, 1997]

However, the descriptive convenience gained by using rule bases comes at a cost: the global behaviour of the interacting rules is difficult to predict or to analyze automatically.

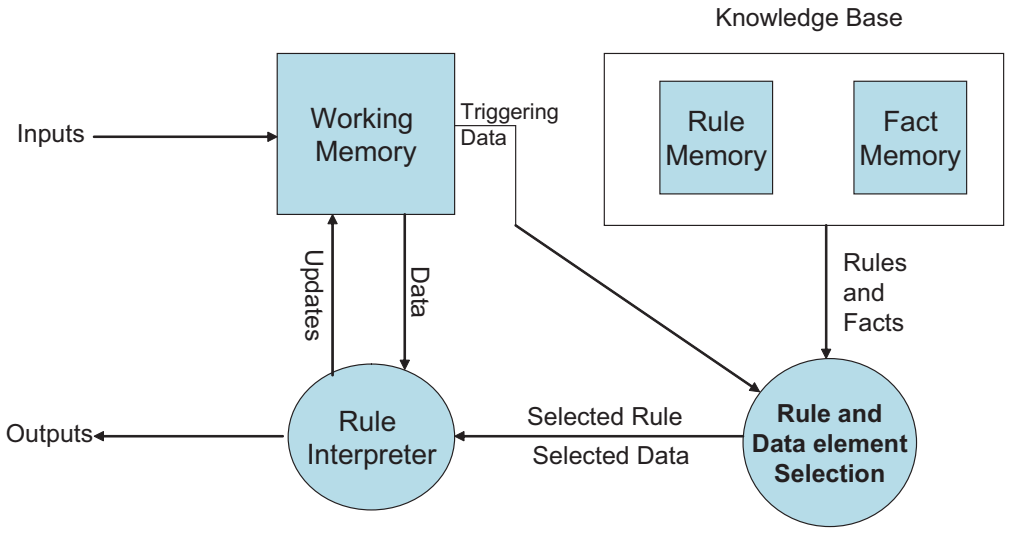

Figure 2.4: The Basic Features of an RBS

To create a rule-based system for a given problem, you must have (or create) the following [url, a]:

1. A set of facts to represent the initial working memory. This should be anything relevant to the initial state of the system.

2. A set of rules. This should encompass any and all actions that should be taken within the scope of a problem, but nothing irrelevant. The number of rules in the system can affect its performance.

3. A condition that needs to be satisfied by the final solution and to indicate that a solution is not possible. This is necessary to terminate rule-based systems that may find themselves in infinite loops otherwise.

Why Not Rule-Based Systems

In [Su and Chang, 1992] there is an interesting discussion on difficulties that make a rule-based approach unfavourable to meeting the design goals for a large scale system. Here is the list of problems:

- It is hard to handle uncertainty due to the lack of an objective preference measure. Therefore, it is awkward in dealing with highly ambiguous constructs. Since real text is not always strictly ill-formed, such a system also has little capacity in dealing with ill-formed input.

- It is hard to deal with complex and irregular knowledge. When exceptions are encountered, lexicon specific and ad hoc procedures are commonly used, which make the system difficult to manage and maintain.

- It is hard to maintain consistency of the knowledge bases among different persons at different times. For a practical MT system, which usually takes a long period of development, the inconsistency often introduces diverse effects into the system performance and increases the difficulty and cost of maintenance.

- It is not easy for a rule-based system to attain high coverage for real text. First, the knowledge is usually confined to linguistically interested phenomena; linguistically uninterested phenomena are handled specifically as exceptions.

  Secondly, designers tend to incorporate a lot of specific knowledge and increase the complexity of the model to make the system work for some testing cases. However, due to the lack of robustness, the coverage for real text tends to be poor. Consequently, such approaches usually fail to attain the required coverage for unforeseen text because the required fine-grained knowledge is not included in the over-tuned knowledge base.

- A rule usually takes care of only local constraints. The effects of adding a rule may cause unpredictable side effects over the global performance of the system, and global improvement is not guaranteed. This prevents a system designer from predicting the system performance in terms of well characterized system parameters; therefore, it is hard to tune such a system.

- Finally, there is usually no systematic and automatic approach to acquire the rules for large-scale applications. The lack of acquisition tools may make a formalism unfavorable in real applications even though it is theoretically interesting.

Accordingly, although rule-based systems could be tuned to show impressive performance in small scale, they may not be satisfactory for a large scale system.

"Traditional RBMT systems involve many human cost in formulating rules" [Bennett and Slocum, 1985]. This easily introduces inconsistencies. Rules are usually universal, i.e., they are not domain dependent [Sumita et al., 1990].

Computational Cost

Computational cost is considerable in RBMT. RBMT is really a large-scale rule-based system, which consists of analysis, transfer, and generation modules using syntactic rules, semantic restrictions,structural transfer rules, word selections, generation rules, and so on.

Improvement Cost

In RBMT, it is too difficult to keep all rules consistent because improvement of translation quality is made by modifying rules that are mutually dependent.

System Building Cost

Formulating linguistic rules for RBMT is a difficult job and needs a linguistically trained staff.

Context-Sensitive Translation

RBMT needs another understanding device in order to translate context-sensitively.

Robustness

RBMT works on exact-match reasoning. RBMT fails to translate when it has no knowledge that matches the input exactly. If RBMT included a fail-safe mechanism to search rules which can translate an expression similar to the input, RBMT could then translate by borrowing the rules found.

Reliability factor

RBMT has no device to compute the reliability of the result.

2.4.2 Methods of Rule-Based Systems

According to [url, b] here are reported the two methods available for Rule-Based Systems, that are :

1. Forward-Chaining;
2. Backward-Chaining;

Of the two methods available, forward- or backward-chaining, the one to use is determined by the problem itself. A comparison of conditions to actions in the rule base can help determine which chaining method is preferred. If the average rule has more conditions than conclusions, that is the typical hypothesis or goal (the conclusions) can lead to many more questions (the conditions), forward-chaining is favoured. If the opposite holds true and the average rule has more conclusions than conditions such that each fact may fan out into a large number of new facts or actions, backward-chaining is ideal.

If neither is dominant, the number of facts in the working memory may help the decision. If all (relevant) facts are already known, and the purpose of the system is to find where that information leads, forward-chaining should be selected. If, on the other hand, few or no facts are known and the goal is to find if one of many possible conclusions is true, use backward-chaining.

Forward-Chaining In some problems, information is provided with the rules and the AI follows them to see where they lead. An example of this is a medical diagnosis in which the problem is to diagnose the underlying disease based on a set of symptoms (the working memory). A problem of this nature is solved using a forward-chaining,

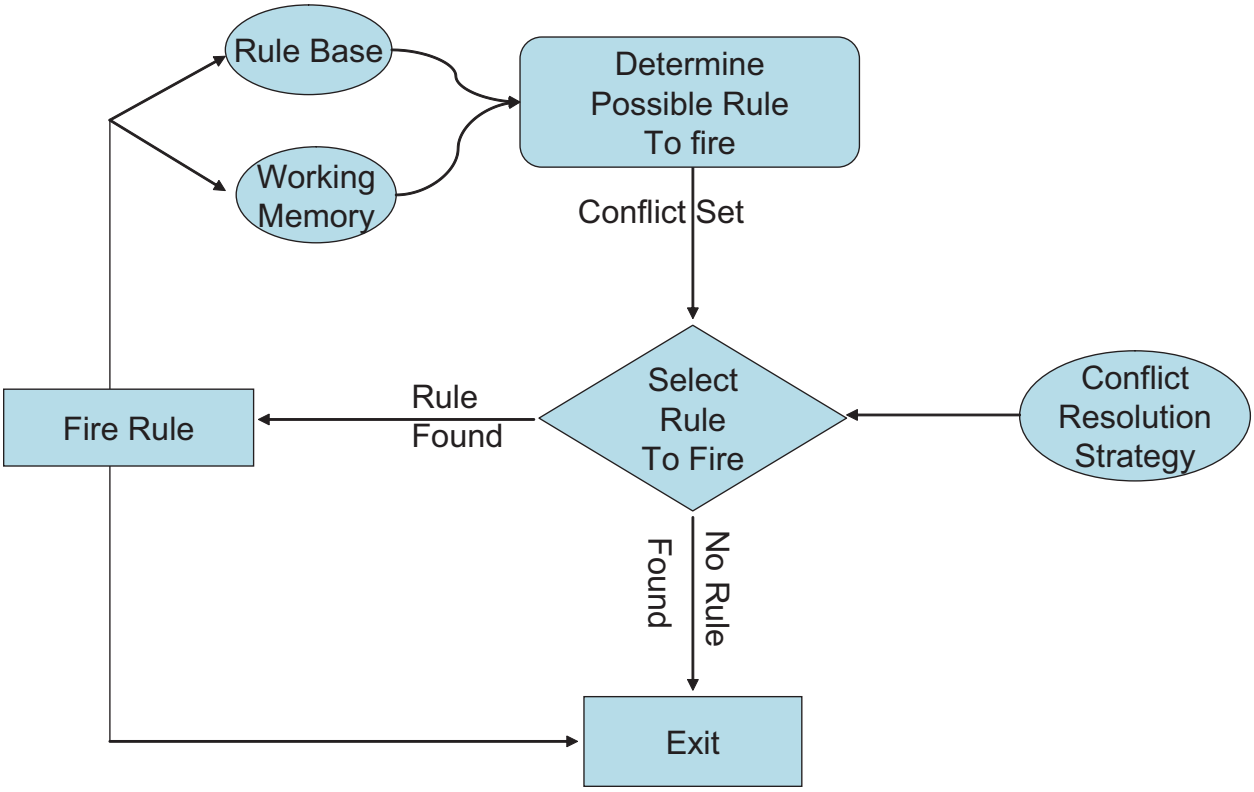

Figure 2.5: Forward-Chaining Procedure

data-driven, system that compares data in the working memory against the conditions (IF parts) of the rules and determines which rules to fire.

Backward-Chaining In other problems, a goal is specified and the AI must find a way to achieve that specified goal. For example, if there is an epidemic of a certain disease, this AI could presume a given individual had the disease and attempt to determine if its diagnosis is correct based on available information. A backward-chaining, goal-driven, system accomplishes this. To do this, the system looks for the action in the THEN clause of the rules that matches the specified goal. In other words, it looks for the rules that can produce this goal. If a rule is found and fired, it takes each of that rule's conditions as goals and continues until either the available data satisfy all of the goals or there are no more rules that match. MYCIN uses backward-chaining.

The strengths of the rule-based method lie in the fact that information can be obtained through introspection and analysis. The weakness of the rule-based method is that the accuracy of the entire process is the product of the accuracies of each sub-stage.

2.4.3 Example-based Machine Translation Systems

The idea for EBMT came out from the paper presented by Nagao [Nagao, 1984] at a conference in 1981 and published only 3 years later. He stressed the notion of detecting *similarity*.

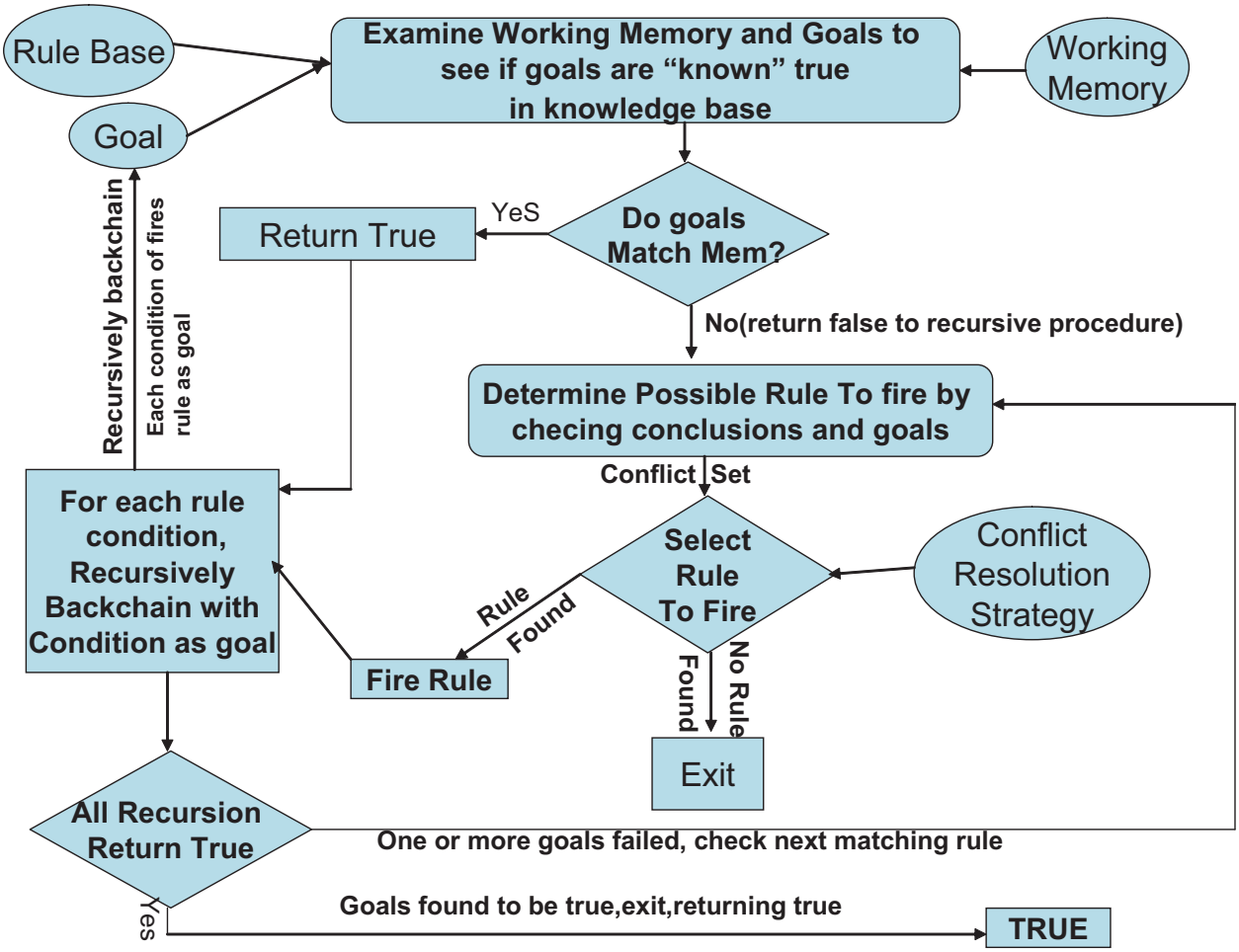

Figure 2.6: Backward-Chaining Procedure

The most important function is to find out the similarity of the given input sentence and an example sentence, which can be a guide for the translation of the input sentence [Nagao, 1984].

[Chunyu Kit and Webster, 1992] show how EBMT works: "EBMT is about how to decode knowledge from bilingual texts, where the knowledge seems to have no overt formal representation or any encoding scheme. The EBMT approach became popular soon after some positive results were published in a number of papers demonstrating its plausibility. Sato and Nagao [Sato and Nagao, 1990] investigated the problem of example selection by approximate (or inexact) matching of input sentences and example sentences, using a similarity measure based (score of the translation based on the score of the matching expression). By around 1993, EBMT had become an established research field of MT and many example-based techniques were applied to various MT tasks. Sato [Sato, 1993] attempted the example-based translation of computer technical terms with respect to the focus term and its surrounding

contexts and reported an overall accuracy of 96%, with an accuracy of 92% for unknown terms."

In the early research of EBMT, e.g., in early 1990's, many researchers tended to focus on examples at the sentence level.

An example is a pair (or couple) of texts in two languages that are a translation of each other. The texts can be of any size at any linguistic unit: words, phrase, clause, sentence, and paragraph. A critical issue that needs to be examined closely in this context is the number of examples over a large-scale bilingual corpus, which can be unlimited in practice. An example can be further decomposed, in more than one possible way, into sub-structures or shorter examples, and that examples can overlap with each other. Therefore, the example number can be exponentially large in respect to the corpus size, if all possible examples from a bilingual corpus are selected. Consequently, the problems of scale of EBMT might arise, because any fragment of a sentence can be an example.

How to control an EB to a reasonable size becomes vitally critical. It has to be determined which examples should be filtered out and which ones should be maintained in the EB, not only for the matter of efficiency but, more importantly, for practicality.

The Four Stages of EBMT

In general, there are four stages of work in EBMT, namely:

Example acquisition Example acquisition is about how to acquire examples from parallel bilingual corpus (i.e., existing translation).

Example base management Example base management is about how examples are stored and maintained.

Example application The example application concerns itself with how examples are used to facilitate translation, which involves the decomposition of an input sentence into examples and the conversion of source texts into target texts in terms of existing translation.

Target sentence synthesis The sentence synthesis is to compose a target sentence by putting the converted examples into a smoothly readable order, aiming at enhancing the readability of the target sentence after conversion.

There are various resources from which we can collect examples. For example, from bilingual dictionaries can be collected examples at the word level. Text alignment is a necessary step towards example acquisition at various levels. The approaches to text alignment can be categorized into two types, namely,

- resource-poor approaches;
- resource-rich approaches.

The resource-poor approach mostly focuses on sentence alignment and relies mainly on sentence length statistics, co-occurrence statistics and some limited lexical information.

The resource-rich approach makes use of whatever is available and useful, in particular, bilingual lexicons and glossaries, to facilitate the alignment."

Once the relevant example or examples have been selected, the corresponding fragments in the target text must be selected. This has been termed alignment or adaptation and involves contrastive comparison of both languages. Once the appropriate fragments have been selected, they must be combined to form a target text, as the generation stage of conventional MT puts the finishing touches to the output.

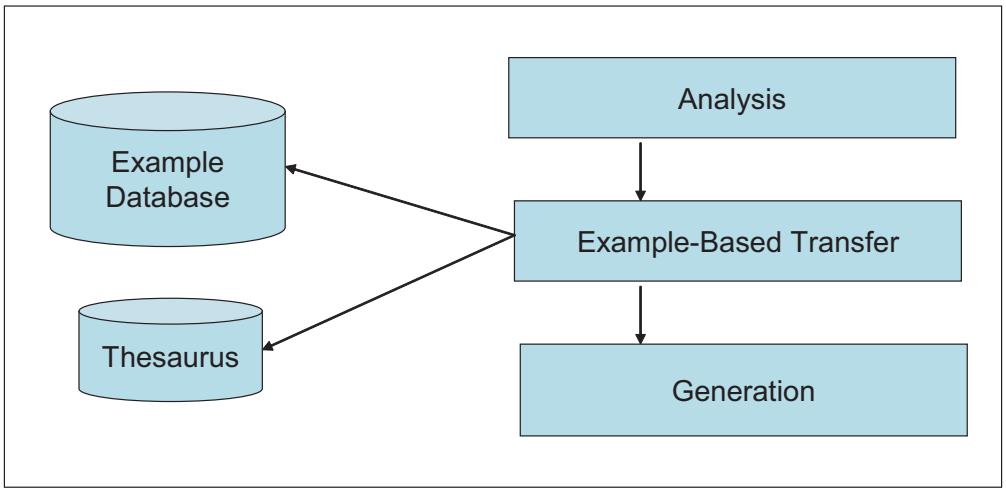

Figure 2.7: EBMT System Configuration

Example-based MT (EBMT) can be seen as a hybrid of RBMT and SMT [Somers, 2004]. Like SMT, it depends on a corpus of already existing translations, which it reuses as the basis for a new translation. In this respect it is similar to (and sometimes confused with) the translator's aid known as a Translation Memory (TM). Both EBMT and TM involve matching the input against a database of real examples, and identifying the closest matches. They differ in that in TM it is then up to the translator to decide what to do with the proposed matches, whereas in EBMT the automatic process continues by identifying corresponding translation fragments, and then recombining these to give the target text.

The process is thus broken down into three stages:

- "matching" fragments against a database of real examples (which EBMT and TM have in common);
- "alignment" identifying the corresponding translation fragments;
- "recombination" to give the target test.

The basic idea of EBMT assumes a database of parallel translations which is searched for the source language sentences and phrases closest matching a new source language sentence.

The translation of the matched phrases is then modified and combined to form a transfer translation of the new sentence. The closeness of the match would be determined by the semantic "distance" between the two content words as measured by some metric based on a thesaurus or ontology. The accuracy and quality of the translation depends heavily on the size and coverage of the parallel database.

EBMT systems differ widely in how the translation examples themselves are actually stored. In the simplest case, the examples may be stored as pairs of strings, with no additional information associated with them. Sometimes, indexing techniques borrowed from Information Retrieval (IR) can be used [Somers, 1998].

"In run time systems, the full database of examples is made accessible and subject to any manipulation as required during matching and extracting processes. Such use of the database is ancillary (however essential) to the basic operation of converting SL input into TL output" [Hutchins, 2005]. In other EBMT systems, the analysis of the database is made in preparatory operations, before actual SL texts (input sentences) are processed for translation.

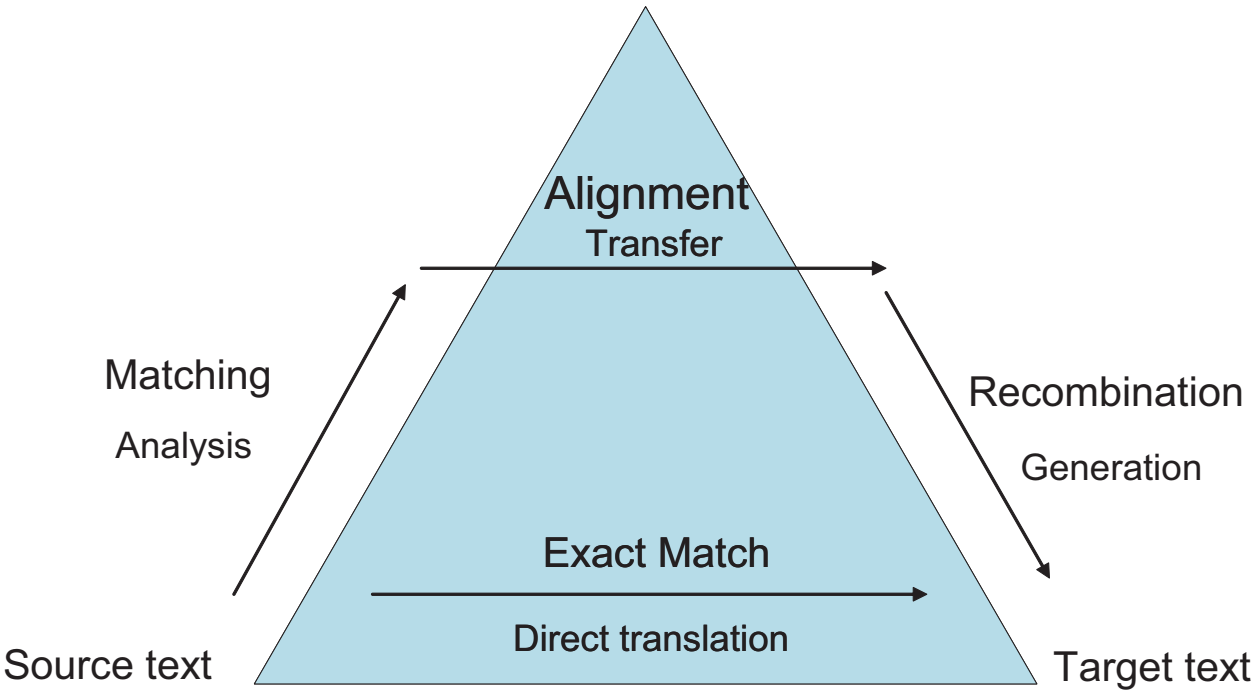

Figure 2.8: The "vauquois [Vauquois, 1968] pyramid" [Somers, 1999] adapted for EBMT.

Many example based translation systems refrain from using syntactic analysis. But syntactical knowledge is crucial especially for relatively free word-order languages like German, since the sequence of words has to be changed in most cases when translating to strict word order languages like English. Also complex sentences containing, e.g., relative clauses, are extremely difficult to handle without any syntactical knowledge" [Engel, 2000]

[Sumita et al., 1990] explains that "the performance of an EBMT system depends on the quality of collected examples and the similarity measure on examples and input sentences. When the matched unit are subsentential structures, the performance of such a system is better than that of a word-level system.

Computational Cost

EBMT directly returns a translation by adapting the examples without reasoning through a long chain of rules. The computational cost of EBMT is less than that of RBMT.

Improvement Cost

EBMT has no rules, thus improvement is effected simply by inputting appropriate examples to the database. In other words, EBMT is easily upgraded.

System Building Cost

EBMT are easy to obtain because a large number of texts and their translations are available. Moreover, as electronic publishing increases, more and more texts will be machine-readable.

Context-Sensitive Translation

EBMT is a general architecture, where incorporating contextual information into example representation provides a way to translate context-sensitively. As for our corpus, i.e., conversation about registering for an international conference, the set of words surrounding examples, the speaker of the examples, and soon, are ready to be used.

Robustness

EBMT works on best-match reasoning. EBMT intrinsically translates in a fail-safe way.

Reliability factor

In EBMT, a reliability factor is assigned to the translation result according to the distance between the input and similar examples found. EBMT can tell when its translation is inappropriate.

Example Independency

EBMT knowledge consists not of rules based on a particular system as in RBMT but rather the linguistic facts themselves. As suggested in [Nagao, 1984], this implies that the knowledge is completely independent of the system, so is usable in other systems and can be analyzed by any linguistic theory."

2.4.4 Statistical Machine Translation

According to [Och, 2000] "statistical MT has been introduced by the research group at IBM [Brown et al., 1990] in 1990. They introduced the concept of alignment models to describe the dependencies between source and target language words [Brown et al., 1993] and developed a search algorithm for these models based on the paradigm of stack decoding. Unfortunately, even for simple translation models, the search problem in statistical MT is NP complete. Various research groups tried to extend the IBM work to develop more efficient search algorithms by using suitable simplifications and applying better optimization methods. A major disadvantage of the baseline IBM alignment models is that they do not take word context into account. A partial solution to this problem, which works for frequent words was introduced by Brown [Brown et al., 1993] and continued by Garcia-Varea[Garcia-Varea et al., 2001] which suggested a maximum entropy based context-dependent lexicon model."

In other terms, a statistical translation model is a mathematical model in which the process of human language translation is statistically modeled [Yamada and Knight, 2001]. Model parameters are automatically estimated using a corpus of translation pairs. Statistical translation models have been used for statistical machine translation:

- word alignment of a translation corpus [Melamed, 2000];
- multilingual document retrieval [Franz et al., 1999];
- automatic dictionary construction [Resnik and Melamed, 1997];
- data preparation for word sense disambiguation programs [Brown et al., 1991];

A key component in statistical machine translation systems is the so called alignment model [url, c]. It structures the dependencies and re-orderings among words between a source language text and its translation.

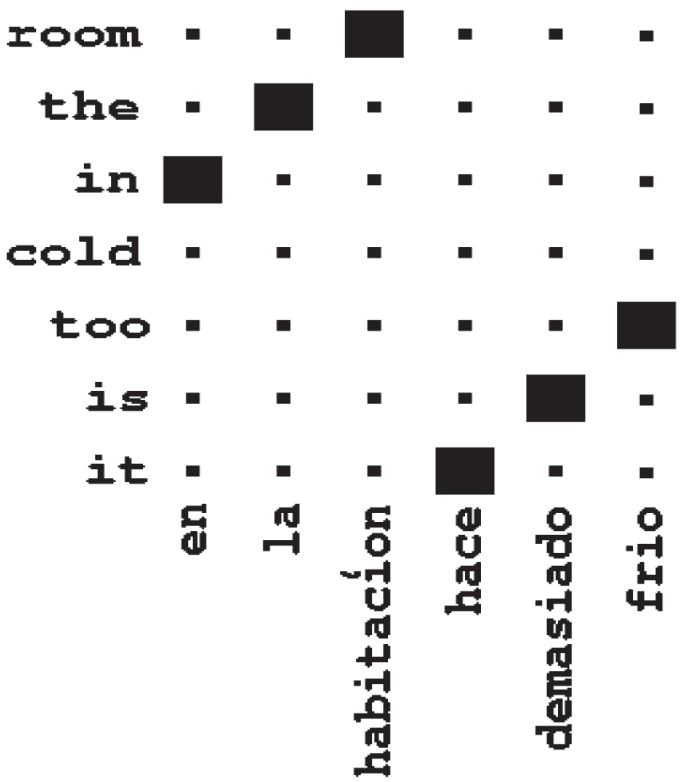

Figure 2.9: Alignment between a source and a target language sentence

Statistical Machine Translation with Alignment Templates

The alignment template system is a machine translation system which is an extension of the baseline single-word based translation models typically investigated in statistical machine translation. The key elements of this approach are the alignment templates which are pairs of phrases together with an alignment between the words within the phrases. The advantage of the alignment template approach over word based statistical translation models is that word context and local re-orderings are explicitly taken into account. This approach produces better translations than the single-word based models. The alignment templates are automatically trained using a parallel training corpus [url, c].

Researchers at IBM first described a statistical TM system.

Their models were based on a string-to-string noisy channel model. The channel converts a sequence of words in one language (such as English) into another (such as French). The channel operations are movements, duplications, and translations, applied to each word independently. The movement is conditioned only on word classes and positions in the string, and the duplication and translation are conditioned only on the word identity. One criticism of the IBM-style TM is that it does not model structural or syntactic aspects of the language. It was only demonstrated for a structurally similar language pair (English and French). It

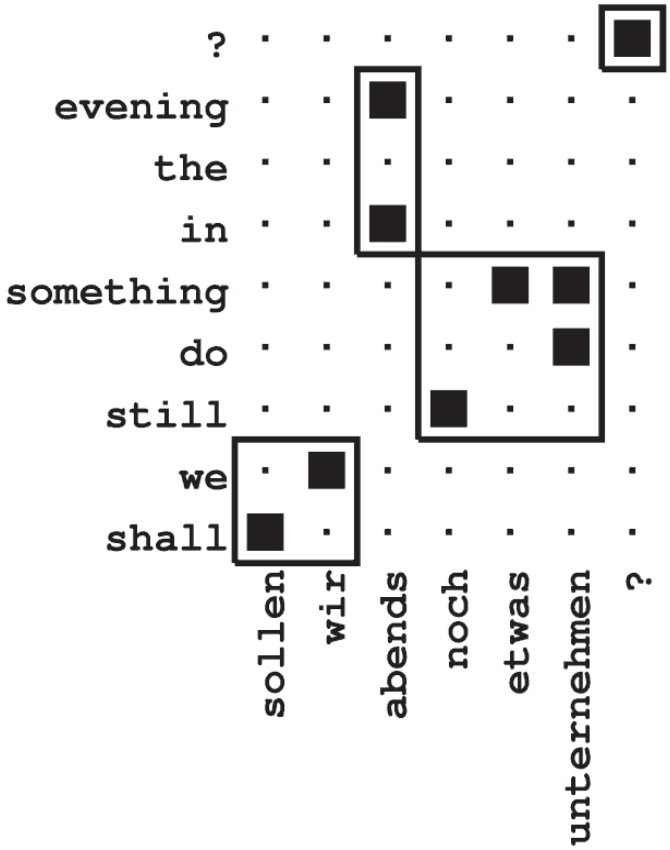

Figure 2.10: Example of Alignment Templates

has been suspected that a language pair with very different word order such as English and Japanese would not be modelled well by these systems.[Yamada and Knight, 2001]

[Su and Chang, 1992] "a statistical approach does not have a strict sense of well-formedness in mind. It does not assume any linguistic models either. Most of the time, the language generation process is simply modelled as a simple stochastic process, such as a Markov chain, and translation is regarded as a decoding process [Brown et al., 1990]. Almost no existing linguistic model is used to characterize the highly simplified stochastic process; and most of such models acquire the parameters directly from surface strings."

The statistical translation models were initially word-based , but significant advances were made with the introduction of phrase-based models. In [url, d] we can find the following description of the two models.

WORD-BASED Model

Most (if not all) of the statistical machine translation systems employ a word-based alignment model, which treats words in a sentence as independent entities and ignores the structural relations among them. Typically, the number of words in translated sentences is different due to compound words, morphology and idioms. The ratio of the lengths of sequences of translated words is called fertility, which tells how many foreign words each native word produces. Simple word-based translation is not able to translate language pairs with fertility

rates different from one. To make word-based translation systems manage, for instance, high fertility rates, the system could be able to map a single word to multiple words, but not vice versa [Wang, 1998].

An example of a word-based translation system is the freely available GIZA++ package, which includes IBM models.

Phrase-Based Model

In phrase-based translation, the restrictions produced by word-based translation have been tried to reduce by translating sequences of words to sequences of words, where the lengths can differ. The sequences of words are called, for instance, blocks or phrases, but typically are not linguistic phrases but phrases found using statistical methods from the corpus. Restricting the phrases to linguistic phrases has been shown to decrease translation quality. [url, d]

Development Cycle of Statistical MT Systems

According to [Och, 2000] "figure 2.11 presents the development cycle of a statistical MT system. A major difference to the development cycle of classical MT systems is that an evolutionary rapid prototyping approach can be pursued. An initial baseline with a reasonable quality can be bootstrapped very quickly if sufficient training data is available. Afterwards, an iterative improvement process starts.
The first step is the collection of training data, the need to obtain parallel texts, perform sentence alignment and extract the suited translation pairs.

In the second step, we perform an automatic training of the MT system. The output of this step is an operative MT system. Typically, this step is quite fast and needs no human supervision.

Afterwards, the MT system is tested and an error analysis is performed."

The evaluation criteria used in the statistical MT systems has to be cheap in its application. If the development and improvement cycle of statistical MT systems takes only a few hours or a few days, then a slow evaluation cycle would be the bottleneck for improving system quality. Hence, performing a time-consuming subjective evaluation of MT quality is not desirable.

Two error criteria can be distinguished:

- objective error criteria;
- subjective error criteria.

The objective error criteria compare the similarity of the produced translation with a set of reference translations.

On the other hand, the subjective criteria depend on a human quality judgment. Objective error criteria that can be used for the test phase are:

- WER (word error rate) [Niessen et al., 2000];

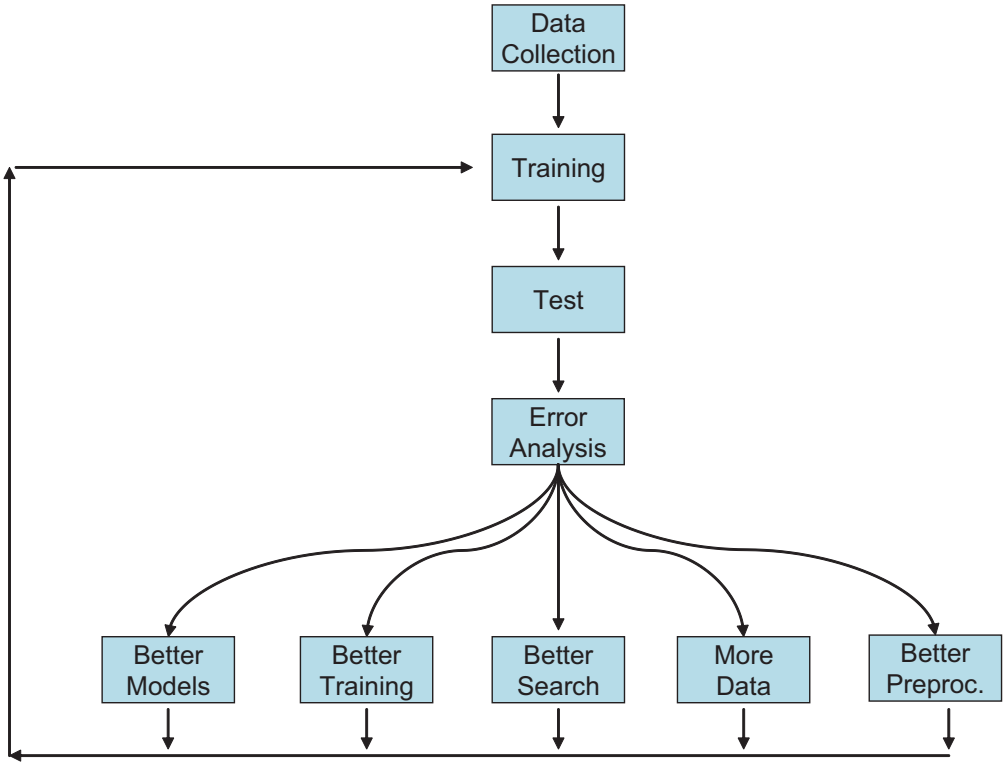

Figure 2.11: Development cycle of a statistical MT system

- BLEU score [Papineni et al., 2001].

Subjective error criteria that can be used for the test phase are:

- SSER (subjective sentence error rate)[Niessen et al., 2000].

Taking into account the architecture of a statistical MT system (Figure 2.3), different error types can be distinguished :

- search errors
- modelling errors
- training errors
- training corpus errors
- preprocessing errors

Depending on the result of this error analysis, various modifications are performed:

Better models: Here, the goal is to develop models, which better capture the properties of natural language and whose free parameters can be estimated reliably from training data.

Better training: training algorithms are often based on maximum likelihood, which is prone to overfitting. For certain model parameters, different parameter values have to be tested with respect to development corpus error rate so called parameter tuning. To do this efficiently, an automatic evaluation procedure is important.

Better search: A search error occurs if the search algorithm produces a translation, which is different from the optimizing translation $\hat{S}$ defined in Equation 2.2. The search problem in statistical MT is typically NP-complete. Therefore, suitable approximations in search need to be performed to obtain a good trade-off between translation quality and efficiency.

More training data: Typically, translation quality improves if the training corpus size increases. The learning curve of an MT system shows how much training data are needed to obtain a certain performance. An additional error source is wrong or too free translation in the training data. To avoid these errors, manual or automatic filtering or correction of these translation examples needs to be done.

Better preprocessing: Various natural language phenomena are notoriously difficult to handle for state-of-the-art statistical approaches. One method for dealing with this problem is to preprocess the text such that the text is better suited for the statistical translation models. Here, rule-based MT technology can be used. Typically, simple text transformations are performed that yield a normalized source and target language."[Och, 2000]

[url, e] states: "what is capturing the interest of statistical MT researchers?
The basic issues are:

- How can a complete calculation be approximated so that the statistics are reliable but at the same time the calculation is possible within the constraints of time and memory?

- How can each of the models be improved (in particular, what parameters can we use beside word form, word form sequences and word form alignments that might improve the estimates?

In the former case, the expectation is that larger corpora (such as the world wide web) will inevitably lead to improved results. In addition, there have been experiments with different alignment techniques which focus on "segments" of sentences (e.g. [Deng et al., 2004]) or bilingual parallel "segments" extracted from non parallel texts (e.g. [Munteanu et al., 2004]).

In the latter case, even such naive additions word stems, part-of-speech or morphological information (case, number, noun-adjective agreement, verb-nominal agreement, etc.) have sometimes lead to improved performance (but not necessarily!). Yamada and Knight [Yamada and Knight, 2001] for instance, describe a technique for developing syntactic transfer systems using aligned corpora in which at least one of the languages is syntactically annotated."

Advantages of the Statistical Approach for MT

"In the following, are summarized various arguments supporting a statistical approach to MT. All these arguments cannot prove a general superiority of the statistical approach over other approaches.

The relationships between linguistic objects such as words, phrases or grammatical structures are often weak and vague. To model those dependencies, a formalism is needed, such as offered by probability distributions, that is able to deal with these dependencies.

To perform MT, typically is needed a combination of many knowledge sources. In statistical MT, a mathematically well-founded machinery exist to perform an optimal combination of these knowledge sources.

In statistical MT, translation knowledge is learned automatically from example data. As a result, the development of an MT system based on statistical methods is very fast compared to the rule-based approach.

Statistical MT is well suited for embedded applications where MT is part of a larger application. For example, in speech translation there is an additional speech recognition engine, which introduces speech recognition errors. Statistical MT seems to be especially well suited for this application as it has a natural robustness. Another example is interactive MT.

The 'correct' representation of syntactic, semantic and pragmatic relationships is not known. In the statistical approach, the modelling assumptions are empirically verified on training data.

One of the major advantages of statistical MT is that it can be learned automatically. This is also the main reason why the process of developing a statistical MT system differs significantly from a classical rule-based system.

Statistical machine translation is a machine translation paradigm where translations are generated on the basis of statistical and information theoretic models whose parameters are derived from the analysis of bilingual text corpora."[Och, 2000]

"As the translation systems are not able to store all native strings and their translations, a document is typically translated sentence by sentence, but even this is not enough. Language models are typically approximated by smoothed n-gram models, and similar approaches have been applied to translation models, but there is additional complexity due to different sentence lengths and word orders in the languages" [url, d].

Disadvantages of the Statistical Approach

[Su and Chang, 1992] introduces the disadvantages of this kind of approach:

- "In general, a statistical model will have poor performance in unseen domain if a large training database is not available. The criteria of being large is measured with respect to the complexity of the underlying models or the size of the parameter space. As a rule of thumb, in order to get reasonable estimate of the statistical parameters, the number of training instances must be several times larger than the number of possible outcomes in the parameter space.

- Since a purely statistical approach does not use any high level linguistic knowledge in constructing a stochastic model, the parameter space is usually too large to be practical. For instance, to gather the word bigram statistics for the most frequently used 10,000 words, the number of possible bigram patterns is 10. The number of training word pairs must be several times larger than this amount. This is one reason why a purely statistical MT, like the one proposed in [Brown et al., 1990], requires a large training corpus even for a small vocabulary. Unless the corpus is large enough, the sparse data problem is inevitable; reliable statistics will not be available under such circumstances.

- Furthermore, to reduce the size of the parameter space, the stochastic model of a purely statistical approach is usually simplified. For instance, because a trigram model requires a training database that is about 10,000 times larger than a bigram model in the previous example, the latter is usually adopted at the expense of less contextual information. Hence, a purely statistical model usually cannot deal with long distance dependency. The purely statistical models also suffer from robustness problems due to sparse data when a complex model is used."

Statistical MT Systems

Statistical MT is exactly what the name indicates: it is a strategy used to make word, phrase, and sentence choices through a statistical analysis of the data that is studied. Statistical methods imply that there are numerous occurrences of each of the items likely to be chosen and that there is a high probability of finding threads of commonalities and similarities that can be used to detect the most probable items in order to make default choices [Allen, 2000].

According to [Wang, 1998] "statistical machine translation is based on a channel model. Given a sentence T in one language to be translated into another language, it considers T

to be the target of a communication channel, and its translation S to be the source of the channel. Basically every sentence S is a possible source for a T target sentence. A probability $P_r(S|T)$ is assigned to each pair of sentences (S, T). The problem of translation is to find the source S for a given target T, such that $P_r(S|T)$ is the maximum. According to the Bayes rule:

$$P_r(S|T) = \frac{P_r(S) \cdot P_r(T|S)}{P_r(T)} \tag{2.1}$$

Since the denominator is independent of S, the translation of T is therefore:

$$\hat{S} = \arg\max_{S} P_r(S) \cdot P_r(T|S) \tag{2.2}$$

There are three sub-problems in statistical machine translation:

- Modelling Problem: How can the process of generating a sentence in a source language be depicted, and what process is used by the channel to generate a target sentence upon receiving a source sentence? The former is the problem of language modelling, and the latter is the problem of translation modelling. They provide a framework to calculate $P_r(S)$ and $P_r(T|S)$

- Learning Problem: Given a statistical language model $P_r(S)$ and a statistical translation model $P_r(T|S)$, how can the parameters in these models be estimated from a bilingual corpus of parallel sentences? Moreover, is there a way to automatically learn the structure of the translation model?

- Decoding Problem: With a fully specified (framework and parameters) language and translation model, given a target sentence T, how can the source sentence S that satisfies Equation 2.2 be most efficiently identified.

Most statistical machine translation systems use n-gram for language modelling. Translation models rely on the concept of alignment. A word alignment is a mapping between the source words and the target words in a set of parallel sentences. Many alignment translation models assume that a target sentence is generated from a source sentence word by word [Wang, 1998]. In alignment translation, each target word can align with only one source sentence word. So far, most of the statistical machine translation systems use word-based alignment models and no structure is involved in the alignment. [Brown et al., 1993] introduced five different word-based alignment models for translation modelling.

Since the alignment between a paired source/target sentence is not marked in a parallel training corpus, the maximum likelihood (ML) estimator cannot be directly applied. EM algorithm is an effective ML estimator for statistical models with hidden variables, which can be applied to the translation models, where alignments are the hidden variables [Wang, 1998]. To estimate the model parameters the EM algorithm can be used. The Expectation Maximization is a general framework for estimating the parameters of a probability model when the data has missing values. This algorithm can be applied to minimally supervised learning, in which the missing values correspond to missing labels of the examples. The EM algorithm consists of the E-step in which the expected values of the missing sufficient-statistics given the observed data and the current parameter estimates are computed, and the M-step in which

the expected values of the sufficient-statistics computed in the E-step are used to compute complete data maximum likelihood estimates of the parameters [Tsuruoka and Tsujii, 2003].

The decoding algorithm is another crucial part in statistical machine translation. Its performance directly affects translation quality and efficiency. Without a reliable and efficient decoding algorithm, a statistical machine translation system may miss the best translation of an input sentence even if it is perfectly predicted by the model." [Wang, 1998]

SMT system is robust in processing partial and well-formed sentences. The computation time in SMT increase potentially with the length of the sentences. In additional, parameters strongly depend on the training corpus.

As a summary [Allen, 2000] is cited: "statistical MT is one possible approach, and that, when combined with other types of MT systems, it can provide improved results. However, the hindering fact is that most of the languages in the world today lack electronic data upon which to test and train systems. Thus, statistics-based methods as a primary approach should be reserved for the international languages that have sufficient data with which and from which to work. Only data collection and data compilation efforts with a significant amount of invested human resources could eventually allow the world's less-supported languages to also benefit more amply from the statistical MT approach."

2.4.5 Hybrid Systems

[Chen and Chen, 1996] state that "many different approaches to machine translation design have been proposed. The traditional rule-based machine translation system [Bennett and Slocum, 1985] is expensive in terms of formulating rules. It easily introduces inconsistencies, and it is too rigid to be robust.

In contrast, the statistics-based machine translation system [Brown, 1992] is based on noisy channel model and is robust in processing partial and ill-formed sentences. The computation time in processing long sentences sharply increases as the number of words increases.

The example-based system heavily depends on the quality of collected examples and the similarity measures between examples and input sentences. When the matched units are subsentential structures (e.g., phrase structures), the performance of such a system is better than that of a word-level system. As for the knowledge-based system the difficulties are in how the knowledge is represented, how fine the knowledge is, and what the inference engine is. In addition, the cost of compiling knowledge is expensive."

The hybrid design is a design effort whose aim is to produce a method that utilises the best features of both methods and that compensates for their weaknesses.

At AMTA 2004 and MT Summit 2005 just about all commercial MT developers also claimed to have hybrid systems.

A lot of current research in machine translation is neither based purely on linguistic knowledge nor on statistics, but includes some degree of hybridization. "Currently the research in this field is directed at the development of hybrid MT systems which integrate more than one approach to MT, the idea being that integration will help achieve properties that combine the advantages of the approaches involved." [Oliver Streiter, 2000]

According to [Oliver Streiter, 2000] "the hybridization of MT approaches attempted, so far, primarily concentrated on two aspects: the technical aspect and the improved translation quality. A prominent example is a parallel run of two or more different MT engines and the combination of their output.

	Advantages	Disadvantages
Rule-Based	effective for core phenomena based on linguistic theories easy to build an initial system	rules are formulated by experts difficult to maintain and extend ineffective for marginal phenomena
Example-Based	extracts knowledge from corpus based on translation patterns in corpus reduces the human cost	similarity measure is sensitive to system search cost is expensive knowledge acquisition is still problematic
Statistics-Based	numerical knowledge extracts knowledge from corpus reduces the human cost model is mathematically grounded	no linguistic background search cost is expensive hard to capture long distance phenomena

Table 2.1: Advantages and disadvantages of the various approaches [Chen and Chen, 1996].

These attempts include:

- integration of statistical information into RBMT systems using various techniques [Nomiyama, 1991], [Shinichi and Maraki, 1992], [Chen and Chen, 1995],[Manny and Bouillon, 1995], [Streiter, 2000], [Streiter and Iomdin, 2000]
- extraction of new translation units out of bilingual text and their compilation into RBMT systems [Streiter and Iomdin, 2000]
- combination of translation memories with RBMT [Heyn, 1996], [Michael, 2000]
- combination of EBMT with RBMT [Michael, 2000]
- combination of translation memory with EBMT [Michael and Hansen, 2000]

Lingstat is a hybrid MT system, combining statistical and linguistic techniques while METIS-II (10–2004 09–2007) [Dirix et al., 2005] is a hybrid machine translation system, in which insights from statistical, example-based, and rule-based machine translation (SMT, EBMT, and RBMT respectively) are used. METIS investigates rule-based and data-driven methods to the extent they can be built and used with relative ease and they complement each other. Rule-based methods are used where representations and decisions can be determined a-priori with high accuracy, for instance, based on linguistic insight. Corpora serve as a basis to ground decisions where uncertainty remains. Data-driven methods are used for target language generation, using only a target language corpus and a bilingual dictionary instead of a parallel corpus [2].

[2]http://www.ilsp.gr/metis2/

Chapter 3

Human Evaluation of MT

3.1 Human evaluation for EuroMatrix

EuroMatrix team annually organizes a challenge: different MT systems perform translations between pairs of some European languages. The results of the challenge are illustrated in a report. In this chapter the criteria used by organizers are analyzed to manually evaluate the output of each system. [Koehn and Monz, 2006] More than 100 volunteers are involved in manual evaluation. Manual evaluation was made using the following criteria:

Indicating two values, one for **fluency** and the other for **adequacy**;

Ranking translated sentences relative to each other;

Ranking translation of syntactic constituent drawn from the source sentence.

3.1.1 Fluency and adequacy

Commonly **fluency** refers to the degree to which the translation is well-formed according to the grammar of the target language. Hovy [Eduard Hovy and Popescu-Belis, 2002] sums some methods proposed to measure if: "Various ways of measuring fluency have been proposed, some focusing on specific syntactic constructions, such as relative clauses, number agreement, etc., others simply asking judges to rate each sentence as a whole on an n-point scale, and others automatically measuring the perplexity of a target text against a bigram or trigram language model derived from a set of ideal translations. The amount of agreement among such measures has never been studied." The EuroMatrix evaluators used the following five point scale:

5 *Flawless English*

4 *Good English*

3 *Non native English*

2 *Disfluent English*

1 *Incomprehensible*

This scale used for evaluating adequacy was developed for the annual NIST Machine Translation Evaluation Workshop by the Linguistics Data Consortium.

Adequacy is used to evaluate the quantity of the information existent in the original text that a translation contains. The scale for EM evaluator is the following:

5 *All*

4 *Most*

3 *Much*

2 *Little*

1 *None*

"Separate scales for fluency and adequacy were developed under the assumption that a translation might be disfluent but contain all the information from the source. However, in principle it seems that people have a hard time separating these two aspects of translation. The high correlation between peoples' fluency and adequacy scores indicate that the distinction might be false. Another problem with the scores is that there are no clear guidelines on how to assign values to translations. No instructions are given to evaluators in terms of how to quantify meaning, or how many grammatical errors (or what sort) separates the different levels of fluency. Because of this many judges either develop their own rules of thumb, or use the scales as relative rather than absolute. For judging we use bilingual raters, volunteers from the groups that participate to the competition. We ask them to exclude the system they produce and judge the others." [Koehn and Monz, 2006]

3.1.2 Ranking translations

Ranking the translations of a sentence has a double scientific goal: firstly, it is possible to use human ranking to assess how well the automatic evaluation works, and then it is useful to examine which metric for manual evaluation is the most representative of the accuracy and quality of a translation. Evaluators for this task have just to "*rank each whole sentence translation from Best to Worst relative to the other choices (ties are allowed)*": this is the only instruction given by EuroMatrix organizer.

3.1.3 Ranking translation of syntactic constituent

This is a new evaluation methodology: the source sentence is parsed to extract some constituents from the tree and people judge the translations of those syntactic phrases. For all the languages, aside from Czech, a parser is used: [**?**] for Spanish, [**?**] for French, [**?**] for German, and [**?**] for English. Once the phrase is selected, it is necessary to extract the corresponding portion from the reference translation and each of the system translations. As in [Koehn and Monz, 2006] "The word alignments were created with Giza++ [**?**], applied to a parallel corpus containing 200,000 sentence pairs of the training data, plus sets of 4,007 sentence pairs created by pairing the test sentences with the reference translations, and the test sentences paired with each of the system translations. The phrases in the translations were located using techniques from phrase-based statistical machine translation which extract phrase pairs from word alignments [**?**]; [**?**]. Because the word-alignments were created

automatically, and because the phrase extraction is heuristic, the phrases that were selected may not exactly correspond to the translations of the selected source phrase." There were three criteria adopted to select the constituents:

- a constituent could not match with the whole sentence;
- a constituent has to be longer than 3 words;
- a constituent has to have a corresponding phrase in the with-consistent word alignment in each translations to reduce the number of alignment errors.

The only instruction provided to the judge was: "*Rank each constituent translation from Best to Worst relative to the other choices (ties are allowed). Grade only the highlighted part of each translation.*
Please note that segments are selected automatically, and they should be taken as an approximate guide. They might include extra words that are not in the actual alignment, or miss words on either end." [Koehn and Monz, 2006]

Beyond the manual evaluation the project provided also an automatic evaluation, which is described in detail in the next chapter.

3.2 Other evaluation metrics

Many other evaluation metrics were used.

3.2.1 Reading time

Reading time is a measure close to fluency. The closer the Words Per Minute (WPM) rate is to the WPM of natural language, the higher is the quality of the translation.

There are two types of reading time measure:

- **Oral reading time** [Dijk, 1979]: for each document, the evaluators should read out loud the first paragraph and count the time it takes. The number of words is then used to calculate the words per minute (WPM) rate:

$$WPM = \frac{number - of - words}{reading - time}$$

- **Closed reading time**: as for oral reading time, the WPM needs to be calculated. This is done in the same way. The level of understanding of the readers also needs to be checked to see if it is *sufficient*. For this check, the reader was requested to answer some basic questions about the text.

3.2.2 Post-editing time

Post-editing time is based on the idea that the time required to change the translation in an acceptable text is inversely proportional to the translation quality. It is necessary to normalize the time by taking into account the size of the text measured in words and then

multiply by a fixed factor in order to obtain a number on a wider scale. The suggested calculation is:

$$correction\_time = \frac{number\_of\_minutes\_spent\_in\_correction}{totalnumber\_of\_words\_in\_text} * 10$$

High values for this measure mean bad translations.
There are negative aspects to consider if we use post-editing time measure. Some of which are:

- the ultimate use of the text influences the amount of correction (a text to be published will be treated with more care than a text for informal communication);
- the nature of errors are different and the time to correct them too;
- some correctors work faster than others.

3.2.3 Cloze test

Cloze test was first proposed as a method for MT evaluation by Crook & Bishop [Crook and Bishop, 1965]. This is a metric used often in psychological studies of reading. The evaluator sees an output sentence with a word replaced by a space (for example, every 8th word might be deleted). Evaluators have to guess the identity of the missing word. Accuracy at the cloze task, i.e. average success of evaluators at guessing the missing words, generally correlates with how intelligible or natural the MT output is. According to [Dabbadie et al., 2002] "two scores are normally computed, one based on the number of answers which comprise exactly the suppressed original word, the other based on the number of answers with a word close in meaning to the original word. The second score has to be interpreted partly in the light of the first score."

$$percentage\_of\_exact\_items\_supplied = \frac{number\_of\_exact\_answers}{number\_of\_deleted\_items} * 100$$

$$percentage\_of\_close\_items\_supplied = \frac{number\_of\_close\_answers}{number\_of\_deleted\_items} * 100$$

3.2.4 Clarity

Clarity score is used to judge the degree to which some discernible meaning is expressed in the sentences of a text: it is not important if the meaning of input text is preserved (fidelity) and neither if each sentence makes sense in the context or if it is well formed.

A 4-value scale is used to evaluate each sentence of a text:

3 Meaning of sentence is perfectly clear on first reading;

2 Meaning of sentence is clear only after some reflection;

1 Some, although not all, meaning is able to be gleaned from the sentence with some effort;

0 Meaning of sentence is not apparent, even after some reflection.

3.2.5 Informativeness

This is a comprehension task. There are two methods for testing. The most common is the reading comprehension exam (usually under 10 questions for the given text). The other is based on the idea that a good translation should mention people, organizations and places and preserve entity relationships in the source text.

These evaluation metrics (reading time, post-editing time, cloze test, clarity, informativeness) were added to fluency and adequacy for LREC 2002 conference ([Dabbadie et al., 2002]). On that occasion fluency had the 4-grade scale described below:

3 Very intelligible: all the content of the message is comprehensible, even if there are errors of style and/or of spelling, and certain words are missing, or are badly translated, but close to the target language;

2 Fairly intelligible: the major part of the message is passed along;

1 Barely intelligible: only a part of the content is understandable, representing less than 50% of the message;

0 Unintelligible: nothing or almost nothing of the message is comprehensible.

3.3 The aim of MT evaluation

An important discussion on MT evaluation concerns the aim of the evaluation: [Koehn and Monz, 2006] treats in depth this question identifying different aims and grouping them in two big categories: the overall aims, with respect to the recipients and to the user of the evaluation, and the specific aims, that include the effectiveness and usefulness of a translation and the capability of a system to improve its quality.

Table 3.3 is what Van Slype proposes to identify the appropriate criteria to evaluate a translation according to the aims and the group involved.

On the other hand in [Koehn and Monz, 2006] an MT system is expected to guarantee a sufficient level to enable a reader whose mother language is the target language to post-edit the translation without risk of disaster.

Groups involved	Aims of evaluation	Criteria
Final user of row MT	- effective transfer of information from one language to another	- fidelity - intelligibility
	- acceptability	- legibility - reading time
	- service conditions	- cost - product time
Post-editors correcting MT	- acceptability (allied to the scale and type of corrections)	- post-editing rate - post-editing time
Decision-makers (responsible for the development of an MT system)	- potential market	- acceptability - cost - improvability (in synthesis)
System technicians (data processing in specialist, linguists, coders)	- error diagnosis (by type elements of the specific MT system concerned)	- errors by causes
	- correctibility	- improvability (analytical)
Linguists	- errors diagnosis (by type elements of grammatical and stylistic)	- errors by linguistic type
Head of translation services	- number, perhaps classified by type	- post-edition rate
	- comparison of the features of the MT/post-edition circuit with those of human translation/revision circuit	- cost - production time

Table 3.1: Table suggested by Van Slype in [Dijk, 1979]

Chapter 4

Objective Evaluation of MT

We saw that manual evaluation has different advantages compared to automatic evaluation. Nonetheless automatic evaluation has some peculiarities we cannot ignore. One of the most important peculiarities is that a good evaluation should provide the same evaluation values for two perfectly equal texts (even if they are the output of different translation systems): yet two human evaluators that have to judge the same text could give two different evaluations, as might the same evaluator at different moments. Also, an algorithm used for evaluating texts is reusable for every text, while on the contrary manual evaluation techniques require considerable time and people. Miller and Beebe-Center [Miller and Beebe-Center, 1958] suggested that a good automatic translation is very similar to a human translation for the same text. Since it is possible to translate the same text in different ways we can compare system translation with more than one human translation and so consider a set of perfect translations.

"In order to be both effective and useful, an automatic metric for MT evaluation has to satisfy several basic criteria. The primary and most intuitive requirement is that the metric have very high correlation with quantified human notions of MT quality. Furthermore, a good metric should be as sensitive as possible to differences in MT quality between different systems, and between different versions of the same system. The metric should be consistent (same MT system on similar texts should produce similar scores), reliable (MT systems that score similarly can be trusted to perform similarly) and general (applicable to different MT tasks in a wide range of domains and scenarios)" [Banerjee and Lavie, 2005]

We could summarize the advantages of automatic evaluation in this way:

- fast and cheap;
- no need for bilingual speakers;
- minimal human labour;
- can be used on an on-going basis during system development to test changes.

Following is a list of the disadvantages of automatic MT:

- metrics are very crude;
- do not distinguish well between subtle differences in systems;

- individual sentence scores are not very reliable, aggregate scores on a large test set are required;

According to [Cole et al., 1997] we can distinguish three interdependent levels of specificity to evaluate a system:

- Criterion: what is the main target of the evaluation? Speed, error rate, precision, recall, fluency, accuracy, etc.
- Measure: which specific property of system performance should be reported to get the chosen criterion? For example, for speed report *processes per second*, for precision report *the number of retrieved relevant data to all retrieved data*, for recall report *the number of retrieved relevant data to all relevant data*, etc.
- Method: how to determine the appropriate value for a given measure and a given system. For example, post analytic measurement of a system's behaviour over some benchmark task.

4.1 Criteria for automatic MT evaluation

[Eric H. Nyberg, 1994] identifies 3 major criteria:

- **Completeness**: a system is complete if for each input string there is a corresponding one in the output. Completeness could be of three types: *lexical* (complete if the system has source and target language lexicon entries for every word or phrase in the translation domain), *grammatical* (complete if the system can analyze of the grammatical structures in the source language and generate all of the grammatical structures necessary in the target language translation) and *mapping role* (complete if the system assigns an output structure to every input structure in the translation domain).
- **Correctness**: a system is correct if it assigns a correct output string to every input string it is given to translate. There are three types of correctness: *lexical* (words in the target sentence should be correctly chosen for the sense in input), *syntactic* (there should be no grammatical errors) and *semantic* (compositional meaning of target sentence should be equivalent to the meaning of the source sentence).
- **Stylistics**: system evaluation may go beyond correctness and test additional, interrelated stylistic factors, such as *syntactic style* (the output sentence should not contain a correct grammatical structure which nonetheless inappropriate for the context), *lexical appropriateness* (words chosen should be appropriate for the context), *usage appropriateness* (the most conventional or natural expression should be chosen), *formality*, *level of difficulty of the text*, etc.

4.2 Measures for automatic MT evaluation

[Taylor and White, 1998] suggests four steps for the development of measure in MT:

- identifying the text-handling tasks that users perform with translated material as input;

- discovering the order of text-handling task tolerance, i.e., how good a translation must be in order for it to be useful for a particular task;
- analyzing the translation problems (both linguistic and non-linguistic) in the corpus used in determining task tolerance;
- developing a set of source language patterns which correspond to diagnostic target phenomena.

4.3 Methods for automatic evaluation in EuroMatrix

The most common algorithm for evaluating machine translation is Bleu, used for ACL workshop. A study by Koehn and Monz ([Koehn and Monz, 2006]) shows that Bleu systematically underestimates the quality of rule-based MT systems. That is why the automatic evaluation of the EuroMatrix challenge used eleven different measures:

- BLEU
- METEOR
- General Text Matcher
- Translate Error Rate
- ParaEval precision and ParaEval recall
- Dependency overlap
- Semantic role overlap
- Word Error Rate over verbs
- Maximum correlation training on adequacy and on fluency

4.3.1 BLEU

BLEU was developed by Kishore Papineni and others in 2001 [Papineni et al., 2001].
The basic idea of BLEU is that the closer the machine translation is to a professional human translation, the better it is. The motivation for Papineni's work has come from the industry need of quick MT results in order to take out the bad ideas and leave the good ones, so BLEU was presented as an alternative for human evaluation that can be used when quick and frequent evaluations are required.
IBM described the BLEU metric in July 2001 TIDES PI meeting in Philadelphia. It is based on the average of matching n-grams between a proposed translation and one or more reference translations, and it seems to correspond well with human judgments on accuracy and fluency.

BLEU is an automatic evaluation technique which is a geometric mean of n-gram matching. To compute the BLEU score, one has to count the number of n-grams in the test translation that have a match in the corresponding reference translations. The formula used to calculate the n-gram precision is simple. The words from a candidate translation that

match with a word in the reference translation (human translation) are counted, and then divided by the number of words in the candidate translation.

To check how close a candidate translation is to a reference translation, a n-gram comparison is done between both translations. However, because the evaluation is based on n-gram comparison with reference sentences, it is possible to make sentences with completely different meaning by switching words/n-grams and still get high scores [url, f].

Multiple reference translations are also used to increase accuracy where paraphrases may exist. Having this number divided by the total number of n-grams in the test translation, one can get the n-gram precision. BLEU uses a modified n-gram precision, called pn. This precision clips the count for each n-gram in any test translation to prevent it from exceeding the count of this n-gram in the best matching reference translation. Because BLEU is precision based, and because recall is difficult to formulate over multiple reference translations, a brevity penalty is introduced to compensate for the possibility of proposing high precision hypothesis translations which are too short. IBM's formula for calculating BLEU score is as follows [Finch et al., 2005]:

$$BLEU = BP \times exp\ (\sum_{n=1}^{4} \frac{1}{n}\ \log(pn))$$

where brevity penalty is calculated as:

$$BP = \min(1, e^{1-r/c})$$

where c is the length of the corpus of hypothesis translations, and r is the effective reference (is calculated as the sum of the single reference translation from each set which is closest to the hypothesis translation) corpus length.

The n-gram precision is calculated as:

$$p_n = \frac{\sum_{i=1}^{I} \sum_{ngram \in s_i}\ count(ngram)}{\sum_{i=1}^{I} \sum_{ngram \in s_i}\ count_{sys}(ngram)}$$

$count(ngram)$ is the count of n-grams found both in s_i and r_i.
$count_{sys}(ngram)$ is the count of n-grams found only in s_i.
Example [Zwarts, 2005]

Translation
How are you gentlemen !! All your base are belong to us.
Reference
You seem to be preoccupied, gentlemen. With the kind cooperation of the Federation forces, all of your bases now belong to us.

1-grams:
(you) $(gentlemen)$ || (all) $(your)$ $(belong)$ (to) (us) $(.) = \frac{2+6}{5+8} = \frac{8}{13}$
2-grams:
|| $(belong, to)$ (to, us) $(us, .) = \frac{0+3}{4+7} = \frac{3}{11}$
3-grams:
|| $(belong, to, us)$ $(to, us, .) = \frac{0+2}{2+6} = \frac{2}{9}$

$$BP = e^{1-r/c} = e^{1-26/13}) = 0.368$$

$$BLEU = BP \times exp\,(\frac{1}{3}log(8/13)\ + \frac{1}{3}log(3/11)\ + \frac{1}{3}log(2/9)) = 0.229$$

BLEU is insensitive to syntactic changes.

A stumbling block for BLEU is the length of the candidate translation. If the difference in length between the two translations is large, then it is doubtful that the candidate translation is in fact an accurate translation. Turian [Turian et al., 2004] tested this theory, and found that BLEU was in fact consistently poor in evaluating shorter documents [Ahmed, 2006].

BLEU is known to perform poorly (i.e. not agree with human judgments of translation quality) when evaluating the output of commercial systems like Systran against N-gram based statistical systems, or even when evaluating human-aided translation against machine translation [C.Callison-Burch et al., 2006].

BLEU is the best known and best adopted Machine Evaluation for (machine) translation. However it has the weakness that judgment is based not on the fact whether an algorithm captures and translates the meaning, but on how well it scores against references.

4.3.2 METEOR

As in [Russo-Lassner et al., 2005]"Meteor [Banerjee and Lavie, 2005] is a machine translation evaluation metric developed at Carnegie Mellon University. It is based on a word-to-word alignment between the machine-generated translation and the reference translation. This metric assigns a score equal to the harmonic mean of unigram precision (that is, the proportion of matched ngrams out of the total number of n-grams in the evaluated translation) and unigram recall (that is, the proportion of matched n-grams out of the total number of n-grams in the reference translation). Meteor also includes a fragmentation penalty that accounts for how well-ordered the matched unigrams of the machine translation are with respect to the reference. The alignment between machine translation and reference translation is obtained through mapping modules that apply sequentially, linking unigrams that have not been mapped by the preceding modules:

- the exact module maps words that are exactly the same;
- the porter-stem module links words that share the same stem;
- the WordNet synonymy module maps unigrams that are synonyms of each other."

[Banerjee and Lavie, 2005] explains that METEOR was designed to explicitly address the weaknesses in BLEU. The main principle behind IBMs BLEU metric [Papineni et al., 2001] are:

- the measurement of the overlap in unigrams (single words)
- higher order n-grams of words, between a translation being evaluated
- a set of one or more reference translations

If more than one reference translation is available, the given translation is scored against each reference independently, and the best score is reported.

Given a pair of translations to be compared (a system translation and a reference translation), METEOR creates an alignment between the two strings. An alignment is a mapping between unigrams, such that every unigram in each string maps to zero or one unigram in the other string, and to no unigrams in the same string. Thus in a given alignment, a single unigram in one string cannot map to more than one unigram in the other string. This alignment is incrementally produced through a series of stages, each stage consisting of two distinct phases. In the first phase an external module lists all the possible unigram mappings between the two strings. In the second phase of each stage, the largest subset of these unigram mappings is selected such that the resulting set constitutes an alignment as defined above (that is, each unigram must map to at most one unigram in the other string). If more than one subset constitutes an alignment, and also has the same cardinality as the largest set, METEOR selects that set that has the least number of unigram mapping crosse.

METEOR score is computed as follows:

First unigram precision (P) is computed as the ratio of the number of unigrams in the system translation that are mapped to the total number of unigrams in the system translation.

Similarly, unigram recall (R) is computed as the ratio of the number of unigrams in the system translation that are mapped to the total number of unigrams in the reference translation.

Next Fmean is computed by combining the precision and recall via a harmonic-mean that places most of the weight on recall. A harmonic mean of P and 9R is used. To take into account longer matches, METEOR computes a penalty as follows [Banerjee and Lavie, 2005]:

$$METEOR = Fmean * (1 - Penalty)$$

$$Penalty = 0,5 * \frac{\sharp chunks}{\sharp unigrams - matched}$$

$$Fmean = \frac{10PR}{R + 9P}$$

Meteor is a Precision and Recall Metric.

We can find Meteor as a automatic evaluation metric in the IWSLT 2005-2006 Evaluation Campaign.

4.3.3 General Text Matcher (GTM)

GTM [Melamed et al., 2003] is based on accuracy measures as precision, recall and F-measure. GTM measures the similarity between texts.

"GTM allows the calculation of standard precision and recall scores for automatically produced translations. It also calculates an f-measures score, which combines both the precision and recall score for a given translation."[**?**]

GTM was used in the IWSLT 2004-2005 Evaluation Campaign.

4.3.4 Word Error Rate over verbs

Niessen in 2000 [Niessen et al., 2000] proposed a performance evaluation method with which system performance can be evaluated automatically and quickly: Word error rate (WER).

Word error rate is expressed as the minimum edit distance between hypothesis and reference at word level.

The Levenshtein distance proposed by Vladimir Levenshtein in 1965 [Levenshtein, 1965] between two strings is given by the minimum number of operations needed to transform one string into the other, where an operation is an insertion, deletion, or substitution of a single character.

The edit distance has the great advantage of being automatically computable, and as a consequence, the results are inexpensive to get and reproducible, because the underlying data and the algorithm are always the same. This approach involves no human intervention, so the evaluation cost is fairly low.

The great disadvantage of the WER metric is the fact that it depends fundamentally on the choice of the sample translation.

"The Word Error Rate (WER) is computed as the sum of insertions, substitutions and deletions, normalized by the length of the reference sentence. A WER of 0 means the translation is identical to the reference. One problem with WER is that this "rate" is in fact not guaranteed to be between 0 and 1 and in some settings a wrong translation may yield a WER higher than 1" [Goutte, 2006].

A differently normalized WER, denoted WERg , normalizes the sum of insertions, substitutions and deletions by the length of the Levenshtein alignment path, insertions, substitutions, deletions and matches. The advantage of this metric is that it is guaranteed to lie between 0 and 1, where 1 is the worst case (no matches) [Goutte, 2006].

$$WER(s_i, r_i) = \frac{I(s_i, r_i) + D(s_i, r_i) + S(s_i, r_i)}{|r_i|}$$

where $I(s_i, r_i)$,$D(s_i, r_i)$,$S(s_i, r_i)$ are the number of insertions, deletions and substitutions respectively [Finch et al., 2005].

Word Error Rate over verbs An extension of WER measure proposes to calculate the WER measure over POS classes in order to estimate the inflectional errors and the distribution of missing word over POS classes. In [Popovic and Ney, 2007] the measure is so explaned: "The dynamic programming algorithm for WER enables a simple and straightforward identification of each erroneous word which actually contributes to WER. Let *errk* denote the set of erroneous words in sentence k with respect to the best reference and p be a POS class. Then *n(p, err k)* is the number of errors in err k produced by words with POS class p. It should be noted that for the substitution errors, the POS class of the involved reference word is taken into account. POS tags of the reference words are also used for the deletion errors, and for the insertion errors the POS class of the hypothesis word is taken. The WER for the word class p can be calculated as:

$$WER(p) = \frac{1}{N^*_{ref}} \sum_{k=1}^{K} n(p, err_k)$$

The sum over all classes is equal to the standard overall WER."

The example shown in that article is:

Reference: *Mister#N Commissioner#N ,#PUN twenty-four#NUM hours#N sometimes#ADV can#V be#V too#ADV much#PRON time#N .#PUN*

Hypothesis: *Mrs#N Commissioner#N ,#PUN twenty-four#NUM hours#N is#V sometimes#ADV too#ADV much#PRON time#N .#PUN*

reference errors	hypothesis errors	error type
Mister#N	Mrs#N	substitution
sometimes#ADV	is#V	substitution
can#V		deletion
be#V	sometimes#ADV	substitution

Table 4.1: Distribution on errors in the example above

4.3.5 Translate Error Rate (TER)

Translate Error Rate, proposed by Snover and Dorr in [Snover and Dorr, 2006], is a more intuitive measure to evaluate MT. This measure represents the number of edits needed to change a hypothesis in one of the references, normalized on the length of the references.

$$TER = \frac{number\_of\_edits}{average\_of\_reference\_words}$$

Possible edits include the insertion, deletion, substitution of single words and shifts of word sequence.
A good example of this measure is the following one form [Snover and Dorr, 2006]:
"Consider the reference/hypothesis pair below, where differences between the reference and hypothesis are indicated by upper case:

- `REF: SAUDI ARABIA denied THIS WEEK information published in the AMERICAN new york times`
- `HYP: THIS WEEK THE SAUDIS denied information published in the new york times`

Here, the hypothesis (HYP) is fluent and means the same thing (except for missing "American") as the reference (REF). However, TER does not consider this an exact match. First, we note that the phrase "this week" in the hypothesis is in a shifted position (at the beginning of the sentence rather than after the word denied) with respect to the hypothesis. Second, we note that the phrase "Saudi Arabia" in the reference appears as "the Saudis" in the hypothesis (this counts as two separate substitutions). Finally, the word "American" appears only in the reference.

If we apply TER to this hypothesis and reference, the number of edits is 4 (1 Shift, 2 Substitutions, and 1 Insertion), giving a TER score of 4 13 = 31%. BLEU also yields a poor score of 32.3% (or 67.7% when viewed as the error-rate analog to the TER score) on the hypothesis because it doesnt account for phrasal shifts adequately. Clearly these scores do not reflect the acceptability of the hypothesis, but it would take human knowledge to determine that the hypothesis semantically matches the reference."

TER is different from Word Error Rate (WER) because it treats shifts of contiguous multi-word sequences as a single operation.

4.3.6 ParaEval precision and ParaEval recall

[Zhou et al., 2006] proposes to use paraphrases to improve the quality of machine translation evaluation. A first step is to acquire a large collection of paraphrases. The main assumption is that two sentences that have the same meaning could have the same translation in a foreign language. Once acquired a two-tier matching strategy for MT evaluation is adopted. "At the top tier, a paraphrase match is performed on system-translated sentences and corresponding reference sentences. Then, unigram matching is performed on the words not matched by paraphrases. Precision is measured as the ratio of the total number of words matched to the total number of words in the peer translation."

4.3.7 Dependency overlap

[Amigo et al., 2006] suggest dependency trees to evaluate the quality of an automatic translation. In [Koehn and Monz, 2006] dependency overlap is described in the following way: "This metric uses dependency trees for the hypothesis and reference translations, by computing the average overlap between words in the two trees which are dominated by grammatical relationships of the same type".

4.3.8 Semantic role overlap

[Gimenez and Marquez, 2007] underline that "Most metrics used in the context of Automatic Machine Translation (MT) Evaluation are based on the assumption that *acceptable* translations tend to share the lexicon (i.e., word forms) in a predefined set of manual reference translations. This assumption works well in many cases." In that paper they suggest other metrics: one of them is the Semantic Role (SR) Overlap. This metric analyses the similarity between hypothesis and reference translations by comparing the SRs that occur in the text. The types of SR are:

- Arguments associated with a verb predicate, defined in the PropBank Frames scheme.
- Causative agent
- Adverbial (general-purpose) adjunct
- Causal adjunct
- Directional adjunct
- Discourse marker
- Extent adjunct
- Locative adjunct
- Manner adjunct
- Modal adjunct
- Negation marker

- Purpose and reason adjunct
- Predication adjunct
- Reciprocal adjunct
- Temporal adjunct

4.3.9 Maximum correlation training on adequacy and on fluency

A description of this metric is in [Liu and Gildea, 2007]. "Maximum Correlation Training (MCT) is an instance of the general approach of directly optimizing the objective function by which a model will ultimately be evaluated. In our case, the model is the linear combination of the component metrics, the parameters are the weights for each component metric, and the objective function is the Pearsons correlation of the combined metric and the human judgments. The reason to use the linear combination of the metrics is that the component metrics are usually of the same or similar order of magnitude, and it makes the optimization problem easy to solve. Using w to denote the weights, and m to denote the component metrics, the combined metric x is computed as:

$$x(w) = \sum_j w_j m_j$$

Using hi and $x(w)_i$ to denote the human judgment and combined metric for a sentence respectively, and N to denote the number of sentences in the evaluation set, the objective function is then computed as:

$$Pearson(X(w), H) = \frac{\sum_{i=1}^{N} x(w)_i h_i - \frac{\sum_{i=1}^{N} x(w)_i \sum_{i=1}^{N} h_i}{N}}{\sqrt{(\sum_{i=1}^{N} x(w)_i^2 - \frac{(\sum_{i=1}^{N} x(w)_i)^2}{N})(\sum_{i=1}^{N} h_i^2 - \frac{(\sum_{i=1}^{N} h_i)^2}{N})}}$$

Now our task is to find the weights for each component metric so that the correlation of the combined metric with the human judgment is maximized. It can be formulated as:

$$w = argmax_w Pearson(X(w), H)$$

The function Pearson(X(w),H) is differentiable with respect to the vector w. The derivative can be computed analytically and gradient ascent performed. The objective function is not always convex (one can easily create a non-convex function by setting the human judgments and individual metrics to some particular value). Thus there is no guarantee that, starting from a random w, the globally optimal w using optimization techniques such as gradient ascent will be obtained. The easiest way to avoid ending up with a bad local optimum to run gradient ascent is to start from different random points. In the experiments performed, the difference in each run is very small, i.e., by starting from different random initial values of w, one ends up with, not the same, but very similar values for Pearsons correlation.

4.4 Other methods for MT evaluation

4.4.1 Simple String Accuracy and Generation String Accuracy

Two general methods to calculate accuracy are:

- Simple string accuracy (SSA)
- Generation string accuracy (GSA)

$$SSA = 1 - \frac{I + D + S}{|R|}$$

where I is the number of insertions, D the number of deletions, S the number of substitutions and R the number of tokens in the string. This metric, which has already been used to measure the quality of Machine Translation systems [Alshawi et al., 1998], penalizes twice words which are misplaced, because it counts this error as one deletion and one insertion. As a consequence, the number of insertions and deletions can be larger than the actual number of tokens. In this case, the result of the metric may be negative. To avoid this, another variable (M) which counts the number of misplaced tokens is added to treat the misplaced words separately in the new formula.

$$GSA = 1 - \frac{M + I + D + S}{|R|}$$

4.4.2 Multiple reference word error rate (MWER)

Niessen [Niessen et al., 2000] in 2000 proposed an identical approach to WER, but this new one considers several references for each sentence to be translated. The idea of computing the difference to more than one reference has the advantage that the set of reference sentences comes for free as the database is enlarged [Alshawi et al., 1998]. Besides, the new reference sentences produced by the translation systems under consideration are more adequate for the purpose of word-by-word comparison, because human translators tend to translate more or less freely, frequently resorting to synonyms and sentence restructuring [Tomast et al., 2003].

mWER is computed as follows: a translation $\boldsymbol{t}$ is compared to each reference out of a set of references of the test sentence $\boldsymbol{s}$ and the edit distance of $\boldsymbol{t}$ and the most similar reference is used for the computation of the mWER. Let $\mathbf{R(s)}$ be the set of reference translations for $\boldsymbol{s}$ and $\mathbf{d(t,\ r)}$ the edit distance between a translation candidate $\boldsymbol{t}$ and a reference $\mathbf{r} \in \mathbf{R(s)}$. $\mathbf{d(t,R(s))}$ is the minimal edit distance of $\boldsymbol{t}$ compared to any reference of $\boldsymbol{s}$:

The mWER of a set of translations $t_1^n = t_1....t_n$ for a test corpus $s_1^n = s_1...s_n$ can then be defined as follows [Niessen, 2002]:

$$mWER(s_1^n, t_1^n) = \frac{\sum_{i=1}^{n} d(t_i, \mathcal{R}(s_i))}{\sum_{i=1}^{n} \frac{1}{|R(s_i)|} \cdot \sum_{r \in \mathcal{R}(s_i)} |r|}$$

where $|r|$ is the length of the reference $\boldsymbol{r}$ and $|R(si)|$ is the number of references for the *i-th* test sentence s_i.

Applications of mWER can be found in the IWSLT 2004-2005 and CESTA Evaluation Campaign.

4.4.3 Inversion word error rate (invWER)

The Inversion Word Error Rate is a new automatic evaluation measure proposed by Leusch, Ueffing, and Hermann Ney in 2003 [Leusch et al., 2003] as an extension of the WER that

takes block reordering into account. WER does not admit any changes in order, PER does not put any constraints on reordering, and invWER takes an intermediate position between WER and PER by allowing recursive block inversions.

The classical Levenshtein distance has been extended by block transpositions in order to allow for moves of symbol blocks at constant cost.

The inversion edit distance between a source sentence s_I^1 and a target sentence t_J^1 to be the minimum cost of the set $T(s_I^1,\ t_J^1)$ of all parse trees generated by the bracketing transduction grammar BTG [Wu, 1995] for this sentence pair [Leusch et al., 2003]:

$$d_{inv}(s_I^1,\ t_J^1) = \min_{\tau \in\ T(s_I^1,\ t_J^1)} c(\tau)$$

4.4.4 All references word error rate (aWER)

"aWer measures the number of words, which are to be inserted, deleted or replaced in the sentence under evaluation in order to obtain a correct translation. It can also be seen as a particular case of the mWER, but taking for granted that all the possible references are at our disposal. Since it is impossible to have a priori all possible references, the evaluator will be able to propose new references, if needed. The evaluation process can be carried out very quickly, if one takes as the starting point the result obtained by the WER or the mWER. The idea consists of visualizing the incorrect words detected by one of these methods (editing operations). The evaluator just needs to indicate whether each of the marked items is an actual error or whether it can rather be considered as an alternative translation" [Tomast et al., 2003].

4.4.5 Position independent word error rate

Leusch in 2003 [Leusch et al., 2003] proposed a related measure called position-independent word error rate (PER).

The Position-independent Error Rate does not take into account the ordering of words in the matching operation. In fact it considers the translations and the reference as bag-of-words and computes the differences between them, normalized by the reference length. "Depending on whether the translated sentence is longer or shorter than the target translation, the remaining words result in either insertion or deletion errors in addition to substitution errors. The PER is guaranteed to be less than or equal to the WER" [Finch et al., 2005].

$$PER(s_i, r_i) = \frac{\max[diff(s_i, r_i), diff(r_i, s_i)]}{|r_i|}$$

where $diff(si, ri)$ is the number of words observed only in si.

Applications of PER can be found in the IWSLT 2004-2005 and CESTA Evaluation Campaigns.

4.4.6 Dice coefficient

The Dice coefficient [Jianmin et al., 2002] demonstrates the intuition that good translations tend to have more common words with references than bad ones. This is especially true for

0 ≡ nonsense 1 ≡ some aspects of the content are conveyed 5 ≡ understandable with major syntactic errors. 9 ≡. Only slight errors in register or style or minimal syntax errors. K = 10 ≡ perfect

Table 4.2: Definition of scores for human evaluation

example based machine translation for localization purpose. Dice is a position independent word error rate.

The Dice coefficient of element sets of strings s1 and s2:

$$Dice(s1, s2) = 2 \times \frac{|s1 \cap s2|}{|s1| + |s2|}$$

4.4.7 Sentence error rate (SER)

"SER indicates the percentage of sentences, whose translations have not matched in an exact manner those of reference. It shows similar advantages and shortcomings as WER" [Tomast et al., 2003].

4.4.8 Subjective Sentence Error Rate (SSER)

The approach for this technique [Niessen et al., 2000] is to turn the evaluation task into a classification task. The evaluation technique depends on the training data provided. The translations are scored by classification into a small number of quality classes, ranging from perfect to absolutely wrong. "In comparison to the WER, this criterion is more reliable and conveys more information, but to measure the SSER is expensive, as it is not computed automatically but is the result of laborious evaluation by human experts. Besides, the results depend highly on the persons performing, and the comparability of results is not guaranteed.

Another disadvantage is the fact that the length of the sentences is not taken into account: The score of the translation of a long sentence has the same impact on the overall result as the score of the translation of a one-word sentence."[**?**]

Each sentence is scored from 0 to 10, according to its translation quality [Niessen et al., 2000].

An example of these categories is shown in table 4.2:

The subjective sentence error rate (SSER) of a set of translations t_1^n for a test corpus s_1^n can then be defined as follows [Niessen, 2002]:

$$SSER(s_1^n, t_1^n) = 1 - \frac{1}{K_n} \cdot \sum_{i=1}^{n} v(s_i, t_i)$$

The evaluation scheme is defined such that each translation $\boldsymbol{t}$ for an input sentence $\boldsymbol{s}$ is assigned a score $v(s,t)$ ranging from 0 points (nonsense) to K points (perfect).

4.4.9 CDER metric

Lopresti and Tomkins in 1997 [Lopresti and Tomkins, 1997] showed that finding an optimal path in a long-jump alignment grid is an NP-hard problem.

Leusch, Ueffing and Ney [Leusch et al., 2006] in 2006 experiments showed that the calculation of exact long jump distances becomes impractical for sentences longer than 20 words. A possible way to achieve polynomial runtime is restricting the number of admissible block permutations. This has been implemented by Leusch in 2003 [Leusch et al., 2003] in the inversion word error rate (see section **??**).

CDER can be seen as a measure oriented towards recall, while measures like BLEU are guided by precision. The CDER is based on the CDCD distance introduced in [Lopresti and Tomkins, 1997] The authors demostrate that the problem of finding the optimal solution can be solved in $O(I*L)$ time, where I is the length of the candidate sentence and L the length of the reference sentence.[**?**]

CDER shows better correlation with human assessment than BLEU and WER on both corpora.

Another interesting topic in MT evaluation research is the question of whether a linear combination of two MT evaluation measures can improve the correlation between automatic and human evaluation.

Particularly, in their experiments, was expected the combination of CDER and PER to have a significantly higher correlation with human evaluation than the measures alone. The two measures were combined through linear interpolation. The result was consistent across all different data collections and language pairs: a linear combination of about 60% CDER and 40% PER has demonstrated a significantly higher correlation with human evaluation than each of the measures alone.

4.4.10 X-Score metric

Rajman and Hartley in 2001 introduced the new metric: X-Score. The X-Score metric [Hartley and Rajman, 2001] is based on the distribution of elementary linguistic information within a text, such as morphosyntactic categories, or syntactic relationships. The authors' hypothesis is that this distribution of linguistic information is similar from one text to another within a given language. Depending on the nature of the linguistic information selected, the metrics precision will vary. For instance, working with syntactic dependencies will be much more precise than working with morphosyntactic categories only. Whichever type of information is selected, the X-Score measures the grammatical correctness of a text, comparing the distribution of the selected linguistic information within this text to a representative measure of the same information distribution within the whole language.

This metric remains experimental and it can be expected to be highly dependent on many parameters. In particular, it depends on the nature of the selected linguistic information, on the tool used to extract this information, and on the training corpus, to cite only the main factors. In the CESTA 2004-2005-2006 Evaluation Campaign this metric has been investigated for different types of linguistic information, with different tools.

4.4.11 D-Score metric

In 2001 Rajman and Hartley [Hartley and Rajman, 2001] introduced another new metric: D-Score.

The D-Score measures the preservation of a texts semantic content throughout the translation process. First the authors create semantic vector space models, of both the source language and of the target language. Then the position of any given source document is computed within the source language vector space, and the position of its translation is also computed within the target language vector space. Finally, the distance between these two positions is used to compute the D-Score measure. It is a highly experimental metric, subject to high variations due to a number of parameters. In particular, it is highly dependent on the method used to reduce the representation space of the terms, the usage of tools such as stemmers or lemmatizers to normalize the terms, and the training corpus used to build the source and target language models. Applications of D-SCORE can be found in the CESTA 2004-2005-2006 Evaluation Campaign.

4.4.12 Weighted N-gram Model

The Weighted N-gram Model, or WNM was proposed by Babych and Hartley in 2004 [Babych and Hartley, 2004]. Their proposal was to extend BLEU and the computation of proximity scores (i.e. the distance measure between the evaluated translation and the references) by introducing weights coming from the statistical relevance of the words inside the text.

This extension gives additional information about evaluated texts; in particular it allows to measure translation Adequacy, which, for statistical MT systems, is often overestimated by the baseline BLEU method. In this case the model uses a single human reference translation, which increases the usability of the proposed method for practical purposes.
The model suggests a linguistic interpretation which relates frequency weights and human intuition about translation Adequacy and Fluency. Using weighted N-grams is essential for predicting adequacy, since correlation of Recall for non-weighted N-grams is much lower.

A preliminary experiment [Babych and Hartley, 2004] proved that WNM results for recall are well correlated (even better than BLEU) to human judgments about adequacy.
WNM was used in the Cesta 2004-2005 Evaluation Campaign.

NIST Machine Translation Evaluation In the NIST Machine Translation Evaluation exercise that has been running annually for the last five years - as part of DARPAs TIDES program, the quality of Chinese-to-English and Arabic-to-English translation systems has been evaluated both by using BLEU score and by conducting a manual evaluation. The evaluation exercise of 2005 was startling in that the BLEUs rankings of the Arabic-English translation systems failed to fully correspond to the manual evaluations.

A number of prominent factors contribute to BLEUs crudeness
[C.Callison-Burch et al., 2006]:

- Synonyms and paraphrases are only handled if they are in the set of multiple reference translations.
- The scores for words are equally weighted so missing out on content-bearing material brings no additional penalty.

- The brevity penalty is a stop-gap measure to compensate for the fairly serious problem of not being able to calculate recall.

Each of these failures contributes to an increased amount of inappropriately indistinguishable translations in the analysis presented above. Given that BLEU can theoretically assign equal scoring to translations of obvious different quality, it is logical that a higher BLEU score may not necessarily be indicative of a genuine improvement in translation quality.

BLEU may not correlate with human judgment to the degree that it is currently believed to do. It is not necessary to receive a higher Blue score in order to be judged to have better translation quality by human subjects.

The BLEU score has been shown to correlate well with human judgment, when statistical machine translation systems are compared [Doddington, 2002], [Przybocki, 2004].

In the paper of Deborah Coughlin [Coughlin, 2001], there is a more in depth analysis of the correlation of BLEU and human judgment.

For example, the standard evaluation size (250 sentences) is a limiting factor in determining BLEUs ability to correlate with human evaluations. When a larger dataset has been used, the correlation coefficient with human assessments, was extremely high.

Another proof was that of grouping sentences by their unigram scores. BLEU gives a score of 0.0 to several files at the lower end. This inability to distinguish between very low scoring corpora suggests a weakness in the BLEU metric. Because BLEU uses the geometric mean of n-gram scores at its foundation.

BLEU does a poor job of scoring files whose sentences are poorly (or too freely) translated and share no trigrams or 4- grams with the reference sentences.

When comparing two systems run on the same test sentences, BLEU reliably agree with human relative assessments, correlating rather strongly.

What conclusions can be drawn from this? Should one give up on using BLEU entirely?

The advantages of BLEU are still very strong; automatic evaluation metrics are inexpensive, and do allow many tasks to be performed that would otherwise be impossible. The important thing therefore is to recognize which uses of BLEU are appropriate and which uses are not.

Appropriate uses for BLEU include tracking broad, incremental changes to a single system, comparing systems which employ similar translation strategies (such as comparing phrase-based statistical machine translation systems with other phrase-based statistical machine translation systems), and using BLEU as an objective function to optimize the values of parameters such as feature weights in log linear translation models, until a better metric has been proposed.

Inappropriate uses for BLEU include comparing systems which employ radically different strategies (especially comparing phrase-based statistical machine translation systems against systems that do not employ similar n-gram-based approaches), trying to detect improvements for aspects of translation that are not modelled well by BLEU, and monitoring improvements that occur infrequently within a test corpus [C.Callison-Burch et al., 2006].

4.4.13 NIST

The National Institute for Standards and Technology (NIST) score is another evaluation tool for machine translation systems. NIST scores MT systems in a manner very similar to BLEU, in that it uses n-gram co-occurrence statistics to obtain an accuracy score. In fact, NIST was commissioned based on BLEU, as BLEU was illustrated by IBM to have a strong correlation between automatically generated scores and human judgments of translation quality.

The formula for information weight is [Finch et al., 2005]:

$$NIST = \sum_{n=1}^{N} BP \times \frac{\sum_{\substack{all\ ngram \\ that\ co-occur}} info(ngram)}{\sum_{ngram \in s_i} 1}$$

where info(ngram) is:

$$info(ngram) = \log_2 \frac{count((n-1)ngram)}{count(ngram)}$$

It computes the arithmetic mean of the n-gram precisions, also with a length penalty. A significant difference with BLEU is that n-gram precisions are weighted by the n-gram frequencies, to put more emphasis on the less frequent (and more informative) n-grams.

NIST is based on the same ideas as the BLEU metric of IBM, and it can be seen as an upgrade to this metric. The difference between these two methods is that the NIST score takes the "information gain" from each n-gram into account. The idea behind "information gain" is that NIST believes that n-grams which are more informative should carry more weight than those which are less informative. If a very rare n-gram occurs simultaneously in both a source and translated text, then NIST gives this n-gram a higher score than a very common n-gram. The information weightings used to decide these scores are calculated using an equation, and the final value is also calculated using a different equation [Ahmed, 2006].

NIST is also an n-gram counting metric, but the aim is to fix two of the problems in the BLEU metric we previously mentioned:

Firstly, BLEU uses a geometric mean of n-grams. The weights for the different pn are chosen to be uniform: $w_n = 1/N$

According to NIST this can lead to counterproductive variances due to low co-occurrences for the larger values of N.

Secondly, BLEU treats all n-grams equally. This means that n-grams which occur often and have little information (for example the bi-gram ["in" "the"]) have as much impact on overall precision as information rich n-grams (for example the bi-gram ["counter" "productive"] has much more information than the previous bi-gram). The value on how information rich an n-gram is, is based on how often it occurs [url, f].

Using NIST scores in evaluations, corpus size must be kept constant because NIST scores increase logarithmically with corpus size [Coughlin, 2001].

Summary

One potential problem with n-gram methods is that a low n-gram score is not necessarily indicative of a poor translation, although a high n-gram score (where the definition of 'high' depends on the number of reference translations and other factors) is probably indicative of a good translation. A second point is that it is critical to control the type of translation represented by the reference translations.

The human translations that scored poorly are generally freer translations.

One reason why some of the human translations get relatively low scores is that the human translators often did not translate particularly faithfully, and made changes even when there is no clear necessity to do so.

The pool of translations favours the stricter translations over the freer one. In order to make the best use of an n-gram metric, the reference translations should be roughly the same style as the translation to be judged. This implies that only fairly strict translations can be judged by this metric for the time being.

A third potential problem is that with only two reference translations, it is harder to distinguish good and not so good translations. With four reference translations, the contrast between good and not so good translations is more clearer. More reference translations make the evaluation scores more discriminative [Culy and Riehemann, 2003].

Results confirm that a greater number of reference translations does give rise to higher scores, across the board.

It can be concluded that automatic metrics are most appropriate when evaluating incremental changes to a single system, or comparing systems with very similar architectures.

4.4.14 RED

RED is an automatic ranking method based on edit distances to multiple reference translations, proposed by Akiba in 2001 [Akiba et al., 2001]. RED consists of:

- a learning phase
- an evaluation phase

In the learning phase, in order to estimate a rank from edit distances, RED learns a Decision Tree for the ranking (hereafter, ranker) from ranking examples by a Decision Tree learner [Quinlan, 1993].

In the evaluation phase, RED assigns a rank to each MT output by using the ranker. Each ranking example is encoded by using multiple edit distances and a median rank among the ranks assigned by three or more human evaluators. On the other hand, each translation to be ranked is encoded by using only multiple edit distances before being assigned a rank.

Each edit distance is measured by one of sixteen variations of the basic edit distance measure, ED1, with three edit operators: insertion, deletion and replacement.

	RED	BLEU
Evaluation unit	An utterance	A segment
Evaluation target	An utterance	A document
Evaluation results	Ranking	Scoring
Learning strategy	Supervised	Not learning
Approach	Edit distances	N-gram matching
Robustness to replacing or swapping words	Relatively weak	Strong
Long-distance co-occurrence	Strong	Weak

Table 4.3: The different features of automatic evaluators: RED and BLEU

The different features of automatic evaluators: RED and BLEU Some experiments have been done [Akiba et al., 2001] to determine how close the evaluation results by each automatic evaluator RED and BLEU are to the average evaluation results by human evaluators, following the ATR standard of MT evaluation. The main lessons learnt from the experiments are:

- The evaluation results by RED are close to the average evaluation results by humans. The ratio at which RED agrees with the average evaluation by humans is in the range of 90%, even when different types of MT systems are compared.
- The ratio of how much BLEU agrees with the average evaluation by humans reaches 100%, but only when the same type of MT systems are compared by using some sets of reference translations, because BLEU is very sensitive to the choice of reference translations.

4.4.15 F-measure

The F-Measure has been proposed as a more comprehensible alternative for MT evaluation [Melamed et al., 2003], and can be defined as a simple composite of unigram precision and recall. It is a metric developed at New York University in 2003. It is designed to eliminate the double counting done by n-gram based metrics such as the NIST and BLEU (which penalize the same word insertion, deletion or movement as it occurs in a unigram, a bigram, etc.). It is designed to eliminate the double counting done by ngram based metrics such as the NIST and BLEU n-gram based metrics (which penalize the same word insertion, deletion or movement as it occurs in a unigram, a bigram, etc.). It uses two scores, precision and recall, computed separately for each candidate sentence. Both precision and recall are defined in terms of the maximum match size, which is the weighted sum of the lengths of the longest matching text blocks between candidate and reference sentences. Precision is the maximum match size divided by the length of the candidate sentence; recall is the maximum match size divided by the length of the reference sentence. The maximum match size can be adjusted to weight longer matches more or less heavily by using a different exponent. This metric punishes variation in sentence length less than BLEU and NIST, so it would be more closely correlated with human judgments for variation generation [Stent et al., 2005].

$$Precision(candidate|reference) = \frac{|reference \cap candidate|}{|candidate|}$$

$$Recall(candidate|reference) = \frac{|reference \cap candidate|}{|reference|}$$

$$F - Measure = \frac{2 \times P \times R}{P + R}$$

The final F-measure is the harmonic mean of both the precision and the recall.

Turian in [Melamed et al., 2003] claims to have higher correlation with F-Measure than either BLEU or Nist has. It is good not to have an arbitrary (maximum) length of the n-grams, because this metric automatically rewards larger chunks of text which are identical to (one of the) references texts [url, f].

4.4.16 Information item error rate (IER)

How to evaluate long sentences consisting of correct and wrong parts?

IER attempts to find a solution to this question.

The test sentences are segmented into information items. For each of them, the human examiner decides if the candidate translation includes this information item. The translation of this information item is judged as correct, if the intended information is conveyed and there are no syntactic errors [Niessen et al., 2000].

Each item of the sentence is marked with OK, error, syntax, meaning or others.

The metric IER (Information Item Error Rate) [Tomast et al., 2003] can be calculated as the percentage of badly translated items (not marked as OK).

Each input sentence in the database is partitioned into segments representing the relevant items of information to be conveyed. Let $II(s)$ be the set of information items for $\boldsymbol{s}$. The information item error rate (IER) is the rate of information items not evaluated as "OK" for a set of translations t_1^n [Niessen, 2002]:

$$IER(s_1^n, t_1^n) = \frac{\sum_{i=1}^{n} |\{ii|ii \in II(s_i), ii \neq \text{“}OK''\}|}{\sum_{i=1}^{n} |II(s_i)|}$$

4.4.17 HTER, Human-targeted translation error rate

HTER [Snover and Dorr, 2006] is an approach that employs human annotation to make TER a more accurate measure of translation quality. HTER involves a procedure for creating targeted references to find the closest possible reference to the hypothesis from the space of all possible fluent references that have the same meaning as the original references [Snover et al., 2005].

ROUGE

Recall-Oriented Understudy for Gisting Evaluation [Lin, 2003] [1] was proposed in 2003, and is a very recent adaption of the IBM BLEU for Machine Translation that uses unigram co-occurrences between summary pairs.

[1] http://www.isi.edu/ cyl/ROUGE/

ROUGE is recall oriented, in contrast to the precision oriented BLEU script, and separately evaluates 1, 2, 3, and 4-grams. Also, ROUGE does not apply any length penalty (brevity penalty), which is natural since text summarization involves compression of text and thus rather should reward shorter extract segment as long as they score well for content. ROUGE has been verified for extraction based summaries with a focus on content overlap.

It includes measures to automatically determine the quality of a summary by comparing it to other summaries created by humans. The measures count the number of overlapping units such as n-gram, word sequences, and word pairs between the computer-generated summary to be evaluated and the ideal summaries created by humans. There are four different types of measures ROUGE-N, ROUGE-L, ROUGE-W and ROUGE-S. The measures count the number of overlapping n-grams, word-sequences and word pairs. Rouge-N is calculated as:

$$ROUGE_n = \frac{\sum_{c \in \{ModelUnits\}} \sum_{n-gram \in C} Count_{match}(n-gram)}{\sum_{c \in \{ModelUnits\}} \sum_{n-gram \in C} Count\ (n-gram)}$$

N stands for the number of n-grams and count the maximum number of n-grams co-occurring in a candidate summary and a set of human summaries.

Chapter 5

Conclusions

This survey is a general presentation of the techniques used in the field of MT evaluation, and of some of the major problems related to it. The initial section deals with an analysis of how MT systems work, and discusses three different approaches to the problem: direct approach, transfer approach and interlingua approach.

Historically, the direct approach was the first developed and it is the simplest methodology for translation. Using the direct approach, the work necessary for the translation process is limited to the bare minimum, such as a word-by-word translation from one language to another. An example of a system based on this type of approach is the Georgetown Automatic Translation (GAT) project, which was initiated in 1952 but became fully operational only in 1964.

A more complex methodology is the transfer approach. Using this approach the translation process is divided into three steps:

1. analysis (an abstract representation of the input sentence is produced);

2. transfer (the abstract representation is transferred in the corresponding representation in the target language);

3. generation (from the representation to a sentence in the target language). TAUM (a project of 1965) is an example of system based on transfer approach.

TAUM (a 1965 project) is an example of a system based on the transfer approach.

To use an interlingua approach means to translate the source text into an artificial language designed to capture the various types of meanings of the source, and then to transfer it from the interlingua into the target language.

MT systems can be rule-based or follow the empirical approach. In the first instance experts use a set of rules to describe the translation process. The fact that rules are individually written by linguistic specialists makes this a very expensive approach.

The empirical approach analyzes example translations to develop an MT system. This implies that, with sufficient data, MT systems for new language pairs or domains are quickly developed.

In the development of a system, evaluation remains an essential activity. Both manual and automatic evaluation has some advantages and disadvantages: human evaluation is expensive and not always effective, but there is no obvious bias. On the other hand automatic evaluation is biased but is repeatable and always gives the same judgment whenever the same input are given.

The most common metrics used in manual evaluation are fluency and adequacy. High values of fluency show that the text is well-formed, according to the grammar of the target language. On the other side adequacy is used to evaluate the quantity of information a translation contains that existed in the original text. Other evaluation metrics are: reading time, post-editing time, cloze test, clarity, informativeness.

Several automatic metrics are proposed to evaluate MT. The idea is to compare the hypothesis with the reference translation. The closer they are, the better the hypothesis is. The most commonly used evaluation system is BLUE. BLUE calculates the number of n-grams in the hypothesis and in the reference translation, then it applies a brevity penalty if the hypothesis is too short compared to the reference. Future work could focus on making human evaluation more effective or automatic evaluation less biased.

Automatic evaluation could be improved by exploiting linguistic knowledge about semantic equivalence (e.g. equivalent constructions) or giving different weights to different kinds of words (the meaning of a sentence is more influenced if a "not" is missed rather than an article).

An interesting topic is the relation between human judgments and automatic metrics: in [Koehn and Monz, 2006], EuroMatrix data were used to check how well automatic evaluation metrics correlate with human judgments, and human assessment was considered the authoritative standard. In order to be able to make a better assessment of the automatic metrics for evaluation, this topic needs to be further investigated, and the causes of the differences between need to be studied further.

Bibliography

[url, a] http://ai-depot.com/Tutorial/RuleBased.html.

[url, b] http://ai-depot.com/Tutorial/RuleBased-Methods.html.

[url, c] http://www-i6.informatik.rwth-aachen.de/web/Research/MTResearch_frame.html.

[url, d] http://en.wikipedia.org/wiki/Statistical_machine_translation.

[url, e] http://www.fti.uab.es/tradumatica/revista/num4/articles/06/06art.htm.

[url, f] http://www.ics.mq.edu.au/~szwarts/MT-Evaluation.php.

[Ahmed, 2006] Ahmed, N. (2006). Evaluation of machine translation systems. Master's thesis, University of Sheffield. Available at http://www.dcs.shef.ac.uk/intranet/teaching/projects/archive/msc2006/abs/acp05na.htm.

[Akiba et al., 2001] Akiba, Y., Imamura, K., and Sumita, E. (2001). Using multiple edit distances to automatically rank machine translation output. In *Proc. MT Summit VIII*.

[Allen, 2000] Allen, J. (2000). What about statistical-based machine translation? *International Journal for Language and Documentation*.

[Alshawi et al., 1998] Alshawi, H., Bangalore, S., and Douglas, S. (1998). Automatic acquisition of hierarchical transduction models for machine translation. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics, Montreal, Canada*.

[Amigo et al., 2006] Amigo, E., Gimenez, J., Gonzalo, J., and Marquez, L. (2006). MT evaluation: Human-like vs. human acceptable. In *Proceedings of COLING-ACL06*.

[Babych and Hartley, 2004] Babych, B. and Hartley, A. (2004). Extending bleu mt evaluation method with frequency weighting. In *Proceedings of ACL*.

[Banerjee and Lavie, 2005] Banerjee, S. and Lavie, A. (2005). METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL 2005 Workshop on Intrinsic and Extrinsic Evaulation Measures for MT and/or Summarization.*

[Bennett and Slocum, 1985] Bennett, W. and Slocum, J. (1985). The LRC machine translation system. *Computational Linguistics*, pages 111–119.

[Boitet, 1982] Boitet, C. (1982). Implementation and conversational environment of ariane. In *Proceedings of COLING-82*.

[Brown, 1992] Brown, P. (1992). Analysis, statistical transfer, and synthesis in machine translation. In *Proceedings of TMI-92,*, pages 83–100.

[Brown et al., 1991] Brown, P., Cocke, J., Pietra, S. D., Jelinek, F., Mercer, R., and Roossin, P. (1991). Word-sense disambiguation using statistical methods. In *ACL-91*.

[Brown et al., 1990] Brown, P. F., Cocke, J., Pietra, S. D., Pietra, V. J. D., Jelinek, F., Lafferty, J. D., Mercer, R. L., and Roossin, P. S. (1990). A statistical approach to machine translation. *Computational Linguistics*, 16(2):79–85.

[Brown et al., 1993] Brown, P. F., Pietra, S. D., Pietra, V. J. D., and Mercer, R. L. (1993). The mathematic of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

[C.Callison-Burch et al., 2006] C.Callison-Burch, M.Osborne, and P.Koehn (2006). Reevaluating the role of BLEU in machine translation research. In *Proceedings of EACL-06*.

[Chen and Chen, 1996] Chen, K. and Chen, H. (1996). A hybrid approach to machine translation system design. *Computational Linguistics and Chinese Language Processing*.

[Chen and Chen, 1995] Chen, K. H. and Chen, H. H. (1995). Machine translation: An integrated approach. In *Proceedings of the 6th International Conference on Theoretical and Methodological Issues in Machine Translation*.

[Chunyu Kit and Webster, 1992] Chunyu Kit, H. P. and Webster, J. J. (1992). Example-based machine translation: A new paradigm. Available at `http://personal.cityu.edu.hk/~ctckit/papers/EBMT-review-CUHK.pdf`.

[Cole et al., 1997] Cole, R., Mariani, J., Zaenen, A., and Zue, V. (1997). Survey of the state of the art in human language technology.

[Coughlin, 2001] Coughlin, D. (2001). Correlating automated and human assessments of machine translation quality. In *Proceedings of MT Summit IX*.

[Crook and Bishop, 1965] Crook, M. N. and Bishop, H. P. (1965). Evaluation of machine translation. *Final Report*.

[Culy and Riehemann, 2003] Culy, C. and Riehemann, S. (2003). The limits of n-gram translation evaluation metrics. *Machine Translation Summit IX*.

[Dabbadie et al., 2002] Dabbadie, M., Hartley, A., King, M., Keith J. Miller, W. M. E. H., Popescu-Belis, A., Reeder, F., and Vanni, M. (2002). A hands-on study of the reliability and coherence of evaluation metrics. *lrec 2002*.

[Deng et al., 2004] Deng, Y., Kumar, S., and Byrne, W. (2004). Bitext chunk alignment for statistical machine translation. Technical report.

[Dijk, 1979] Dijk, B. M. V. (1979). Critical study of methods for evaluating the quality of machine translation.

[Dirix et al., 2005] Dirix, P., Schuurman, I., and Vandeghinste, V. (2005). METIS-II: Example-based machine translation using monolingual corpora - system description. In *Proceedings of MT Summit X*.

[Doddington, 2002] Doddington, G. (2002). The NIST automated measure and its relation to IBMs BLEU. In *Proceedings of LREC-2002 Workshopon Machine Translation Evaluation: Human Evaluators Meet Automated Metrics*.

[Eduard Hovy and Popescu-Belis, 2002] Eduard Hovy, M. k. and Popescu-Belis, A. (2002). Principles of context-based machine translation evaluation.

[Engel, 2000] Engel, R. (2000). Chunky: an example based machine translation system for spoken dialogs. Available at `http://www.dfki.de/~rengel/papers/icslp2000.pdf`.

[Eric H. Nyberg, 1994] Eric H. Nyberg, Teruko Mitamura, J. G. C. (1994). Evaluation metrics for knowledge-based machine translation. Available at `http://www.lti.cs.cmu.edu/Research/Kant/PDF/evaluate.pdf`.

[Finch et al., 2005] Finch, A., Hwang, Y.-S., and Sumita, E. (2005). Using machine translation evaluation techniques to determine sentence-level semantic equivalence. *IWP2005*.

[Forgy, 1982] Forgy, C. L. (1982). Rete : A fast algoritm for the many pattern / many object pattern match problems. *Artificial Intelligence*, 19.

[Franz et al., 1999] Franz, M., McCarley, J., and Ward, R. (1999). Models of translational equivalence among words. In *TREC-8*.

[Garcia-Varea et al., 2001] Garcia-Varea, I., Och, F. J., Ney, H., and Casacuberta, F. (2001). Refined lexikon models for statistical machine translation using a maximum entropy approach. In *ACL*, pages 204–211.

[Gimenez and Marquez, 2007] Gimenez, J. and Marquez, L. (2007). Linguistic features for automatic evaluation of heterogenous mt systems. In *Proceedings of ACL Workshop on Statistical Machine Translation*.

[Goutte, 2006] Goutte, C. (2006). Automatic evaluation of machine translation quality.

[Hartley and Rajman, 2001] Hartley, A. and Rajman, M. (2001). Automatically predicting MT systems rankings compatible with fluency, adequacy or informativeness scores. In *Proceedings of the 4th ISLE Workshop on MT Evaluation, MT Summit VIII*.

[Heyn, 1996] Heyn, M. (1996). Integrating machine translation into translation memory systems. In *European Association for Machine Translation - Workshop Proceedings*.

[Hiroshi, 1993] Hiroshi, U. (1993). Interlingua for multilingual machine translation. In *MT Summit IV*.

[Hirschman and Thompson, 1996] Hirschman, L. and Thompson, H. S. (1996). Overview of evaluation in speech and natural language processing. *Survey of the State of the Art in Human Language Technology*.

[Hovy, 2002] Hovy, E. (2002). Principles of context-based machine translation evaluation. *Machine Translation*.

[Hovy et al., 2002] Hovy, E., King, M., and Popescu-Belis, A. (2002). An introduction to machine translation. In *Workshop at the LREC 2002 Conference*.

[Hutchins, 1978] Hutchins, J. (1978). Machine translation and machine-aided translation. *Journal of Documentation*, 34:119–159.

[Hutchins, 2005] Hutchins, J. (2005). Towards a definition of example-based machine translation. Available at `http://www.hutchinsweb.me.uk/MTS-2005.pdf`.

[J.Dorr et al., 1998] J.Dorr, B., W.Jordan, P., and W.Benoit, J. (1998). A survey on current paradigm of mt. *Tecnical Report*.

[Jianmin et al., 2002] Jianmin, Y., Jing, Z., Tiejun, Z., and Sheng, L. (2002). An automatic evaluation method for localization oriented lexicalised ebmt system. In *Proceedings of the 19th international conference on Computational linguistics*.

[Jordan, 1991] Jordan, P. W. (1991). A first-pass approach for evaluating machine translation systems. In *Proceedings of the Evaluators Forum*.

[Kaji, 1988] Kaji, H. (1988). An efficient execution method for rule-based machine translation. In *Proceedings of the 12th conference on Computational linguistics*.

[Koehn and Monz, 2006] Koehn, P. and Monz, C. (2006). Manual and automatic evaluation of machine translation between european languages. In *Proceedings of the Workshop on Statistical Machine Translation,*.

[Leusch et al., 2003] Leusch, G., Ueffing, N., and Ney, H. (2003). A novel string-to-string distance measure with applications to machine translation evaluation. In *Proceedings of MT Summit IX*.

[Leusch et al., 2006] Leusch, G., Ueffing, N., and Ney, H. (2006). CDER: Efficient MT evaluation using block movements. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics*.

[Levenshtein, 1965] Levenshtein, V. (1965). Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics - Doklady 10*.

[Lin, 2003] Lin, C. Y. (2003). ROUGE: Recall-oriented understudy for gisting evaluation. `http://berouge.com/`.

[Liu and Gildea, 2007] Liu, D. and Gildea, D. (2007). Source-language features and maximum correlation training for machine translation evaluation. In *Proceedings of NAACL*.

[Lopresti and Tomkins, 1997] Lopresti, D. and Tomkins, A. (1997). Block edit models for approximate string matching. *Theoretical Computer Science*.

[Manny and Bouillon, 1995] Manny, R. and Bouillon, P. (1995). Hybrid transfer in an english-french spoken language translator. In *Proceedings of IA-95*.

[Melamed, 2000] Melamed, I. (2000). Models of translational equivalence among words. In *Computational Linguistics*.

[Melamed et al., 2003] Melamed, I., Green, R., and Turian, J. (2003). Precision and recall of machine translation. In *Proceedings of the HLTNAACL 2003*.

[Michael, 2000] Michael, C. (2000). Towards a model of competence for corpus-based machine translation. In *IAI Web-based Working Paper*.

[Michael and Hansen, 2000] Michael, C. and Hansen, S. (2000). Linking translation memories with example-based machine translation. In *Hybrid Approaches to MT*. IAI Web-based Working Paper.

[Miller and Beebe-Center, 1958] Miller, G. A. and Beebe-Center, J. G. (1958). Some psychological methods for evaluating the quality of translations. *Mechanical Translation*.

[Munteanu et al., 2004] Munteanu, D., Fraser, A., and Marcu, D. (2004). Improved machine translation performance via parallel sentence extraction from comparable corpora. In *Proceedings of HLT/NAACL*.

[Nagao, 1984] Nagao, M. (November 1984). A framework of a mechanical translation between japanese and english by analogy principle. In Elithorn, A. and Banerji, R., editors, *Artificial and Human Intelligence*, pages 173–180. Amsterdam: North-Holland.

[Nakamura, 1984] Nakamura, J. (1984). Grammar writing system (GRADE) of mu-machine translation project and its characteristics,. In *Proceedings of COLING-84*.

[Niessen, 2002] Niessen, S. (2002). Improving statistical machine translation using morpho-syntactic information.

[Niessen et al., 2000] Niessen, S., Och, F., Leusch, G., and Ney, H. (2000). An evaluation tool for machine translation: Fast evaluation for mt research. In *Proceedings of the 2nd International Conference on Language Resources and Evaluation*.

[Nirenburg and Wilks, 2000] Nirenburg, S. and Wilks, Y. (2000). Machine translation. *Advances in Computers*, 52:160–189.

[Nirenburgq et al., 1985] Nirenburgq, S., Raskin, V., and Tucker, A. B. (1985). Interlingua design for TRANSLATOR. In *Proceedings of the Conference on Theoretical and Methodological Issues in Machine Translation of Natural Languages*.

[Nomiyama, 1991] Nomiyama, H. (1991). Lexical selection mechanism using target language knowledge and its learning ability. In *IPSJ-WG, NL86-8*.

[Nyberg et al., 1994] Nyberg, E., Mitamura, T., and Carbonell, J. G. (1994). Evaluation metrics for knowledge-based machine translation. In *Proceedings of COLING-94, Kyoto, Japan*, pages 95–99.

[Och, 2000] Och, F. J. (2000). Statistical machine translation: From single-word models to alignment templates. *Technical Report*.

[Oliver Streiter, 2000] Oliver Streiter, Michael Carl, L. I. (2000). A virtual translation machine for hybrid machine translation. Available at `http://www.iai.uni-sb.de/docs/sci00.pdf`.

[Papineni et al., 2001] Papineni, K., Roukos, S., Ward, T., and Zhu, W. (2001). Bleu: a method for automatic evaluation of machine translation.

[Popovic and Ney, 2007] Popovic, M. and Ney, H. (2007). Word error rates: Decomposition over pos classes and applications for error analysis. In *Proceedings of ACL Workshop on Statistical Machine Translation*.

[Przybocki, 2004] Przybocki, M. (2004). NIST machine translation 2004 evaluation summary of results. In *Machine Translation Evaluation Workshop*.

[Quinlan, 1993] Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann.

[Resnik and Melamed, 1997] Resnik, P. and Melamed, I. (1997). Semi-automatic acquisition of domain-specific translation lexicons. In *ANLP-97*.

[Rosetta, 1994] Rosetta, M. T. (1994). Compositional translation. *International Series in Engineering and Computer Science, Kluwer Academic Publishers*.

[Russo-Lassner et al., 2005] Russo-Lassner, G., Lin, J., and Resnik, P. (2005). A paraphrase-based approach to machine translation evaluation. *Technical Report*.

[Sato, 1993] Sato, S. (1993). Example-based translation of technical terms. In *TMI-93: The Fifth International Conference on Theoretical and Methodological Issues in Machine Translation*.

[Sato and Nagao, 1990] Sato, S. and Nagao, M. (1990). Towards memory based translation. In *Coling-90: papers presented to the 13th International Conference on Computational Linguistics*, volume 3, pages 173–180.

[Satoshi Shirai and Takahashi, 1997] Satoshi Shirai, F. B. and Takahashi, Y. (1997). A hybrid rule and example-based method for machine translation. Available at `http://www.kecl.ntt.co.jp/mtg/members/bond/pubs/1997-nlprs-hybrid.pdf`.

[Shinichi and Maraki, 1992] Shinichi, D. and Maraki, K. (1992). Translation ambiguity resolution based on text corpora of source and target language. In *Proceedings of COLING-92*.

[Slocum, 1985] Slocum, J. (1985). A survey of machine translation: Its history, current status, and future prospect. *Computational Linguistics*.

[Snover and Dorr, 2006] Snover, M. and Dorr, B. (2006). A study of translation edit rate with targeted human annotation. In *AMTA 2006: Proceedings of the 7th Conference of the Association for Machine Translation in the Americas*.

[Snover et al., 2005] Snover, M., Dorr, B., Schwartz, R., Makhoul, J., Micciulla, L., and Weischedel, R. (2005). A study of translation error rate with targeted human annotation. Technical report.

[Somers, 1998] Somers, H. (1998). New paradigms in MT: the state of play now that the dust has settled. In *Proceedings of the 10th European Summer School on Logic, Linguistics and Information*.

[Somers, 1999] Somers, H. (1999). Review article: Example-based machinetranslation. *Machine Translation*.

[Somers, 2004] Somers, H. (2004). Machine translation and Welsh: the way forward. Technical report, A Report for The Welsh Language Board.

[Stent et al., 2005] Stent, A., Marge, M., and Singhai, M. (2005). Evaluating evaluation methods for generation in the presence of variation. In *Proceedings of CICLing*.

[Streiter, 2000] Streiter, O. (2000). Reliability in example-based parsing. In *International Workshop on Tree Adjoining Grammars and Related Formalisms*.

[Streiter and Iomdin, 2000] Streiter, O. and Iomdin, L. (2000). Learning lessons from bilingual corpora: Benefits for machine translation. *Journal of Corpus Linguistics*.

[Su and Chang, 1992] Su, K.-Y. and Chang, J.-S. (1992). Why corpus-based statistics-oriented machine translation. Available at `http://www.mt-archive.info/TMI-1992-Su.pdf`.

[Sumita et al., 1990] Sumita, E., Iida, H., and Kohyama, H. (1990). Translating with examples: A new approach to machine translation.

[Taylor and White, 1998] Taylor, K. and White, J. (1998). Predicting what mt is good for: User judgments and task performance. In *Proceedings of the Third Conference of the Association for Machine Translation in the Americas on Machine Translation and the Information Soup*.

[Thouin, 1982] Thouin, B. (1982). The METEO system. In Lawson, V., editor, *Practical Experience of Machine Translation, Proceedings of a Conference, London, UK, 5-6 November 1981*.

[Tomast et al., 2003] Tomast, J., Mas, J. A., and Casacuberta, F. (2003). A quantitative method for machine translation evaluation.

[Tsuruoka and Tsujii, 2003] Tsuruoka, Y. and Tsujii, J. (2003). Training a naive bayes classifier via the em algorithm with a class distribution constraint. In *Proceedings of CoNLL-2003*.

[Turian et al., 2004] Turian, J. P., Shen, L., and Melamed, I. D. (2004). Evaluation of machine translation and its evaluation.

[Vauquois, 1968] Vauquois, B. (1968). A survey of formal grammars and algorithms for recognition and transformation in machine translation. In *IFIP Congress-68, Edinburgh*.

[Wang, 1998] Wang, Y.-Y. (1998). Grammar inference and statistical machine translation.

[Wu, 1995] Wu, D. (1995). An algorithm for simultaneously bracketing parallel texts by aligning words. In *Proceedings of the 33th Annual Meeting of the Assoc. for Computational Linguistics*.

[Yamada and Knight, 2001] Yamada, K. and Knight, K. (2001). A syntax-based statistical translation mode. In *Proceedings of ACL*.

[Zhou et al., 2006] Zhou, L., Lin, C.-Y., and Hovy, E. (2006). Re-evaluating machine translation results with paraphrase support. In *Proceedings of EMNLP*.

[Zwarts, 2005] Zwarts, S. (2005). Evaluation techniques for machine translation. Available at `www.ics.mq.edu.au/~szwarts/MTEvaluation.pdf`.