[*Machine Translation Today: Translating and the Computer 15*. Papers at a conference…18-19
November 1993 (London: Aslib)]

INTERACTIVE CORPUS-BASED TRANSLATION DRAFTING TOOL (TRANSLEARN)

Stelios Piperidis
Institute for Language and Speech Processing
22 Margari Street, 115 25 Athens, Greece
email: Stelios.Piperidis@eurokom.ie

ABSTRACT

This paper describes the research and development activities carried out in the framework of the Translearn project. The aim of the project is to build a translation memory tool and the appropriate translation work environment. Translearn's application corpus consists of regulations and directives of the European Union (EU), extracted from the CELEX database, the EUs documentation system on EU law and the language versions it concentrates on are English, French, Portuguese and Greek. The development of the prototype tool for the envisaged system proves the application's usefulness in the translation process of international multilingual organisations as well as in the localisation-internationalisation process of international enterprises.

I. INTRODUCTION

This paper describes the research and development activities carried out in the framework of the LRE/Translearn project. The project's conception stems from the observation that translation work is very frequently characterised by two parameters: repetition and high demand on quality. This is particularly true for translation of technical and administrative documentation, becoming more evident in the case of law documents (contracts, regulations, etc.) and product documentation (manuals, etc.) where repetition of blocks of text may reach a rate of 70% and sometimes higher.

The aim of this project is to tackle this problem by providing a computational environment, in more practical terms a toolbox that will:

- rid translators of the repetitive part of their work by reusing existing human translations and learning from them
- enhance quality and consistency of translation by being able to integrate ancillary translation tools.

Appropriate storage of pairs of source language (SL) and target language (TL) blocks of text and provision of means for retrieval of applicable solutions and means for post-editing them would increase the productivity of a translator and at the same time improve the quality and consistency of the translation (Freibott 92) (Ishida 94).

The project's descriptive goal is to develop a machine translation aid tool dedicated to managing repetition phenomena in the translation of specific types of text. Its methodological goal is to employ sophisticated text matching techniques in order to identify the longest coherent part of source language text that is identical or similar to an input to-be-translated-text and retrieve from the memory the corresponding target language text.

The key issues of the approach revolve around three major axes :

- organisation of multilingual parallel corpora, i.e. texts in different languages, one being the translation of the other,
- alignment of parallel texts, i.e. establishment of correspondences between units of parallel texts
- text matching techniques

The targeted "end product" is a prototype translation memory tool and the appropriate translation work environment for machine assisted translation in multilingual professional environments like translation departments of international organisations and enterprises.

In section II, an overview of the approaches to the key issues of Translearn is discussed. In section III, text preprocessing and in particular the techniques adopted for text alignment are presented together with examples of aligned text derived from the application corpus. In section IV the text matching tool is discussed, while in section V the overall system architecture is sketched. In section VI the application of the translation memory tool on the CELEX database is discussed.

II. BACKGROUND

The technology underlying translation memory applications stems from what has been described in the literature as example-based machine translation (EBMT). EBMT is based on the idea of performing translation by imitating translation examples of similar sentences (Nagao 84). In this type of translation system, a large amount of bi/multi-lingual translation examples has been stored in a textual database and input expressions are rendered in the target language by retrieving from the database that example which is most similar to the input.

There are three key issues which pertain to example-based translation :

- establishment of correspondence between units in a bi/multi-lingual text at sentence, phrase or word level, i.e. alignment of parallel texts
- a mechanism for retrieving from the database the unit that best matches the input
- exploiting the retrieved translation example to produce the actual translation of the input sentence

Several different approaches have been proposed tackling the alignment problem at various levels. Catizone's technique (Catizone 89) was to link regions of text according to the regularity of word cooccurrences across texts. (Brown 91) described a method based on the number of words that sentences contain. Moreover, certain anchor points and paragraph markers are also considered. The method has been applied to the Hansard Corpus and has achieved an accuracy between 96%-97%.

(Gale 91) proposed a method that relies on a simple statistical model of character lengths. The model is based on the observation that longer sentences in one language tend to be translated into longer sequences in the other language while shorter ones tend to be translated into shorter ones. Although the apparent efficacy of the Gale-Church algorithm is undeniable and validated on different pairs of languages (English - German - French - Czech - Italian), it seems to be awkward when handling complex alignments. Complex alignments are defined to be alignments in which the 1-1 correspondence between text units in the parallel texts does not hold, and they are usually due to mergers of sentences occurring during the translation process. In the Gale-Church algorithm the 2-1 alignments had five times the error rate of 1-1. The 2-2 category disclosed a 33% error rate, while the 1-0 or 0-1 alignments were totally missed.

(Simard 92) argues that a small amount of linguistic information is necessary in order to overcome the inherited weaknesses of the Gale-Church method. He proposed using cognates, which are pairs of tokens of different languages which share "obvious" phonological or orthographic and semantic properties, since these are likely to be used as mutual translations. (Papageorgiou 94), proposed a generic alignment scheme invoking surface linguistic information coupled with information about possible unit delimiters depending on the level at which alignment is sought. Each unit, sentence, clause or phrase, is represented by the sum of its content part of speech tags. The results are then fed into a dynamic programming framework that computes the optimum alignment of text units.

In establishing a mechanism for the best match retrieval two crucial tasks are identified:
(i) determining whether the search is for matches at sentence or sub-sentence level, that is determining the "text unit", and
(ii) the definition of the metric of similarity between two text units.

As far as the decision about the text unit is concerned, the obvious choice is to use as text unit the sentence. This is because, not only are sentence boundaries unambiguous, but also translation proposals at sentence level is what a translator is usually looking for. Sentences can, however, be quite long. And the longer they are, the less possible it is that they will have a perfect match in the translation archive, and the less flexible the EBMT system will be.

On the other hand, if the text unit is the sub-sentence, we face one major problem, that is the possibility that the resulting translation of the whole sentence will be of low quality, due to boundary friction and incorrect chunking. In practice, EBMT systems that operate at sub-sentence level involve the dynamic derivation of the optimum length of segments of the input sentence by analysing the available parallel corpora. This requires a procedure for determining the best "cover" of an input text by segments of sentences contained in the database (Nirenburg 93). It is assumed that the translation of the segments of the database that cover the input sentence is known. What is needed, therefore, is a procedure for aligning parallel texts at sub-sentence level (Kaji 92), (Sadler 90). If sub-sentence alignment is available, the approach is fully automated but is quite vulnerable to the problem of low quality as mentioned above, as well as to ambiguity problems when the produced segments are rather small. Despite the fact that almost all running EBMT systems employ the sentence as the text unit, it is believed that the potential of EBMT lies on the exploitation of fragments of text smaller than sentences and the combination of such fragments to produce the translation of whole sentences (Sato 90). Automatic sub-sentential alignment is, however, a problem yet to be solved.

Turning to the definition of the metric of similarity, the requirement is usually twofold. The similarity metric applied to two sentences (by sentence from now on we will refer to both sentence and sub-sentence fragment) should indicate how similar the compared sentences are, and perhaps the parts of the two sentences that contributed to the similarity score. The latter could be just a useful indication to the translator using the EBMT system, or a crucial functional factor of the system as will be later explained.

The similarity metrics reported in the literature can be characterised depending on the text patterns they are applied on. So, the word-based metrics compare individual words of the two sentences in terms of their morphological paradigms, synonyms, hyperonyms, hyponyms, antonyms, pos tags (Nirenburg 93) or use a semantic distance d (0<d<l) which is determined by the Most Specific Common Abstraction (MSCA) obtained from a thesaurus abstraction hierarchy (Sumita 91). Then, a similarity metric is devised, which reflects the similarity of two sentences, by combining the individual contributions towards similarity stemming from word comparisons.

The word-based metrics are the most popular, but other approaches include syntax-rule driven metrics (Sumita 88), character-based metrics (Sato 92) as well as some hybrids (Furuse 92) (Cranias 94). The character-based metric has been applied to Japanese, taking advantage of certain characteristics of Japanese. The syntax-rule driven metrics try to capture similarity of two sentences at the syntax level. This seems very promising, since similarity at the syntax level, perhaps coupled by lexical similarity in a hybrid configuration, would be the best an EBMT system could offer as a translation proposal. The real time feasibility of such a system is, however, questionable, since it involves the complex task of syntactic analysis.

The third key issue of EBMT, that is exploiting the retrieved translation example, is usually dealt with by integrating into the system conventional MT techniques (Kaji 92), (Sumita 91). Simple modifications of the translation proposal, such as word substitution, would also be possible, provided that alignment of the translation archive at word level was available.

III. TEXT PREPROCESSING

In order to be able to make full use of parallel corpora, the corpora have to be rendered in an appropriate form. To this end, corpora have to be normalised, handled and aligned. Normalisation consists in extraction of the multilingual corpus body of all those sections or information that is not exploitable for text translation purposes.

Text handling can be seen as a sophisticated interface between input text streams and various text manipulation modules. At the stage of analysis, the text handler has the responsibility of transforming a text from the original form in which it is found into a form suitable for the manipulation required by the application; at the stage of synthesis, it is responsible for the reverse process, i.e. for converting the output text from the form used by the application into a form equivalent to that of the input text. The main operations usually associated with the text handler include:

- analysis of the format of the physical appearance of the input text (as evidenced by the word-processing and/or typesetting commands, such as bold and italic characters, indentation, etc.)

and mapping of these into a standardised markup language or a canonical form recognised by the application

- identification of textual units at the level of paragraphs and sentences
- identification of extra-linguistic elements, such as dates, abbreviations, acronyms, list enumerators, numbers, etc.
- at the stage of synthesis, conversion of the output of the application into the same format recognised at the stage of analysis; e.g. italicised characters, centred phrases, etc. must be given to the user in their original form.

The texts contained in the Translearn application corpus are in simple unstructured ascii format, i.e. word-processing and/or typesetting information has already been excluded.

As already discussed briefly above, alignment consists in establishing correspondence links between units in a bi/multi-lingual text. The heart of the alignment scheme, adopted in Translearn, is a method for aligning sentences based on a simple statistical model of character lengths (Gale 91). The method relies on the assumption that longer sentences in the source language tend to be translated into longer sentences in the target and that shorter sentences in the source are translated into shorter sentences in the target. A probabilistic score is assigned to each pair of proposed sentence pairs, based on the ratio of lengths of the sentences and the variance of this ratio. This probabilistic score is used in a dynamic programming framework in order to find the maximum likelihood alignment of sentences. The whole process proceeds in two steps. First, paragraphs are aligned and then sentences within a paragraph are aligned. Apparently, for the method to work well, the texts should have exactly the same number of large regions, bearing the same structure. In case sentences have been added or deleted during the translation of source into target, this method is expected to fail. It would be desirable for the method to provide ways for setting anchors between the two texts and be able to align texts above or below the anchors. Extensions, some of which follow from the Translearn text structures, proposed by (Brown 91) have also been taken into account. Instead of measuring sentence lengths in characters, they are measured by the number of words they contain. Additionally, certain points of the texts can be anchored thus dividing the texts into smaller sections to be aligned. Besides anchors, paragraph markers are also considered. Anchor points are specific to the text to be aligned and they usually appear in both texts. They are divided into major and minor anchors and alignment proceeds in two steps, first aligning major anchor points and then minor anchor points. In the first step, alignments of major anchors are assigned a cost. A dynamic programming algorithm finds the alignment of major anchors in the two texts with the least total cost. This first step outputs the texts as chunks of text between aligned major anchors. In the second step, chunks of text are retained that contain the same number of minor anchors which divide the remaining pieces into smaller sections that may extend from one to many sentences. Then, the pieces lying between minor anchors are aligned at sentence level using a hidden Markov model that generates aligned pairs with the assumption that a sentence in one language can yield zero, one or two sentences in the other language. The method has been applied to the Canadian Hansard (parallel English-French) corpus, which is structured and in which anchor points are easily detected. The approach, however, also works where anchors are rare.

Experiments with the statistical techniques applied on Translearn's application corpus showed that alignment can achieve a rate higher than 96%. Not unexpectedly, the rate of complex alignments (2-1, 1-2, 2-2, 0-1, 1-0) resulting from the application corpus was low, fact attributable to the corpus type. To illustrate, in Figure 1, aligned sentences taken from the English-French pair of the application corpus are presented. The format shows sentences in alternate languages; each English sentence is aligned with the French sentence that follows it. Markers in angled brackets (<S>) are used for sentence-end annotation.

COMMISSION REGULATION (EEC) No 486/89 of 27 February 1989 on the sale by the procedure laid down in Regulation (EEC) No 2539/84 of beef held by certain intervention agencies and intended for export, amending Regulation (EEC) No 569/88 and repealing Regulation (EEC) No 3627/88 <S>

RÈGLEMENT (CEE) No 486/89 DE LA COMMISSION du 27 février 1989 relatif à la vente, dans le cadre de la procedure définie au règlement (CEE) no 2539/84, de viandes bovines détenues par certains organismes d'intervention et destinées à être exportées, modifiant le règlement (CEE) no 569/88 et abrogeant le règlement (CEE) no 3627/88 <S>
#
THE COMMISSION OF THE EUROPEAN COMMUNITIES, <S>

LA COMMISSION DES COMMUNAUTÉS EUROPÉENNES, <S>
#
Having regard to the Treaty establishing the European Economic Community, <S>
vu le traité instituant la Communauté économique européenne, <S>
#
Having regard to Council Regulation (EEC) No 805/68 of 27 June 1968 on the common organization of the market in beef and veal (1), as last amended by Regulation (EEC) No 4132/88 (2), and in particular Article 7 (3) thereof, <S>

vu le règlement (CEE) no 805/68 du Conseil, du 27_juin_1968, portant organisation commune des marchés dans le secteur de la viande bovine (1), modifié en dernier lieu par le règlement (CEE) no 4132/88 (2), et notamment son article 7 paragraphe 3, <S>
#

Figure 1 : CELEX aligned sentences

Depending on the availability of corpus linguistic annotators in the languages represented in the multilingual corpus, the corpus is lemmatised and tagged for grammatical category (part of speech, pos). **Lemmatisation** consists in deriving the lemma or canonical form of each wordform while **tagging** consists in labelling each wordform, with its grammatical category or part of speech. Ambiguities stemming from multiple possible lemma and tag assignments are not resolved and all possible values are stored in the memory.

IV. TEXT MATCHING

The core of the system is its text matching tool. Having rendered the corpus in the appropriate form, and aligned it so that the system knows for each database sentence in a source language A the corresponding database sentence in a target language B, the matching tool can search for database sentences of language A that are identical or only similar to an input sentence (in source language A) and retrieve the equivalent sentence or sentences, if more than one exist in the target language. The approach adopted to text matching is based on computations of common elements between an input sentence and a database sentence and computation of consecutive elements in them. The level at which computations of common elements are performed can vary between wordform level and lemma-tag level, i.e. computations are either based on wordforms and their respective position in the compared sentences or on lemma-tag tuples of each word in the compared sentences as well as their respective positions in them. The level of computations depends on the availability of linguistic processors for the language pair at hand. In case linguistic processors are available, the level of computation is externally configurable by the user.

The matching tool first searches for perfect matches between the input and the database sentences. In doing so, it does not take into account extra-linguistic tokens of the sentences like dates and numbers, so that linguistically real perfect matches are not missed due to minor differences. If no perfect match is found, the matching tool searches for database sentences that are similar to the input, i.e. for fuzzy matches. In doing so, the tool considers either wordforms only or surface linguistic data (lemmas and tags) in order to search for similar sentences and identify their common parts. The parts of the database and input sentences that are different are computed and displayed to the user so that he/she knows where to intervene in the proposed translation. In addition, the system computes a similarity score between the compared sentences, based on the importance of the differences between them. The similarity score is externally configurable in that it can accept a minimum value for the similarity score in case of fuzzy matches. The modifications that the user may make to the proposed translations are then stored in the system for future use, thus enabling the system to learn new translations.

The input data of the matching mechanism are classified in different categories and extracted from different modules of the TM system. Input data consist in :

- the string of characters of the input sentence.
- the sentences as annotated by the text handler. Extra-linguistic tokens, like numbers and dates appearing in a sentence, are annotated as such by the text handler.
- lemmas and part-of-speech (pos) tags, i.e. grammatical categories, as extracted from the linguistic processors
- the database sentences and their translations. Furthermore, other pieces of information such as the position of the words in the sentences, the number of characters and words of a sentence are used in order to accelerate the matching process.
- minimum similarity score, a boundary value for the similarity score and is given by the user. Matches that correspond to values that fall below this threshold value are rejected by the matching mechanism.

The output of the matching mechanism is:

- a database sentence or a set of database sentences that have a certain similarity to the input sentence.
- similarity scores. Each of the database sentences which is close to the input sentence is associated with a similarity score so that alternative solutions are accordingly ranked. The similarity score expresses the degree of similarity between the input and the database sentence. The greater the similarity score, the more similar the sentences. The similarity score is

expressed as a percentage value. A 100% match means that there is a sentence in the TM which is identical to the input sentence.

- common words and parts of sentences between the input and the database sentences. This information is provided to the user so that he can later adapt the suggested translation in an efficient manner.

The matching mechanism consists of two processes:

(i) the perfect match process by which the system finds a database sentence (and its translation) in the Translation Memory which is identical to the input sentence, and
(ii) extraction of candidate sentences and the fuzzy match process. The fuzzy match process aims at extracting from the TM a number of sentences and their translations which resemble the given input sentence above a certain minimum degree (percentage).

Perfect Match Mechanism

The mechanism that looks for perfect matches is a module of the TM system. In the case, where a perfect match is found, the output of this process is a database sentence which is identical to the input one.

The input to the perfect match module is the input sentence as annotated by the text handler as well as meta-information about the database sentences stored in the database. The output of the perfect match algorithm is a sentence which perfectly matches the input sentence, if such a sentence exists.

Fuzzy Match Mechanism

The aim of the second phase of the matching mechanism is to find a sentence or a set of sentences in TM which are as similar as possible to the input sentence. This phase of the matching mechanism uses the results of the morphological analyser and other meta-information stored in the database. The output data of this mechanism is a sentence or a set of sentences in TM which are as similar as possible to the input sentence. For each database sentence a similarity score is computed. The module also computes indications of the common parts or words between the input and the database sentences as the user needs this information in order to adapt efficiently the suggested translation.

The second phase of the matching process is separated into two stages:

(i) extraction of candidate sentences,
(ii) fuzzy match procedure.

The aim of the first stage is to extract a list of candidate sentences from the database which have some common characteristics with the input sentence. This stage is used in order to reduce the search space and to speed up the system. Numbers of words and numbers of content words have been alternately studied and used in this stage. The underlying assumption is that two sentences with the same number of words (or content words) may be more similar than two sentences whose lengths are different

The aim of the fuzzy match procedure is to compute the number of common elements and the number of consecutive common elements between the input and the database sentence. In the simplest case an element corresponds to a word. The procedure can be expanded to encapsulate surface linguistic information, if it is available. In such a case, the element is a combination of a word and a lemma (and/or a pos tag). Furthermore, it computes the similarity score between the two sentences compared.

Exemplary cases of fuzzy matches computed by the matching tool include : (Sa, Sb, Sc, Sd stand for segments of sentences extending over a number of words identified in the input sentence (IS) and database source language (SL) sentences).

IS : Sa Sb SL: Sa Sb Sc	IS : Sa Sb Sc SL: Sa Sb	IS : Sa Sc SL: Sa Sb Sc
IS : Sa Sb Sc SL: Sa Sb Sd	IS : Sa Sb Sc SL: Sa Sc Sb	IS : Sa Sb Sc SL: Sa Sc

Figure 2 : Possible fuzzily matching segments

The input to the fuzzy match module is:

- the elements of the input sentence.
- the elements of the database (candidate) sentence.

The output of the fuzzy match module is:

- a similarity score.
- if the similarity score is greater than the threshold the user has set, the matching parts of IS and SL appropriately marked
- the TL sentence associated with the SL in the TM and, if possible, the matching segments between the SL and the TL.

In cases where fuzzy matches accepted by the user are found, the user is asked to render in the target language those parts of the SL sentence that have not matched. In this way, the user can render the exact translation of the input sentence he wants to translate, reusing the existing translations for parts of it. The new emerging pair of translation units is then stored in the translation memory database for future use. In cases where no match can be found, including cases where matches exist but their score is below the user's desired threshold, the user is asked to provide the translation of the IS which is again subsequently stored in the TM database. Thus, the translation memory system starts learning new translation pairs in an interactive mode.

V SYSTEM ARCHITECTURE

The above described tools have been implemented in the Translearn environment. Figure 3 shows the configuration of the Translearn tools and their communication.

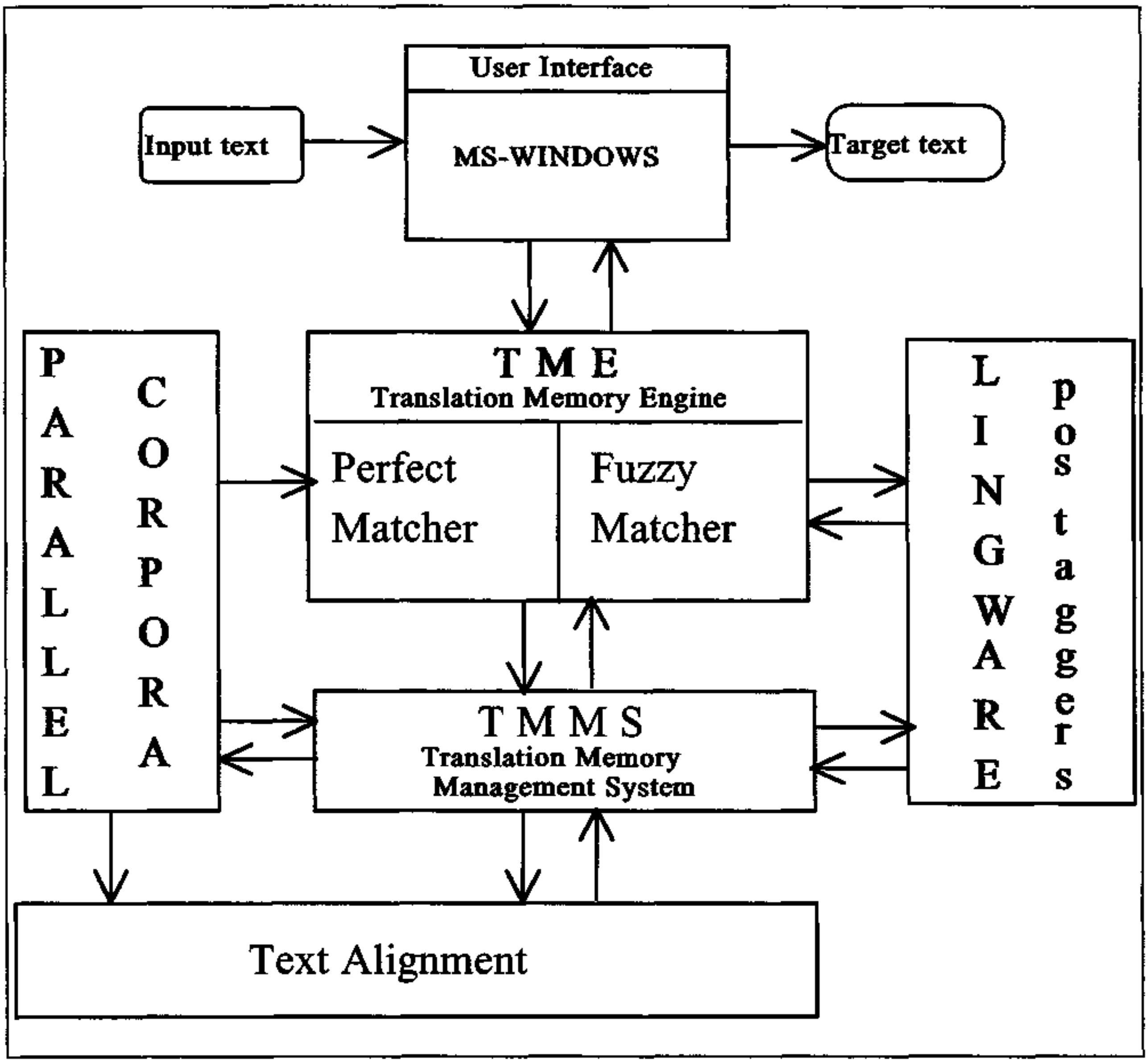


Figure 3 : The Translearn Tool Configuration

The Translearn tools are integrated in a translation environment operating on a client-server architecture. In the standard client-server architecture, one or more clients and one or more servers, along with the underlying operating system and interprocess communication systems, form a composite system allowing distributed computation, analysis and presentation. Each client runs an application on a workstation but does database access from the server. This process is depicted in Figure 4.

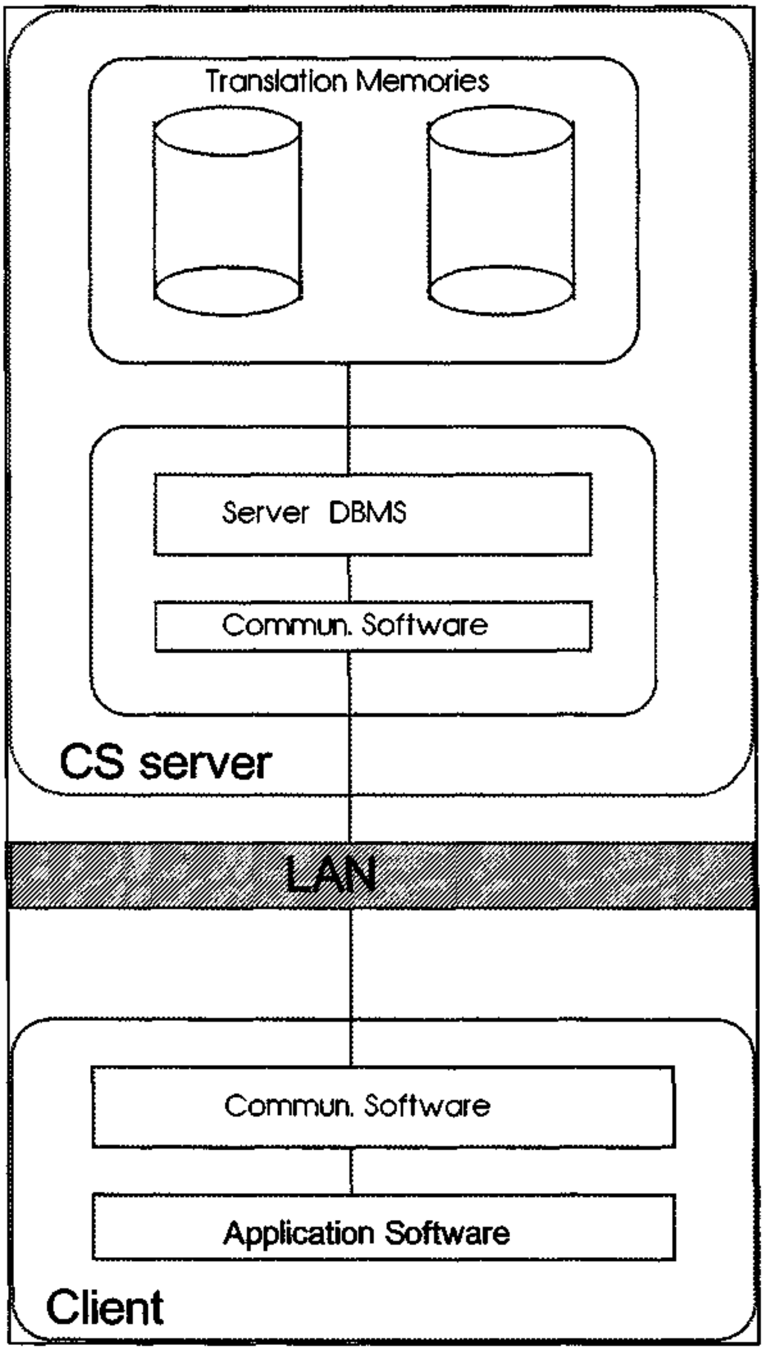


Figure 4 : Client-Server Architecture

The Client presents a graphical user interface (Microsoft Windows-based). This interface is the sole means of garnering user translation requests, as well as the means of presenting the results of one or more translation alternatives. In the translation environment, the Client performs the necessary handling of a text that a user has opened in order to translate, without any involvement of the server. Furthermore, in real mode operation, the client invokes the appropriate linguistic processors, if available, to fill in the linguistic information that text matching demands. The server (UNIX based) stores the multilingual corpus meta-data (linguistic meta-data, statistical and alignment data) and transmits them over the network upon a client's request.

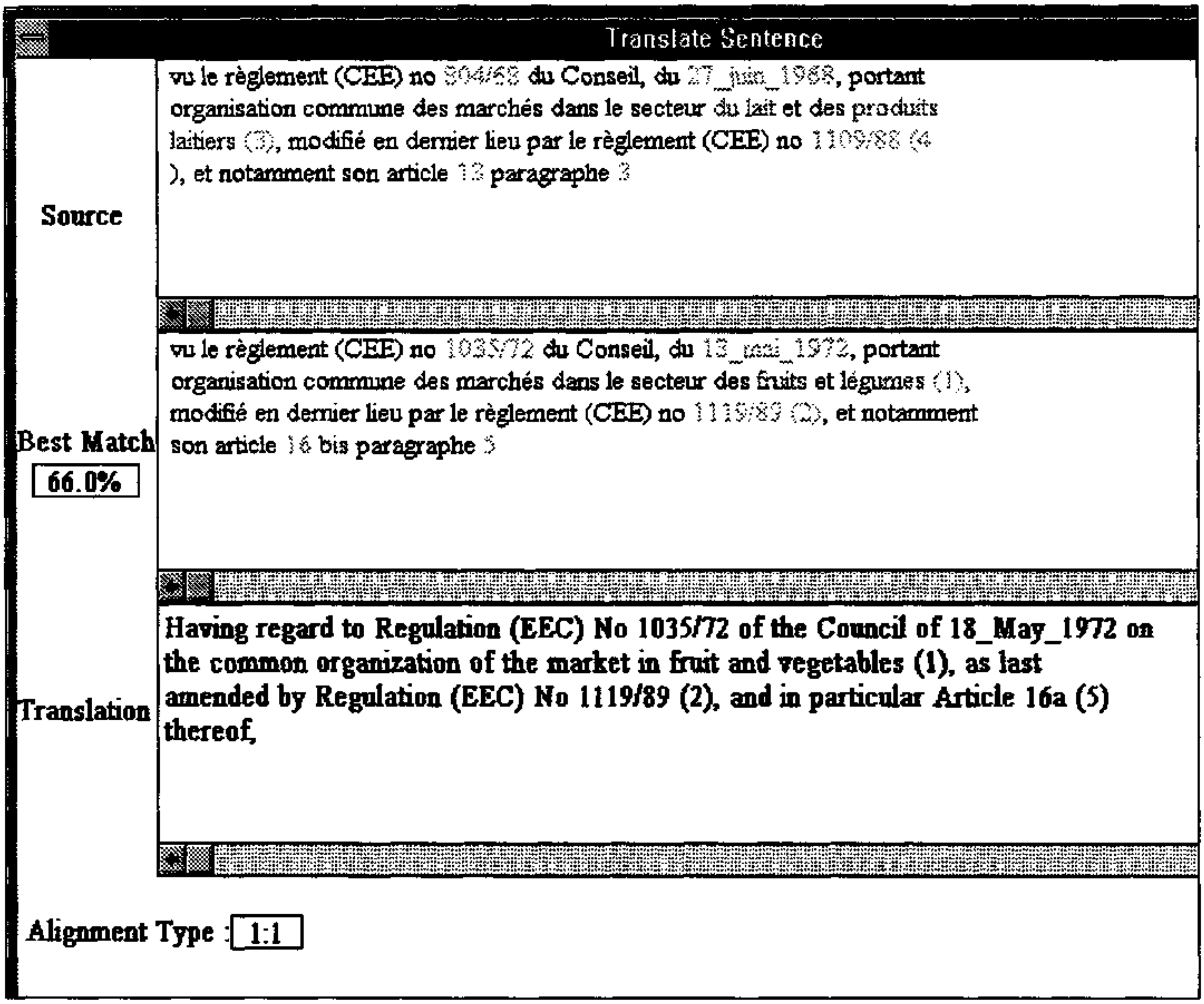
VI. APPLICATION

Translearn has collected and investigated a large body of parallel ascii texts, between 5 and 6 million words, for each language, English, French, Portuguese and Greek. The corpus has been extracted from the CELEX database, the European Union's (EU) documentation system on EU law. The characteristics of the administrative sublanguage span the whole corpus, while technical/financial sublanguage is used depending on the subject matter of each text. The corpus texts are of regulatory type with slight variations, while the structure of almost all texts is the same. The corpus by itself validates the usefulness of the project by the high percentage of frequently recurring pieces of text that need not be retranslated since one can reuse existing human translations. In parallel, samples of texts extracted from software manuals have been studied revalidating the usefulness of the approach.

The corpora have also been aligned so that each paragraph and sentence in the French, Portuguese and Greek version is linked to the corresponding paragraph and sentence of the English version. The alignment software that was developed, based on techniques considering mainly statistical information, computed 96% correct alignments while methods for improvements and increasing robustness are currently being explored. Experiments for alignment below the level of sentence have also been made, yielding promising results. The new methods combine the power of statistical modelling and surface linguistic information in order to establish correspondences between phrases/clauses across multilingual texts. The alignment software is used not only for translation data preparation but it constitutes an integral utility of the Translearn environment so that if the future user has translated texts available (s)he will be able to align them, store them and reuse them.

The corpus has been lemmatised and tagged at part-of-speech (pos) level. Lemmatisation is performed by access to a morphological dictionary. The tagsets used are compatible with the TEI and NERC guidelines, catering at the same time for the peculiarities of each language. Lemmatisation and tagging return for each word of the text the combination <lemma, pos>. If multiple such combinations are valid for a word, then all possible combinations are output. Combinations of more than one <lemma, pos> tuples are then grouped together to form a morphologically ambiguous class and these ambiguity classes are treated as tags of their own. Lemma and pos tag information is later utilised in the text matching process in order to determine identical or similar sentences and subsequently rank their similarity.

In Figure 5, we illustrate the text matching process operating on French-English. The example sentences are taken from the CELEX application corpus. In the upper window, the input sentence to be translated is presented, in the middle window the database sentence best matching the input and in the lower window the translation equivalent of the database sentence (that of the middle window). In the upper and middle windows the differing segments are shown in different colours. In addition, segments having the same lemma form are indicated by different colours. In this way, the system indicates to the translator which segments have to be changed and what types of changes have to be made. In the lower window the translator can then make the appropriate changes, adopt and store the new translation pair in the database, thus enabling the system to "learn" new translations.



Translate Sentence

Source

vu le règlement (CEE) no 804/68 du Conseil, du 27_juin_1968, portant organisation commune des marchés dans le secteur du lait et des produits laitiers (3), modifié en dernier lieu par le règlement (CEE) no 1109/88 (4), et notamment son article 13 paragraphe 3

Best Match 66.0%

vu le règlement (CEE) no 1035/72 du Conseil, du 13_mai_1972, portant organisation commune des marchés dans le secteur des fruits et légumes (1), modifié en dernier lieu par le règlement (CEE) no 1119/89 (2), et notamment son article 16 bis paragraphe 5

Translation

Having regard to Regulation (EEC) No 1035/72 of the Council of 18_May_1972 on the common organization of the market in fruit and vegetables (1), as last amended by Regulation (EEC) No 1119/89 (2), and in particular Article 16a (5) thereof,

Alignment Type : 1:1

Figure 5 : Translating from French into English using Translearn

ACKNOWLEDGMENTS

This work was supported by the Commission of the European Union as part of the Linguistic Research Engineering Programme. Collaborators in the project were University College London (UK), Istituto de Linguistica Teorica e Computacional, ILTEC (Portugal), Sonovision Itep Technologies, SITE (France), Knowledge S.A. (Greece) and the Institute for Language and Speech Processing, ILSP (Greece) which coordinated the project.

REFERENCES

Brown P. F. et al, (1991). "Aligning Sentences in Parallel Corpora". *Proc. of the 29th Annual Meeting of the ACL,* pp 169-176.

Catizone R., Russell G., Warwick S., (1989), "Deriving translation data from bilingual texts" *Proc. of the First Lexical Acquisition Workshop,* Detroit 1989

Cranias L., Papageorgiou H. and Piperidis S., (1994), "A matching technique in Example-Based Machine Translation" *Proc. of Coling,* pp 100-105.

Freibott G.P., (1992), "Computer Aided Translation in an Integrated Document Production Process: Tools and Applications", *Translating and the Computer 14,* pp 45-66.

Furuse O. and Iida H. , (1992). "Cooperation between Transfer and Analysis in Example-Based Framework". *Proc. Coling,* pp 645-651.

Gale W. A. and Church K. W., (1991). "A Program for Aligning Sentences in Bilingual Corpora". *Proc. of the 29th Annual Meeting of the ACL.,* pp 177-184.

Ishida R., (1994), "Future translation workbenches: some essential requirements", *Aslib Proceedings, vol.46, no. 6,* June 1994, pp 163-170

Kaji H., Kida Y. and Morimoto Y., (1992). "Learning Translation Templates from Bilingual Text". *Proc. Coling.,* pp 672-678.

Nagao M., (1984). "A framework of a mechanical translation between Japanese and English by analogy principle". *Artificial and Human Intelligence, ed. Elithom A. and Banerji R., North-Holland,* pp 173-180.

Nirenburg S. et al, (1993). "Two Approaches to Matching in Example-Based Machine Translation". *Proc. of TMI-93, Kyoto, Japan.*

Papageorgiou H., Cranias L. and Piperidis S., (1994), "Automatic alignment in parallel corpora", *Proc. of the 32nd Annual Meeting of the ACL*

Sadler V. and Vendelmans R., (1990). "Pilot Implementation of a Bilingual Knowledge Bank". *Proc. of Coling,* pp 449-451.

Sato S. and Nagao M., (1990). "Toward Memory-based Translation". *Proc. of Coling,* pp 247-252.

Sato S., (1992). "CTM: An Example-Based Translation Aid System". *Proc. of Coling,* pp 1259-1263.

Simard M., Foster G. and Isabelle P., (1992), "Using cognates to align sentences in bilingual corpora", *Proc. of TMI*

Sumita E. and Iida H., (1991). "Experiments and Prospects of Example-based Machine Translation". *Proc. of the 29th Annual Meeting of the Association for Computational Linguistics,* pp 185-192.

Sumita E. and Tsutsumi Y., (1988). "A Translation Aid System Using Flexible Text Retrieval Based on Syntax-Matching". *TRL Research Report, Tokyo Research Laboratory, IBM.*