Case-Based Translation: First Steps from a Knowledge-Light Approach Based on Analogy to a Knowledge-Intensive One

Yves Lepage, Jean Lieber

Case-Based Translation: First Steps from a Knowledge-Light Approach Based on Analogy to a Knowledge-Intensive One

Yves Lepage and Jean Lieber*

Waseda University, IPS, 2-7 Hibikino, 808-0135 Kitakyushu, Japan
yves.lepage@waseda.jp
Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France
jean.lieber@loria.fr

Abstract. This paper deals with case-based machine translation. It is based on a previous work using a proportional analogy on strings, i.e., a quaternary relation expressing that "String A is to string B as string C is to string D". The first contribution of this paper is the rewording of this work in terms of case-based reasoning: a case is a problem-solution pair (A, A') where A is a sentence in an origin language and A', its translation in the destination language. First, three cases (A, A'), (B, B'), (C, C') such that "A is to B as C is to the target problem D" are retrieved. Then, the analogical equation in the destination language "A' is to B' as C' is to x" is solved and $D' = x$ is a suggested translation of D. Although it does not involve any linguistic knowledge, this approach was effective and gave competitive results at the time it was proposed. The second contribution of this work aims at examining how this prior knowledge-light case-based machine translation approach could be improved by using additional pieces of knowledge associated with cases, domain knowledge, retrieval knowledge, and adaptation knowledge, and other principles or techniques from case-based reasoning and natural language processing.

Keywords: Analogy, Machine translation, Knowledge-light case-based reasoning, Knowledge-intensive case-based reasoning

1 Introduction

Right after the advent of computers, machine translation was the very first non-numerical application envisaged for these machines [39]. The first approach consisted in word-to-word translation, relying on large bilingual dictionaries that contained assembly instructions for insertion, deletion or movement of words relying on the inspection of close context. It was rapidly understood that the focus should move from bilingual dictionaries to monolingual grammatical descriptions of languages and the design of parsers and generators, hence the so-called rule-based approach, in which translation itself took place in a transfer phase, working at a higher level of description [4].

* The first author is supported by a JSPS Grant-In-Aid 18K11447: "Self-explainable and fast-to-train example-based machine translation using neural networks."

1.1 Data-Oriented Approaches to Machine Translation

The idea of example-based machine translation was introduced for the first time in the seminal paper of Nagao [19]: translation should be performed by comparing a new sentence to be translated (the target problem) to existing examples of translations (a source problem and its solution, i.e., its translation). Although some research in example-based machine translation was started, it was rapidly overwhelmed by the stream of statistical machine translation (SMT), an approach that also entirely relies on the availability of aligned parallel corpora, i.e., sets of translated sentences. In the statistical approach to machine translation, several types of knowledge are extracted from aligned corpora: mainly dictionaries of corresponding short sequences of words with associated translation probabilities, probabilities for how they should be reordered, and probabilistic language models for the fluency in the destination language.

1.2 Availability of Data for Machine Translation

Statistical machine translation systems require aligned bilingual corpora to extract the above-mentioned knowledge. In 1988, the founders of the approach, IBM researchers [5], used the Hansard corpus of proceedings of the Canadian parliament for French-English. The need for such corpora intensified their production. In 2002, ATR officially announced a multilingual corpus with 160,000 sentences in Japanese, Chinese and English [32]. The European Parliament speeches corpus (Europarl) contained at least 400,000 sentences in combination with English for 23 other languages in its 3$^{\text{rd}}$ version in 2005 [13]. Evaluation campaigns then collected and released corpora of more that 1 million sentences (WMT 2006 et seq., IWLST 2014 et seq.). With the operational deployment of systems on the Web, large companies or institutions were able to collect very large corpora: Google is claiming 1 billion aligned sentences in French-English in 2016 [10,29]; the World Intellectual Property Organisation (WIPO) is also claiming hundreds of millions of aligned sentences extracted from patent families in various language pairs [42]; the DGT-Translation memory of the Directorate-General for Translation of the European Commission released 6.8 million translation units in March 2018 in addition to the several million units already released. Subtitles also constitute an invaluable resource of multilingual aligned data [33,16].

The statistical machine translation approach, which had been dominant in research approximately from 2005 to 2015, was in turn drowned under the tsunami of the neural network approach to machine translation (NMT). This last approach requires even larger amount of data than statistical machine translation systems (and also enormous computation time and power in comparison to statistical machine translation), but, for well-resourced languages, this is no more a problem. Indeed, very large amounts of data are available for such languages and part of such data is not even used during training. For instance, [29] reports that Google used only 15 % of the total of the French-English data at their disposal for their neural machine translation system.

1.3 The Challenge of Less-Resourced Languages: an Opportunity for Case-Based Reasoning

As enough data is available for well-resourced languages or language-pairs, the consciousness about less-resourced languages is raising among researchers in the natural language processing community. There exist more than 6,000 languages in the world and only slightly more than 100 are available with Google Translate. The Linguistic Data Consortium is aware of the lack of data for the majority of the languages of the world and is starting to explore less expensive ways to collect data for such languages, e.g., through gamification [7]. Other techniques which are being proposed are in the vein of zero-shot translation, i.e., the possibility of mapping data across independently learnt neural network models [10].

Another possibility could well be the use of case-based reasoning, which is supposed to be a remedy when not so many examples are available. As explained above in Section 1.1, applying case-based reasoning to machine translation was indeed present in Nagao's proposal in 1984, however its first mention with its official name, or at least under the form of *memory-based reasoning,* is to be found in Kitano's description of massive parallel artificial intelligence later in 1993 [12].

1.4 Purpose of the Paper

The purpose of this paper is twofold. Its first objective is to reword example-based machine translation, in particular the approach described in [14], in terms of case-based reasoning, so as to open opportunities for CBR researchers to tackle machine translation for less-resourced language pairs. We will show that this particular approach to example-based machine translation corresponds indeed to a knowledge-light CBR approach using analogies.

The second objective is to open paths for improving this approach to example-based machine translation, as such rewording will open opportunities to CBR researchers to easily spot possible places where improvement can be brought. In particular, we see opportunities for CBR researchers to work on more elaborated description of cases, or introducing and representing domain knowledge, or knowledge dedicated to retrieval and adaptation.

These two objectives are addressed in Sections 3 and 4. They are preceded by a section giving some preliminaries, Section 2.

2 Preliminaries: Definitions, Notations, Assumptions

2.1 On Case-based Reasoning

A Reminder of the Main Notions. Case-based reasoning (CBR [25]) aims at solving a new problem—the *target problem*, denoted by `tgt`—with the help of cases, where a case represents a problem-solving episode. In this paper, a case is denoted by an ordered pair $(\texttt{pb}, \texttt{sol}(\texttt{pb}))$ where `pb` is a problem and $\texttt{sol}(\texttt{pb})$ is a solution of `pb`. However, it may occur that some additional pieces of information are associated with a case. The *case base* is a finite set of cases

and constitutes an essential source of knowledge of the CBR system. A *source case* $(\texttt{srce}, \texttt{sol}(\texttt{srce}))$ is an element of the case base, $\texttt{srce}$ is a *source problem*.

The *process model* of CBR consists usually in four steps: retrieval, adaptation, correction and memorization (also known as retrieve, reuse, revise and retain in the 4Rs model of [1]). Retrieval consists in selecting one or several source case(s). Adaptation uses this or these case(s) to propose a first solution $\texttt{sol}(\texttt{tgt})$ to $\texttt{tgt}$. This solution $\texttt{sol}(\texttt{tgt})$ is possibly corrected, e.g. by confrontation to a human expert. Finally, the newly formed case $(\texttt{tgt}, \texttt{sol}(\texttt{tgt}))$ is memorized in the case base if this is judged to be useful.

The *knowledge model* of CBR decomposes its knowledge base in four containers [24]. The first one is the case base, already mentioned. The *domain ontology* contains knowledge about the objects and properties used to represent the cases in the application domain. It can be considered as a representation of necessary conditions for a case to be licit. The *retrieval knowledge* is used during the retrieval step, the *adaptation knowledge*, during the adaptation step.

The First Approach to Example-Based Machine Translation. The seminal paper in example-based machine translation by Nagao in 1984 [19], was indeed "case-based reasoning comes early." In its introduction, the problem of translation is stated as follows: given a sentence in a language to be translated into another language, use another sentence in the same language that differs by only one word and for which the translation in the other language is known, change the word that differs in the other language to get the final translation. Given the above description of example-based machine translation, one can imagine a CBR process based on the use of a bilingual dictionary for managing several mismatches between the source and the target problems. This was the approach explored in [28]. An entry of such a dictionary is a pair (w^o, w^d) where w^o (resp., w^d) is a word in the origin language (resp., the destination language). It also contains the pair $(\varepsilon, \varepsilon)$: the empty string ε is considered as a particular word in both languages. The principle of this approach is as follows:

Retrieval Find a case $(\texttt{srce}, \texttt{sol}(\texttt{srce}))$ that is similar to $\texttt{tgt}$ in that a minimal number of words have to be substituted in $\texttt{srce}$ to get $\texttt{tgt}$.

Adaptation For each word substitution $w_s^o \rightsquigarrow w_t^o$ from $\texttt{srce}$ to $\texttt{tgt}$ in the origin language, the word substitution $w_s^d \rightsquigarrow w_t^d$ is built in the destination language, using the dictionary entries (w_s^o, w_s^d) and (w_t^o, w_t^d). Then, these substitutions are applied on $\texttt{sol}(\texttt{srce})$ to get $\texttt{sol}(\texttt{tgt})$.

For example, with French as origin language and English as destination language:

$$\begin{aligned} \texttt{tgt} &= \textit{Amenez-moi à Pluton.} \\ \texttt{srce} &= \textit{Amenez-moi à votre chef.} \\ \texttt{sol}(\texttt{srce}) &= \textit{Take me to your leader.} \\ \text{hence } \texttt{tgt} &= \sigma^o(\texttt{srce}) \text{ with } \sigma^o = \textit{chef} \rightsquigarrow \textit{Pluton} \circ \textit{votre} \rightsquigarrow \varepsilon. \end{aligned}$$

Given the entries $(\mathit{Pluton}, \mathit{Pluto}), (\mathit{chef}, \mathit{leader}), (\mathit{votre}, \mathit{your})$ and $(\varepsilon, \varepsilon)$

it comes $\sigma^d = \mathit{leader} \rightsquigarrow \mathit{Pluto} \circ \mathit{your} \rightsquigarrow \varepsilon$

hence $\texttt{sol}(\texttt{tgt}) = \sigma^d(\texttt{sol}(\texttt{srce})) = \mathit{Take\ me\ to\ Pluto.}$ (correct translation)

Note that this approach is likely to propose a large number of incorrect translations among the proposed solutions, in particular because a single word in the origin language can be translated in different ways and no context is used here to select the appropriate word.

A more elaborate approach to example-based machine translation was proposed in [27] in which additional pieces of information are added to cases in the form of their dependency parses. Adaptation is constrained by the shape of the dependency trees, in that only sub-sequences of words which correspond to a sub-tree in dependency trees can be substituted for. It then becomes crucial to be able to align dependency sub-trees across languages and to perform fast approximate retrieval of sub-trees. The Kyoto EBMT system implemented such an approach [20].

2.2 On Strings and Texts

An *alphabet* $\mathcal{A}$ is a finite set. A *character* is an element of $\mathcal{A}$. A *string* of length $\ell \geq 0$ on $\mathcal{A}$ is a finite sequence $\alpha_1\alpha_2\ldots\alpha_\ell$ of characters. The set of strings is denoted by $\mathcal{A}^*$. It contains the empty string ε. *Edit distances* on strings are distance functions on $\mathcal{A}^*$, defined as follows. An *edit operation* is a function from $\mathcal{A}^*$ to $\mathcal{A}^*$. Common edit operations are the following ones:

- *Deletions* consist in removing a character of a string. For example, the deletion of the 3rd character of the string *case* yields *cae*.
- *Insertions* consist in inserting a character into a string at a given position. For example, inserting *s* after position 4 of string *case* yields *cases*.
- *Substitutions* consist in replacing a character of a string with another character. For example, the substitution of *c* with *b* at the 1st position of *case* yields *base*. A substitution can be written as the composition of a deletion and an insertion. In the example: $\mathit{case} \mapsto \mathit{ase} \mapsto \mathit{base}$.
- *Swaps* consist in swapping two contiguous characters. For example, swapping *a* with *s* in *case* yields *csae*.
- *Shifts* are extension of swaps to non-necessarily contiguous sequences of characters. The length of the gap is usually taken into account to compute the weight of shifts. For example, shifting *se* with *ca* in *case* yields *seca*.
- Etc.

An *edit path* is a sequence $\texttt{P} = \texttt{e}_1\ \texttt{;}\ \ldots\ \texttt{;}\ \texttt{e}_{p-1}\ \texttt{;}\ \texttt{e}_p$ of edit operations $\texttt{e}_i$. Such a path *relates* a string S_1 to a string S_2 if $\texttt{e}_p(\texttt{e}_{p-1}(\ldots(\texttt{e}_1(S_1))\ldots)) = S_2$. Let `weight` be a function that associates to an edit operation `e` an integer $\texttt{weight}(\texttt{e}) > 0$. This function is extended on edit paths by $\texttt{weight}(\texttt{e}_1\ \texttt{;}\ \texttt{e}_2\ \texttt{;}\ \ldots\ \texttt{;}\ \texttt{e}_p) = \sum_{i=1}^{p} \texttt{weight}(\texttt{e}_i)$. Given a set of edit operations and a function `weight`, the edit distance from a string S_1 to a string S_2 is defined as

$$\texttt{dist}(S_1, S_2) = \min\{\texttt{weight}(\texttt{P}) \mid \texttt{P}\text{: path from } S_1 \text{ to } S_2\}$$

The Levenshtein distance is an edit distance that considers deletions, additions and substitutions only, each with a weight of 1. The LCS distance (longest common subsequence) is simpler in that it considers deletions and insertions only, with a weight of $\texttt{weight}(\texttt{e}) = 1$ for every operation (in this setting, substitutions have a weight of 2). The computation of the Levenshtein or the LCS distance is quadratic in the worst case [2] (proving that the Levenshtein distance can be computed in lesser time would imply $P = NP$ [38]). Better behaviours can be obtained for felicitous cases; for instance, the computation of the distance between two equal strings is of course linear in the length of the string by Ukkonen's algorithm [37].

It is worth noting that edit distances have been used in CBR on other structures, such as temporal sequences [26] or graphs [6,15].

3 A Knowledge-Light Approach to Case-Based Translation Using Analogies

In [14], an implementation of example-based machine translation was proposed and evaluated. The approach was effective: at that time, it delivered comparable results to nascent statistical methods. It worked only on the string level and did not involve any linguistic knowledge. The purpose of this section is to describe it anew, but this time, in terms of knowledge-light CBR. Before relating it to CBR, an analogical relation between strings is introduced.

3.1 Analogy Between Strings

A *proportional analogy* is a quaternary relation between four objects A, B, C and D denoted by $A : B :: C : D$. In all generality, we call *conformity* the operation denoted by the sign :: and *ratio* the operation denoted by the sign :. An analogy should satisfy the following properties (for any objects A, B, C and D of the same type):

Reflexivity of conformity: $A : B :: A : B$;
Symmetry of conformity: if $A : B :: C : D$ then $C : D :: A : B$;
Exchange of the means: if $A : B :: C : D$ then $A : C :: B : D$.

An *analogical equation* is an expression of the form $A : B :: C : x$ where A, B and C are given objects and x is the unknow. Solving such an equation for x consists in finding the objects x satisfying this equation.

A proportional analogy between numbers is defined by $A : B :: C : D$ if $B - A = D - C$, i.e., conformity is equality and ratio is subtraction. A proportional analogy between n-tuples of numbers ($A = (a_1, a_2, \ldots, a_n)$) is defined by $A : B :: C : D$ if $a_i : b_i :: c_i : d_i$ for each $i \in \{1, 2, \ldots, n\}$. For instance, (0, 2) : (3, 3) :: (1, 6) : (4, 7).

A proportional analogy between strings is defined as follows. First, let $\mathcal{A} = \{\alpha_1, \alpha_2, \ldots, \alpha_p\}$ be a predefined finite set of characters, i.e., an alphabet. For a given string $S \in \mathcal{A}^*$, where $\mathcal{A}^*$ is the set of all strings on $\mathcal{A}$, let $\pi(S) =$

$(|S|_{\alpha_1}, |S|_{\alpha_2}, \ldots |S|_{\alpha_p})$ be the Parikh vector of string S, i.e., the vector of the number $|S|_{\alpha_i}$ of occurrences of each character α_i in S. Then, four strings A, B, C, D are in proportional analogy, i.e., $A : B :: C : D$, if $\pi(A) : \pi(B) :: \pi(C) : \pi(D)$ and $\texttt{dist}(A, B) = \texttt{dist}(C, D)$, with $\texttt{dist}$ the LCS distance. If $A : B :: C : D$, it can be proven that $\texttt{dist}(A, C) = \texttt{dist}(B, D)$ also holds, thanks to the exchange of the means. For example, it can be easily checked that the following strings make a proportional analogy:

$$\begin{array}{ll} A = \textit{to reason} & B = \textit{reasoning} \\ C = \textit{to do} & D = \textit{doing} \end{array}$$

In particular, $|B|_r - |A|_r = 1 - 1 = |D|_r - |C|_r = 0 - 0$, $|B|_n - |A|_n = 2 - 1 = |D|_n - |C|_n = 1 - 0$, $\texttt{dist}(A, B) = \texttt{dist}(C, D) = 6$, and $\texttt{dist}(A, C) = \texttt{dist}(B, D) = 6$.

Two words of caution: On integers, any analogical equation has exactly one solution ($D = B - A + C$ always exists). By contrast, on strings, it can have zero, one or multiple solutions. For example, $a : b :: ac : x$ has two solutions: bc or cb. Also, notice that, on strings, conformity is not transitive in the general case: $A : B :: C : D$ and $C : D :: E : F$ do not imply $A : B :: E : F$ in general.

3.2 Case Representation

In the domain of machine translation, a problem `pb` is given by a sentence in an "origin" language (e.g., French) and a solution of `pb` is a sentence `sol(pb)` in a "destination" language (e.g., English). A case is a pair $(\texttt{pb}, \texttt{sol(pb)})$, without additional information. Fig. 1 illustrates a case base containing such cases, i.e., pairs of translated sentences.

`srce`	`sol(srce)`
As-tu sauté au plafond ?	*Did you hit the roof?*
Elle évite ce chien.	*She avoids this dog.*
Elle évite les chiens.	*She avoids dogs.*
Il veut faire ça.	*He wants to do that.*
Je peux faire du vélo aujourd'hui.	*I can ride my bicycle today.*
J'aime ce chat.	*I like this cat.*
J'ai sauté au plafond.	*I hit the roof.*

Fig. 1. A toy case base of translations from French into English.

3.3 No Domain Knowledge Used

It is worth mentioning that this approach uses no domain knowledge (no linguistic knowledge about any of the two languages involved or about their relationships): the knowledge is contained only in the cases. This makes the approach independent of any language: only the case acquisition has to be carried out to apply it to a new pair of origin and destination languages.

3.4 Retrieval

Let tgt be the French sentence to be translated. In the running example, the following French sentence is chosen:

$$\texttt{tgt} = \textit{Je veux faire du vélo.}$$

Retrieval aims at finding one or several *triples* of source cases $((\texttt{srce}_A, \texttt{sol}(\texttt{srce}_A)), (\texttt{srce}_B, \texttt{sol}(\texttt{srce}_B)), (\texttt{srce}_C, \texttt{sol}(\texttt{srce}_C)))$ such that $\texttt{srce}_A : \texttt{srce}_B :: \texttt{srce}_C : \texttt{tgt}$. With the running example, the source problems could be

$$\begin{aligned}\texttt{srce}_A &= \textit{Tu peux le faire aujourd'hui.}\\ \texttt{srce}_B &= \textit{Tu veux le faire.}\\ \texttt{srce}_C &= \textit{Je peux faire du vélo aujourd'hui.}\end{aligned}$$

If no such triple can be found, an alternative approach can be applied (see Section 3.6).

3.5 Adaptation

Given a target problem and a source case triple that has been retrieved, the adaptation is based on the following principle: if four sentences in the origin language are in proportional analogy then it is plausible that their translations in the destination language are also in proportional analogy. Based on this idea, the adaptation of the source case triple to solve the target problem consists in solving the following analogical equation:

$$\texttt{sol}(\texttt{srce}_A) : \texttt{sol}(\texttt{srce}_B) :: \texttt{sol}(\texttt{srce}_C) : x$$

In the running example, the English sentences translating the French sentences $\texttt{srce}_A$, $\texttt{srce}_B$ and $\texttt{srce}_C$ are

$$\begin{aligned}\texttt{sol}(\texttt{srce}_A) &= \textit{You can do it today.}\\ \texttt{sol}(\texttt{srce}_B) &= \textit{You want to do it.}\\ \texttt{sol}(\texttt{srce}_C) &= \textit{I can ride my bicycle today.}\end{aligned}$$

The equation is solvable and gives the following solution which is a correct translation of tgt:

$$\texttt{sol}(\texttt{tgt}) = \textit{I want to ride my bicycle.}$$

Since there may be several retrieved source case triples and, for each of them, several solutions to the analogical equation in the destination language, the approach may propose a set of solutions sol(tgt) (possibly repeated a number of times), not necessarily all correct. The translation examples in [14] suggest that the quality of the solutions should be correlated with their output frequency.

3.6 Recursive Application of the CBR Process

The main bottleneck of the above approach lies in the fact that the first step of retrieval is obviously prone to fail in the majority of cases, unless a more flexible definition of analogy between strings is provided. We will discuss more flexible approaches later (Sections 4.1 and 4.5). With the purely symbolic approach described above in Section 3.1 for analogy between strings, there is statistically very little chance to find a source case triple of sentences which makes an analogy with a given target problem (the input sentence to translate), even in a dense data set consisting of very short sentences from similar restricted domains exhibiting a large number of commutations (like the BTEC corpus: *My tooth hurts.*, *My head hurts.*, *My head hurts badly.*, etc.). In [14] where a purely symbolic approach was adopted, a recursive application of the method was proposed to remedy this problem: instead of triples, pairs of cases $(\texttt{srce}_A, \texttt{srce}_B)$ are retrieved. Solving $\texttt{srce}_B : \texttt{srce}_A :: \texttt{tgt} : x$ for x yields $\texttt{srce}_C$. When $\texttt{srce}_C$ does not already belong to the case base, it is considered a new target on which to apply the CBR process recursively. This recursive application is different from [31] where recursive reasoning is applied to sub-components, hence on the hierarchical structure of the cases. Recursive CBR on sub-parts of sentences seems also a promising topic for translation. We think that all this opens new avenues to study: how to combine retrieval with a recursive application of the CBR process itself on entire cases or sub-parts of cases so as to lead to a solution of the target problem as fast as possible?

4 Towards a Knowledge-Intensive Approach to Case-Based Translation Using Analogies

The approach presented in the previous section is effective, though it only uses simple cases, the LCS edit distance and no other knowledge containers. The aim of this section is to examine how this approach can be improved thanks to a more flexible definition of analogies (Section 4.1), a richer case representation (Section 4.2), the use of domain knowledge (Section 4.3), and the modification of the following CBR steps using some additional knowledge: retrieval and adaptation (Sections 4.4 and 4.5). Other techniques related to CBR or to natural language processing could be used as well to improve the system, such as case maintenance techniques or textual CBR techniques (see e.g. [30] and [40]).

4.1 Using More Flexible Analogies

Word vector representations may allow for a more flexible definition of analogies between sentences, considered as sequences of words. One of the recent breakthroughs in natural language processing is the use of (shallow) artificial neural networks for the fast computation of distributional semantic word vector representations (word embeddings) from large corpora [18,22,23]. This offers the possibility of solving semantic analogies, as illustrated by the hackneyed example

man : *woman* :: *king* : *x* leads to *x* = *queen* [18], through the computation of semantic similarities between words. Vector representations of words had in fact already been proposed [36,34] to answer (SAT) questions using a model inspired by Gentner's structural mapping engine [9], called the Latent Relation Mapping Engine [35]. As for sentences, so-called soft alignment matrices give the word-to-word distance between each pair of words in two sentences. Figure 2 illustrates how it could be possible to use such representations to solve analogical equations between sentences. Some attempts have already been made either using soft alignment representations [11] or vector representations of sentences [17].

We think that this general problem is relevant for the CBR community as it falls within the topic of computational analogy: how to solve analogical equations between sentences in a truly semantic way?

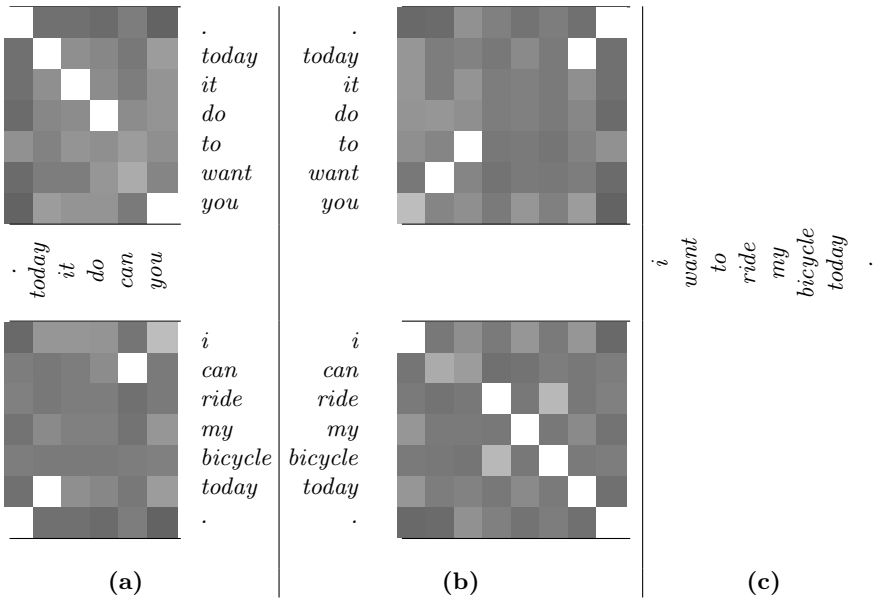

Fig. 2. Soft alignment matrices for analogies between sequences of words (example from Sect. 3.5). A cell in a matrix is the distance between the two corresponding words (arccos of the cosine of their vectors in the word embedding space), the closer the whiter. The two matrices in **(a)** are computed from three given sentences. How to compute the matrices in **(b)** from the matrices in **(a)** is an open problem. The solution of the analogy in **(c)** should be computed from the matrices in **(b)**.

4.2 Enriching the Case Representation

The approach of [14] shows that "raw cases", i.e. pairs of translated sentences only, can already be effective. However, additional pieces of information to a case

can improve its re-usability through CBR. In particular, linguistic information can be used. For instance, parts of speech or morphological features like verb tenses (see an example in Section 4.3) or the fact that a sequence of characters constitutes a noun in singular form (example in Section 4.5), can be used.

This is related to the various steps of the CBR process and raises several issues. In particular, the following question is worth studying, from a case acquisition effort perspective: when is it more beneficial to (manually or automatically) acquire additional information on cases instead of acquiring new raw cases?

4.3 Taking into Account Domain Ontology

The domain ontology (or domain knowledge) is used for several purposes. First it expresses a vocabulary in which cases can be expressed. For example, in a cooking application (such as Taaable [8]), queries, cases and other knowledge units are expressed with terms like `citrus_fruit`, `lemon`, etc. Here, this would be the vocabulary used to represent features of the cases in an enriched representation.

Second it expresses *integrity constraints* about this vocabulary. In the cooking application, this could be for example $\varphi = \texttt{lemon} \Rightarrow \texttt{citrus\_fruit}$.[1] Indeed, φ can be read as the integrity constraint "There is no recipe with lemon and without citrus fruit.", formally: "$\varphi \wedge \texttt{lemon} \wedge \neg \texttt{citrus\_fruit}$ is insatisfiable." Back to the machine translation application, what could be such an integrity constraint and how could it be used to better solve translation problems? One possible answer is the use of linguistic knowledge about the destination language that will recognize that a sentence is not correct (because of a non existing word or because of an ungrammatical construction); this can be used to simply rank the possible solutions. For example, the sentence *I have gone.* should be preferred to *I have goed.*, if both are produced. Standard NLP techniques would, e.g., determine that future tense is used consistently in both languages, so that translation of future tense into future tense should be preferred. For example, the French sentence *Je me lèverai.* (future tense) could be translated into *I will get up.* and *I am going to get up.* According to this criterion, the first sentence could be preferred to the second one though, in fact, both translations are acceptable here. This is why the less preferred solution should be given a lower rank but not necessarily discarded.

What are the NLP techniques case-based translation using analogies can profit from when it is applied to less-resourced languages? E.g., when not enough data is available to build reliable N-gram language models, can linguistic knowledge like incomplete linguistic parsers help to efficiently rank the possible translations? Can case-based translation using analogies identify lacking linguistic descriptions of less-resourced languages?

[1] In this section, the ideas are illustrated in propositional logic, but could easily be expressed in other formalisms.

4.4 Taking into Account Retrieval Knowledge

In a standard CBR setting, the complexity of retrieval is typically $O(n)$ to pick up the most similar case, or $O(n \times \log n)$ to sort the entire case base, of size n, by similarity to the target problem. For the model described in Section 3.4, the complexity of retrieval is the price to pay for the lightness of knowledge and the null cost of adaptation: the previous complexities become $O(n^3)$ or $O(n^3 \times \log n^3) = O(n^3 \times \log n)$. In the case of the approach described in Section 3.6, this complexity is quadratic or more, because a recursive application of the CBR process has a cost.

A way to reduce the computational cost of retrieval is to explicitly compile retrieval knowledge in advance, e.g., in the form of analogical clusters, i.e., series of source pairs which stand for the same transformation. For instance (English meaning below French):

Il peut faire ça aujourd'hui. : *Il veut faire ça.*
'He can do that today.' 'He wants to do that.'
Je peux le faire aujourd'hui. : *Je veux le faire.*
'I can do it today.' 'I want to do it.'
Tu peux la voir aujourd'hui ? : *Tu veux la voir ?*
'Can you see her today?' 'Do you want to see her?'

This technique has never been applied to example-based machine translation, but it has been used for statistical machine translation to create new pairs of aligned sentences (in CBR terms, source problems and their solutions) so as to augment the training data (the case base in CBR terms) [41]. Analogical clusters identify well attested transformations, which should thus be reliable. It is then possible to choose to generate new source cases $\texttt{srce}_C$ from such clusters only, by simultaneously solving all possible analogies formed by the set of the case pairs in a given cluster in conjunction with `tgt`, as illustrated below (English meaning of `tgt` below: 'I want to ride a bicycle').

$\texttt{srce}_B$	:	$\texttt{srce}_A$	::	`tgt`	:	$\texttt{srce}_C$
Il veut faire ça.		*Il peut faire ça aujourd'hui.*				
Je veux le faire.	:	*Je peux le faire aujourd'hui.*	::	*Je veux faire du vélo.*	:	x
Tu veux la voir ?		*Tu peux la voir aujourd'hui ?*				

leads to x = *Je peux faire du vélo aujourd'hui.*
'I can ride my bicycle today.'

During retrieval, computing the similarity of `tgt` to each sentence in the case base reduces the cost of retrieval to $O(n)$ or $O(\log n)$. The most similar cases can lead directly to the clusters they belong to using an inverse index. This avoids redundancy firstly in the retrieval of source case pairs, secondly in the generation of $\texttt{srce}_C$, and thirdly in adaptation, because the generation of the same $\texttt{srce}_C$ is factored once in comparison with several generations in the absence of clusters. This should thus considerably speed up the overall process.

As enriching not only the case base but also the retrieval knowledge should be an essential feature of a knowledge-intensive CBR approach, the question of

managing the dynamic aspect of retrieval knowledge in the recursive application of the CBR process (Sect. 3.6) is a challenging question: when and how should new retrieval knowledge be compiled and added to profit from new cases of the type $(\texttt{srce}_C, \texttt{sol}(\texttt{srce}_C))$ that are added to the case base along the recursive application of the CBR process?

4.5 Using Adaptation Knowledge

The proportional analogy on strings defined in Section 3.1 is used for adapting three cases in order to solve a target problem. It covers a wide variety of situations. However, some situations that are recognized as analogies are not covered by it, hence the usefulness of defining specific edit operations or even specific edit distances for specific languages, constituting therefore new adaptation knowledge.

Let us exemplify with the case of marked plural forms of nouns in Indonesian or Malay. Marked plurals are expressed by repetition: the marked plural form of a noun w is w-w. For the sake of simplicity, let us express the case using English: the marked plural of *cat* (*several cats*) would be *cat-cat*. Therefore, in such languages, the analogy $A : B :: C : D$ between the following strings makes sense:

$$\begin{array}{ll} A = \textit{I like this cat.} & B = \textit{She likes cat-cat.} \\ C = \textit{I avoid this dog.} & D = \textit{She avoids dog-dog.} \end{array} \tag{1}$$

The definition of proportional analogy of commutation presented in Section 3.1 does not cover this case, because, e.g., $|B|_t - |A|_t = 2 - 2 \neq |D|_t - |C|_t = 0 - 1$. In order to take this phenomenon into account, the idea is to define an edit distance `dist` whose edit operations are the ones of the LCS distance, plus an edit operation `repeat_noun` that would replace a substring w that is recognized as a noun in singular form, with its marked plural form w-w (in a manner reminiscent of what was done for consonant spreading in Arabic in the framework of two-level morphology [3]) and its reverse edit operation, replacing w-w with w. Each of the above edit operations should be assigned a cost of 1. Another change to the proportional analogy of Section 3.1 is the fact that the number of occurrences $|S|_c$ of character c in string S is considered only for the strings obtained by removing the nouns w and w-w involved in the computation of the two new edit operations. With these changes the sentences in (1) are in analogy.

Another language-dependent procedure that can be integrated in the adaptation process is the use of correction techniques in the destination language. For example, for $\texttt{tgt} = \textit{As-tu mangé une orange ?}$, a retrieved triple of source cases can be

$$(\texttt{srce}_A, \texttt{sol}(\texttt{srce}_A)) = (\textit{J'ai sauté au plafond.}, \textit{I hit the roof.})$$
$$(\texttt{srce}_B, \texttt{sol}(\texttt{srce}_B)) = (\textit{J'ai mangé une orange.}, \textit{I ate an orange.})$$
$$(\texttt{srce}_C, \texttt{sol}(\texttt{srce}_C)) = (\textit{As-tu sauté au plafond ?}, \textit{Did you hit the roof?})$$

This leads to the proposed solution $\texttt{sol}(\texttt{tgt}) = \textit{Did you ate an orange?}$, which is incorrect. A spell-checker, such as the ones used in some word processors,

can be used to correct `sol(tgt)` in such a situation. It is noteworthy that domain knowledge can play a role in a correction process: linguistic knowledge can be used to examine what makes a sentence incorrect. If such an automatic correction process fails, a human user can correct the sentence, giving birth to a *correction case*: (*Did you ate an orange?*, *Did you eat an orange?*) in the example. Research in correcting SMT errors using, e.g., a NMT system trained on correction cases already exists [21]. Sets of correction cases are already available[2]. But basically, SMT and NMT systems are not traceable, which should be contrasted to case-based translation systems using analogies: used cases can easily be traced and the adaptation and correction knowledge is explicit.

This last topic is directly of interest to the CBR community: can we implement MT systems which are true explainable AI systems, i.e., systems where human-readable linguistic knowledge is easy to integrate and leads directly to visible improvement and where translation results can be intuitively explained?

5 Conclusion

The application aimed in this paper is machine translation (MT), especially MT for language pairs for which corpora of examples are small, relatively to the size of the corpora used in nowadays neural network approaches to MT. Indeed, it is our working hypothesis that case-based MT is competitive in such a context. This hypothesis is based on the prior work of [14] that is reformulated here in terms of case-based MT. This reformulation constitutes the first contribution of this paper. This approach is knowledge-light in the sense that the only language-dependent pieces of knowledge are the cases, which are raw cases, representing only the problem and the solution (the sentences in the origin and destination languages), without any additional information. This approach is based on the transfer of proportional analogies found in the origin language onto the destination language.

The second contribution of this work is a theoretical examination of the question: "How can this knowledge-light case-based MT approach be improved by incorporating some new pieces of knowledge and other principles, methods, and techniques from CBR or NLP?" Obviously, the answers given in this paper are at an embryonic stage. Therefore, future directions of work are obvious: implementing and testing the ideas presented and developing new ideas for knowledge-intensive case-based MT.

Our impression is that this issue of case-based MT, though little explored nowadays, deserves much research: there are certainly many ways to improve it and it is worth doing so with the aim of developing competitive MT systems that are not limited to pairs of languages with very large corpora. One way, still under investigation, is to continue this research through a contest, similar to the Computer Cooking Contest. Such contests already exist for MT, but they

[2] E.g., `https://www.matecat.com/`. The authors of this paper are currently working on a slightly different scenario and are collecting such correction cases for use in a case-based correction system.

focus on very large corpora. The idea would be to organize such a contest on smaller corpora like the small ones offered by the Tatoeba project[3] and to use the off-the-shelf automatic evaluation techniques of the MT community.

References

1. Aamodt, A., Plaza, E.: Case-based reasoning: Foundational issues, methodological variations, and system approaches. AI Commun. **7**(1), 39–59 (1994)
2. Backurs, A., Indyk, P.: Edit distance cannot be computed in strongly subquadratic time (unless SETH is false). In: STOC'15. pp. 51–58. ACM (2015)
3. Beesley, K.R.: Consonant spreading in Arabic stems. In: COLING-ACL'98. vol. I, pp. 117–123. Montréal (1998)
4. Boitet, C.: Current state and future outlook of the research at GETA. In: MT Summit I. pp. 26–35. Hakone (1987)
5. Brown, P., Cocke, J., Pietra, S.D., Pietra, V.D., Jelinek, F., R, M., Roossin, P.: A statistical approach to machine translation. In: COLING 1988. pp. 71–76 (1988)
6. Bunke, H., Messmer, B.: Similarity measures for structured representations. In: S. Wess, K.-D. Althoff, M.M.R. (ed.) Topics in case-based reasoning. EWCBR-93. LNAI, vol. 1168, pp. 106–118. Springer, Berlin (1993)
7. Cieri, C.: Addressing the language resource gap through alternative incentives, workforces and workflows (invited keynote lecture). In: LTC'17. Poznań (2017)
8. Cordier, A., Dufour-Lussier, V., Lieber, J., Nauer, E., Badra, F., Cojan, J., Gaillard, E., Infante-Blanco, L., Molli, P., Napoli, A., Skaf-Molli, H.: Taaable: a Case-Based System for personalized Cooking. In: Montani, S., Jain, L.C. (eds.) Successful Case-based Reasoning Applications-2, Studies in Computational Intelligence, vol. 494, pp. 121–162. Springer (2014)
9. Gentner, D.: Structure mapping: A theoretical model for analogy. Cognitive Science **7**(2), 155–170 (1983)
10. Johnson, M., Schuster, M., Le, Q.V., Krikun, M., Wu, Y., Chen, Z., Thorat, N., Viégas, F.B., Wattenberg, M., Corrado, G., Hughes, M., Dean, J.: Google's multilingual neural machine translation system: Enabling zero-shot translation. CoRR (2016)
11. Kaveeta, V., Lepage, Y.: Solving analogical equations between strings of symbols using neural networks. In: Computational Analogy Workshop at ICCBR-16. pp. 67–76. Atlanta, Georgia (2016)
12. Kitano, H.: Challenges of massive parallelism. In: IJCAI'93). IJCAI'93, vol. 1, pp. 813–834. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1993)
13. Koehn, P.: Europarl: A Parallel Corpus for Statistical Machine Translation. In: MT Summit X. pp. 79–86. Phuket (2005)
14. Lepage, Y., Denoual, E.: Purest ever example-based machine translation: detailed presentation and assessment. Machine Translation **19**, 251–282 (2005)
15. Lieber, J., Napoli, A.: Using classification in case-based planning. In: Wahlster, W. (ed.) ECAI 96. pp. 132–136. John Wiley & Sons, Ltd. (1996)
16. Lison, P., Tiedemann, J.: OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In: LREC 2016. Paris, France (2016)
17. Ma, W., Suel, T.: Structural sentence similarity estimation for short texts. In: FLAIRS-29. pp. 232–237 (2016)

[3] `https://tatoeba.org/`

18. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. CoRR **abs/1301.3781** (2013)
19. Nagao, M.: A framework of a mechanical translation between Japanese and English by analogy principle. Artificial & Human Intelligence pp. 173–180 (1984)
20. Nakazawa, T., Kurohashi, S.: EBMT system of KYOTO team in patentMT task at NTCIR-9. In: NTCIR-9. pp. 657–660. Tokyo, Japan (2011)
21. Pal, S., Naskar, S.K., Vela, M., van Genabith, J.: A neural network based approach to automatic post-editing. In: ACL'16. pp. 281–286 (2016)
22. Pham, H., Luong, T., Manning, C.: Learning distributed representations for multilingual text sequences. In: 1st Workshop on Vector Space Modeling for NLP. pp. 88–94. Denver, Colorado (2015)
23. Řehůřek, R.: Making sense of word2vec, available on line
24. Richter, M.: Knowledge containers (2003), available on line
25. Riesbeck, C.K., Schank, R.C.: Inside Case-Based Reasoning. Lawrence Erlbaum Associates, Inc., Hillsdale, New Jersey (1989), available on line
26. Rougegrez-Loriette, S.: Prédiction de processus à partir de comportement observé : le système REBECAS. Ph.D. thesis, Université Paris 6 (1994)
27. Sadler, V., Vendelmans, R.: Pilot implementation of a bilingual knowledge bank. In: COLING-90. vol. 3, pp. 449–451. Helsinki (1990)
28. Sato, S.: Example-based Machine Translation. PhD thesis, Kyoto University (1991)
29. Schuster, M.: The move to neural machine translation at Google (invited talk). In: IWSLT 2017 (2017)
30. Smyth, B., Keane, M.T.: Remembering To Forget. In: IJCAI'95. vol. 1, pp. 377–382. Montréal (1995)
31. Stahl, A., Bergmann, R.: Applying recursive CBR for the customization of structured products in an electronic shop. In: EWCBR 2000. pp. 291–308 (2000)
32. Takezawa, T., Sumita, E., Sugaya, F., Yamamoto, H., Yamamoto, S.: Toward a broad coverage bilingual corpus for speech translation of travel conversation in the real world. In: LREC 2002. pp. 147–152. Las Palmas (2002)
33. Tiedemann, J.: News from OPUS - A collection of multilingual parallel corpora with tools and interfaces. In: RANLP-09, vol. V, pp. 237–248. John Benjamins, Borovets, Bulgaria (2009)
34. Turney, P., Pantel, P.: From frequency to meaning: Vector space models of semantics. J. of Artificial Intelligence Research **37**, 141–188 (2010)
35. Turney, P.D.: The latent relation mapping engine: Algorithm and experiments. J. Artif. Int. Res. **33**(1), 615–655 (2008)
36. Turney, P.D., Littman, M.L.: Corpus-based learning of analogies and semantic relations. Machine Learning **60**(1–3), 251–278 (2005)
37. Ukkonen, E.: Algorithms for approximate string matching. Information and Control **64**, 100–118 (1985)
38. Wagner, R.A., Fischer, M.J.: The string-to-string correction problem. Journal of the Association of Computing Machinery **21**(1), 168–173 (1974)
39. Weaver, W.: Translation. Tech. rep., The Rockfeller Foundation, New York (1949)
40. Weber, R.O., Ashley, K.D., Brüninghaus, S.: Textual case-based reasoning. The Knowledge Engineering Review **20**(3), 255–260 (2005)
41. Yang, W., Shen, H., Lepage, Y.: Inflating a small parallel corpus into a large quasi-parallel corpus using monolingual data for Chinese–Japanese machine translation. Journal of Information Processing **25**, 88–99 (2017)
42. Ziemski, M., Junczys-Dowmunt, M., Pouliquen, B.: The United Nations Parallel Corpus v1.0. In: LREC 2016. Paris, France (2016)