Non-hybrid Example-based Machine Translation Architectures

Daniel Jones

Centre for Computational Linguistics
UMIST
Manchester
UK
danny@ccl.umist.ac.uk

Abstract

A general definition of rationalist and empiricist natural language processing is attempted. A classification of empiricist machine translation systems is given based on the rationalist/empiricist distinction. Examples of approaches falling into the two different strategies are discussed. Research results are reported from attempts to break new ground in what is referred to as "pure" or non-hybrid example-based machine translation.

Keywords: Rationalist; Empiricist; Analogical; Example-based; Translation.

1. Introduction

There has, in recent years, been an increase in the amount of interest shown (at least in the field of Machine Translation) in what has been called example- or memory-based translation. Advocates of this new approach say that this is because of the advantages that analogy-based processing offers e.g. robustness, ease of augmentation, accuracy, etc.
Although it is possible to get an intuitive feel for what example-based approaches entail, a more accurate picture of the fundamental issues involved can be gained by contrasting the two approaches to language processing by introducing the concepts of rationalism and empiricism. The following quotation from Connolly [4] introduces some important concepts in understanding the critical differences between what can be referred to as rule-based and example-based language processing.

> When linguists describe languages, they naturally employ the technical apparatus of descriptive linguistics .. .they attempt to capture structural and functional regularities in the language they are investigating ... analysing data so as to reveal the regularities displayed therein .. .and setting the results down in the form of explicit statements
> .. These .. .data may be generated by linguists themselves .. .and/or the data may be obtained from other informants, [p. 222]

Connolly describes this process as externalising the internal linguistic competence of language users and also observes the following characteristics of this pursuit summarised briefly below:

- The externalised knowledge is incomplete.
- The externalised and internalised knowledge are not isomorphic i.e. the methods of representation of the two are different. Not only is internalised knowledge stored differently - it would appear from psycholinguistic experiments that there is a certain amount of

redundancy in the way the brain stores linguistic information -- it is assumed that the highly technical frameworks which contemporary linguists employ are not used in the human brain itself.

- Externalised knowledge is open to inspection, in contrast to internalized knowledge.

It is also noted that:

> The internalised competence is knowledge *of* a language, whereas the explicit, externalised description represents knowledge *about a* language. In other words, the externalised description is a direct embodiment not of *linguistic* knowledge as such, but of *metalinguistic* knowledge. [ibid].

If we assume that the same principles of externalisation apply to *computational* linguistics, then we have a definition of a *rationalist* approach to language processing i.e. the goal of the computational linguist is to discover the internalised knowledge of language users, and represent it in a formal metalinguistic manner[1] with the aim of making the representation computationally tractable. Examples of these representations would be grammars written employing, for example, Government Binding Theory[3], Lexical Functional Grammar[1], Generalized Phrase Structure Grammar[6], etc.

A definition for *empiricist* approaches is also implicitly introduced by Connolly. If the method of storing and representing linguistic knowledge is non-isomorphic with respect to rationalist metalinguists, then, perhaps, an empiricist approach would attempt to eradicate or at least reduce this asymmetry. The empiricist view of language might then be encapsulated by a viewpoint which states that the externalisation process is at best unnecessary and, at worst, wrong. As language in its written and spoken forms contains all the information necessary for language users (human beings) to analyse, generate, and translate, the job of the empiricist is merely to describe what is there. In other words, the major difference between a rational and empiricist approach is the latter does not attempt to create a metalinguistics.

2. Rule-based versus Example-based Processing

The above definition of empiricist methodology is a rather extreme one. The suggestion that empirical (computational) linguists deny the need for a metalinguistic framework leads to some interesting questions e.g. Where are the rules[2] which allow the processing to occur? How do we know how to manipulate the linguistic knowledge we have analysed and described? The central concern here is the use of rules and it is this feature which characterises the two fundamentally different approaches to the processing of natural language. Rationalism's use of metalinguistics is typically realised by the use of rules to predict and determine how language can and should be analysed and generated.[3] In contrast, empiricist (or what can also be called example/memory-based) approaches do not seek to use rules as a *necessary* feature of the linguistic knowledge of the system.

3. Types of Empirical Natural Language Processing

Example-based systems do not reject rules out-of-hand. Indeed rules are frequently employed to some degree or other in systems which can be classified as example-based. However, the important point is that they *may not be used.* The distinction can be clarified with reference to existing systems which fall within the example-based paradigm.

[1] The formal methodology for this process is, of course, heavily influenced by the seminal work of Chomsky.

[2] Where "rule" is used as a general pattern-and-associated-action configuration. The predictive and explanatory capacity of such formulations provide much of the underpinning generality of many contemporary natural language processing systems.

[3] The fact that these rules may have been derived by corpus analysis is irrelevant because the fact remains that the rules themselves (regardless of their origin) are the main embodiment of both linguistic and executive knowledge.

3.1 Hybrid Example-based Systems

A hybrid example-based system is one where some element or module of the overall architecture employs non-rule-based processing. Sumita et al.[l4] describe the use of a module to translate noun/preposition/noun constructs from Japanese into English. They state that although constructing a rule or rules to do this task would appear to be straightforward, in reality, the task is quite difficult for a rule-based approach. Consequently, they suggest the use of an example-based method [4] to execute this sub-task as its performance is clearly superior to the deployment of "conventional" transfer rules. Sumita et al.'s belief is that:

> ... it is not yet clear whether EBMT [Example-Based Machine Translation] can/should deal with the whole process of translation. We assume that there are many kinds of phenomena; some are suitable for EBMT and others are not. Thus, it is more acceptable for users if RBMT [Rule-Based Machine Translation] is first introduced as a base system which can translate totally, then its translation performance can be improved incrementally by attaching EBMT components as soon as suitable phenomena for EBMT are recognized. [p. 211]

This is the clearest form of hybrid system but there are degrees to which rule and example-based approaches can be mixed. Sumita et al.'s approach gives more emphasis to rule-based as opposed to example-based algorithms whereas Sato & Nagao[12] suggest a reversal of this emphasis by suggesting that, for translation purposes at least, example-based processing should, in fact, deal with the *whole process* of translation. They propose a system based on pairs of translation equivalents i.e. whole sentences stored as dependency trees. The task of the system is to identify which translation units (subtrees) can be used as matching points with the input. The concept of a restricted (syntactic) environment[5] is used to calculate the most suitable matching points across the example data set which is defined as:

> .. .the summation of the similarity values [taken from the thesaurus] between corresponding nodes in two restricted environments at the best matching. [Sato & Nagao, p.250]

In suggesting that example-based methods can be used autonomously, Sato & Nagao realise that there is a need to *recombine*[6] parts of translation examples:

> The ability to combine some fragments of translation examples is essential to example-based translation. A lack of this ability restricts the power of example-based translation, [ibid., p.247]

Recombination can be used for monolingual as well as bilingual language processing. Monolingual recombination processing will be used purely to gain the highest degree of matching with the input, a process which can be regarded as one of *cloning.* As the aim of the analysis phase is to map the required information from the described examples onto the input, the flow of information is from the dataset of language examples to the input. If the dataset of examples is regarded as not a static set of discrete entities but a permutable and flexible interactive set of process modules, we can envisage a control architecture where each process (example) attempts to clone itself with respect to (parts of) the input. This cloning process may indeed involve some degree of recombination. In a monolingual scenario the result of the cloning process may be some (quasi) logical form for database query purposes. In a bilingual environment, instead of a logical form, we would generate a translation.

[4] A distance measure is used to determine the best translation candidates derived from a parallel corpus. A thesaurus is also used in order to capture lexical similarities if identical matching is not possible.

[5] This consists of nodes in the dependency tree one mother node up from that of the translation unit itself.

[6] The term "recombination" is used as it is assumed that the examples under analysis are already combined in some sense. Hence, a process of permutation would, strictly speaking, be re-activating a previously executed process.

3.2 Pure Example-based Systems

In contrast to what have been called hybrid example-based systems, there are approaches which can be regarded as coming as close as possible to employing wholly analogical methods. The most well-known of these are connectionist models[7].

It is clear why connectionist models should be regarded as pure example-based systems whether designed for processing natural language or otherwise. Because the connectionist network is trained on a succession of inputs and their corresponding outputs i.e. what the model should produce when presented with that input, the model learns by example how to deal with new inputs once the training stage has been completed. Hence, the "judgements" such a model makes about any input received are solely based on examples the model has seen before. Recent research has shown that such models can be used successfully for analysis purposes. McClelland & Kawamoto[9] have demonstrated the effectiveness of a model trained to assign case roles to sentential constituents and Jain[7] has developed a system which:

> ... *learns* to parse complex sentences presented one word at a time by acquiring a statistical grammar based on a combination of semantic and syntactic clues. [p. 111] (emphasis original)

Both active and passive sentences are dealt with as well as centre-embedded constructions. Another "pure" approach to processing natural language by example is offered by Skousen's "Analogical Modeling"[13]. In a similar fashion to connectionism, Skousen advocates using a network of examples. Each example is represented as a set of features which represent characteristics of the language segments ranging from phonemic units through to sociolinguistic factors of age and social status. Each example has an associated "outcome" which can be regarded as the functional effect of applying an example in a particular context e.g. a voiced/voiceless distinction with respect to a "phonetic" context. The dataset is constructed from a network of the examples (with their outcomes) with pointers from each outcome to every other outcome in the dataset. When an input is received, the analogical processing attempts to find which examples across the dataset have the greatest *analogical effect* on the input. These examples are assigned a percentage probability of analogical effect so that the results can be scored.

Skousen notes the similarity between connectionism and analogical modelling stating that:

> Both approaches dispense with the need for rules, yet still account for "rule governed" behavior. They can both predict behavior when the data is ill-formed or missing crucial information. [ibid., p. 81]

However, it is clear that an analogical model does not go through a competitive learning procedure and it has more of the feel of a conventional linguistic database from which appropriate examples can be directly extracted.

The reason both connectionism and analogical modelling can be thought of as embodying pure example-based architectures is not that they deny the use of rules during processing (any algorithm has to be expressed in terms of rules). Rather, it is the assumptions of where the knowledge comes from to make a descriptive judgement about a given piece of natural language. This difference can be stated informally as follows:

1. Rule-based maxim: Language behaviour can be described by a set of discrete metalinguistic rules. Language can therefore be processed directly by the *compositional* application of these discrete rules.
2. Example-based maxim: Language behaviour can be described by a set of appropriately represented examples of language which *compete together as a whole* to "passively" imbue their associated descriptions onto some language fragment.

Although research in connectionism and analogical modelling has not, until very recently, been carried out in the field of machine translation, research carried out by Brown et al.[2] has

[7] Also referred to as Parallel Distributed Processing (PDP) or Neural Networks.

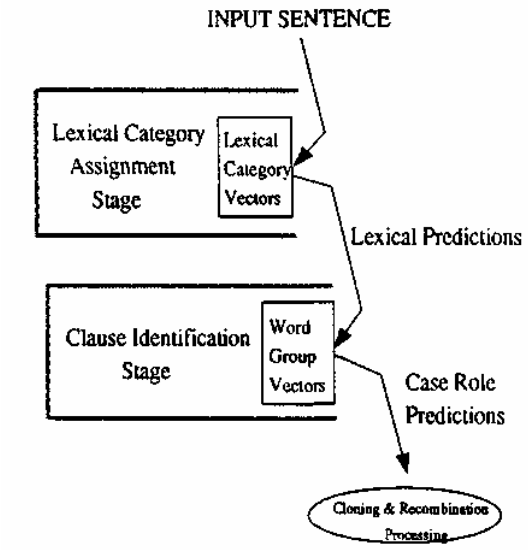
demonstrated how effective parallel examples of text can be in translating between two languages. By aligning sentences of a bilingual corpus it is possible to predict which elements of a source text input are most likely to be translates of the target language:

> We assign to every pair of sentences *(S,T)* a probability, *Pr(T|S),* to be interpreted as the probability that a translator will produce T in the target language when presented with S in the source language. We expect *Pr(T|S)* to be very small for pairs like *(Le matin je me brosse les dents | President Lincoln was a good lawyer)* and relatively large for pairs like (Le *president Lincoln était un bon avocat |\ President Lincoln was a good lawyer).* [ibid., p. 79]

4. Pure Example-Based Machine Translation

It is interesting to note that systems which I have classified as *hybrid* have been designed with the aim of translating between languages whereas the *pure* approaches (apart from Brown's work) have been restricted to monolingual processing tasks.
There is a suspicion that the performance of a purely statistically-based translation system would improve with the inclusion of linguistic descriptions or more generally abstract information about linguistic function. As the other purely stochastic techniques, i.e. connectionism and analogical modelling, rely on descriptive detail as a fundamental requirement, it would be interesting to see how these techniques perform with respect to the task of translation.

Figure 1: Cascaded Analysis Architecture



4.1 Rule-less Analysis

Experiments with the analysis phase of the translation task have been attempted without conventional rule-based parsing algorithms by the author [8]. Analogical Modelling was used to provide an example-based machine translation system with a means to measure the probabilistic distance between an input text and the set of translation examples.
A cascaded analysis architecture was used to successively pass along the results of different levels of analysis until a sufficiently detailed amount of information had been obtained for the "transfer" stage. The overall concept of the system is similar in nature to that proposed by Nagao[10], (and later Sumita & Nagao[12]) as well as Sadler[11], but the emphasis here is on non-rule-based analysis. Previous research has tended to assume some parsing process would

be used which may well have been rule-based. The aim in these experiments was to attempt to be as strictly example-based as possible.
There are two stages through which the input (in this case a sentence) must pass -- the lexical category assignment stage and the clause identification stage. See Figure 1.
The examples for the lexical category assignment task were derived from a small number of business letter texts by the following means. Sets of three consecutive words at a time were taken from the corpus. Each triplet can be represented as *(W1, T, W2)*. The last three letters from W1, the last four letters from T, and the first three letters from W2 were taken to form a vector of ten characters. The *outcome* for this vector was the actual lexical category of T in its context of occurrence giving the following representation[8]:

$$[char_1, char_2, \ldots, char_{10}] \rightsquigarrow Lexical\ Category \quad (1)$$

If any of W1, T, or W2 had too few characters, the missing characters were signalled by a null character in the example vector. The lexical categories used to describe both the outcomes of the lexical category predictions and the representation of word group vectors are given below (see Tables 1-4):

Table 1: Pronoun Descriptions

NUMBER	REPRESENTATION
singular	Pro+s
plural	Pro+p

Table 2: Determiner Descriptions

DEFINITENESS	REPRESENTATION
indefinite	Det+idef
definite	Det+def

Table 3: Noun Descriptions

NUMBER	(NON)HUMAN	EXAMPLE	REPRESENTATION
singular	human	"girl"	N+s+h+c
plural	non-human	"dogs"	N+p+h+a

The lexical category data was constructed semi-automatically. Plain ascii text files were automatically segmented into the word triples. As the outcomes of these particular example vectors were lexical categories, i.e. information not already available in the non-tagged corpus, in order to maintain accuracy of labeling they were entered by hand (although this process, too, could be automated with an automatic tagger). However, as the number of examples in the experiment was small (around 250), the overhead of semi-automatic construction was thought to be acceptable. The examples had, at some point, to be encoded in 'C' but this was achieved by a compiler compiler thereby divorcing the database creator from low-level implementation issues. In contrast to the creation of lexical examples, word group vectors were constructed manually. Texts were manually segmented and labeled for appropriate outcomes. This was the most time-consuming part of example creation.
Even though the scale of the experiments was small, the results were surprisingly good. In most cases, the correct lexical assignments were made to the words in input sentences. Also of interest was the fact that correct "guesses" were made about words which the analogical model did not have as direct examples in its dataset.[9]

[8] Where the symbol "$\rightsquigarrow$" means "can lead to" or "can represent".

[9] The ability to make best guesses when receiving noisy or ill-formed input is a major feature of these types of example-based system.

Table 4: Preposition Descriptions

FUNCTION	EXAMPLE	REPRESENTATION
Source	"from"	Pre+source
Location	"at"	Pre+location
Direction	"to"	Pre+direction
Quality	"with"	Pre+quality
Time	"by"	Pre+time
Reason	"for"	Pre+reason

Some results from lexical matching are shown below.

Table 5: Lexical Assignments for "You can import widgets from America"

INPUT WORD	PROBABLE CATEGORY	BEST PROBABLE "DISTANCE"
You	Pro+s	100%
can	Modaux	95%
import	Vfu+a	43%
widgets	N+p+nh+c	35%
from	Pre+so	95%
America	Moduax	50%
"	Adj	14%
"	N+s+nh+c	11%
"	Pro+s	11%
"	Det+def	11%

Interestingly, The system proposes (guesses) that the lexical item "widgets" is a plural, non-human, count noun (N+p+nh+c) even though this word is unknown to the system. The ability of the system to do this is derived from the comparison of subparts of all examples along with their outcomes which weight global probability for a particular outcome. In the case of "widgets", the best guess of the system was based on a number of examples of nouns ending in "-ts" plus the relatively large proportion of nouns present in the database. While this feature of the system gives rise to robustness in that probability of assignment will always be given, if the database is not represented properly the system will become too unconstrained in its predictive ability. Once these assignments had been made to the words in the input string, the newly classified lexical items were passed through to the clause-level network. The same principle of example preparation was used here as with the lexical processing. Phrasal groupings from the sentences of the business letter data were taken, and a corresponding vector was generated consisting of a fixed number of variables signalling the presence or absence of the particular lexical category, e.g. noun, adjective, adverb, etc. The outcome of the vector was the case role of the word group i.e.

$$[lex\_category_1, lex\_category_2, \ldots, lex\_category_n] \rightsquigarrow Case\ Role \quad (2)$$

The input was segmented by the use of a simple heuristic in order to associate word groupings with a level of sentential description based on Functional Grammar (see Dik[5]) predicate frame structures[10]. Word group boundaries were said to occur where the analogical effect for a particular case role or phrasal outcome fell after the inclusion of a lexical item. As words from the input sentence are added to the word group input vector, one particular outcome may prevail but then fail to do so at a given point.

[10] The use of a linguistic formalism for descriptive purposes does not make an example-based system hybrid. It is how the descriptions are used that is crucial in this respect.

This cascaded process was demonstrated with a different dataset derived from a small sample of travel messages. For example, the phrase "No problems reported" will be analysed in three stages i.e. "No", "No problems", and "No problems reported" once the probabilistic lexical assignments have been made 11. See Tables 6 and 7 below.

Table 6: Lexical Category Assignment Results

Word	Outcome probabilities	Individual Example Probabilities
No (No_)	25.000000% : [noun, adj] 50.000000% : [neg]	25.000000% : ["North" , "Normal", "Not" , "No"]
problems (prob)	100.000000% : [noun]	100.000000% : ["problems"]
reported (repo)	100.000000% : [verb]	100.000000% : ["reported"]

Table 7: Clause Identification Results

Word	Outcome probabilities	Individual Example Probabilities
No	33.333332% : [AdjP] 66.666664% : [NP]	33.333332% : ["advise", "divert", "be"]
No problems	100.000000% : [NP]	33.333332% : ["lead_to"] 66.666664% : ["report"]
No problems reported	40.000000% : [PP] 60.000000% : [NP]	25.000000% : ["operate", "require", "be", "divert"]

The third column of tables 6 and 7 shows the origin of the data on which the probable matches are predicted. In table 6, these are lexical items, and in table 7, predicate frames. The names given in table 7 are the predicate names which govern the example word groupings.
The experiment shows the probability for the outcome NP rising until "reported" is encountered giving the structure [[No problems] reported] to the input.
The example predicate frame is chosen based on the lexical item predicted to be the main verb of the input sentence by the lexical category prediction phase i.e. "reported" (see table 6). If the predictions of the analogical matching between proposed word groups and current example predicate frame are the same for all frame slots then that example predicate frame (source language part) would have a high analogical effect on the input sentence. The most likely predicate frame associated with the word grouping "No problems" is "require" which corresponds with the proposed lexical category prediction for the main verb. There is an exact correspondence between the two levels of processing and the relevant examples can be said to have cloned well with respect to the input.

5. Summary

The main distinction between rationalist and empiricist approaches to machine translation (and natural language processing in general) is that rationalists attempt to build a metalinguistic model which is supposed to capture and predict (as elegantly as possible) all possible "legal" expressions in a given language. The empiricist assumption is that the construction of a suitable executive mechanism plus the appropriate description of real (as opposed to invented) language data is a better way of approaching the complex issues surrounding human language. A distinction, based on this premise, has been discussed which differentiates between two types of empiricist (or example-based) MT systems i.e., hybrid and pure. Hybrid systems employ a

[11] The lexical category assignments shown in Table 6 were achieved with different data than that shown in Table 5. Rather than a (W1, T, W2) configuration, isolated words were used with only the first N letters retained. The experiment shown in Table 6 had N equal to 4.

mixture of rationalist (or rule-based) assumptions, representations, and algorithms along with example-based modules. Pure systems reject any rule-based processing. It is noted that those approaches which can be characterised as pure have not been used for translation purposes. Consequently, research experiments have been reported which have explored the use of one of these approaches (analogical modelling) in the analysis phase of a transfer-based MT architecture.

References

[1] Bresnan, J. *The Mental Representation of Grammatical Relations.* MIT Press, 1982.

[2] Brown, P., Cocke, J., Della Pietra, S., Della Pietra, V., Jelinek, F., Lafferty, J.D., Mercer, R., and Roossin, P. A Statistical Approach to Machine Translation. *Computational Linguistics,* Volume 16(Number2):79--85,1990.

[3] Chomsky, N. *Lectures on Government and Binding: The Pisa Lectures. Studies in Generative Grammar.* Number 9. Foris Publications, 1981.

[4] Connolly, J. H. and Dik, S. C., editors. *Functional Grammar and the Computer.* Functional Grammar Series Number 10. Foris Publications, 1989.

[5] Dik, S. *Functional Grammar.* North-Holland Linguistic Series. North-Holland, 1978.

[6] Gazdar, G., Klein, E., Pullum, G.K., and Sag, I.A. *Generalized Phrase Structure Grammar.* Blackwell, 1985.

[7] Jain, A.N. Parsing Complex Sentences with Structured Connectionist Networks. *Neural Computation,* 3(1):110-120,1991.

[8] Jones, D.B. *The Processing of Natural Language by Analogy with Specific Reference to Machine Translation.* PhD thesis, The University of Manchester Institute of Science and Technology, 1991.

[9] McClelland, J.L. & Kawamoto, A.H. Mechanisms of Sentence Processing: Assigning Roles to Constituents. In McClelland, J.L. & Rumelhart, D.E., editor, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition,* volume 2: Psychological and Biological Models, pages 272-333. MIT Press, 1986.

[10] Nagao, M. A Framework of a Mechanical Translation between Japanese and English by Analogy Principle. In A. Elithorn, editor, *Artificial and Human Intelligence.* Elsevier, 1984.

[11] Sadler, V. The Textual Knowledge Bank: Design, Construction, and Applications. In *Proceedings of International Workshop on Fundamental Research for the Future Generation of Natural Language Processing,* ATR Telephony Research Labs, 1991.

[12] Sato, S. and Nagao, M. Towards Memory-based Translation. In Hans Karlgren, editor, *Proceedings of COLING 90,* Helsinki, 1990. University of Helsinki. Vol. 3.

[13] Skousen, R. *Analogical Modeling of Language.* Kluwer Academic Publishers, 1989.

[14] Sumita, E., Iida, H., and Kohyama, H. Translating with Examples: A New Approach to Machine Translation. In *Proceedings of the Third International Conference on Theoretical and Methodological Issues in Machine Translation of Natural Languages,* Linguistics Research Center, University of Texas at Austin, 1990.