Corpus-Based Acquisition of Grammars and Transfer Functions for Machine Translation

Patrick Juola
juola@cs.colorado.edu

CU-CS-756-95 January 1995

University of Colorado at Boulder

Technical Report CU-CS-756-95
Department of Computer Science
Campus Box 430
University of Colorado
Boulder, Colorado 80309

Corpus-Based Acquisition of Grammars and Transfer Functions for Machine Translation

Patrick Juola
juola@cs.colorado.edu

January 1995

Abstract

One major weakness with automatic learning of transfer functions is that the formalisms usually employed are psychologically implausible as well as difficult for a human to understand, while more understandable formalisms are often very difficult to infer empirically. This work describes a set of linguistic universals that are computationally useful for NLP and also empirically discoverable from a large corpus of text.

These universals have been incorporated into a family of systems (METLA) which perform this task on a small-scale in a language-independent fashion. Results for several language pairs and several corpora are presented. These results tend to be good and supportive of the notion of guiding acquisition by psycholinguistics, the validity of the notion of computational accessibility, and of the usefulness of the exact universals employed.

1 Introduction

Machine translation can be viewed as an almost textbook example of a problem where neither the empirical nor the theoretical approaches have been completely successful. In large part, this is due to the tremendous volume and complexity of the data to be described and modeled. Traditional, rule-based systems find it difficult to capture the idiosyncracies and innumerable exceptions, while more data-oriented systems rely on extracting data from huge corpora and are likely to overlook fundamental regularities of the sort easily noticed and identified by linguists. Recent conferences[25, 19, 11] have seen a resurgence of interest in the development of techniques to either analyze and correct the results of such statistical systems, or to provide automatic ways of developing rules, in the hopes of providing systems with greater generalizability, understandability, robustness, and explainability than either approach alone.

Unfortunately, the mathematical formalisms most amenable to statistical parameterization and learning, such as neural networks and Markov chains, are by their very nature difficult for humans to understand and modify. Furthermore, they are sufficiently non-linguistic that it is difficult to apply known theoretical constraints. Many recognized linguistic principles such as compositionality or basic word order[15] are difficult even to describe in terms of such systems. On the other hand, systems that are designed to focus on linguistic universals tend to lose the ability to directly identify those universals for themselves, as the universals are themselves described in high-level linguistic terms that are not easily extracted from simple sentences as found in a corpus.

This work describes an approach to the automatic extraction of transfer functions from an aligned bilingual corpus that relies heavily on low-level universals that can be easily identified and extracted. In theory, by compiling general linguistic principles into the system, the same system can be easily changed to a novel language pair by simply changing the training data. Because the transfer functions are described in linguistically plausible terms, it is easy to modify and correct the learned transfer functions by anyone with knowledge of the relevant languages. Two prototype systems have been built and tested which follow this approach, and the experimental results are presented in sections 9 and 12.

These results indicate that psycholinguistic plausibility need not be a barrier to the empirical acquisition of transfer functions for machine translation or to corpus-based NLP in general. Furthermore, the structure provided by linguistic universals, and particularly the Marker Hypothesis, can add to the efficiency of such programs by restructuring and restricting the search space. Finally the incorporation of such linguistic plausibility adds to the understandability and maintainability of such NLP systems by insuring that any solutions found tend to be describable and understandable in the terms that humans have independently developed to describe and solve such problems.

2 Example-based translation

The problems associated with rule-based machine translation are too well-known to require a detailed description. The reader is referred to [21, 39, 33] for a detailed survey of approaches and problems. In general, the greatest problem is the "Knowledge Acquisition Bottleneck"— the amount of detail it is necessary to represent is both idiosyncratic and too large to comfortably handle. On the other hand, the necessary information can often be found by simply searching through large corpora.

Nagao[32] outlined what would become a major force in machine translation with his proposal of what became known as "example-based translation." Rather then generating explicit rules, translation systems would store a huge collection of translation examples that would provide coverage in context for the input. For a typical input sentence, the system would parse it into its components, examine the database for examples which were semantically similar to the components of the input sentence, then construct the translated sentence by putting together the translations of the input components according to the grammatical rules of the target language.

A simple example of this technique was developed by Sumita et al. [44, 43] and called EBMT (for Example-Based Machine Translation). This system translated only phrases of the form "$noun_1$ NO $noun_2$," where NO is the general partitive adposition in Japanese. In most contexts, it is similar to the English word "of," but can also be translated as "in" (as in "the conference in Kyoto"), "'s" ("the teacher's pencil"), "by," possessive pronouns, and several other potential translations depending upon the nouns surrounding it. Sumita et al. incorporated a commercial thesaurus into their system and imposed a notion of semantic distance upon the classes of potential nouns the system understood. Using 2250 pairs of Japanese phrases and their English translation, the system would select the semantically closest pair to the desired source sentence and translate the source phrase into a structure similar to the matched English phrase.

This method has several distinct advantages over more typical approaches. It is much better at selecting the "correct" translation from a set of synonyms or near-synonyms. It is typically

more robust than a fully rules-based translation system, can use translator expertise well, and can assign a confidence or reliability factor based on the nearness of the semantic matches. On the other hand, it still requires significant skill to develop a translation database, and still more skill to develop a suitable parser, generator, and thesaurus for calculating semantic similarity. There is no easy way for a computer to recognize two words as being semantically similar or distinct, which creates a new knowledge acquisition bottleneck.

A similar corpus-based approach was developed by Brown et al.[3]. Using a bilingual corpus, they attempted to solve the translation problem as a mapping between Markov chains, asserting that every sentence is a possible translation of every other sentence, then calculating the most probable translation from the statistics of the corpus, defined as the product of the probabilities of each word in the source language producing a particular word or set of words in the target language.

The limitations of such a word-by-word translation approach are many and varied. There is no idea of grammar, only a simple Markov chain. There is no context-sensitivity, and no notion of selecting the most appropriate translation from a set of near-synonyms. This method, however, appears to require relatively little human expertise or time to produce its translations.

A similar approach has been used by Koncar[26], who used a large neural network to infer a translation method between a set of English sentences and their Serbo-Croat translations. The advantages and limitations of artificial neural networks are well-known; of particular significance to this work is that, again, they are very weak in their ability to handle grammatical constructions, and it is very difficult to analyze the inferred functions in any sort of linguistically plausible or understandable fashion.

A final example of this sort of data mining can be seen in the work of Jones et al.[42, 20] on the automatic extraction of translation functions by alignment. In many regards, this is the most linguistically sophisticated and plausible data-mining system produced to date. It preserves, for example, the notion of compositionality in the sense of the final translation being produced by a compounding of (potentially many) translated fragments. However, the fragments themselves are not produced by any sort of a parse scheme, and are instead produced by simple dynamic programming routine rather similar to the UNIX diff(1) program. The fragments identified are not necessarily linguistically useful or well-formed. Finally, the transfer process itself is nearly as opaque as those produced by Koncar's neural network, as there is no easy-to-describe method for performing the recompounding or rearrangement of the translated texts.

3 Linguistic universals

The obvious weakness of these approaches is that they do not make enough use of known linguistic principles. An easy example of this is the so-called basic word order of a language. If a random sample of ten clauses shows that all ten have the verb preceding the subject, then it is a good bet that most if not all of the clauses will have the verb preceding the subject. Furthermore, the work of Greenberg[15] demonstrates that such verb-initial languages tend to have prepositions that precede their nouns instead of postpositions that follow their nouns. Similarly, Keenan and Comrie[23] have demonstrated that languages that permit relative clauses to be made on oblique objects ("The woman [that I went to Boston with REL] is my sister") will also allow relativization on direct objects ("The hat [that I bought REL] is made of felt"), but not necessarily the reverse. By observing a few sample sentences, a sufficiently sophisticated computer program can set a small number of parameters corresponding

to a high-level description of the observed language and, in theory, acquire the language very quickly. This approach is typified in the work of Dorr[8] and Niyogi[34] as the "Principles and Parameters" approach. In this approach, a researcher or engineer sets up a metalinguistic system characterized by a set of principles that describe the total space of languages. Each language, in turn, is characterized by the instantiation of those principles into particular slots. French, for instance, can be parameterized as having two genders (masculine and feminine), from the possible (principled) space of {no gender variance, two genders, three genders}. The exact selection of principles and parameters depends crucially on the linguistic formalisms employed. Dorr, in particular, is developing an MT system based upon the principles and parameters expressed in the linguistic theory called Government-Binding Theory[5].

It should be clear that careful observation of these linguistic principles can make NLP tasks, and particularly language learning tasks, much easier. First, they can prevent systems, and the engineers who build them, from spending time on blind alleys. No human language is known that makes a grammatical distinction between prime and composite numbers (the way some languages make distinctions between zero, one, and two-or-more). For this reason, computer programs should not waste time examining adjective agreement systems based on whether the number of items is prime or composite. Engineers should not spend their time designing and implementing a fast and robust primality testing algorithm. On the other hand, languages do make relative clause distinctions, but they do so in a principled way. By using the universals described above, systems can explore the approximately ten attested systems instead 2^{10} which are naively possible.

Unfortunately, many of these principles are difficult to identify and acquire. As an example, one principle that has been incorporated into Dorr's system (Trace Theory) determines whether a noun phrase can be coidentified with an invisible and unexpressed trace element across more than one intervening clause boundary. This is a very powerful restriction, but is almost impossible to identify automatically without tremendous amounts of information about the structure and semantics of the language. A simple rephrasing of this parameter setting can make the difficulties clearer — this parameter determines a noun phrase's reference to the same object as an imaginary word (which is not present in the sentence itself) can be blocked by an imaginary boundary (which is also not present in the sentence itself). To adequately represent this principle requires that the system be able to correctly infer locations for these null elements and to classify them correctly. This ability has been provided to Dorr's system by carefully hand-coding the universals, which creates a new bottleneck of defining and describing languages in terms of a predefined parameter set.

One need not go this deep into theories to find useful-but-inaccessible universals. Consider the following sentence in a presumably unknown language (Urdu) : *topi dukan se lao.* It is not clear at first glance which word is the verb, which one is the subject, or even if this language requires nouns to form a grammatical sentence. Greenbergian basic word order can be difficult to acquire without a priori knowledge of which lexical items are verbs, nouns, adjectives, and so forth. What is needed is a set of linguistic universals that are easily accessible without an abundance of previous knowledge of the particular language, or, in other words, a set of principles and parameters that can be easily instantiated by a computer without creating a new knowledge acquisition bottleneck to create the linguistic superstructure. By using such a set, the system can gain the advantages of using linguistic principles without the requiring expensive, difficult, and time-consuming detailed analysis as done by humans. Although several such principles may exist, this work focuses on two in particular, the first being a direct

computational expression of a universal called "the Marker Hypothesis," and the second being a syntactic/lexical formalism called Categorial Grammar (described in section 11).

The Marker Hypothesis, as developed by Green[14] and others[29, 30], states in its simplest form that natural languages are "marked" for grammar at surface level—that there exists in every language a small set of words or morphemes that appear in a very limited set of grammatical contexts and that can be said, in a sense, to signal that context. As an example of this principle, consider a basic sentence in English :

> The Boulder Faculty Assembly announced a list of ten faculty awards at its Thursday meeting, with more awards for excellence in teaching than expected.

In this sentence, taken at random from a Boulder newspaper, two noun phrases began with determiners, two with quantifiers, and one with a possessive pronoun. The set of determiners and possessive pronouns in English is very small (less than fifteen words, depending upon how one counts[1]), and the set of quantifiers is equally recognizable[2] Similarly, every word in this sentence ending with '-ed' is a past tense verb. The Marker Hypothesis presumes the converse of these observations, e.g. that words which end in '-ed' are very often past tense verbs, and the word 'the' usually heralds the appearance of a noun phrase. Or, more generally, that concepts and structures like these will have similar morphological or structural marking in all languages.

Proponents of the Marker Hypothesis go further, however, claiming not only that these "marker words" could signal the occurrence of particular contexts, but that they do—that marker words form an important cue to psycholinguistic processing of structure. Experiments with miniature languages have backed up this claim. When human subjects are presented with the task of learning a small artificial language from sentences in the language, they learn more accurately and faster if the artificial language has cues of the sort described above. Green[14] showed this effect in artificial languages with and without specific marker words as attested in Japanese. Morgan et al.[29] demonstrated it in languages with and without phrase-level substitutions, as of pronouns for full noun phrases. Mori and Moeser[30] examined the effect of case marking on the pseudowords of the languages. In these and other experiments, evidence confirming the Marker Hypothesis was always found.

Other evidence for the psychological utility of marker words can be found in typological evidence. The original statement of the Marker Hypothesis was based upon the typological observation that every natural language has such constructs, whether in derivational morphology or separate marker words. Even pidgins and creoles have such constructs. For example, [37] lists examples from a pidgin called Russenorsk. In this language, sentences tend to be very simple strings of words, without grammatical inflection. Even in this language, however, verbs are marked with a special '-om' marker, which presumably helps hearers of this language identify the basic concept expressed in a given utterance (and from that determine the appropriate roles of the other words in the sentence).

Other psycholinguistic evidence for such the Marker Hypothesis can be taken from child language acquisition. Constructs which are easily and readily marked (e.g., regular verbs) tend to be learned early and strongly, and may even override other irregular forms which have been learned by rote memorization. The classic child's sentence "*I goed to the store"[3] is an

[1] e.g., is 'thy' worth putting into a translation system?

[2] Although in theory there are an infinite number of quantifiers, words like '635' or 'heptillion' are rare and easy to process. See [6, p. 98 et seq.] for a discussion of number markedness.

[3] The asterisk, as used here and elsewhere, is standard linguistic representation for a "wrong" sentence.

obvious example of this sort of overgeneralization. The child has learned that events which have already happened are described by verbs marked with the '-ed' morpheme. Slobin[38] lists dozens of psycholinguistic principles that may describe how children focus on important bits of the language to learn. Many of these (for example, "pay attention to the ends of words") are direct descriptions of phenomena the Marker Hypothesis would predict.

Finally, there is psychological evidence not only about the universality of marker words and morphemes, but also about their cross-linguistic similarity. Certainly, such concepts as case marking, gender, and tense seem to be concepts found in a large variety of languages. Talmy[45] suggests that, in fact, there are certain cognitive aspects or concepts that are inherently likely to be expressed grammatically (using marker morphemes or structural cues) and others that are universally expressed lexically. For example, many languages have inflections on nouns to express the number. On the other hand, there is no known language where morphemes exist to differentiate nouns referring to red objects from nouns referring to blue ones. Color, then, is not a concept expressed grammatically. The typological evidence of number agreement, that prime/composite is not a useful grammatical distinction, has already been presented. The implication is not only that marker constructs exist, but that the semantic concepts and distinctions that they express tend to be expressed in other languages by other marker constructions.

It can be concluded that the Marker Hypothesis can also be used to bolster the psycholinguistic plausibility of grammatical formalisms, and that these grammatical formalisms, can, in turn, be used as an explanation of certain grammatical properties associated with the Marker Hypothesis. Assuming, then, that the Marker Hypothesis is an accurate description of a useful property of natural languages, it is reasonable to use this property in an attempt to build a system that will naturally acquire the grammar of the source and target languages of interest to a machine translation system. The next section describes a computational formalism to do exactly that, based on the notion of acquisition of translation functions from large corpora as described above.

4 Computational implications of the Marker Hypothesis

What useful properties would marker words[4] have? As described above, they may signal grammatical structure *if* the information can be properly teased out. Smith and Witten[40] used a related hypothesis about "function words" to do inference of the grammar describing a large corpus. Observing that function words tend to be among the most common words in all languages, they gathered the most frequent 1% of all word types in a large document and used a series of strong statistical filters and n-gram models to infer a grammar of English. In their words, "the result is a relatively compact grammar that is guaranteed to cover every sentence in the source text that was used to form it." In addition, they found that the inferred grammar was plausible under current syntactic theories, unlike many large-corpora projects. These results were obtained despite a relatively inaccurate and implausible definition of function word. For example, one of the 1% most frequent words in *Moby Dick*, unsurprisingly, is the word "whale," A more accurate definition, perhaps including length and morphological complexity criteria, would presumably result in more accurate and general inductions.

[4] Or morphemes. The current work only focuses on marker *words*, but future developments will include morphological analysis from large corpora as a part of marker identification[22].

Another advantage of the Marker Hypothesis, particularly with regard to translation, is the way it isolates content words, which tend to have few translations. Although the many words to many words problem in translation is difficult, most of the difficulty originates not in the translation of words like "computer" or "kidnapping" but in words like "of" or "the." Context-dependencies are typically defined in terms of the syntactic nature of the surroundings, i.e. in terms of the marker words, and can therefore be solved with a more complex theory of marker word translation rather than a more complex theory of translation in general. Finally, Talmy's theories of grammaticalization[45] indicates that the structures indicated by marker words, although they may vary in their structure between languages, will indicate the same general properties and can therefore be incorporated into the target marker words rather than requiring major vocabulary shifts in the target language.

How, then can the Marker Hypothesis be formally incorporated into a computational theory of language in a way that allows it to be easily used? As described in section 3, the crucial property for this work is the existence of identifiable classes of marker words. Specifically, the formalism and system as described below assumes first that the languages of interest can be approximated by a context-free grammar, and second, that these languages can be naturally described by CFGs in *marker-normal form*, as defined below.

The computational background to this project can be summed up in the following mathematical result :

Theorem 1 *To every CFG* Γ *there corresponds an equivalent grammar in marker-normal form, where every production is of one of the following forms :*

$$A \rightarrow \epsilon$$

$$A \rightarrow a$$

$$A \rightarrow A_0 a_1 A_1 a_2 A_2 \cdots$$

$$A \rightarrow a_1 A_1 a_2 A_2 \cdots$$

(As usual, upper-case letters are nonterminal symbols, lower-case letters are terminal symbols, and ϵ *is the null string of zero length.) All right-hand sides of productions are either a single terminal symbol, or are an alternating sequence of terminals and nonterminals.*

Proof: Greibach's theorem[16, 17] states that for every CFG, there exists an equivalent grammar in which all productions are of the form $A \rightarrow aBCDEF\cdots$ (so-called *Greibach-normal form*). For a grammar in this form, all right-hand sides consist of exactly one terminal symbol, followed by by zero or more nonterminal symbols. For any grammar of interest, begin by finding an equivalent Greibach-normal form grammar Γ for it. This will then be transformed into an equivalent marker-normal form grammar.

Replace every production $A \rightarrow a\beta$, where β is a string of two or more nonterminal symbols, with two productions involving a new nonterminal : $A \rightarrow aX$ and $X \rightarrow \beta$. At this point, all productions involving the original nonterminals of Γ are in the required form for marker-normal form.

Now, consider a variable B that appears in the right-hand side of a rule $X \rightarrow \beta$. If B is the left-hand side of several production rules, create multiple production rules for the nonterminal X with the right-hand side of each production rule for B. Repeat this process with the Cartesian

product of all original nonterminal symbols of Γ. At the end of this process, every rule of the original grammar with multiple nonterminals has been replaced with a rule of the form $A \rightarrow aX$, with a single (marked) nonterminal variable. As the right-hand side of X did not contain any of the new nonterminals, every nonterminal in has been replaced by a marked nonterminal, and so the right side of every X-production is also completely marked.

To convert this entirely to marker-normal form may require the addition of another nonterminal between two terminals in the right-hand side of a production. Simply add the rule $\Omega \rightarrow \epsilon$, for a novel nonterminal Ω, and replace all such right-hand sides γ with $\Omega\gamma$ creating the initial nonterminal as required. This grammar is clearly in marker-normal form and also clearly equivalent to the Greibach-normal form grammar from which it was derived. *(q.e.d)*

Theorem 1 has an immediate corollary to reduce the necessary size of the production rules, at the expense of the number of such rules :

Corollary 1 *To every CFG there corresponds an equivalent grammar form, where every production is of one of the following forms :*

$$A \rightarrow \epsilon$$

$$A \rightarrow a$$

$$A \rightarrow A_0 a_1 A_1$$

$$A \rightarrow A_0 a_1 A_1 a_2 A_2$$

Proof: Exercise 4.11 of [16, p. 66] states that every context-free language can be generated by a grammar of the form

$$A \rightarrow a$$

$$A \rightarrow aB$$

$$A \rightarrow aBC$$

The construction of Theorem 1, when applied to the above grammar, produces a grammar of the desired form. If necessary, an ϵ-generating non-terminal symbol can be prepended to any production rules that do not already begin with one. *(q.e.d)*

This construction clearly results in much larger and potentially less-coherent grammars than the more standard Chomsky- and Greibach- normal forms. However, the Marker Hypothesis implies that explicitly marked grammars such as these are more psychologically plausible and thus that these grammars are likely to be more natural and understandable for human languages. In particular, natural language should tend to have relatively simple descriptions in which the set of terminal symbols that appear alone in productions is distinct from the set of terminal symbols that appear in a marking context; in other words, that the set of marker words is distinct and identifiable. The existence of marker-normal form provides a framework for attempting to solve natural language problems by focusing on the marker words. In addition, the symbolic, plausible, and understandable nature of these grammars makes it easier to incorporate other principles as constraints upon the grammar.

5 The problem

Given a bilingual text, it should be possible to extract enough information from the text to translate novel sentences. (The process of extracting an algorithm to fit a set of input/output pair could, in fact, be seen as a very general definition of learning) However, given that the source and target text are both written in natural language, with its well-known and well-studied properties, it seems only natural to use those properties to guide and improve the learning process. The METLA (Machine Engineered Translation by Language Acquisition) system is a family of experimental prototypes to investigate some computational applications of psycholinguistic principles and constraints to the problem of automatic learning of machine translation.

Specifically, the application of these principles should result in several significant improvements over the current state of the art. For example, the transfer functions produced by METLA are more linguistically cogent and plausible than those produced by purely statistical methods (e.g. [26]). At the same time, the amount of (human) work necessary to produce a system for a new language pair should be greatly reduced over that necessary for more typical MT systems, and it is easy to incorporate new linguistic principles and phenomena as they are discovered. The final, and potentially the greatest, new strength is that the system should be able to operate with partial linguistic descriptions. For example, it should be easy to modify a working system to incorporate new vocabulary items. Similarly, the system can start from an incomplete set of linguistic parameters (e.g. language X is known to be a subject-object-verb language, with postpositions and mandatory case marking on adjectives), and use them to speed and/or direct the inference process for a more accurate and easily found set of transfer functions.

6 Design considerations

The METLA system infers a grammar and symbolic transfer functions from an aligned bilingual corpus of sentences. More accurately, the system infers a set of parameters which collectively describe a grammar and transfer functions. These parameters, in turn, are derived from and express psycholinguistic theories and constraints. These parameters include :

- A context-free grammar or equivalently strong formalism describing the source language
- A context-dependent bilingual dictionary describing the relationships among lexical types in the two languages
- A set of permutation relations describing the necessary syntactic reconstruction to convert sentences in the source language into their translations in the target language

The system begins with a random set of parameters describing a skeletal grammar and transfer functions. Over many (potentially millions or billions of) passes through the training corpus, the parameters are tuned to reduce the differences between the translated source sentences (as translated by the current transfer functions) and the correct translation as given in the training corpus. The final set of tuned parameters can then be tested for generalization and/or used in a standalone translation system.

Once the system has been tuned to an appropriate set of parameters (or during the tuning phase as part of performance measurement), the parameters are used in a general translation

function as follows : The parse formalism is applied to an appropriately sized unit of text, typically a sentence, to produce a parse tree. Each leaf of the tree is translated by looking up the appropriate translation in the bilingual dictionary, and then leaves are successively permuted and concatenated until the entire tree has been concatenated into the desired target sentence.

The exact nature of the components in this general description is undefined. For example, the parsing could be done in a variety of ways and the tuning and inference task can be performed by any of a dozen algorithms. This work focuses on the design and development of two instantiations of the METLA framework (METLA-1 and METLA-2) which are functionally identical but differ in the exact makeup of their subsystems and parameter sets. In particular, the grammatical formalism of METLA-1 and METLA-2 differ, resulting in different parsing schema and different methods for storing grammatical information.

7 Components of METLA

The first step in the translation process, obviously, is to come up with a description of the source sentence(s) in some form amenable to further processing. By assumption and design, this should be something psycholinguistically plausible while still being easily inferrable. In practical terms, this means a context-free grammar or an equivalently strong formalism, at a minimum. The exact nature of this formalism and the parsing algorithm used comprises the main difference between the two METLA systems (METLA-1 and METLA-2) developed and discussed in this work, and for this reason will be examined in detail later. In brief, METLA-1 uses a greedy, top-down parser that is a direct extension of the marker-normal form formalism, while METLA-2 uses a more general and powerful formalism based on Categorial Grammar (defined in section 11 below).

In either system, the final parameters constitute a formal description of the syntactic properties of the lexical items that can be used to parse novel sentences in preparation for the restructuring and translation phases of the process.

As discussed in section 3, languages differ fundamentally in the syntactic structures that they use to represent similar semantic concepts. A single language, though, usually displays a relative regularity in its structures. For example, English is an SVO language, while Japanese is an SOV language. Syntactically speaking, then, English sentences tend to be composed of a (subject) NP, a verb and an (object) NP, in that order. A similar Japanese sentence would be composed of two NPs and a verb, in that order. Assuming that the grammar inferred by the system can successfully parse and and identify the two NPs and the verb from an English sentence, each of these components can be translated as a unit and their translations conjoined in a different order to form the Japanese translation. Numbering the components from left to right, the Japanese is produced by appending the first, third, and second components (after translation). This operation is immediately recognizable as a simple permutation.

A similar permutation could be carried out at every point of application of every grammatical rule in the source grammar. By repeating this translate-permute-concatenate operation recursively, any sentence in the source language can be restructured into a corresponding target structure. To complete the translation process, then, it remains only that the recursion have a base case, i.e. that the individual lexical items be translatable.

Neglecting the difficulties inherent in world-knowledge and pro-drop languages such as Spanish, every bit of semantic information expressed in the source sentence must be present in the

target sentence. The difficulty arises from the possibility of a different and ambiguous encoding in either or both languages. For example, the word 'that' in English can either be a demonstrative determiner ("Put that hat over in the box.") or a marker for a relative clause ("The hat that my cat likes to sleep on is covered in fur.") This lexical ambiguity does not have a similar ambiguity in French – the first 'that' would be translated as *ce*, while the second would be translated as *que*. Such ambiguity makes it difficult to develop a bilingual dictionary to translate English words into their corresponding French words.

For many languages, a simple one-to-one dictionary will cover large fractions of the vocabulary. For those words with multiple translations, many of the ambiguities can be resolved by looking at the context, generally speaking, in which they appear. For the example of 'that' above, for instance, the demonstrative determiner appears as part of a noun phrase (NP) while the relativizer separates a noun phrase from a relative clause. It is relatively easy to generalize the notion of a single correspondence to multiple correspondences by developing multiple one-to-one correspondence sets and selecting among them on the basis of context.

The METLA systems use a multiple dictionary formalism to produce a context-dependent dictionary for lexical selection. Within METLA, every grammatical context carries with it information to select one of of a fixed number of dictionaries to use. Within an NP, then, the translation system will use a dictionary in which the lexical entry for 'that' is *ce*, while using a different dictionary with a different entry when translating relative clauses. As another example, the English word 'one' translates to the French word *une* in the context of a feminine noun phrase, *on* when alone in the subject of a sentence, and *un* in other contexts. By developing three separate dictionaries, this word can be correctly translated in all three contexts.

Further sophistication has been added by the incorporation of ϵ (epsilon, or the null string) as an additional lexical type in all languages. This allows words to be deleted (translated to ϵ) deleted in some contexts, or for ϵ to be translated to another word in specific contexts to insert words as appropriate. An obvious example of this feature in use can be seen in the English→Urdu experiments described later, where there is no Urdu word corresponding to the English word 'the'.

Implicit in the above formalisms is the notion of describing them by parameter sets. For example, each of the several dictionaries can be seen as a function mapping words (or ϵ) to other words or as a function mapping numerical tags to other numerical tags. Each domain element can be individually mapped and changed to fit the bilingual data. Similarly, the choice of dictionaries can be described in numerical terms—in such and such a grammatical context, use dictionary number three. The end result of such description is a large number of relatively independent and tunable parameters which collectively describe a transfer function between the source and target language.

Setting the parameters at random, of course, will typically result in complete gibberish. However, by translating the source sentences in the database and comparing the translated results with the "correct" translation also listed in the database, one can produce a measure of the relative fitness of a given parameter set. Standard optimization techniques can then be applied to maximize the fitness of the parameter set.

Both systems use a multivariate optimization algorithm called simulated annealing[28, 24]. This technique was originally designed as a model of crystal growth and metal annealing, but has found widespread use in a variety of contexts and has the advantage of being well-known, well-studied, and reliable. In simplest terms, it is a variant of a random walk through the event-space of interest. At each step, the algorithm considers a random change to the set of

parameters and measures the quality of the translations produced by the changed set. If the changed parameters result in improved performance, the system accepts the new parameter set for further work. Even if the parameters reduce performance a bit, the system *may* still accept the new parameters as long as the performance loss isn't too great. As the algorithm progresses, the notion of "too great" is gradually tightened until the algorithm accepts only improving moves and eventually will find the global performance maximum.

As all of the parameters are described in symbolic or numerical terms, the changes are simple to define and implement. The available operations for the current METLA systems are:

- The translation of a (random) lexical item in a (random) dictionary may be changed to a new item.
- The dictionary to be used in a (random) context may be changed.
- The permutation-restructuring of a particular syntactic context may be changed by swapping any two items. For example, a sentence may be changed from SOV to SVO or VOS, but not to VSO (which would involve two swaps).[5]
- (METLA-1 only) The set of marker words separating two non-terminals in a grammar rule may be changed by the insertion or deletion of a single lexical item.
- (METLA-1 only) Any single non-terminal in the expansion of a grammar rule may be replaced by any other non-terminal.

The operations which are not currently permitted, although a system could in theory incorporate them include such operations as changing the number of productions in a grammar, changing the number of dictionaries, and so forth.

Upon inspection, it should become clear that the selection of simulated annealing as a "learning algorithm" may be controversial. Although simulated annealing produces good results, there is no particular psycholinguistic reason for its use. Numerical optimization is a long-studied problem and there are many other algorithms which have been developed. Early prototypes of the METLA-1 system used a variant of genetic algorithms[2], where the system maintains a large pseudopopulation of parameter sets and crosses and mutates them in a simulation of evolutionary selection. Another optimization technique, called tabu search[12, 13] (primarily a variant of hill climbing with momentum) has also been studied. In all cases, the results have been comparable or slightly worse than the results obtained from simulated annealing, and because of the size of the parameter sets and the large dimensionality of the search space, simulated annealing wins on the purely engineering consideration of efficiency.

Hidden in the description of the optimization task above is the idea of measuring the quality of the translations produced by a given parameter set. For obvious reasons, this measurement must be done numerically and automatically by the computer. Automatic measurement of translation performance is unfortunately difficult to perform. Many psycholinguistically plausible measurements are computationally expensive or technologically impossible. However, the sort of tasks that are computationally viable may produce "false positives" which appear to be related to the correct translation but in fact are very different. (Consider the effect of adding or deleting a 'not' from an English sentence.) After several experiments, the system uses a

[5] Formally speaking, of course, subject and object are semantic roles and only partially syntactic. The idea of permuting two NPs and a verb should be clear, however.

modified greatest-common-subsequence formalism[31], which should be familiar to most UNIX programmers as the *diff(1)* algorithm. Specifically, this measures the number of changes (insertions or deletions) that distinguish the translated source sentence from the desired target sentence. Sentences are thus graded on the number of words that would need to be added to or deleted from them to produce the exact form in the examples, an approximate measurement of the amount of work a human editor would need to do. Again, this measurement is not psycholinguistically motivated or defensible but has been empirically selected for engineering reasons from a larger set of candidates.

8 Description of experiments

The standard procedure for most modern learning systems (e.g.[27, 7, 26] among many others) is to produce two separate sets of data, a training set and a testing set. The system is trained to some criterion, usually measured either in terms of performance or else a set number of training epochs, and then the actual performance measurements are taken on novel data to which the system has not been exposed. This prevents the system from merely memorizing the input data and provides a better measure of learning performance, but also requires that the researchers acquire two sets of data. In the case of METLA, this would of course be two similar aligned corpora on the same language pair, or more simply two halves of the same corpus.

Because both METLA systems, as defined in sections 9 and 12, use simulated annealing as their primary inference mechanism, there is no practical possibility of "incremental" learning to a desired criteria, and the systems are trained over a fixed number of epochs as designated in the annealing schedule. The actual learning is run multiple times to prevent conspiracies in the random numbers from having an undue effect on the final outcome.

The experiments themselves have been run on a variety of language pairs, covering an equally varied style of writing. Both natural and artificial corpora have been used, and the difficult level ranges from elementary pre-primers to literature. The exact details of the various corpora are presented in the individual sections on each system.

9 METLA-1 parsing algorithm

As the name implies, METLA-1 was the first version of a translation system developed using the formalism from section 6. Because METLA-1 was designed only to be a prototype, it is only capable of dealing with very small grammars and vocabularies. For instance, both the source and target language must have vocabularies of thirty-one words or smaller, to permit word sets to fit within a single four-byte integer variable. Similarly, the number of grammatical rules and categories is restricted to a small compile-time constant (for these experiments, typically between five and ten).

The parsing algorithm used by METLA-1 is a direct expression of the marker-normal form mathematics developed in section 4. Specifically, every non-terminal symbol (of which there are a fixed number, defined at compile time) is associated with a production rule in a modified marker-normal form. The primary modification used is that sets of marker words, instead of individuals, are used to separate the various constituents. A secondary modification is that marker words are attached, for purposes of further parsing, to the constituents that follow them. For example, a sample rule for English might be

Sentence → NP aux V det NP

where 'det' is any of the set of {a, an, the} and 'aux' is any of the set of auxiliary verbs {be, have, will, can,...} in any of their inflected forms.

Formally, the grammar can be characterized as a fixed set of rules, numbered from zero to $N-1$. Each of these rules has a fixed fanout of non-terminal symbols k, so every rule in the grammar is of the form

$$A_i \rightarrow A_x m_{i,1} A_y \cdots m_{i,k-1} A_z$$

In this notation, each $A_?$ is a non-terminal in the set $A_0 \ldots A_{N-1}$ and each $m_{?,?}$ is a set of marker words that marks the separation between the various constituents of A_i.

Parsing is done in a rather simplistic fashion. A_0 is by fiat designated as the starting symbol of the grammar, and the training sentences are parsed in a strict top-down fashion. Each sentence is partitioned into its constituents at the appearance of the leftmost element of each marker set, in order of appearance in the rule of grammar. For the sample rule above, this would divide a sentence at the first auxiliary, and then at the first determiner following. These constituents are then recursively parsed in accordance with the single rule corresponding to their nonterminal, and so on, until the sentence has been broken down into only lexicalized items. These items are, of course, translated using the various context-sensitive dictionaries the system has developed.

Because of the strict top-down nature of this parsing as well as the fact that each non-terminal is associated with only one production rule, the nature and accuracy of the grammars that METLA-1 can infer is strictly limited. In addition, because of the focus of METLA-1 only on marker *words*, many morphologically marked structures will not be found. Finally, the system starts from a randomly-chosen starting grammar and vocabulary, rather than using any techniques to select a good starting point, which slows the inference task considerably.

10 METLA-1 experiments

METLA-1 was tested on two corpora, one involving English and Urdu, the other involving English and French. The first was an English→Urdu text taken from [46]. This book is one of many such books written in the early 20th century for the benefit of British officers in various colonies and provides a quick and easy introduction to Urdu for native English speakers[6]. The training corpus consisted of a vocabulary list and the set of example sentences (and their translations) taken from lesson 2, while the testing corpus was the set of exercises (which were translated by hand and confirmed by a native speaker of Urdu). Typologically, Urdu is an Indo-European language with a heavy influence from Arabic. Structurally, it has basic word order SOV, postpositions instead of prepositions, and no definite/indefinite article distinction.

The second corpus was an artificial English→French corpus designed to test the performance of the system on a small vocabulary but with greater syntactic complexity than the Urdu corpus. The training set consisted of forty-three sentences selected from a thirty word vocabulary and included gender distinctions, embedded relative clauses, words with ambiguous translations, reflexive and non-reflexive verbs, and multiple subcategorizations of verbs. The testing data consisted of similar sentences produced by a different experimenter using the same vocabulary.

[6] Aravind Joshi, personal communication of 15 September, 1994

All translations were confirmed by a native speaker. Typologically, French is an Indo-European language with the same basic word order and structure as English, but a more pronounced gender agreement system.

For both experiments, the system was presented with the training data and allowed to infer (through simulated annealing) long enough to achieve a complete "freezing" of the system. At each point, the system attempted to translate the entire training corpus, sentence by sentences, and the total amount of changes necessary to correct the translation was measured as the error. The annealing schedule used was a simple geometric progression, where every new temperature was 1% smaller than the previous temperature, and between 50 thousand and 150 thousand changes were made at each temperature step. The annealing itself progressed over three orders of magnitude of temperature changes, typically resulting in between eight and sixteen hours of computation time on an HP Snake.

One of the difficulties involved with the development of a machine translation system is the evaluation of the end product. Is it better, for instance, to produce an ungrammatical translation that nonetheless seems to capture the meaning of what the original said, or to produce a grammatically flawless sentence that states something completely different from the original? How should the system respond to unusual, metaphoric, or ungrammatical inputs?

For many fully self-automated translation systems (e.g. [3]), the problem can be made worse by the relative opacity of the inferred translation system. There is no easy way to examine the internal workings of the algorithm to determine the nature and causes of a translation error or to identify how to repair the error. And for translation systems using Markov models[3] and similar oversimplified grammatical structures, it may not be possible to understand the cause of the error even after a lengthy and extensive analysis of the translation parameters, as the underlying model is too distant from people's intuitive understanding of how languages are put together.

Nonetheless, it is possible to do some sort of a black box analysis of the output of the system. Brown et al.[3], for instance, performed their analysis on the basis of hand-classification of sentences into five types, ranging from "Exact" (Identical to what the Hansard translator chose), through "Alternate" (Different phrasing but the same idea expressed), down to "Ungrammatical." This sort of hand-classification for final system evaluation is useful because it directly measures the appropriateness of the final product in a way that more automatic measures (such as diff) cannot. For the METLA-1 prototype, though, this particular classification was less useful than the classification actually used. Because of the limited vocabulary and grammar in the experiments, very few different grammatical ways to express the same idea were available. It was therefore more useful and appropriate to classify sentences (again by hand) into the categories "Correct," "Minor errors," and "Gibberish." The first category corresponds to "Exact," above. The third category describes sentences that were so syntactically ill-formed as to be unintelligible and would be a subset of Brown's "Ungrammatical" sentences. The second category would be classified by [3] sometimes as "Alternate" and sometimes as "Ungrammatical." These tend to be syntactically invalid but semantically understandable. They also tend to reflect (subtle) properties of the source language that are slightly changed in the target language. In fact, they closely resemble typical errors of first-year language students. Examples of these from the English→French experiments include deletion of sentence complementizers[7], deletion of reflexive particles, or gender errors.

[7]The 'that' in the English sentence "I believe (that) rocks sink" is optional. The corresponding 'que' in its French translation is required.

Category	Training	Testing	Limited
Exact	61%	36%	41%
Minor	29%	21%	19%
Gibberish	10%	44%	41%

Table 1: Results from black-box analysis of METLA-1 French experiments

When this sort of analysis is performed on the results of the English→Urdu experiments, the system learned the original training corpus (the example sentences from the lessons) perfectly and could reproduce it without errors. Testing on novel sentences (the exercises) revealed 72% completely correct, and only 7% translated as "gibberish." Upon further analysis (as described later in this section), the training corpus was shown to be unrepresentative of the test corpus, and in particular was missing coverage in context for several words. When the training corpus was updated to include coverage for the missing items, the system could still learn the training corpus perfectly and the percentage correct on novel items of the same forms increased to 100%.

The English→French experiment, because of the higher syntactic complexity in conjunction with the limited scale of the prototype, performed less well overall. Typical performance for the system on the training corpus was approximately 61% correct. On the test data, performance was lower, with only 36% correct and a full 44% gibberish. However, when the test sentences that presented structures unrepresented in the grammar were excluded, the performance improved, up to 41% correct. Although cross-system and cross-corpus comparisons can be problematic, or even meaningless, the percentage correct for the METLA-1 system is in the approximate area of the results from [3], where an early version of the system was able to correctly translate 48% of the test data based on a much larger training (and testing) corpus. These results are summarized in tabular form in table 1.

Clearly, much additional work will be required before METLA turns into a commercial-quality translator. However, given the known structural limitations of the implementation and the small grammars that it used for these experiments, these still represent a significant accomplishment in the development of an empirical but psycholinguistically plausible MT system. Perhaps equally significantly, to convert the system from one language to another required approximately an hour of human effort to type in the training data, and no system modifications. This indicates that language-independent induction of transfer functions may be a viable approach to machine translation.

A major advantage of a psycholinguistically plausible approach is that, if properly done, the output of the system can be directly converted into a grammar and dictionaries for the appropriate languages. This makes it possible to directly analyze the plausibility and appropriateness of the various transfer rules and to improve them by human intervention. Some of the simplifications made in the course of developing METLA-1 have made it more difficult to perform this task, but one can still examine the source grammar and transfer functions which the system developed and use this information to change the transfer rules or training data.

For example, in the English→Urdu experiment, the training data consisted of copula-locatives ("the hat is on the chair", "the man is in the shop") and imperative sentences ("wait in the office," "send the knife to the house"). Each of these had to be rearranged into verb-final form, and the prepositions had to be converted to postpositions. In addition, all the determiners ('a,' 'the,' 'this,' etc.) needed to be deleted, so the final result of translation would be something like the word-for-word translation of the string "*knife house to send."

Upon examination, the word classification and translation methods make sense. For an example, one of the early experiments initially divided all sentences into two parts based on the first appearance of a determiner or preposition. This divided imperatives ("wait in the office") into their verb components followed by one or more arguments which were translated by another set of rules. The translation of the verb was permuted to follow the rest of the sentence, giving the necessary verb-final form. On the other hand, declarative sentences ("the book is on the table") are passed through this initial rule unchanged, to be divided later at 'is' into subject, verb, and location, and permuted appropriately. This sort of analysis can be carried out to any desired level of detail.

Even this simplified analysis, however, is enough to demonstrate the advantage of a psycholinguistically plausible and symbolic representation. The statement "to be divided later at 'is'" is, in point of fact, slightly inaccurate. Using the first version of the training data, the system accurately inferred that 'is' serves to mark the boundary between subject and verb. However, it also inferred (wrongly) that 'knife' and 'man' were also part of that same marker group. This resulted in a small number of incorrect translations of the testing sentences.

Further examination of the input corpus showed the reason that these errors had been made. Although the system was presented with a full vocabulary list ('man'/'admi', 'house'/'ghar', and so forth) of individual words, only a subset of those words had been presented in the context of a phrase or sentence. Although the system, then, had learned that 'man' translated to 'admi,' it had no evidence about the part of speech of 'man.' The system had no way of knowing, for example, that the word 'man' was not an alternate form of the copula. In general, the lists of marker words are obviously of one or more grammatical classes, with potentially a few outliers that represent words that have never been seen in that context and therefore may or may not be relevant. With this observation, it becomes/became obvious that the input examples were not representative of the testing data, and that some new input was required. After adding two more sentences to provide context for these words, the percentage correct increased in later experiments to 100%.

Similar analysis can be done for the more grammatically-complex English→French experiments. Because of the greater syntactic complexity, the system as built proved to be oversimplified in several important regards and some errors were in that sense inevitable. For example, consider the following set of sentences :

(the man) (kisses) (the woman)
(le homme[8]) (embrasse) (la femme)

(the man) (kisses) (her)
*(le homme) (embrasse) (la)

*(the man) (her) (kisses)
(le homme) (la) (embrasse)

In the first pair of sentences, the pattern subject–verb–object is used in both French and English, so the identity permutation is appropriate. In the second and third pairs, the pattern becomes subject–object–verb in French, so the identity permutation is no longer appropriate. However, as each non-terminal symbol (sentence, in this case) has only one rule and one permutation

[8]The process that converts, for example, 'le homme' into 'l'homme' is almost purely phonological and was ignored for simplicity in the French1 corpus.

bring the letter from the shop (bring) ((the letter) (from the shop)) (lao) ((chitthi) (dukan se)) chitthi dukan se lao
wait in the office (wait) (in the office) (thairo) (daftar men) daftar men thairo
put the box on the table (put) ((the box) (on the table)) (rakho) ((sanduq) (mez par)) sanduq mez par rakho

Table 2: Sample English→Urdu translations with partial analysis

associated with it, the system is forced to select one and only one of object-final or verb-final structure.

On the other hand, the system correctly learned appropriate translation structure for a large part of the input corpus. For example, the original sentences are parsed into three pieces based upon the existence first of a verb, and then of a determiner or pronoun. Noun phrases (which begin with a determiner in the input corpus) are themselves partitioned into classes of masculine/feminine noun phrases so that the gender of the determiner is correctly set.

The major error made by the English→French system was that it found a local maximum in reusing one of the production rules. Because any translation system should allow for recursive structures ("John said that Mary told him that Susan said that ..."), the system is permitted to call rules that have already been called. The system tended to find a local maximum where the rule used to separate masculine from feminine nouns was the same rule used to parse the original sentence, and so it conflated the two categories of verbs and feminine nouns. This meant, in turn, that sentences such ase "that woman washes a car" were divided not as "(that woman) (washes) (a car)" but instead as "*(that) (woman washes) (a car)." This error could presumably be rectified by allowing the system to use more production rules, but is more appropriately solved by a better parsing algorithm in general.

Some sample results are attached as tables 2 and 3. Each table shows a number of sample sentences (in the nearly opaque parenthesized format) along with their primary division into constituents, the translations of those constituents, and the final translation after it has been permuted and concatenated.

The errors in table 3 should be explained. First, note that the division of the third sentence is incorrect—"the man that touches the car" is an entire component and the main verb of the sentence is the *second* token of 'touches.' This is an artifact of the admittedly limited METLA-1 parsing algorithm, which divides at the first appearance of a given token. That this sentence is correctly translated at all is a tribute to the remarkable structural similarity between this sentence and its French translation. The fifth sentence is an example of a so-called "reflexive" verb; the proper translation should be "ce chat se lave," where 'se' is a general pronoun meaning 'self.' In English, certain verbs can be intransitive when the subject and

the glass touches a car (the glass) (touches) (a car) (le verre) (touche) (une voiture) le verre touche une voiture
she washes a cat (she) (washes) (a cat) (elle) (lave) (un chat) elle lave un chat
the man that touches a car touches a glass (the man that) (touches) (a car touches a glass) (le homme qui) (touche) (une voiture touche un verre) le homme qui touche une voiture touche un verre
that man washes a car that she creates (that man) (washes) (a car that she creates) (ce homme) (lave) *(une voiture qui elle creee) *ce homme lave une voiture qui elle creee
this cat washes (this cat) (washes) () (ce chat) (lave) () *ce chat lave

Table 3: Sample English→French translations with partial analysis

object of the verb are the same—for example, "I shave (myself) every morning," "I wash[9]," and so forth. Some of these verbs, in turn, *must* be expressed with the reflexive particle in French but with an ordinary direct object otherwise. This leads, in turn, to another example of the multiple-necessary-permutation problem discussed above.

The fourth sentence is more interesting. The word 'qui' in the fourth example sentence is a relative pronoun used only for people (like 'who'). As an inanimate object, "a car" should have taken the relative pronoun 'que' as a translation of 'that'. However, notice should be taken of the mistake that the system did not make. The other token of 'that' in the sentence was a demonstrative determiner, which was correctly translated as 'ce', taking into account the gender of 'man'. The system correctly identified the second 'that' as a relative pronoun and not a demonstrative determiner. Similarly, the third sentence indicates an ability to distinguish between feminine nouns ("une voiture") and masculine ones ("un verre"), a relatively subtle grammatical point. These results, then, indicate an ability on the part of METLA-1 to determine remarkably small grammatical structures and to appropriately account for and to produce them as needed in the translation process.

11 Categorial Grammar

The notion, as expressed by the Marker Hypothesis, of marking expected constituents can easily be generalized beyond simple closed-class words and morphemes. For example, verbs are usually viewed being associated with subcategorization frames, a shorthand for the notion that

[9]In some dialects, not including the author's, this concept would be expressed as "I wash up."

John	knew	that	Paul	was	a	poor	man
n	s/(n)[n]	n/[s]	n	s/(n)[n]	n/[n]	n/[n]	n
n	s/(n)[n]	n/[s]	n	s/(n)[n]	n/[n]	n	
n	s/(n)[n]	n/[s]	n	s/(n)[n]	n		
n	s/(n)[n]	n/[s]		s			
n	s/(n)[n]			n			
				s			

Table 4: Sample grammatical analysis in CG

a particular verb is usually associated with a particular set of syntactic and/or thematic roles to be filled in a grammatical sentence. Nouns, on the other hand, are typically viewed only as slot fillers, and carry no expectations about the other words in the sentence. Categorial Grammar (CG) is a grammatical formalism that attempts to derive all forms of syntactic regularities from such individual lexical properties.

First developed by Bar-Hillel[1] and used by many later researchers (e.g. [48]), CG can be seen as a method of describing syntactic regularities purely on the basis of properties of the individual words in a language. The basic units of analysis are entities (nouns, typically), and propositions (sentences), which can carry truth-values about entities. Other words are described in functional terms—for example, a transitive verb is simply a function which converts two noun phrases into a sentence. The basic structure of English derives partially from the fact that the two noun phrases expected by a transitive verb are on opposite sides of the verb itself, while in Urdu, the two noun phrases expected by a transitive verb are both to the left of the verb.

Wood[47] gives an example of this process reproduced here as table 4. The notation $x/(y)[z]$ describes the lexical (sub)categorization associated with a word. In particular, (y) means that this word expects to be immediately preceded by a word or phrase of category y, while $[z]$ describes similar expectations for the following word or phrase. Either or both of these sorts of expectations may be present; if all are met, the resulting conjunction is of category x. As hinted above, the basic categories for this particular analysis are 's' (for sentence) and 'n' (used here for "everything else")—the exact choice of atomic categories is a hotly debated topic among CG researchers.

In brief, words such as 'man', 'John', and 'Paul', which represent entities, are categoried as 'n'. Words like adjectives ('poor') and determiners are categorized as 'n/[n]', functions mapping a noun to another, more complex noun. Only one parse is available for phrases such as "a poor man," reflecting the necessity for every modifier to have an object to modify. Verbs, as have already been discussed, are mappings from one or more nouns into sentences, while the complementizer 'that' can convert a sentence into an entity in its own right for further discourse. From a formal standpoint, CG is clearly of equivalent power to standard CFGs—every lexical item and categorization can be easily converted into a CFG production, and conversely, every production in a CFG in Greibach normal form can be described as a CG lexical subcategorization for the initial nonterminal symbol in that production.

Categorial grammar defines an easily-computable but linguistically viable approach for describing syntactic structures. Because of its strong focus on the lexicon, it is relatively simple for a computer to identify a potential set of categorial expectations or to define novel words in terms of other words with similar categorization. These properties may make CG an ideal formalism for the investigation of corpus-based linguistic phenomena.

CG, of course, can be viewed as the ultimate extension of the Marker Hypothesis, as grammatical structures by assumption cannot exist except as the expectations of lexical items. It is worth investigating what other psycholinguistic predictions CG or a similar "radically lexical"[41] grammatical formalism would make.

For example, a well-known psycholinguistic phenomenon is the difficulty of parsing so-called "garden path" sentences, sentences in which the initial part of the sentence forms a valid but syntactically different sentence. The standard example of such a sentence is "the horse raced past the barn fell," where the desired interpretation is that of the sentence "the horse (that was raced past the barn) fell."

In CG, lexical ambiguity (such as between relative clauses and main verbs) is handled by multiple subcategorizations for each lexical item. For instance, in the example sentence, the choices can be described as an ambiguity between two senses of 'raced', one with a noun phrase preceding it as the agent and one without, as an embedded passive construction. The parsing difficulty can be explained by assuming that the agentive sense is preferred for some reason (presumably related to frequency of occurrence) and that the evidence that it does not parse the sentence correctly is sufficiently non-local that the dispreferred sense cannot be chosen sufficiently fast. Thus, CG provides, at least at first glance, a plausible explanation for garden path phenomena.

Of course, analysis of isolated phenomena are useless without some idea of a more fundamental relationship between the two systems. The major strength of the Marker Hypothesis is that it provides a natural distinction between function words and content words and describes the differences in their respective purposes. It is possible to describe this distinction fairly easily in terms of Categorial Grammars. For instance, (in English) there must be some sort of categorical difference between bare (count) nouns such as 'car' and determined count noun phrases such as "the car". Construction Grammar, as developed by Fillmore[10], describes this in terms of a property called 'maximality', which simply states whether or not a noun phrase can be used to build other constructions. Categorial grammar, on the other hand, would state that the lexical categorization of 'the' includes (np/[n]) — or in other words, that 'the', functionally, takes a bare noun and produces a noun phrase. Adjectives, however, which do not change the maximality of their respective nouns, are categorized as (n/[n]).[10]

In general, this distinction can be extended into a general classification of words. Content words, under CG, appear to fall into one of the following three categories :

- Nouns (entities), which are basal elements of the grammar,
- Verbs, which are functions from something into a sentence (s), and
- other things (adjectives, adverbs, &c.), which provide semantic but not syntactic information. [11] In CG terms, these are words which convert a structure of one category into a structure of the same category, i.e. n/[n], s/[s], et cetera.

Function and/or marker words, then, are simply words which serve to transform categories (such as np/[n]) or conjoin categories (such as 'and' np/(np)[np]). This observation provides

[10] I am grateful to Dr. Laura Michaelis for her lucid explanation of how Construction Grammar approaches this.

[11] N.b. The converse, a word which provides no semantic information, is almost the standard definition of a function/marker word.

a simple explanation of numerous properties in terms of an easily-formalizable theory. For a simple example, consider the case system in German. Using a naive view of case, the difference between an uncased noun and a cased noun is expressed in the choice of the determiner. The determiner, then, serves as a word to hold and mark the case of the following noun. In functional terms, the determiner can be viewed as a function from uncased nouns to nouns of a particular case, which in turn is subsumed into the subcategorization framework of the verb. A more sophisticated view of case (such as Fillmore's/[9]) produces largely the same results – under this framework, the verb assigns case to each of its syntactic constituents and the determiner provides a place for this case to be marked, again providing, functionally, a mapping from uncased nouns to cased ones. Similar arguments could provide functional explanations for quantifiers, prepositions, conjunctions, and most other marker words. A simple generalization to the notion of words, allowing morphemes to have lexical expectations and subcategorizations, would provide an equally powerful explanation of marker morphemes.

12 METLA-2 parsing algorithm

The METLA-2 system represents a further development of the translation approach defined above, with the avowed intention of developing a more robust and viable parsing algorithm while still retaining psycholinguistic plausibility and learnability. The new system has many of the characteristics of the first version, but the parsing system has been completely redesigned to use a formalism based on Categorial Grammar, as described above.

Every word in the source language is associated with one or more "senses" which, in turn, represent possible contexts in which that word may appear and the translation appropriate to each context. Contexts, in turn, are represented by a subcategorization frame associated with each sense. Parsing is done by a dynamic programming scheme, where any word for which its subcategorization slots are filled subsumes those slots into a larger tree structure. In the case of entities (nouns), this is a trivial operation; in the case of more complex functional categories, this may be delayed for one or more passes until the categorial expectations can be fulfilled.

Again, an example may help. Consider the following sentence :

The chair is in the office.

Both 'chair' and 'office' are nouns and can be immediately identified as such. The word 'the' subcategorizes for an immediately following noun and creates a larger phrase 'the chair.' 'In' expects to be followed by a noun phrase (or whatever category 'the chair' parses as). The word 'is', of course, has many senses, of which one is the context of s/(np)[pp].

These categorial frames are built, in turn, from a much larger template that defines the maximal extent of categorial expectations. From both an engineering and a linguistic standpoint, the template produces a crucial variation in the inferred grammars. Consider briefly the English verb '(to) put' or any similar ditransitive verb. The semantics of this verb require at least three expectations—a subject (donor), an object, and a location or recipient. Any CG template that does not provide for at least three expectations, one preceding and two following the verb, will be unable to correctly parse sentences containing such verbs. On the other hand, too broad a template results in too many free parameters for the inference to perform well. The METLA-2 system uses, as a compromise, up to four potential constituents in addition to the lexical item, two before and two after the lexical item of interest.

Further sophistication is added by the introduction of ϵ-productions, which allow for a constituency slot to be "filled" by an imaginary lexical item ϵ, which can then be translated into another word. This allows the system to insert words in the process of translation, as is required to correctly translate English 'not' into the French 'ne/pas' construction. Similarly, any word can be deleted by translating it into ϵ.

Permutation, concatenation, and lexical lookup are performed as in the METLA-1 system. The only major change between the two systems is that, in METLA-2, the grammar formalism does not guarantee that the entire input utterance will be conjoined into a single tree. In many cases, especially in dealing with real-world input, this may be a feature, as run-on sentences and misspeakings are relatively common. In any case, the METLA-2 system handles this difficulty by simply translating and conjoining all trees in the order in which they appear in the input, hopefully retaining the metalinguistic pragmatic information expressed by the clause ordering in the source text.

From the viewpoint of the learning/annealing and the parameterization, METLA-2 has almost the same characteristics. The only major difference is in the grammatical formalism, which must be parameterized differently. Instead of using marker words and their corresponding set operations, the parser is parameterized in terms of categorial expectations, resulting in the following new operations possible :

- The syntactic category expected to the left of a (random) lexical item may be changed or eliminated.
- The syntactic category expected to the right of a (random) lexical item may be changed or eliminated.
- Similarly, any syntactic category expected in any subcategorization frame may be changed or eliminated.
- The lexical category resulting from the conjunction of several phrases may be changed to another category.

13 METLA-2 experiments

METLA-2 has been tested on two corpora, with further experiments in progress but unreported here. As with METLA-1, the first experiment involved the Urdu text from lesson 2 of [46]. The second was a simple, but genuinely natural corpus taken from a child's picture book called *Curious George*[35] and its Spanish translation[4]. It should be noted that the two books were published separately and independently, not as a running bilingual text, and that the translation did not necessarily reflect all the properties of the source document. The two texts were aligned by hand, and then all sentences of seven words or fewer (in both versions) were extracted and presented to the METLA-2 system as training data. The testing data, in turn, was taken from the (English) book *Curious George Takes a Job*[36]. All sentences of seven words or fewer were taken from this book, and those which used only words in the training data were presented as test material.

As above, the Urdu experiments resulted in perfect translations of both the training and the testing data. The individual lexical transfer functions can be examined by hand and proved

george was very curious jorge estaba muy curioso
he was too curious también estaba curioso
they opened the door * la
george was fascinated jorge estaba fascinado
this is george éste es jorge
where was george * casa

Table 5: Sample English→Spanish translations

to translate perfectly for the small vocabulary and grammar. The lexical items found were intuitively "correct" in terms of categorial structure, translations, and permutations.

Few corpus-based machine translation experiments have been done with non-technical, non-legal corpora. This undoubtedly due, in part, to the scarcity of available corpora (of the 13 multilingual corpora on the ECI/MCI disk, only 2 short texts are "fiction"). Another part is due to the perceived greater freedom for a translation of fiction rather than legal/technical documents. Certainly, the quality of the interlineal translations of *Curious George* was considerably worse than the quality of the other two corpora. In some cases, the difficulty was purely stylistic and can be easily resolved in context—for example, the sentence "The man was happy too" was translated into *Y el hombre también.*[12] In other cases, however, the translation represented a serious change in the semantics of the target sentences.

For example, the English text used the sentence "The man took off the bag." In context (George, a monkey, has just been captured and placed in a bag), it is clear that the desired interpretation is that George is being "undressed" by having the bag taken off, as one would take the diaper off a baby. The corresponding Spanish sentences is *El hombre lo sacó de la bolsa*, literally "the man took him/it out of the bag." In addition to changing the pragmatic focus of the sentence from the bag to George and thus depersonalizing the monkey, this sentence also describes a slightly different act; in the English version, the bag is moving, which in the Spanish version, George is moving while the bag stays in place. In legal or technical documents, such a minor change could be disasterous, while in the context of fiction, it is barely noticeable.

Despite the evidently poor quality of the test data, the results from the Spanish data are promising. Only six (novel) sentences made it through the stringent selection criterion — although these six sentences, in turn, represented a corpus about 15those six, three were translated accurately (50%), while one more (17%) was translated to a grammatical sentence with a different meaning than the original source sentence. Even here, the error is interesting, significant, and plausible – the testing corpus included the word 'too' only as a synonym for 'also', as in the sentence presented above. The sentence "he was too curious" was translated into *también estaba curioso*, a correct Spanish sentence corresponding to "He too was curious" or "He was curious as well." Some results from the Spanish experiments are attached as table 5.

[12]literally, "and the man [was], too".

From a psycholinguistic standpoint, the results of the Spanish experiments are disappointing. The basic word order of Spanish and English are sufficiently similar that for most sentences, and especially most simple sentences, a mere word-by-word translation suffices. For this reason, many of the sentences, especially the ones that were translated correctly, were simply parsed as a linear sequence of atoms. This results in the somewhat surprising result that psycholinguistic constraints may work *better* on language pairs that are typologically different because of the greater variability, making it easier to tease the marked structures away from the markers.

14 Conclusions

To solve linguistic problems, one needs to understand linguistics. Twenty years ago, that statement would have been uncontroversial. Today, with faster computers, larger disks, and greater memory available, scientists can work more directly with examples from linguistic data. This has led some researchers to use methods that focus more on computational efficiency (e.g. hidden Markov models) and can sometimes be linguistically naive or use very language-specific structures. The work presented above is an attempt to graft psycholinguistic principles onto the EBMT framework in a language-independent fashion. Rather than using human effort to develop an exhaustive analysis of the source and target language, the human effort can be put into identification and incorporation of linguistic principles such as the Marker Hypothesis[14], X-bar theory[18], and structural universals about language[15]. By focusing on the same sets of principles that linguists use to describe novel languages, the same system could be used for many different language pairs, addressing exactly those inter-language differences that linguistic typologists find interesting.

The METLA family of systems are prototypes developed to address and test some of the concepts necessary to produce such a language-independent analysis system. Focusing on a restricted set of linguistic universals (primarily the Marker Hypothesis and Categorial Grammars) and on small sets of data, they nonetheless manage to produce respectable performance on the structural analysis, transformation, and translation of novel sentences. In addition, the structures and grammatical classes used are logical and linguistically sensible—for example, the system picks up readily on the concept of prepositions, correctly gathers the prepositions together in a class, and identifies that the dependent noun of a preposition follows the preposition itself in English.

This work does not exclusively focus on grammatical induction. Although grammatical induction is an important part of the task, neither the problem (translation) nor the approach guarantees that the system will learn anything usable for grammaticality judgements. For a simple example, a system trained to translate (US) telephone numbers from English to French would not necessarily learn that telephone numbers are comprised of seven digits, divided into groups of three and four by a hyphen. At the same time, the system would presumably be robust enough to translate malformed phone numbers without causing system errors. This is clearly an advantage in dealing with real-world input, where typographical errors and misphrasings are not uncommon. At the same time, this system will include grammatical structure which should result in more robust, understandable, and linguistically plausible translation functions than the Markov chains developed by [3].

Finally, although this system uses examples to develop its translation functions, there are several crucial differences between METLA and the more mainstream EBMT paradigm. First, other than the notion of paired sentences, there is no preanalysis of the translation database,

which greatly reduces the load on the human developers of the system. This system also produces a reduced database, explicitly extracting patterns from the example database rather than finding them as needed in on-line examples.

The results of these tentative experiments indicate that induction of transfer functions from untagged, unanalyzed bilingual corpora is a computationally and linguistically viable task. Furthermore, the addition of linguistic information into the algorithm itself produces more understandable and thus maintainable results. In particular, these results seem to show that hours, days, or months of computer time can be substituted for the time of human translators if the appropriate low-level bilingual corpus is available. Further work will hopefully demonstrate that psycholinguistic universals, in general, are computationally useful both for the production of robust NLP systems as well as for the modeling of human language processing.

References

[1] Yehoshua Bar-Hillel. A quasi-arithmatical notation for syntactic description. *Language*, 29:47–58, 1953.

[2] Albert Donally Bethke. *Genetic Algorithms as Function Optimizers.* PhD thesis, University of Michigan, January 1981.

[3] Peter F. Brown, John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Fredrick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roossin. A statistical approach to machine translation. *Computational Linguistics*, 16(2):79–85, June 1990.

[4] José María Catalá and Eugenia Tusquets. *Jorge El Curioso.* Houghton Mifflin Company, Boston, 1990. Translation of (Rey, 1941).

[5] Noam Chomsky. *Lectures on Government and Binding.* Foris Publications, Dordrecht, Holland, 1981.

[6] William Croft. *Typology and Universals.* Cambridge University Press, Cambridge, 1990.

[7] Sreerupa Das, C. Lee Giles, and Guo-Zheng Sun. Learning context-free grammars: Capabilities and limitations of a recurrent neural network with an external stack memory. In *Proceedings of The Fourteenth Annual Conference of the Cognitive Science Society*, 1992.

[8] Bonnie Jean Dorr. *Machine Translation : A View from the Lexicon.* MIT Press, Cambridge, MA, 1993.

[9] Charles Fillmore. The case for case. In *Universals of Linguistic Theory.* Holt, Rinehart, and Winston, New York, 1967.

[10] Charles Fillmore, Paul Kay, and Mary O'Connor. Regularity and idiomaticity in grammatical constructions: The case of let alone. *Language*, 64:510–538, 1988.

[11] Statistical and neural network approaches to natural language processing. workshop following *Advances in Neural Information Processing Systems 1994*, December 1994.

[12] Fred Glover and Manuel Laguna. Tabu search. In *Modern Heuristic Techniques for Combinatorial Problems.* Blackwell Scientific Publications, 1992.

[13] Fred Glover, Eric Taillard, and Dominique de Werra. A user's guide to tabu search. unpublished monograph, 1991.

[14] T. R. G. Green. The necessity of syntax markers: Two experiments with artificial languages. *Journal of Verbal Learning and Verbal Behavior*, 18:481–96, 1979.

[15] Joseph H. Greenberg. Some universals of grammar with particular reference to the order of meaningful elements. In Joseph H. Greenberg, editor, *Universals of Grammar.* MIT Press, Cambridge, MA, 1966.

[16] John E. Hopcroft and Jeffrey D. Ullman. *Formal Languages and Their Relation to Automata.* Addison-Wesley Publishing Company, Reading, Mass., 1969.

[17] John E. Hopcroft and Jeffrey D. Ullman. *Introduction to Automata Theory, Languages, and Computation.* Addison-Wesley Publishing Company, Reading, Mass., 1979.

[18] Ray S. Jackendoff. *$\bar{X}$ Syntax : A Study of Phrase Structure.* MIT Press, Cambridge, MA, 1977.

[19] Daniel Jones, editor. *International Conference on New Methods in Language Processing (NeMLaP).* Centre for Computational Linguistics, UMIST, Manchester, UK, 1994.

[20] Daniel Jones and Melina Alexa. Towards automatically aligning German compounds with English word groups in an example-based translation system. In *Proceedings of the International Conference on New Methods in Language Processing (NeMLaP)*, pages 66–70, Manchester, UK, September 1994.

[21] Patrick Juola. Machine translation and lojban. *Ju'i Lojypli*, 8, February 1989.

[22] Patrick Juola, Chris Hall, and Adam Boggs. Morphological segmentation by information theory. Technical Report unassigned, Computer Science Department, University of Colorado, In preparation.

[23] Edward Keenan and Bernard Comrie. Noun phrase accessibility and universal grammar. *Linguistic Inquiry*, 8:63–99, 1977.

[24] S. Kirkpatrick, C. D. Gelatt, Jr., and M. Vecchi. Optimization by simulated annealing. *Science*, 20:671–80, 1983.

[25] Judith L. Klavans and Phillip Resnik, editors. *The Balancing Act : Combining Symbolic And Statistical Approaches to Language (Proceedings of the Workshop).* Association for Computational Linguistics, 1994.

[26] Nenad Koncar and Gregory Guthrie. A natural language translation neural network. In *Proceedings of the International Conference on New Methods in Language Processing (NeMLaP)*, pages 71–77, Manchester, UK, September 1994.

[27] James L. McClelland, David E. Rumelhart, and the PDP Research Group. *Parallel Distributed Processing : Explorations in the Microstructure of Cognition.* MIT Press, Cambridge, Mass., 1987.

[28] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A.H. Teller, and E. Teller. Equations of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21:1087–92, 1953.

[29] James L. Morgan, Richard P. Meier, and Elissa L. Newport. Facilitating the acquisition of syntax with cross-sentential cues to phrase structure. *Journal of Memory and Language*, 28:360–74, 1989.

[30] Kazuo Mori and Shannon D. Moeser. The role of syntax markers and semantic referents in learning an artificial language. *Journal of Verbal Learning and Verbal Behavior*, 22:701–18, 1983.

[31] Eugene W. Myers. An O(ND) difference algorithm and its variations. *Algorithmica*, 1:251–56, 1986.

[32] Makoto Nagao. A framework of a mechanical translation between Japanese and English by analogy principle. In A. Elithorn and R. Barnerji, editors, *Artificial and Human Intelligence*, pages 173–80. North-Holland, 1984.

[33] Sergei Nirenburg, Jaime Carbonell, Masaru Tomita, and Kenneth Goodman. *Machine Translation : A Knowledge-based Approach.* Morgan Kauffmann Publishers, San Mateo, Calif., 1992.

[34] Partha Niyogi and Robert C. Berwick. A markov language learning model for finite parameter spaces. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics (ACL-94)*, pages 171–180, 1994.

[35] H. A. Rey. *Curious George.* Houghton Mifflin Company, Boston, 1941.

[36] H. A. Rey. *Curious George Takes a Job.* Houghton Mifflin Company, Boston, 1947.

[37] Dan Isaac Slobin. *Psycholinguistics.* Scott, Foresman, and Company, Glenview, Ill., second edition, 1979.

[38] Dan Isaac Slobin. Crosslinguistic evidence for the language-making capacity. In Dan Isaac Slobin, editor, *The Cross-Linguistic Study of Language Acquisition*, volume 2 : Theoretical Issues, chapter 15, pages 1157–1256. Lawrence Erlbaum Associates, Inc., 365 Broadway, Hillsdale, New Jersey, 1985.

[39] Jonathan Slocum, editor. *Machine Translation Systems.* Cambridge University Press, Cambridge, 1988.

[40] Tony C. Smith and Ian H. Witten. Language inference from function words. Technical Report 1993/3, University of Waikato, New Zealand, Jan 1993.

[41] Danny Solomon and Mary McGee Wood. Learning a radically lexical grammar. In *The Balancing Act : Combining Symbolic and Statistical Approaches to Language (post-ACL'94 workshop)*, pages 122–130, Las Cruces, New Mexico, July 1994.

[42] Harold Somers, Ian McLean, and Daniel Jones. Experiments in multilingual example-based generation. In *3rd International Conference on the Cognitive Science of Natural Language Processing (CSNLP-94)*, Dublin, Ireland, July 1994.

[43] E. Sumita and H. Iida. Experiments and prospects of example-based machine translation. *Proceedings of the ACM*, 1991.

[44] E. Sumita, H. Iida, and H. Kohyama. Translating with examples : A new approach to machine translation. In *The Third International Conference on Theoretical and Methodological Issues in Machine Translation of Natural Language*, 1990.

[45] Leonard Talmy. The relation of grammar to cognition. In Brygida Rudzka-Ostyn, editor, *Topics in Cognitive Linguistics*, pages 165–205. John Benjamins Publishing Co., Amsterdam/Philadelphia, 1988.

[46] Aziz ur Rahman. *Teach Yourself Urdu in Two Months.* Azizi's Oriental Book Depot, II, K, 14/4, Nazimabad, Karachi–18, Pakistan, 22nd edition, 1958.

[47] Mary McGee Wood. A categorial syntax for coordinate constructions. Technical Report UMCS-89-2-1, Department of Computer Science, University of Manchester, 1989.

[48] Mary McGee Wood. *Categorial Grammars.* Routledge, London, 1993.