Analysis, Statistical Transfer, and Synthesis in Machine Translation

Peter F. Brown, Stephen A. Della Pietra,
Vincent J. Della Pietra, John D. Lafferty, and Robert L. Mercer

We reinterpret the system described by Brown et al. [1] in terms of the analysis-transfer-synthesis paradigm common in machine translation. We describe enhanced analysis and synthesis components that apply a number of simple linguistic transformations so the transfer component operates from a string of French morphemes to a string of English morphemes. We report the results of a comparison of the new system with the old system on 100 short test sentences. The new system correctly translates 60% of these sentences while the old system correctly translates only 39% of them.

1.1 Introduction

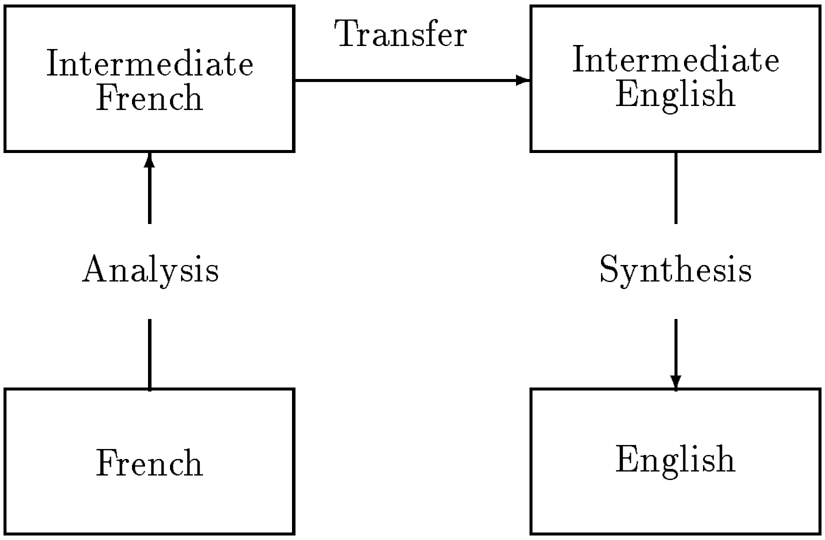
The analysis-transfer-synthesis architecture shown in Figure 1.1 is one of the classical paradigms in machine translation. The analysis component recasts the source sentence into an intermediate form, the transfer component reworks this intermediate form into a second intermediate form more compatible with the target language, and the synthesis component constructs the target language translation of the original source sentence from this new intermediate form.

Brown *et al.* [27] describe a statistical model for generating English sentences and for translating these sentences into French. They show that this model can be combined with a stack-based search strategy to make a system for translating sentences from French to English. Their system is an example of the analysis-transfer-synthesis architecture in which the analysis and synthesis components have become vestigial: their analysis component simply transforms character strings into strings of French words while their synthesis component carries out the reverse, transforming strings of English words into character strings. Their statistically based transfer component carries out the transformation from French words to English words directly.

In this paper, we elaborate the analysis and synthesis components. Figure 1.2 shows the structure of our system including its analysis and synthesis components. Notice that we do not

This paper first appeared in *Proceedings of the Fourth International Conference on Theoretical and Methodological Issues in Machine Translation*, 1992. pp.83–100.

model English character strings directly, but rather the intermediate English text. Brown *et al.* [27] make use of a large collection of aligned French-English sentence pairs [28] as data from which they algorithmically extract the parameters of their translation and language models [33]. In order for us to extract the parameters of our translation and language models, we need a large sample of input-output pairs from the English-to-French translation model. It is important, therefore, that the transformation induced by the synthesis component be invertible, since then, by passing the French member of a French-English sentence pair through the analysis component, and the English member of the pair through the inverse of the synthesis component, we can create an input-output pair for the translation model.



The traditional architecture for a machine translation system consists of the three stages of analysis, transfer and synthesis. A French sentence is first analyzed and thereby converted into an intermediate structure which captures the linguistic relationships between different components of the sentence. This structure is then transferred into an intermediate English structure. Finally, an English sentence is synthesized from the intermediate English structure.

Figure 1.1: Traditional machine translation architecture

From the data processing theorem [53], we know that we can expect neither the analysis component nor the inverse of the synthesis component to add information to a sentence. At best, they can rearrange information that is already present; at worst, they can actually destroy information, mapping two or more distinct sentences into the same intermediate structure. Sometimes, we intend to destroy information as, for example, when we correct misspelled words or choose a canonical spelling for words with several variants, but the main value of these components comes from making available locally to our primitive statistical models information that is manifest from the global structure of a sentence.

In the remainder of the paper, we describe the five steps that make up the analysis component and the inverse of the synthesis component for our new system and present results showing their effect on the performance of the system. These steps are:

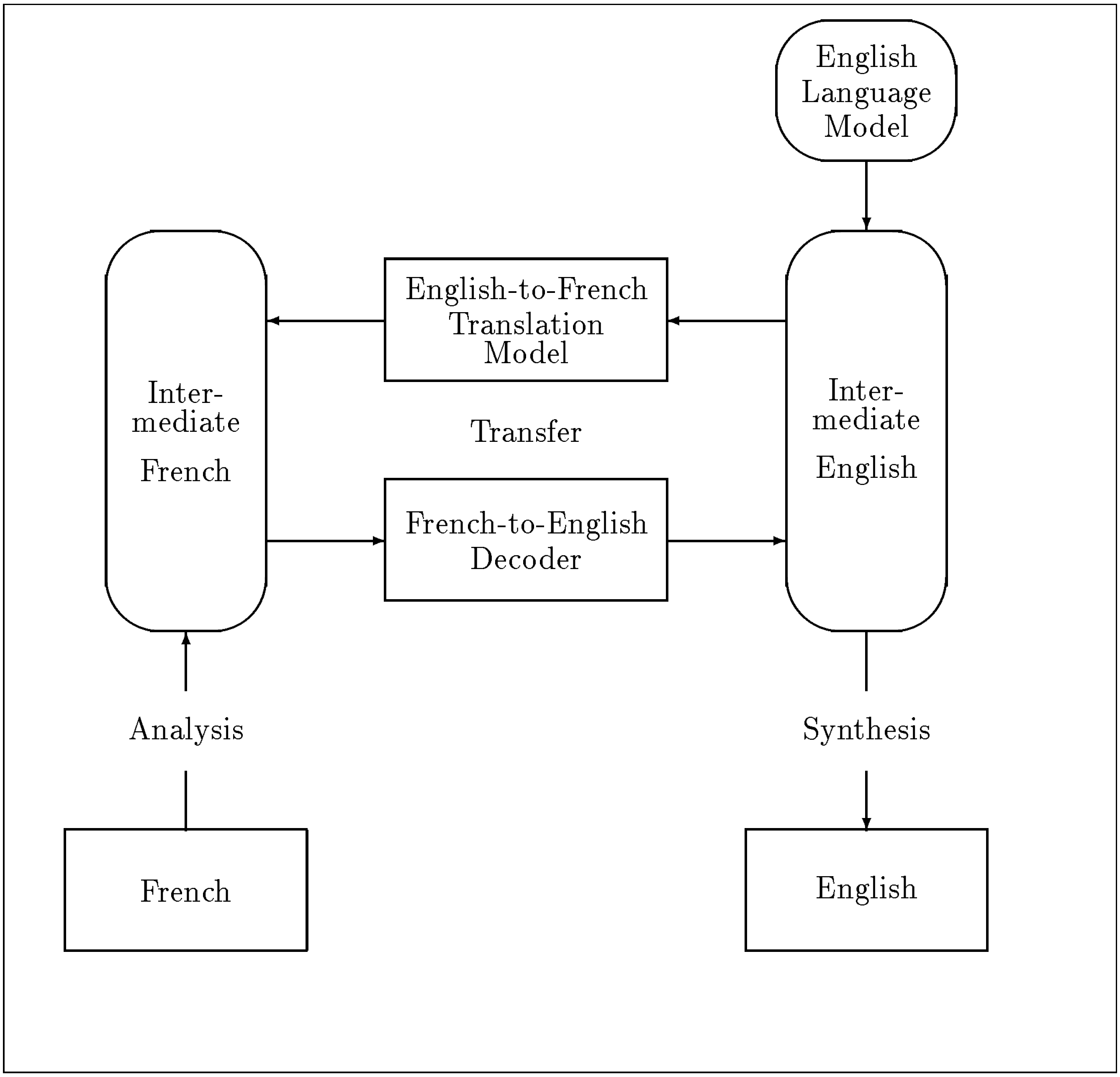


Figure 1.2: Statistical transfer in an analysis-transfer-synthesis architecture

1. Transform character strings to word strings.
2. Annotate words according to their grammatical function.
3. Apply some rudimentary syntactic analysis.
4. Extract inflectional morphology.
5. Assign statistically derived senses to some of the common words.

The resulting intermediate representation, for both the French and the English text, is a string of morphs some of which are limited by sense designations.

1.2 Words from Text

Simplicio is discussing the nature of words with his master Salviati. Let's listen:

Simplicio: How do you find words in text?
Salviati: Words occur between spaces.
Simplicio: What about *however,*? Is that one word or two?
Salviati: Oh well, you have to separate out the commas.
Simplicio: Periods too?
Salviati: Of course.
Simplicio: What about *Mr.*?
Salviati: Certain abbreviations have to be handled specially.
Simplicio: How about *shouldn't*? One word or two?
Salviati: One.
Simplicio: So *shouldn't* is different from *should not*?
Salviati: Yes.
Simplicio: And *Gauss-Bonnet* as in the *Gauss-Bonnet Theorem*?
Salviati: Two names, two words.
Simplicio: But if you split words at hyphens, what do you do with *vis-à-vis*?
Salviati: One word—don't ask me why.
Simplicio: How about *stingray*?
Salviati: One word, of course.
Simplicio: And *manta ray*?
Salviati: One word: it's just like *stingray.*
Simplicio: But there's a space.
Salviati: Too bad!
Simplicio: How about *inasmuch as*?
Salviati: Two.
Simplicio: Are you sure?
Salviati: No.

The intimate familiarity with reading and writing that we all share predisposes us to a ready acceptance of the idea that a body of text is a collection of words strung together when, in reality, it is a collection of characters strung together. In the main, the passage from a string of characters to a string of words is uneventful with words being delineated by blanks and punctuation, but at intervals it becomes tortuous. Successful navigation through a lengthy text demands many, often arbitrary, decisions. We encode many of these decisions in a list of several thousand character sequences that we treat as single words. We also analyze a number of character sequences into two or more words, writing, for example, *a les* for *aux, de le* for *du,* and *it was not* for *'twasn't.* Except for hyphens connecting two words, we treat individual punctuation marks as words. We also treat digits as words.

In order to distinguish between sequences like *1 2* and *12,* we attach a code to each word showing how it is connected to the previous word. For English text, we use three codes according as the word is separated by a space from the previous word, connected directly to it, or separated from it by an intervening hyphen. Thus, *light house* may represent *light house, lighthouse,* or *light-house* depending on the value of the connection code attached to *house.* For French text, we use two additional codes that allow us to represent, for example, *a-t-il* and *qu'il* as *a il* and *que il* depending on the value of the connection code attached to *il.*

1.2.1 Case

We must also deal with the complication presented by uppercase and lowercase letters. Simplicio is again giving his master a hard time:

Simplicio: When do two sequences of characters represent the same word?

Salviati: When they're the same sequence.

Simplicio: So *the* and *The* are different words?

Salviati: Don't be ridiculous! You have to ignore differences in case.

Simplicio: So *Bill* and *bill* are the same word?

Salviati: No. *Bill* is a name and a *bill* is something you pay. With proper names, case matters.

Simplicio: What about the two *May*'s in *May I pay in May?*

Salviati: The first one is not a proper name. It's only capitalized because it's the first word in the sentence.

Simplicio: But how do you know when to ignore case and when not to?

Salviati: You just know!

Sadly, computers do not just know: they have to guess. Happily, the entropy of case is only 0.04 bits per letter [30], so guessing is not entirely out of the question. We imagine each word to consist of an uncased *token* together with a *case pattern* that specifies the case of each letter. The case pattern of a word in context is a corrupted version of the *true-case* pattern that it would have in the absence of typographical error or arbitrary convention. Thus, in *Today, John works for MacPherson at IBm,* the first and last words have as tokens *TODAY* and *IBM,* as case patterns UL^+ and UUL^+, and as true-case patterns L^+ and U^+. The case and true-case patterns agree for the remaining words in this example.

We assign true-case patterns using the following algorithm:

1. If the word is part of a name, choose as its true-case pattern the most probable true-case pattern for the word that also begins with U.

2. Otherwise, if the word belongs to a list of words that have a unique true-case pattern choose that pattern.

3. Otherwise, if the word begins a sentence, choose as its true-case pattern the most probable true-case pattern for the word.

4. Otherwise, choose as the true-case pattern the case pattern.

We recognize names with a finite-state machine that incorporates a list of 12,937 common last names and 3,717 common first names, as well as a number of onomastic antecedents (such as *Mr.*, *Mlle.*, *Dr.*, etc.) and a number of onomastic consequents (such as *Jr.*, *Sr.*, *Ph.D.*, etc.).

We assign a word to the list of words with a unique true-case pattern provided the entropy of case patterns for the word is less than 0.3 bits. In addition, we have examined the 40,000 most frequent English words and assigned a unique true-case pattern to 9,144 of them. We have also examined the 10,000 most frequent French words and have assigned a unique true-case pattern to 3,794 of them.

We have determined the most probable case pattern for each of the remaining words by examining a collection of 67 million English words and a collection of 72 million French words, in each case excluding words that begin sentences.

1.2.2 The Data

Using the steps described above, we have processed 1,778,620 pairs of French and English sentences from our Canadian Hansard corpora [28]. Because many of the words that appear only once in this collection are typographical errors, we excluded all such singletons from our vocabularies. In this way, we arrived at an English vocabulary of 40,806 words, and a French vocabulary of 57,800 words. We replaced all singletons in both texts with the *unknown word.*

1.3 Part-of-Speech Annotation

As a prelude to syntactic and morphological analysis, we tag words in context with parts of speech to show their grammatical function. We use 163 tags for the English text and 157 tags for the French text, roughly categorized as shown in Table 1.1. We employ a statistical, hidden Markov model tagging algorithm [13, 11] embodied in a set of programs developed by Merialdo [77].

Tagging algorithms of this type are most successful when their parameters are extracted from a large body of hand-labelled data. We had at our disposal 1.9 million words of hand-labelled English text, divided about evenly between text from the Associated Press newswire and text from the English half of our Hansard data. This data was labelled at Lancaster University under the direction of Geoff Leech. We used 1,666,191 words of this data for training and 232,090 words for smoothing. On the remaining 23,062 words of test data, the trained system correctly labels 94% of the words.

We also had available 1,283,344 words of French text from a variety of sources collected and labelled by our colleagues at the IBM Paris Scientific Center [47], and a second set of 27,454 hand-labelled words from the French part of our Hansard data. We trained the parameters using the larger set of data and smoothed them using the smaller set [63, 77]. Because of the small quantity of hand-tagged French Hansard data, we took two additional steps in the hope of better imprinting the stamp of Hansard French on our parameters. First, we re-estimated the parameters by running one iteration of the forward-backward algorithm on an additional corpus of 13,433,404 words of untagged data from the French part of our Hansard corpus [15, 8, 77]. Finally, we used the 27,454 words of tagged Hansard data once again to smooth these new estimates. With this system, we correctly tagged 93% of the words in a new set of 24,649 words of Hansard French hand-labelled for us by our colleagues at the IBM Paris Scientific Center.

1.4 Syntactic Analysis

We do not actually perform any syntactic analysis of either the French or the English texts. Instead, we carry out a number of syntactically motivated transformations designed to make sentences in the two languages more similar to one another. Each transformation is made with the aid of a finite state recognizer. Of course, neither English nor French can be described by a simple finite state mechanism. In some cases, therefore, our simple rules will fail to apply where they should or will apply where they should not. While this is regrettable, we take a purely

Broad Category	Number of Refined Tags	
	French	English
Nouns	7	29
Verbs	44	27
Adjectives	4	8
Adverbs	4	16
Pronouns	66	20
Determiners	12	17
Prepositions	2	4
Conjunctions	4	10
Punctuation	2	12
Other	12	20
Total	157	163

Table 1.1: Parts of speech by broad category

pragmatic attitude toward these errors: if the performance of the system improves when we use a transformation, then the transformation is good, otherwise it is bad.

1.4.1 English Transformations

We apply two transformations to English sentences:

1. We undo question inversion when we can find it;

2. We move adverbs out of multiword verbs.

The primary purpose of these transformations is to place the words in a multiword verb in sequence so as to facilitate later morphological analysis. The secondary purpose is to reduce the local statistical variety of English sentences.

Question Inversion

One of the signals of the interrogative in English is the inversion of the subject and the first word of the verb. Speakers of American English prefer to invert the subject with an auxilliary verb rather than a main verb, and so are more comfortable adding some form of the empty auxilliary *do*. It is our intention that our question inversion transformation work as follows:

Has the grocery store any eggs?
⇒ *The grocery store has any eggs QINV*

Will the President run for election again?
⇒ *The President will run for election again QINV*

Why should farmers be growing less wheat?
⇒ *Why farmers should be growing less wheat QINV*

Because of errors in grammatical tagging, compounded with the primitive nature of the rules that we employ to achieve this goal, we succeed only about 40% of the time.

Adverb Movement

We move an adverb that is adjacent to a verb, or contained within a multiword verb, to a position immediately following the verb and mark it to show where it originated. We treat *not* as an adverb for this purpose, and when there is an empty use of a form of *to do* in the vicinity, we combine it with the *not* and treat the combination as an adverb. Thus, we intend the following types of transformations

John does not like turnips.
$\Rightarrow$ *John likes do_not_M1 turnips.*

Iraq will probably not be completely balkanized.
$\Rightarrow$ *Iraq will be balkanized probably_M2 not_M2 completely_M3 .*

Here, the *M1* at the end of *do_not* shows that in the original sentence it preceded the first word (in this case, the only word) of the verb; the *M2* appended to both *probably* and *not* shows that they originally preceded the the second word in the sequence *will be balkanized*; and the *M3* at the end of *completely* shows that it preceded *balkanized.*

We feel that the statistical connection between the subject and the verb is stronger than that between the verb and its object. Therefore, in order to make the best use of the trigram model that we employ for predicting the intermediate English text, we move adverbs to the end of the verb sequence rather than, for example, to the beginning. In this way, the subject and the verb are more likely to fall within the same three-word sequence. Of course, it would be better to move these adverbs out of the way altogether so that we could capture not only the dependence of the verb on its subject, but also the dependence of the object on the verb and on the subject. A more satisfying treatment of this kind must await the development of more general language modelling techniques.

1.4.2 French Transformations

We apply four transformations to French sentences:

1. We undo question inversion.

2. We combine pairs like *ne* ... *pas, ne* ... *rien,* etc. into single words.

3. We move pronouns that function as direct, indirect, or reflexive objects of verbs to a position following the verb and mark them to show their function.

4. We move adjectives to a position preceding the nouns that they modify and adverbs to a position following the verbs that they modify.

These tranformations facilitate the morphological analysis of multiword verbs and also move French a little bit in the direction of English.

Question Inversion

In French as in English, the interrogative is often signalled by inversion of the subject and the verb. Unravelling this is easier in French than in English because, when the subject is a pronoun, the French mark the disturbed words by connecting them with a hyphen. When the subject is not a pronoun, the subject and verb retain their declarative order, but a pronoun that agrees with the subject is added after the verb and attached to it by a hyphen. It is our intention that our question inversion transformation work as follows:

Mangez-vous des légumes?

⇒ *Vous mangez des légumes QINV1*

Où habite-il?
⇒ *Où il habite QINV1*

Le lui avez-vous donné?
⇒ *Vous le lui avez donné QINV1*

Jean mange-t-il comme un cochon?
⇒ *Jean mange comme un cochon QINV2*

In these examples, the digit after *QINV* distinguishes between the case when we invert the verb and its pronoun subject and the case when we make that inversion and then discard the pronoun. We successfully unscramble question inversion about 80% of the time. Because it is sometimes difficult to recognize a complex subject, we make most of our mistakes in the *QINV2*-type questions.

The French can also construct questions by attaching the sequence *est-ce que* to the front of the corresponding declarative sentence. Therefore, we also perform transformation like the following: *Est-ce que vous mangez des légumes?*
⇒ *Vous mangez des légumes EST_CE_QUE*

Est-ce que vous le lui avez donné?
⇒ *Vous le lui avez donné EST_CE_QUE*

Est-ce que Jean mange comme un cochon?
⇒ *Jean mange comme un cochon EST_CE_QUE*

Dealing with *ne ... pas*

Often in a French sentence one finds the verb sandwiched between *ne* and some other word which together serve to negate or otherwise modify the meaning of the sentence. It is our intention to make transformations like the following:

Je ne sais pas.
⇒ *Je sais ne_pas.*

Il n'y en a plus.
⇒ *Il y en a ne_plus.*

Jean n'a jamais mangé comme un cochon.
⇒ *Jean a ne_jamais mangé comme un cochon.*

Sometimes, we fail to find the second member of the pair, and so we succeed only about 75% of the time.

Moving Object Pronouns

In French, the definite articles *le, la, l',* and *les* can also be used as direct objects. In this use, they precede the verb of which they are the object. When we encounter these or other object pronouns before a verb, we move them to a position following the verb and label them according to our understanding of the function that they serve. The following examples should make our intention clear.

Je vous le donnerai.
⇒ *Je donnerai le_DPRO vous_IPRO.*

Vous vous lavez les mains.
⇒ *Vous lavez vous_RPRO les mains.*

Je y penserai.
⇒ *Je penserai à y_PRO.*

J'en ai plus.
⇒ *Je ai plus de en_PRO.*

Notice that when moving the allative and ablative pronominal clitics (*y* and *en*), we also include a preposition. Some pronouns, such as *nous* and *vous* can function either as direct, indirect, or reflexive objects. If we are unsure of which rôle one of these words is playing, we tag it with _CPRO when it is moved. About 5% of the time we mis-tag a pronoun that we have moved.

Moving Adverbs and Adjectives

To make the French structures presented to the statistical models used in transfer as close a possible to the English structures, we move French adjectives in front of the nouns they modify. We do not record the fact that these adjectives have been moved, and so conflate such phrases as *un homme grand* and *un grand homme.* We will remedy this defect in future versions of our system.

We also move French adverbs to a position after the verbs that they modify.

1.5 Morphological Analysis

The translation system described by Brown *et al.* [27] treats words as unanalyzed wholes. From the fact that *parle* is translated as *speaks,* they adduce no evidence for the translation of *parlé* as *spoken.* But even regular verbs in French have many distinct forms, some of which can be quite rare. In a 30 million word sample of French text from our Hansard data, only 24 of the 35 different forms of the verb *parler* actually occur. For less common verbs, fewer than half of the possible forms may be realized in the data. This effusion of disguises for the same underlying object dilutes the effectiveness of our training procedure.

We perform simple inflectional morphological analysis of verbs, nouns, adjectives, and adverbs so that the fraternity of the several forms of the same word is manifest in the intermediate structure. In English, we analyze the several conjugations of the same verb; the singular and plural forms of the same noun; and the positive, comparative, and superlative forms of adjectives and adverbs. In French, we analyze the several conjugations of the same verb; and masculine, feminine, singular, and plural forms of the same noun or adjective. In both languages, we analyze each verb into a tense marker and an infinitive.

The examples below illustrate the level of detail in our morphological analysis.

He was eating the peas more quickly than I.
⇒ *He PAST_PROGRESSIVE to_eat the pea N_PLURAL quick er_ADV than I .*

Nous en mangeons rarement.
⇒ *Nous 1ST_PERSON_PLURAL_PRESENT_INDICATIVE manger rare ment_ADV de en_PRO.*

Ils se sont lavés les mains sales.

$\Rightarrow$ *Ils 3RD_PERSON_PLURAL_PAST laver se_RPRO les sale main N_PLURAL .*

Notice in the last example that we retain no indication of the original number on French adjectives. We also discard any distinction in gender. Thus, in the intermediate French, adjectives always appear in their masculine singular form.

1.6 Sense Disambiguation

In a recent paper, Brown *et al.* [32] describe a method dividing the occurrences of a word in context into a small set of senses so as to achieve a high mutual information between the translation of a word and its sense. In the translation model that we use, we assume that each English word acts independently of the other English words in a sentence to generate a series of French words [27, 33]. By labelling the words in the intermediate French and English structures with senses that reflect the context in which they occur, we provide some global contextual information to what is essentially a local model of the translation process.

We assign senses to 1000 of the most frequent French words. For example, we map *prendre* to *prendre_1* in the sentence

Je vais prendre ma propre voiture,

but to *prendre_2* in the sentence

Je vais prendre ma propre décision.

In the corresponding final step of the inverse of the synthesis component, we assign senses to 1000 of the most frequent English words.

1.7 Experimental Results

We have compared the performance of a translation system incorporating the analysis and synthesis components described above to a simpler system in which the analysis and synthesis components carry out only the first step of the complete five-step procedure. In both systems, we use a trigram language model in the transfer component as compared with a bigram language model as described by Brown *et al.* [27].

We restrict our attention to vocabularies of 40,809 English words and 57,802 French words. In the enhanced system, morphological analysis reduces these to 33,941 English morphemes and 31,115 French morphemes.

We estimated the parameters of the translation model for each system from a set of 1,778,620 pairs of French and English sentences from the Canadian Hansard data [27, 28]. Each of these sentences is 30 words or less in length. We tested both systems on the same set of 100 randomly selected Hansard sentences each containing at most 10 words. We judged as acceptable 39 of the translations produced by the simpler system as compared with 60 of those produced by the enhanced system.

1.8 Discussion

We have described analysis and synthesis components for use in a statistical translation system. Each of the transformations that make up these components is achieved with the aid of a simple finite-state recognizer. Many of them work poorly and yet, together, they produce a system with a significantly higher translation accuracy. Much of the credit for this successful performance in the face of adversity must be laid at the door of the statistical transfer component, which frames no hypotheses but is guided entirely by the training data.

In work of this type, it is desirable to be able ascribe certain increments of performance to certain of the steps in the analysis or synthesis component, and thus to assess the value of the various transformations. Making such an assessment would require of us that we construct a series of analysis and synthesis components with different members of the series including different ones of the steps that make up the complete system. Unfortunately, each such construction must have a differently trained statistical transfer component. Because training is a costly undertaking, we have not made any of these collateral investigations and are, therefore, unable to say which of the new analysis and synthesis steps is the most valuable.