COLING 82, J. Horecký (ed.)
North-Holland Publishing Company
© Academia, 1982

A METRIC SPACE DEFINED ON ENGLISH AND ITS RELATION TO ERROR CORRECTION

James Bradford

School of Computer Science
Acadia University
Wolfville, Nova Scotia
Canada

A distance function is proposed that maps pairs of strings to the real numbers. It has been shown that given suitable constraints the function is a metric over the free monoid generated from a set of grammatical symbols. The necessary constraints modify the metric so that it maps pairs of strings to a lattice of real numbers. Thus for each string the metric defines a countable set of nested neighbourhoods. This aspect of the space has proved useful for the correction of certain kinds of grammatical errors that occur in English sentences. An English parser was written that used the metric to propose corrections to a variety of ungrammatical sentences. Experience with the program suggests that in many cases the intuitive notion of grammatical similarity corresponds closely to the mathematical definition of nearest neighbour in the space.

1. Introduction

Consider a string of grammatical symbols which has been produced by lexical analysis. Each symbol in the string corresponds to a word in the original sentence. The string will be analysed by a parser which compares the sequence of symbols to sequences specified by some grammar, G. If the comparison succeeds then the original sentence is accepted as grammatical. Otherwise, it is rejected and error correction is required.

Definition: Given a grammar G and a string S composed of grammatical symbols from some alphabet A then S is ungrammatical if it is not contained in L(G), the language generated by G.

Ungrammatical in this sense refers to any sentence that was not anticipated by the grammar. In many systems it is possible for a user to produce a proper English sentence within the appropriate domain of discourse and still have the sentence rejected by the parser. This is usually attributed to "holes in the grammar."

This paper will describe a technique for correcting ungrammatical input. The class of errors treated includes both genuine grammatical errors and those resulting from "holes." One of the assumptions tested by this work is that a significant class of errors can be resolved by examination of syntactic structure alone.

An ungrammatical sentence is viewed as a grammatical sentence that has been transformed by one or more error operations.

Definition: An error operation involves either (a) an insertion of a word, (b) a deletion of a word, or (c) an alteration of the word sequence.

In general, the damage done by a single error operation is local and does not significantly alter the global structure.

Thus a comparison of the respective structures of the two sentences is used as the basis for a measure of their similarity. This approach is based on earlier work by Fischer and Wagner.[1] The error correction strategy rests on a measure which expresses structural similarity as a numerical distance. If the parser's analysis of a of a given sentence fails then a search is made for its nearest grammatical neighbour. As various alternatives are found they are presented to the user. The user may elect to continue the search, accept the corrector's proposal or abandon the search and rephrase the input.

The class of errors that can be corrected by a measure of structural similarity are those related to word arrangement. Word arrangement is described by an augmented transition network in which the conditions on the arcs are totally relaxed. Such a net is called a recursive transition network and it defines a context free language.[2] Thus the class of errors treated by this technique are called context free errors.

2. The Measure - Informally

The basis of the distance function is a value called the transformation cost. In essence the transformation cost gives an indication of the number of changes required to convert one string of grammatical symbols into another. The changes are considered under two categories, rearrangement and edition. The cost of a transformation is the sum of the cost of rearrangment and cost of edition.

The rearrangement cost measures the amount of disorder of one string relative to another. Many definitions are possible but most yield asymmetrical costs. One that does not, considers common substrings between the two given strings. The cost is based on the number of gaps between the substrings. For example, if the two given strings match exactly then the rearrangement cost is zero because the two strings match without gaps.

The edition cost considers the symbols that occupy the gaps between substrings. In order to transform one string into another the symbols not part of common substrings in the first must be removed and those in the second that are not common must be inserted into the first. The edition cost is the sum of the costs of insertion and deletion. Clearly the potential for asymmetry exists here as well. However, if the cost of insertion is equal to the cost of deletion for any given symbol then symmetry follows as a consequence.

The most significant element of the formal description of Eta's distance measure is the concept of a match set. Suppose we consider two sequences of words (actually strings of grammatical symbols, each corresponding to a word). The A-sequence will be the input and the B-sequence will be a sentence in the grammar - henceforth called the *test sentence*. Thus a match set M with respect to A and B describes a pairing between words in the input and words in the test sentence. If the two sentences match exactly and are both of length n then the match set denoting the best match will be: $(1,1),(2,2),\ldots,(n,n)$. Notice that the integers comprising each ordered pair are the positions of words in the two respective sentences.

The rearrangement cost (which measures the disorder of one sentence relative to another) is computed from M. Although the cost is related to the number of common substrings of words shared by the two given sentences the actual cost is computed by counting gaps between substrings. For example, suppose two sentences have no words in common. Since there are no shared substrings the word order of the two sentences are not related and thus the rearrangement cost is zero (hence the entire transformation cost will derive from the edit cost). If the two sentences were identical then in this case as well there will exist a match set yielding a rearrangment cost of zero.

The edit cost is also computed from the match set M. In a manner similar to the Fischer/Wagner measure it is assumed that each grammatical symbol has two associated unit costs, the *cost of insertion and the cost of deletion*. The underlying idea is *that after the input has been rearranged* to match the test sentence then nonmatching symbols in the input are removed and unmatched symbols in the test sentence are inserted into the modified input. In practice it is the sum of the unit costs that is used as the edit cost. Because of the nonnegativity and nondegeneracy requirements for a metric the unit edit costs must be positive.

3. The Measure - Formally

Notation 1. If A is a set then the cardinality of A is denoted $|A|$.

2. If $m = (i,j)$ is an ordered pair of integers then $D(m) = i$ and $R(m) = j$

3. If $S = s_1,\ldots,s_n$ is a sequence of symbols then $s^{<i>} = s_i$ where $1 \leq i \leq n$

Definition 1 *Match Set, M*

If A and B are given strings then a match set M with respect to A and B is a set of ordered pairs of integers with the following properties. If $m,n \in M$ and $m \neq n$ then 1. $D(m) \in [1, |A|]$
2. $D(m) \neq D(n)$
3. $R(m) \in [1, |B|]$
4. $R(m) \neq R(n)$
5. $m=(a,b) \rightarrow A<a>=B<b>$

Definition 2 *Inverse Match Set, M^{-1}*

If M is a given match set with respect to two strings A and B then M^{-1} is a match set with respect to B and A such that
1. $|M^{-1}|=|M|$
2. $m \in M \rightarrow \exists n \in M^{-1}$ where (i) $D(m)=R(n)$ (ii) $R(m)=D(n)$

Notation If a,b and n are integers such that $a+b<2n$ then

$$a+_n b = \begin{cases} a+b \text{ if } a+b\leq n \\ a+b-n \text{ if } a+b>n \end{cases}$$

Notice that if $a+b\leqslant 2n$ then

$$a+_n b = [(a+b-1) \bmod n]+1$$

Definition 3 Successor Function, succ(m)

If M is a match set with respect to two strings A and B and if $|A| = a$, $|B|=b$ and $(i,j)\varepsilon M$ then $\mathrm{succ}((i,j)) = (i+_a 1, j+_b 1)$

Recall that the rearrangement cost is based on the number of gaps between substrings. A gap is detected by means of a successor function. The successor of an ordered pair is the pair produced by incrementing each element of the initial pair by 1. Thus the successor of (2,3) is (3,4).

An unusual aspect of the successor function is that for any given sentence, the first word is defined to be the successor of the last. For example, if the length of the two sentences was n, then the successor of (n,n) is defined to be (1,1). A metric must yield a distance from a string to itself of zero. This is the reason underlying the successor function's "wrap around" characteristic.

Definition 4 Gap Set, G

If M is a given match set then G is defined by $G=\{m|\, m\varepsilon M\wedge \mathrm{succ}(m)\not\varepsilon M\}$

Definition 5 Rearrangement Cost, $\Omega(M)$

If M is a match set and G the associated gap set then $\Omega(M)=|G|$

Convention 1 states that the cost of inserting or deleting any grammatical symbol is constant. In other words the cost of a unit edit operation is independent of the symbol being edited.

Convention 1 Let Σ be an alphabet of symbols and c be a positive real constant. Convention 1 requires $\forall s\varepsilon\Sigma\gamma \mathrm{INS}(s)=\gamma \mathrm{DEL}(s)=c$

Recall that the edit cost between two strings A and B is based on the unit costs of inserting and deleting symbols not common to both strings. A definition of the edit cost, $\Gamma(A,B,M)$ based on this is given in reference 3. For two strings A and B and a given match set M, Lemma 1 establishes the equivalence of a more convenient definition. The proof of Lemma 1 is also given in reference 3.

Lemma 1 If M is a match set with respect to two strings A and B and if convention 1 is in force then $\Gamma(A,B,M)=(|A|+|B|-2|M|)c$

Definition 6 Transformation Cost, TCOST(A,B,M)

If M is a match set with respect to two strings A and B then $\mathrm{TCOST}(A,B,M)=\Omega(M)+\Gamma(A,B,M)$

Definition 7 Match Set of Minimal Cost

If U_{AB} is the class of all match sets with respect to A and B then a match set $M\varepsilon U_{AB}$ is said to be a match set of minimal cost (or simply "minimal") iff

$$TCOST(A,B,M)= \min_{N\varepsilon U_{AB}} (TCOST(A,B,N))$$

Definition 8 Edit Distance - Eta, η(A,B)

Let A and B be strings and M be a minimal match set with respect tc A and B. Then $\eta(A,B)=TCOST(A,B,M)$

Example Let A,B and C be strings of letters A = abcd, B = bdac, C = aab
Suppose $\forall s\varepsilon\Sigma$ $\gamma INS(s)=\gamma DEL(s) = 1$
Notice that M1={ (1,3),(2,1),(3,4),(4,2)}is minimal
Let $m\varepsilon M1$

m	succ (m)	succ (m)εM1?
1,3	2,4	False
2,1	3,2	False
3,4	4,1	False
4,2	1,3	True

Hence G ={ (1,3),(2,1),(3,4)}
$\Omega(M1)=3$, $\Gamma(A,B,M1)=0$
TCOST(A,B,M1)=3
Thus $\eta(A,B)=3$
Similarly $\eta(A,C)=4$
and $\eta(B,C)=5$

Theorem 1 If Convention 1 applies then (Σ^*,η) is a metric space. In particular, if $A,B,C\varepsilon\Sigma^*$ then

1. $\eta(A,B)\geq 0$
2. $\eta(A,A)=0$ and $A\neq B\rightarrow\eta(A,B)>0$
3. $\eta(A,B)=\eta(B,A)$
4. $\eta(A,C)\leq\eta(A,B)+\eta(B,C)$

4. Conclusions

A program called Eta (for Error Tolerant Analysis) was written to test the effectiveness of the measure. For a given grammar, G the program searches the metric space in the neighbourhood of the input until a sentence contained in L(G) is found. This sentence is given to the user for confirmation. If the program's proposal is rejected then the search is continued. The neighbourhood searched by the program consists of the set of strings of grammatical symbols within a given distance of the input. By progressively enlarging this distance a partial ordering is applied to the strings in L(G). Thus the user sees alternatives from "near" (structurally similar to the input) to "far" (structurally dissimilar).

Experience with the program suggests that many common grammatical errors can be corrected by relatively short dialogues with the user. Frequently an acceptable alternative is proposed within three or four interactions.

The extent to which the alternatives proposed by Eta are "likely" is, of course, subjective. Nevertheless, the measure does yield alternatives that are structurally similar to input that would otherwise defy analysis. In the majority of the cases seen, grammatical errors do leave much of a sentence's structure intact. Since there is no fixed limit on the number of alternatives that may be presented even pathological cases are correctable with patience.

References

[1] Fischer, M. J. and Wagner, R. A., The String to String Correction Problem, Journal of the ACM, Vol. 21, no. 1. January 1974, 168 - 173.

[2] Woods, W. A., Transition Netword Grammars for Natural Lanuage Analysis, Communications of the ACM, Vol. 13, no. 10, October 1970, 591 - 606.

[3] Bradford, J. H., The Eta Interface - An Error Correcting Parser for Augmented Transition Networds, Ph.D. Thesis, Dpeartment of Computer Science, University of Waterloo, 1982.