Chapter Five
Sequence Analysis Methods

This chapter and the next will focus on methods for analyzing discrete sequences of events. Sequence analysis methods do not require events to be aggregated to interval-level measures, as was done with the Goldstein-scaled data analyzed in Chapter 4, or in studies such as Ashley (1980), Goldstein and Freeman (1990), Ward and Rajmaira (1992), or Goldstein and Pevehouse (1996). Instead, the data are treated as a series of individual events occurring over time. This is far closer to how humans envision political behavior, but quite different than most existing statistical treatments of event data.

Event sequences are a key element in human reasoning about international events. Human analysts "understand" an international situation when they recognize sequences of political activity corresponding to those observed in the past. Empirical and anecdotal evidence point to the likelihood that humans have available in long-term associative memory a set of "templates" for common sequences of actions that can occur in the international system (and in social situations generally). Sequences can be successfully matched by human analysts in the presence of noise and incomplete information, and can also be used to infer events that are not directly observed but which are necessary prerequisites for events that have been observed.

While digital computers are exceptionally adept at numerical computation, they are not very good at pattern recognition. A pattern-recognition problem that is trivial for a human child (or a moth)—distinguishing lettuce from cabbage, for example—is beyond the capabilities of general-purpose computers, and difficult even for specialized equipment. In a similar fashion, it is substantially more difficult to develop computer algorithms for generalizing the characteristics of a set of discrete political event sequences than it is to generalize interval-level measures derived from those sequences. Furthermore most of the work that has been done on these problems occurs in fields quite distant from political science—notably linguistics and biology—rather than in the more familiar territory of economics and psychology. Nonetheless, we believe that the potential utility of these techniques outweighs the disadvantages of their esoteric character, and will therefore illustrate several such methods.

5.1. Analogical Reasoning in Foreign Policy Decision-making

The use analogy or “precedent-based reasoning” has been advocated as a key cognitive mechanism in the analysis of international politics by Alker (1987), Mefford (1985, 1991) and others, and is substantially different from the statistical, dynamic and rational choice paradigms that characterize most contemporary quantitative models of international behavior. Khong (1992) and Vertzberger (1990) review the general arguments in the cognitive psychology literature on use of analogy in political reasoning; May (1973) and Neustadt and May (1986) discuss it from a more pragmatic and policy-oriented perspective. As Khong observes:

> Simply stated, ... analogies are cognitive devices that “help” policymakers perform six diagnostic tasks central to political decision-making. Analogies (1) help define the nature of the situation confronting the policymaker; (2) help assess the stakes, and (3) provide prescriptions. They help evaluate alternative options by (4) predicting their chances of success, (5) evaluating their moral rightness and (6) warning about the dangers associated with options. (pg. 10)

The ubiquitousness of analogical reasoning is supported by a plethora of experimental studies in cognitive psychology in addition to the case studies from the foreign policy literature.

For a human decision-maker, analogical reasoning is a form of bounded rationality because “associative recall”—where the recall of one item naturally activates links to other items that have features in common (Anderson 1983; Kohonen 1984)—is an easy task for the human brain. In particular, associative recall is substantially easier for the human brain than sequential or deductive reasoning.

A “story” or, more formally, an event sequence, is a set of temporally ordered events and an associated context or set of preconditions. These are usually based on simplified versions of history, although some are based on counterfactuals (see Fearon 1991). Stories are easily transmitted and stored by individuals and the use of stories is universal in human culture. Whether sitting around the dying embers of a Neolithic campfire or sitting in the departure lounge of an airport, humans find relaxation in a good yarn.

The use of stories as a means of knowledge representation is strongly associated with the work of Roger Schank (e.g. Schank and Abelson 1977; Schank 1990):

> The form of memory organization upon which our arguments are based is the notion of episodic memory…organized around personal experiences or episodes rather than around abstract semantic categories.… [O]ne of the principal components of memory must be a procedure for recognizing repeated or similar sequences. When a standard repeated sequence is recognized, it is helpful in ‘filling in the blanks’ in understanding. (Schank and Abelson 1977, 18)

Schank and Abelson also relate sequences—their term is “scripts”—to the fundamental process of “understanding”

> In order to understand the actions that are going on in a given situation, a person must have been in that situation before. That is, understanding is knowledge-based. The actions of others make sense only insofar as they are part of a stored pattern of actions that have been previously experienced.
>
> … Understanding is a process by which people match what they see and hear to pre-stored grouping of actions that they have already experienced. New information is understood in terms of old information. (Schank and Abelson 1977, 67)

In international politics, understanding includes information that has been learned or deduced rather than experienced, but the principle is the same. “Understanding” a political situation means fitting observed events into a pre-existing event structure. The analyst’s “intuitive feel” is simply the mental ability to match a set of observed events to sequences he or she already knows.

Common story patterns are assigned to general categories, for example “crisis”, “war”, “coup”, or “revolution”. At the lowest level of aggregation, the elements of a story sequence are events—interactions that can be described by transitive verbs—but typically stories are constructed hierarchically, with complex sequences can be built out of simpler subsequences. Conversely a large amount of detail can be compressed into a few statements by absorbing the details into commonly known patterns.

For example, the Cuban Missile Crisis could be described in very general terms by the sequence:

USSR builds missile launchers in Cuba

USA discovers missile launchers

USA blockades Cuba

USA and USSR negotiate

USSR promises not to deploy missiles in Cuba

USA promises not to attack Cuba

This rendition is very simple but still sufficient to distinguish the Cuban Missile Crisis from, say, Desert Storm or the SALT negotiations. The event “USA blockades Cuba” could be expanded to

President Kennedy convenes Executive Committee of the National Security Council

ExComm considers six possibilities: do nothing, bomb, invade and blockade, negotiate internationally, negotiate with Castro.

“Do nothing” option is rejected because…

“Bomb” option is rejected because…

and so forth.

In the international conflict literature, Lebow’s “justification of hostility crisis” provides an example of a general episodic structure.

1. Exploit a provocation to arouse public opinion.
2. Make unacceptable demands upon the adversary in response to this provocation.
3. Legitimize these demands with reference to generally accepted international principles.
4. Publicly deny or understate your real objectives in the confrontation.
5. Employ the rejection of your demands as a *casus belli*. (Lebow 1981,29)

Lebow develops this sequence using crises such as the Austria/Serbia in 1914, Japan/China in 1931, Germany/Poland in 1939 and USA/Vietnam in 1964. The sequence also fits nicely the actions of Iraq towards Kuwait in the summer of 1990.

Table 5.1: Justification of Hostility Crises

Lebow sequence	July 1914 Crisis	Iraq-Kuwait 1990
Exploit provocation	Assassination in Serbia 28 June	Iraq accuses Kuwait of disregarding OPEC quotas, 17 July
Make unacceptable demands	Austrian ultimatum to Serbia 23 July	Iraq demands Kuwait forgive loans, make reparations, $25 oil price 23-25 July
Understate objectives	Austria claims it does not wish to destroy Serbia	Iraq assures Egypt it will not invade Kuwait
Use rejection of demands as casus belli	Serbia rejects some Austrian demands, 25 July	Breakdown of Jeddah talks, 1 August
War	Austria declares war, 28 July	Iraq invades, 2 August

Stories are generalized into ideal cases that we will call “templates”. When a decision-maker refers to the danger of a coup in El Salvador, this usually refers not to a specific coup but coups in general[1]. In order to apply a template to a specific case, a decision-maker uses substitution principles in combination with historical or idealized sequences of international events to create analogies:

Analogy = precedent + substitution principles

Substitution principles are primarily based on declarative knowledge about the actors involved—for example does the actor have allies; is it a major power; where is it

1 An exception occurs when there is a clear and obvious precedent: for example U.S. policy towards Ferdinand Marcos in the Philippines was discussed in terms of Marcos as "another Somoza", the Nicaraguan dictator whose fall led to the establishment of the Sandinista regime opposed by the United States.

located—although they may also involve contextual knowledge about the historical circumstances of the story (for example, did the story occur before, during or after the Cold War period).

Consider the template

[Tension between X and Y]
[Political instability in X]
[Y invades X]
[X consolidates power and repels Y's invasion]

If X=Iran and Y=Iraq, this describes the initial phases of the Iran-Iraq War circa 1980; if X=France and Y=assorted European monarchies it describes Europe circa 1790; if X=Russia and Y=assorted capitalist states it describes the allied intervention in the Russian Revolution in 1918-1920; if X=Bulgaria and Y=Serbia, it describes the Serbo-Bulgarian war in 1885. Sometimes these underlying general patterns are discussed explicitly, more commonly they are used implicitly in arguments based on precedent and analogy.

Substitution principles often derive simply from the natural language content of a statement itself—for example in the case given above, X and Y would be any pair of mutually antagonistic states. However, the allowed substitutions might be specific to an individual or organization. For example, when United States decision-makers accepted the Munich analogy as a guide to dealing with Vietnam, the substitutions

Southeast Asia 1965 = Europe 1938
North Vietnam = Germany
Ho Chi Mihn = Hitler
Ngo Dinh Diem = Churchill

was at least implicit in the argument and occasionally it was explicit. North Vietnam's preferred analogy:

Southeast Asia 1945 = North America 1775
French Indochina = British colonies
Ho Chi Mihn = George Washington
Ngo Dinh Diem = Benedict Arnold

was not accepted. The U.S. also invoked the Munich analogy (endlessly…) when dealing with Iraq's invasion of Kuwait in 1990-91; but this analogy was not invoked when dealing with Israel's occupation of Gaza, the West Bank and the Golan Heights in 1967 or with Turkey after its invasion of Cyprus in 1974.

The problem of generalizing sequences is particularly salient to the analysis of international political behavior in the late 20th century because, due to current changes in the international system, many contemporary situations do not have exact historical analogs. Yet human analysts are clearly capable of making analogies based on some characteristics of those behaviors. For example, because of its unusual historical circumstances, the situation in Zaire in 1997 had a number of unique characteristics, but during the crisis analysts pieced together sufficient similarities to a variety of earlier crises in Africa and elsewhere to come to the correct conclusion that Zaire had entered a period of rapid political change. The key to this analysis, however, was the ability to use general analogies: if one insisted on an analogy to a single case—which a human analyst would almost never do, but a computer might—then the Zairian case would be nearly impossible to analyze using analogies.

Analogies is not, of course, necessarily good policy, and there are times when a deductive argument might be superior. Weigley provides an interesting case of precedent over-ruling deductive argument:

> In 1966 Walt Rostow called President Johnson's attention to the effects of sustained aerial attack on Germany's petroleum facilities late in World War II and argued "With an understanding that simple analogies are dangerous, I nevertheless feel it is quite possible the military effects of systematic and sustained bombing of [petroleum supplies] in North Vietnam may be more prompt and direct than conventional intelligence analysis would suggest." The intelligence analysis in question indicated that North Vietnam depended so little on petroleum … that bombing … would not much affect the war in the South or compel North Vietnam to make peace. But the Joint Chiefs agreed with Rostow's analogy, and so the aerial campaign against North Vietnam's petroleum was attempted. (Weigley 1973,387)

The bombing campaign eventually failed largely for the reasons suggested in the deductive intelligence analysis, but the analogical argument prevailed[2].

Stories are also a means of inferring “motive”: A motive is the end point of a sequence. Associated with the motive is a series of sequences that terminating in a specific end-point. Motive sequences can be used either inductively or deductively. Inductively, one has a set of facts that could match any of several different end points (e.g. is an arms control proposal intended to reduce arms or to weaken oneself prior to an opponent initiating hostilities?); the decision-maker then searches for information to differentiate between those sequences. Deductively, an end point can be assumed and one can seek to differentiate between the various paths that might be used to reach the end point and thwart them (e.g. you know an opponent is trying to weaken your alliances, but how?). In both instances the use of stories dramatically reduces the information processing task by identifying only those items of information necessary for prediction.

Finally, stories provide means of correcting for noise and missing information. This is particularly important in the political environment that is subject to low information and deliberate deception. Pennington and Hastie observed in an experimental study of individuals summarizing trial evidence:

> …spontaneous interview protocols do exhibit story structures. … Juror’s stories were not simple lists of evidence. They always contained most components of the episode schema (initiating events, psychological and physical states, goals, actions and consequences) in appropriate causal relations. Jurors typically inferred missing components for these structures when they were not contained in direct testimony at trial. Evidence not related to a story of what happened was systematically deleted from discussion. (Pennington and Hastie 1986,252; quoted in Boynton 1991).

2 Rostow seems particularly fond of analogical argument: Wirtz's (1989) discussion of analogies in the Vietnam War opens with a discussion of Rostow's approval of a memo comparing North Vietnam in 1967 with the American Confederacy in 1863-64.

In summary, sequences are one of the primary means by which analysts solve the fundamental problem of short-term prediction in determining the likely consequences of their own policies and the intentions of their opponents. Political analysts have, in associative memory, a large number of sequences acquired through experience and the study of history, and "understand" observed political events when they can match those events to a sequence stored in memory.

5.2. Computational Sequence Recognition

Because analogies are so prevalent in human political reasoning, it would be helpful to have some computational method of determining them. That, in turn, requires determining some means of ascertaining the general characteristics of a set of sequences. In human pattern recognition, we have a general idea of what a category of event sequences look like—the archetypal war, the archetypal coup, and so forth—and probably match to these generalized ideal sequences rather than to clusters of sequences. In a sense, ideal sequences are the centroid of a cluster of sequences, but that centroid is a sequence rather than a point. If a method could be found for constructing such a sequence, the cluster could be represented by the single ideal sequence, which would substantially reduce computing time and provide some theoretical insights as to the distinguishing characteristics of a cluster.

Creating such general sequences, however, is a difficult problem because of the differences between human and digital processing. Creating a system of generalized sequence recognition for political analysis thus presents a substantial challenge. First, the required amount of information is very large. Foreign policy analysts have a tremendous store of sequence-based information, including formal political knowledge such as the history of the Cold War and the evolution of the Westphalian nation-state system, current information on past activities of individual actors—Saddam Hussein, Boris Yelstin; the differences in Japanese and British foreign policy bureaucracies—and "common sense" knowledge about human behavior, usually learned informally—"hit someone and they won't be happy, and they may hit you back, or if they are smaller than you they may find someone else to hit you back.". The quantity of such information is unclear, but it probably runs to tens of thousands of sequences of varying lengths.

The techniques that we illustrate in this chapter begin to develop some machine learning methods that will allow a program to do three things:

- Recognize that two sequences are similar; in other words, simulate the basic sequence recognition function;
- Break a sequence down into its component parts;
- Use sequence similarity and parts to classify sequences into general categories such as war/nonwar.

Because of the complexity of international behavior, a robust system is ultimately going to need to have the ability to learn from example, though probably at the expense of detail.

The ideal sequence recognition system would require three components. First, one needs a knowledge representation structure for the sequences themselves: for this we will use event data. Human sequence recognition in all likelihood tags event sequences with some additional contextual information concerning the national and international environment—the outbreak of the "Soccer War" between El Salvador and Honduras in 1969 is classified differently than the outbreak of the Russo-Afghan war in 1979—but the isolated sequence provides a starting point and the problem of matching contextual information is not likely to be more complex than matching the sequences. Second, one needs a metric that will indicate the degree of similarity between two sequences. Finally, one needs to have a very large number of historical sequences in memory.

Given these three components, the sequence recognition problem can be reduced to a nearest-neighbor problem: Take an observed set of events, compute the distance between that sequence and all of the sequences in memory (or a set of archetypal centroids representing general categories of behavior), and classify the sequence using its nearest neighbor.

5.3. The Levenshtein Metric

The Levenshtein metric (Sankoff and Kruskal 1983) is a sequence comparison technique that originated in information theory and is now commonly used to analyze sequences of sound or DNA; Mefford (1984, 1985) proposed using it as a means of sequence comparison in international relations. The Levenshtein metric uses a large matrix of numerical weights to determine the distance between two sequences; these weights can be set, for example, to produce small distances between sequences of the similar type and long distances between sequences of dissimilar type.

We will demonstrate this method using the crises in the Behavioral Correlates of War (BCOW: Leng 1987) event data set; discriminating crises that don't involve war from those that do. The weight will be determined using an example-counterexample machine learning protocol . To learn to discriminate between two classes of objects, the machine is presented with examples from each class and adjusts its knowledge structure—the matrix of Levenshtein weights—on the basis of those examples. The knowledge structure of the Levenshtein metric is sufficiently complex that it can achieve 100% discrimination among the training cases, so it is validated with split-sample testing.

The Levenshtein distance between two sequences is the sum of the weights of the operations needed to convert one sequence into another. If a and b are two sequences [a_1 a_2 a_3 ... a_m] and [b_1 b_2 b_3 ... b_n], the Levenshtein approach converts one sequence to the other using the operations

Delete an element from a
Insert an element into b
Substitute b_j for a_i

Using the example in Sankoff and Kruskal (1983,11), one could convert the sequence "W A T E R" to "W I N E" by the operations

W A T E R

Substitute I for A

W I T E R

Substitute N for T

W I N E R

Delete R

W I N E

The operations used in computing the Levenshtein distance are those that minimize the sum of the weights. A dynamic programming algorithm for determining this minimum is presented below.

Algorithm for Computing Levenshtein Distances

```
Function Leven_dist(a,b:seq):real;
{ This code follows the Levenshtein distance algorithm described in Kruskal (1983) }
    {a,b are arrays containing the sequences; the 0th element holds the length of the }
{sequence;}
{weight[i,j] gives the insertion, deletion and substitutions weights;}
{dist[i,j] is the matrix used to compute the distance }

var ka,t,r,c        : integer;
    min             : real;
    max_r,max_c     :integer;

begin
  dist[0,0]:=0.0;
  for ka:=1 to a[0] do dist[ka,0]:=dist[ka-1,0] + weight[a[ka],0];
  for ka:=1 to b[0] do dist[0,ka]:=dist[0,ka-1] + weight[0,b[ka]];

{ The code in the “t” loop goes through the matrix starting in the upper left corner } {then
filling by moving down and to the left,ending at the lower right corner.  r is the } {row, c
the column.}

  max_r:=a[0];
  max_c:=b[0];
  ka:=max_r + max_c;
  for t:=2 to ka do begin
    r:=1;
    if t-r<max_c then c:=t-r
                 else begin
                        c:=max_c;
                        r:=t-c;
                      end;
    repeat
              { Determine the operation which adds the minimum to the weight at each point }
      if dist[r-1,c]<dist[r,c-1] then min:=dist[r-1,c]
                                 else min:=dist[r,c-1];
      if dist[r-1,c-1]<=min then min:=dist[r-1,c-1];
      dist[r,c] := min + weight[a[r],b[c]];
      r:=r+1;
      c:=c-1;
    until (c<1) or (r>max_r);
  end;
  Leven_dist := dist[a[0],b[0]];
end; { Leven_dist }
```

The knowledge structure of a Levenshtein metric lies in the insertion, deletion and substitution weights. Changes in a sequence that reflect important differences should have high weights; those reflecting trivial differences should have low weights. For example, in linguistics, it is clear that as words migrate from language to language, vowels are more likely to change than consonants, and if consonants change, they are likely to change only slightly (an “s” might change to “c” or “z” but probably not “b” or “t”). Thus we see similarities between the English "peace", French "paix" and Latin "pax", and similarities between the Hebrew "shalom" and Arabic "salaam", but see considerable differences between the two groups of words.

The extension of this principle to international event sequences is straightforward (Mefford 1984). Certain international events are quite comparable—for example "mediate" (BCOW code 12142) versus "negotiate" (BCOW 12121)—whereas others are very different—for example "peace settlement" (BCOW 12361) and "continuous military conflict" (BCOW 11533). One would also expect that common events (e.g. the ubiquitous "consult" and "comment" codes in events data) should be inserted and deleted with little cost, whereas rare events such as agreements or the beginning of military conflict would be costly to insert. Two international event sequences would be considered similar if one sequence could be converted to the other using operations that substitute like event for like event; the two sequences would be quite different if they could only be converted by substituting unlike events,.

Schrodt (1984, 1985a) reports a feasibility test for using Levenshtein distances to discriminate between general types of dyadic behavior in 1982 using WEIS-coded events (McClelland 1976). Dyads were compared using the distribution of distances from a sample of randomly chosen sequences each containing ten events. The weighting scheme used the fact that two-digit WEIS codes, while technically nominal, are virtually ordinal, so substitution weights were set to the difference between the WEIS codes. Thus the substitution weight of "Force" (WEIS code 22) and "Yield" (WEIS 1) is 21, whereas the substitution of "Force" and "Expel" (WEIS 20) is 3. Insertion and deletion weights were based on the rank order of the frequency of a code: frequent events had low insertion and deletion weights; infrequent events had high weights. While arbitrary and *ad hoc*, this scheme produced plausible differentiation between dyads. For example, the USA/UK and USA/PRC dyads were measured as showing similar behavior; the USA/UK and Iran/Iraq dyads as showing very different behavior.

The clear disadvantage of this approach was the arbitrariness of the weights. Nonetheless, deriving weights for a complex coding scheme on *a priori* theoretical grounds would be difficult: for example, what should be the relationship between BCOW 13551 (Reach Economic

Agreement) and BCOW 12641 (Assume Foreign Kingship)? The alternative is induction: weights determined by what one wants to do with the Levenshtein measure itself.

The algorithm demonstrated here is based on the Widrow-Hoff or "delta rule" training method used in training neural networks[3]. The machine is given BCOW cases in two categories: war crises and nonwar crises. The training objective is finding weights that produce small distances between the sequences within each set, and larger distances between sequences in different sets.

To create the weights, the distances between each pair of sequences is computed using the Levenshtein algorithm. Any weights used in computing the distance between a pair of *like* sequences are *decreased* by a small amount. Weights used in computing the distance between *unlike* sequences are *increased* by the same amount. This process is iterated a number of times.

Using this approach, the weights of operations invoked only in the comparison of like sequences are reduced; the weights of operations invoked only in the comparison of unlike sequences are increased; and the weights of operations invoked in comparing both like and unlike sequences remain about the same, since the increase and decrease cancel out. As a consequence, the distances within the groups should decrease, while the distances between the groups should increase. The learning has to be done iteratively since the choice of operations used in computing the Levenshtein distance may change as the weights change, because the Levenshtein algorithm chooses the operations that have the smallest weights.

In the experiments described below, the weights were initialized in two different ways. In frequency-based initialization, the insertion and deletion weights are set to the rank-order of the event frequency in the set of all sequences used to train the system. The most frequent event had a weight of 1, the second most frequent a weight of 2 and so forth. This is consistent with the coding used in Schrodt (1984) and is based on the information theory argument (Pierce 1980) that frequent events have little discriminating value and can be replaced with little cost, whereas rare

3 The original method was discussed in Widrow and Hoff (1960); Rumelhart *et al* (1986) provide an extensive discussion of variations in the context of neural networks.

events should have a higher cost[4]. The substitution cost was initialized as $|r_a - r_b|$, the absolute difference of the ranks of codes a and b. Thus it is less costly to replace a frequent event with another frequent event than it is to replace a frequent event with a less frequent event.

Alternatively, weights were initialized to a constant. These different initializations had no effect on the learning algorithm, although the constant weights proved somewhat less useful for doing discrimination.

The learning scheme produces several peculiarities in terms of the regularities expected of a "distance" (Sankoff and Kruskal 1983:22)—in fact technically speaking, it is not a distance in the mathematical sense—but these cause no interpretive problems in the discrimination test. First, the metric is completely arbitrary, without a zero point, and the "distance" may be negative since many weights become less than zero by progressive subtraction as their operations are repeatedly used in comparing like sequences. This is completely consistent with the substantive interpretation of the weights, since it allows the matching of elements that are important in determining similarity to cancel out mismatches of less important elements. Second, the distance between two identical sequences is not necessarily zero: in fact this tends to be negative because the training algorithm sets the weights for exact matches to negative values.

Finally, unless the weight matrix is symmetric, the distance between A and B is not necessarily the same as the distance between B and A. Because the changes in weights are done while the sequences are compared, a different weight matrix is used to compare A to B than comparing B to A during the training. As a consequence, the weight matrix is not symmetric. There are no substantive excuses for this: it is simply a quirk in the algorithm. These differences seem to get reinforced in the training—note the asymmetries in Table 5.2—although not so badly as to keep the technique from functioning as intended.

4 No adjustment was made for ties: events tied in frequency were randomly ordered within that tie. The frequency of BCOW events generally follows a rank-size law.

5.3.1. Discriminating War and Nonwar Crises

This system was tested using the BCOW sequences described in the chapter appendix; the short names of the crises (e.g. *pastry*) correspond to the BCOW file identifiers. The training sequences were used to determine the weights; the system was tested with the remaining sequences.[5] Events within a single day are sorted by code so that if identical events occur within a single day in two sequences they will be in the same order in both sequences.

The basic protocol was to run the algorithm on the ten training cases, iterating until the two groups were separated. The resulting weight matrix was then applied to the ten test cases by computing the distance between each test case and the ten training cases. The expectation was that the nonwar cases would, on average, be closer to the nonwar cases in the training set, and similarly for the war cases. The training algorithm showed a monotonic increase in the separation of two groups. This increase in separation was mostly linear with a slight leveling-off that would be expected in a classical learning curve.

Table 5.2 gives the results of testing the comparisons with the training cases[6]. The expectation that the Levenshtein distance would discriminate between the war and nonwar crises is fully met, and the discrimination is almost perfect. Only one crisis—Schleswig-Holstein—is not strongly classified into the correct group, although even this case errs only in being almost as close to the nonwar crises as the nonwar crises are from each other; it has the expected negative distances to the other war crises. The war crises cluster strongly, with large negative distances within the group and large positive distances between the groups.

Table 5.2. Distances in Training Set

5 As noted in the chapter appendix, only the physical events reported in BCOW were used; the sequences were also filtered to eliminate common events and a prefix was added to the BCOW event code to indicate which of five types of dyadic relations were involved in the event.

6 Table 5.2 was generated with 20 iterations of the training set, constant initial weights set at 10.0, and the weight increment of 1.1.

	rhin	1mor	fash	2mor	bosn	schl	spam	cent	ital	chac
rhine:		330	144	139	311	603	1771	1081	1946	1446
1stmor	190.		-148	189	-275	532	1540	1162	1577	1393
fashod	-288	-77		-177	308	437	1496	754	690	803
2ndm	203	142	-168		351	479	1154	628	1298	972
bosnia	258	-155	307	337		773	-467	662	1493	521
schles	358	432	561	389	540		-2874	-2286	-1301	-1080
spam	1413	1432	1521	1407	-533	-3277		-6399	-5747	-6724
centam	1064	1158	773	613	421	-2371	-6277		-3363	-4641
italet	1903	1781	1586	1648	1594	-1357	-4941	-3145		-3216
chaco:	1184	1154	1028	715	474	-1346	-5076	-3161	-2510	

```
Average within-group distance    =  -1729.4
Average between-group distance   =    987.6
Separation                       =   2717.0
```

Table 5.3 reports the split-sample test using the difference in distances:

Difference = (Average distance to war crises) — (Average distance to nonwar crises)

Results for both frequency-based and constant initializations are reported. As Table 5.3 indicates, the discrimination using the frequency-based weights is perfect: all of the war crises in the target set are closer to the war crises in the training set than to the nonwar crises in the training set; the reverse is true for the nonwar crises. In terms of the rank-order of the distances, the same is true when constant initial weights are used but two of the nonwar crises are actually closer to the war group than to the nonwar group. Despite this, there is still a good differentiation using constant weights: the closest nonwar crisis has a distance difference of -821 whereas the furthest war crisis has a difference of -2932. In this test, frequency based initialization seems to produce a better discrimination matrix than does constant initialization.

Table 5.3. Distances in Test Sets

War distance minus nonwar distance

Crisis	Type	frequency	constant
palest	War	-2253	-4494
balkan	War	-1264	-4189
bangla	War	-994	-2932
kash1	War	-624	-3685
kash2	War	-557	-3226
munich	Nonwar	21	-708
berair	Nonwar	210	291
pastry	Nonwar	436	-821
anschl	Nonwar	713	316
brtprt	Nonwar	1645	552

Table 5.4 shows the event pairs that had the maximum and minimum changes from the initial values of their substitution weights.[7] Neither group is particularly surprising. The minimum weights, which indicate strong similarity, are usually similar or identical events, particularly those involving military conflict. The maximum weights, which indicate strong dissimilarity, primarily involve substitution of a cooperative action such as consultation or negotiation for a military action such as “Show of Strength” or “Take POWs”. The magnitude of the largest minimum weights is substantially greater—almost by a factor of ten—than the magnitude of the maximum weights. The first digit is a dyad identification code so, for example, the “Mobilization/Mobilization” pair in the maximum weights is the substitution of “mobilization by an ‘other’ actor against one of the principals” for “mobilization by one of the principals against the other principal.” Unsurprisingly, most of the large weights occur in the frequent events, and thousands of substitution weight were never changed from their initial value.[8]

7 The weights in Tables 5.4 and 5.5 were produced with 51 iterations under the same conditions as Table 5.2. The reported weights were selected from the set of maximum and minimum substitution weights for each event rather than from the maximum and minimum events for the entire table.

8 There were 171 distinct dyad-prefixed event codes in the sequences, so the total size of the matrix, including the insertion and deletion weights, was 172^2, or 29,584.

Table 5.4. Minimum and Maximum Weights

Minimum Weights

Code	Meaning	Code	Meaning	Weight
111633	Military victory	111633	Military victory	-687.5
111513	Clash	111633	Military victory	-579.6
121143	Change in force	121143	Change in force	-446.6
111663	Take POWs	111523	Attack	-299.1
111533	Continuous conflict	111533	Continuous conflict	-278.3
111523	Attack	111523	Attack	-237.6
211131	Military coordin.	112111	Consult	-162.7
312521	Reach Agreement	411313	Show of Strength	-128.6
111553	No code	512111	Consult	-119.8
112111	Consult	112111	Consult	-111.1
312213	Violate Territory	111313	Show of Strength	-109.9
212521	Reach Agreement	412111	Consult	-103.3
211553	No code	111353	Mobilization	-103.3
312142	Mediate	111533	Continuous conflict	-94.5
112631	Attend Internatnl Event	123151	Change Trade	-90.1

Maximum Weights

Code	Meaning	Code	Meaning	Weight
512111	Consult	111313	Show of Strength	74.9
212521	Reach Agreement	111663	Take POWs	70.5
112121	Negotiate	512111	Consult	69.4
111663	Take POWs	112121	Negotiate	69.4

114113	Subversion	111313	Show of Strength	69.4
112213	Violate Internatnl Law	111313	Show of Strength	68.3
212111	Consult	111633	Military victory	67.2
111353	Mobilization	311353	Mobilization	67.2
312173	Expel Foreign Rep	112213	Violate Internatnl Law	66.1
321133	Change Force Level	512111	Consult	63.9
311313	Show of Strength	512111	Consult	60.6
112521	Reach Agreement	514143	Assassinate	60.6
111443	Military Intrusion	111663	Take POWs	59.5
111523	Attack	112111	Consult	58.4
111513	Clash	112111	Consult	57.3

Table 5.5 reports the insertion and deletion weights for the most frequent events. These tend to be symmetric for insertion and deletion, inversely proportional to their rank orderings and substantially higher than the substitution weights. This pattern holds whether frequency-based or constant initial weights are used, and is opposite from the expectation in information theory that frequent events would have the smallest insertion and deletion weights. These high values imply that at some point in the training process, insertions and deletions were being used frequently—otherwise they would not have the high values—but their high values relative to the substitution weights would lead one to expect that eventually the sequence comparisons would be dominated by substitutions. A few codes showed negative weights, usually on the order of 10 to 100, but almost all of the insertion and deletion weights were positive.

Table 5.5. Insertion and Deletion Weights

Code	Meaning	Weights	
		Delete	Insert
212111	Consult	2608.1	2638.9
312111	Consult	2393.6	2545.4
512111	Consult	1548.8	1951.4
111313	Show of Strength	2208.8	2077.9
111353	Mobilization	1763.3	2124.1
111633	Military victory	2083.4	1675.3
112111	Consult	862.4	652.3
111523	Attack	1180.3	1020.8
311313	Show of Strength	597.3	108.8
112121	Negotiate	731.5	735.9
112521	Reach Agreement	994.4	906.4
111653	Occupation	895.4	914.1

111513	Clash	1298.0	1285.9
212121	Negotiate	789.8	699.6
111663	Take POWs	902.0	412.5

5.3.2. Discrimination between multiple cases

The experiment above discriminated between only two categories, war and nonwar. Human decision-makers discriminate between a greater number of categories, so the obvious question is whether a single Levenshtein matrix can be used to handle multiple discrimination.

In an early article using the BCOW crisis set, Gochman and Leng (1983) classify crises into four different categories of bargaining behavior: “fight”, “resistance”, “standoff” and “prudence”. There is sufficient overlap between the crises analyzed here and the Gochman and Leng set that a simple test can be done of that discrimination. The following crises were used:

Fight:	*balkan* (twice), *chaco, kash2*
Resistance:	*brtprt, spam, bosnia, italet*
Standoff:	*bangla, 1stmor, fashoda, centam*
Prudence:	*rhine, 2ndmor, schles, kash1*

This is an imperfect test of the Gochman-Leng categories in at least two respects. First, Gochman and Leng base their characterization on all of the behavior in the crisis, whereas this test looks only at the physical behavior. Second, the BCOW files do not correspond exactly to the crises discussed by Gochman and Leng: *chaco* contains both the 1928-29 dispute, which is classified as “fight” and the 1932 dispute, which is classified as “resistance”; *kash2* is a superset of the 1965 Rann of Kutch dispute which is classified as “fight”.

The algorithm used to handle multiple classifications is identical to that used for the binary classification: weights are reduced in comparisons of similar cases and increased for dissimilar cases. The number of cases in each set have to be identical since otherwise weights that are used to match sequences within a large set will be reduced simply by virtue of their being subject to a greater number of opportunities for reduction. To equalize the cases in each set, the *balkan* set was duplicated in the “fight” category; the unusually short *pastry* case was eliminated from “standoff”, and the *ansch* and *munich* cases were eliminated from the “prudence” set.

Table 5.6 shows the results of this after 31 iterations. The Gochman-Leng categories are generally differentiated, although the results are less than spectacular. The “resistance” category appears to be the unusual case: it is the category most distant from all of the other groups, and is unique among the groups in not having its within-group distance being less than the between-group distances. The “standoff” and “prudence” categories are clearly discriminated from the “fight” and “resistance” categories, which may reflect the fact that these involve acts of violence that appear as physical events. Letting the algorithm run for 81 iterations produced virtually no additional changes[9]: the distances expanded by an average of 25% but the relative distances between the pairs of groups remained the same, and the average distance from “resistance” to “standoff” and “prudence” was still less than the average distance within “resistance”. These failures are, perversely, reassuring since they indicate some falsifiability to the method: it is not capable of differentiating any grouping through pure brute force, even with a very large number of iterations.

This is merely a feasibility test but shows a potential for doing multiple discrimination of categories of sequences using a single Levenshtein matrix, much as neural networks are able to do in a single matrix of weights. The training in a multiple discrimination case takes considerably longer than that required for a single discrimination case: 27 iterations are required before the final discrimination pattern stabilizes; the binary discrimination problem took only about 15 iterations to achieve a comparable level of separation.

9 The only exception was that the rank order of the distance of "fight" and "resistance" from "standoff" reversed.

Table 5.6. Discriminating Multiple Groupings—Average Distance between Crises by Group

	Fight	Resistance	Standoff	Prudence
Fight	1967.952			
Resistance	2708.714	2562.451		
Standoff	2158.901	2206.600	1461.201	
Prudence	2018.164	2067.513	1490.401	1266.701

5.4. Parallel Event Sequences

One of the problems involved in interpreting a stream of events such as those found in an event data set or newswire feed is the fact that these events are the result of multiple, parallel political initiatives. This section extends earlier work by Bennett and Schrodt (1987) that used a subset of 7000 WEIS events involving Middle East states and the North Atlantic major powers to construct common subsequences on the basis of nondirected dyad pairs (e.g. USA ↔ USSR) using two-digit WEIS codes. These subsequences were constructed by first scanning the event sequences for the most common 2-event subsequences, then using a fraction of those to construct 3-event subsequences, then using a fraction of those to construct 4-event subsequences and so forth.

The subsequences found by this technique were very successful at covering the WEIS sequences: as a general benchmark, a set of 10 4-event subsequences could account for about 35% of the data. However, these common subsequences were very repetitive and concentrated heavily on the most common events found in this subset of WEIS: uses of force, accusations and agreements. The system discussed here modifies that earlier work by looking explicitly for event subsequences found in multiple crises coded in the BCOW data set. The BCOW data are denser and more varied than the WEIS data, and filtering is used to eliminate the common events.

The four subsets of the BCOW crises were analyzed; these are listed in Table 5.A.2 in the appendix. The "Threats" set contains crises that did not result in war because one side backed down; the two "War" sets contain crises that involved wars; and the "Mixed" set contains five nonwar crises and five wars. Common subsequences within each of these sets will be determined first, then those subsequences will be used to differentiate the different categories of crises.

5.4.1. Algorithm

The algorithm used to construct subsequences is a fairly simple search technique that focuses first on finding event codes common to as many of the target sequences as possible, then on minimizing the distance between the consecutive events in a subsequence. When a subsequence has been determined, it is eliminated from all of the target sequences where it occurs, then the remaining events in the target sequences are searched for additional subsequences. The algorithm is given below in pseudocode.

To allow for the possibility of an unreported (or nonexistant) event in an otherwise complete sequence, the algorithm does not insist on the perfect matching of a subsequence. The multiple elimination of subsequences allows subsequence to be repeated, for example, when a negotiation is broken off and then reinitiated, or two short periods of hostilities occur. The core of the algorithm is the coverage-maximizing/distance-minimizing search; the remaining idiosyncratic features such as multiple subsequence elimination provide some additional coverage and change slightly the resulting subsequences but are not of critical importance.

The algorithm runs quite quickly because it is deterministically constructing subsequences rather than using a nested (i.e. exponentially expanding) search or using random experimentation. It is also quite short, about 500 lines in Pascal. The number of event codes common across the target sequences tends to be around 100 in each of the sets, so the time required to find the subsequences is generally a linear function of the total length of the target sequences.

```
1. Filter and recode the BCOW sequences (see Appendix)
REPEAT
```

```
1. Set the current point in each sequence to the beginning; set the
   subsequence to the null string
      REPEAT
      1.Evaluate each possible event and each sequence and select the
         event E' which:
         a. maximizes the number of occurrences in the target sequences
         b. minimizes the average distance between the current point and
            the next occurrence of the event subject to (a).
            Events which have already been eliminated by previous
            subsequences are not counted in the distance.
      2.Add E'  to the subsequence being constructed
      3.If E' is in a sequence, reset the current point  of that
      sequence to the location of E'.
      UNTIL   there is no event which occurs beyond the current point in
              at least a fixed number (3) of the sequences

2. Record the subsequence;

3. With each sequence, eliminate all of the events which have been
   matched by the subsequence.  A subsequence can be applied multiple
   times until less than half of its events occur in the sequence

UNTIL size of subsequence is less than or equal to a fixed number (4)
```

5.4.2. Results

The subsequences found in each of the four data sets are listed in Table 5.7; Table 5.8 shows an example of how the subsequences nest within the original sequences. All subsequences that contained four or more events and were found in at least three of the target sequences are listed in Table 5.7.[10] The table presents both the 6-digit code (dyad type + 5-digit BCOW event code) and the BCOW description of the event. Note that in many cases events with the same BCOW code refer to different dyad types: for example in subsequence E in the "War2" set there are three "Reach Agreement" events prior to the "Clash" but these agreements are with "Other" parties, not between the two sides, and quite likely involve consultation with supporters prior to initiating conflict. Similarly many of the frequent "Consult" codes are not consultations between

10The three-event subsequences War1-D and War2-G were found because the algorithm terminated when it could only find a subsequence less than or equal to 4 events in length, and the two War sets have no 4-event subsequences. These are listed in Table 5.7 because they were used when computing the coverage statistics.

the sides of the dispute but with others or between others. A 000000 code in Table 5.8 indicates an event which occurred only once in the set and had been recoded to zero to save storage.

Table 5.7. Parallel Subsequences

Threat Data Set

A. 111313 111313 111313 111313 312111 112111 212111
Show of Strength :: Show of Strength :: Show of Strength :: Show of Strength :: Consult ::Consult :: Consult

B. 212111 312111 111353 121133 311313 212111 312111 212111 112521 312111
Consult :: Consult :: Mobilization :: Change Force Level :: Show of Strength :: Consult :: Consult :: Consult :: Reach Agreement:: Consult

C. 112121 111333 212121 112111 212521
Negotiate :: Alert :: Negotiate :: Consult :: Reach Agreement

D. 512111 112213 114213 123151 512111
Consult :: Unknown* :: Antiforeign demonstration :: Change in trade relations :: Consult

E. 212521 312521 111653 112111
Reach Agreement :: Reach Agreement :: Occupation :: Consult

War1 Data Set

A. 112521 212111 212521 112521 112121 112121 212111
Reach Agreement :: Consult :: Reach Agreement :: Reach Agreement :: Negotiate :: Negotiate:: Consult

B. 111353 312111 111523 311313 512111 512111
Mobilization :: Consult :: Attack :: Show of Strength :: Consult :: Consult

C. 111523 111523 111533 111533 311353
Attack :: Attack :: Continuous Military Conflict :: Continuous Military Conflict :: Mobilization

D. 212111 511313 512521
Consult :: Show of Strength :: Reach Agreement

War2 Data Set

A. 512111 212111 312111 212111 312111 212111 312111 212111 312111 212521 212111 312111 212111 111523 312111
Consult :: Consult :: Consult :: Consult :: Consult :: Consult :: Consult :: Consult :: Consult :: Reach Agreement :: Consult :: Consult :: Consult :: Attack :: Consult

B. 111333 111313 111533 121133 111633 111513 112213 121143
Alert :: Show of Strength :: Continuous Military Conflict :: Change Force Level :: Military Victory (partial) :: Clash :: Unknown* :: Change in Combat Force Level

C. 114123 111523 212111 111513 512111 512111
Discrete Attack :: Attack :: Consult :: Clash :: Consult :: Consult

D. 112121 112121 112521 114213 311313 311313
Negotiate :: Negotiate :: Reach Agreement :: Antiforeign demonstration :: Show of Strength:: Show of Strength

E. 212521 312521 312521 111513 111513 111313 111523

Reach Agreement :: Reach Agreement :: Reach Agreement :: Clash :: Clash :: Show of Strength :: Attack

F. 312111 321111 112111 111633 111533
Consult :: Military Grant :: Consult :: Military Victory (partial) :: Continuous Military Conflict

G. 512521 111523 111313
Reach Agreement :: Attack :: Show of Strength

Mixed Data Set

A. 112521 311313 111523 212111 212111 312111
Reach Agreement :: Show of Strength :: Attack :: Consult :: Consult :: Consult

B. 212111 312111 111313 112121 112111 111313 512111 112121 111353
Consult :: Consult :: Show of Strength :: Negotiate :: Consult :: Show of Strength :: Consult :: Negotiate :: Mobilization

C. 111353 311313 111663 212111 111533 112521
Mobilization :: Show of Strength :: Take POWs :: Consult :: Continuous Military Conflict :: Reach Agreement

D. 512111 114213 212521 112213 111653
Consult :: Antiforeign demonstration :: Reach Agreement :: Unknown* :: Occupation

E. 112121 212121 111513 212121 212521 312521 312111
Negotiate :: Negotiate :: Clash :: Negotiate :: Reach Agreement :: Reach Agreement :: Consult

F. 111523 412111 312121 111523
Attack :: Consult :: Negotiate :: Attack

*”Unknown” corresponds to code 12213, which is in the data but not the codebook; it may be “Violate territory”, which the codebook states is 12223

Table 5.8. Subsequence Positions within Sequences

	pastry	1stmor	fashod	2ndmor	bosnia	brtprt	anschl	rhine	munich
1:	AAAAAA	EEEEEE	312173	AAAAAA	000000	BBBBBB	000000	321133	412111
2:	111523	EEEEEE	312173	111443	CCCCCC	AAAAAA	BBBBBB	BBBBBB	BBBBBB
3:	BBBBBB	000000	112521	000000	112521	114223	BBBBBB	AAAAAA	AAAAAA
4:	BBBBBB	BBBBBB	000000	CCCCCC	123151	DDDDDD	DDDDDD	412111	412111
5:	BBBBBB	BBBBBB	132143	CCCCCC	311353	DDDDDD	114213	BBBBBB	121133
6:	DDDDDD	AAAAAA	111521	EEEEEE	111353	114223	DDDDDD	BBBBBB	AAAAAA
7:	DDDDDD	212161	332143	BBBBBB	BBBBBB	CCCCCC	114251	AAAAAA	121133
8:	000000	112631	132143	AAAAAA	AAAAAA	AAAAAA	214251	412521	112111
9:	111433	AAAAAA	111553	411313	000000	BBBBBB	DDDDDD	000000	111333
10:	414113	321121	EEEEEE	BBBBBB	123151	114213	212213	AAAAAA	112183
11:	112161	BBBBBB	EEEEEE	412111	DDDDDD	000000	112161	DDDDDD	AAAAAA
12:	111433	511313	132143	112121	111353	314251	112213	412111	412111
13:	DDDDDD	212121	000000	312213	321121	212213	114251	CCCCCC	121133
14:	BBBBBB	512121	000000	411313	412521	112631	CCCCCC	BBBBBB	221133
15:	BBBBBB	BBBBBB	AAAAAA	000000	111353	000000	000000	111443	121133
16:	000000	AAAAAA	112521	AAAAAA	CCCCCC	314123	112521	112183	112111
17:	CCCCCC	AAAAAA	112183	CCCCCC	CCCCCC	000000	212111	AAAAAA	114151
18:	BBBBBB	DDDDDD	332143	312121	312121		312111	BBBBBB	412111
19:	000000	212161	EEEEEE	AAAAAA	000000		AAAAAA	000000	121133
20:	111993	512521	AAAAAA	AAAAAA	311353		000000	CCCCCC	114251
21:	112152	CCCCCC	132143	BBBBBB	EEEEEE		DDDDDD	312521	114223
22:	112363	AAAAAA	AAAAAA	CCCCCC	EEEEEE		114251	212111	321121
23:	DDDDDD	000000	BBBBBB	CCCCCC	000000		AAAAAA	312111	412111
24:	BBBBBB	312631	CCCCCC	312121	111353		221133	AAAAAA	114251
25:	111521	AAAAAA	AAAAAA	DDDDDD	311313		112121	DDDDDD	412521
26:	112152	BBBBBB	000000	112121	EEEEEE		DDDDDD	000000	414151
27:	111521	EEEEEE	BBBBBB	BBBBBB	CCCCCC		DDDDDD	000000	114251
28:	000000	AAAAAA	DDDDDD	AAAAAA	EEEEEE		DDDDDD	212111	DDDDDD
29:	AAAAAA	BBBBBB	111553	BBBBBB	312121		212111	312111	AAAAAA
30:	BBBBBB	CCCCCC	AAAAAA	311333	211313		312111	000000	BBBBBB
31:	112173	000000	112173	000000	AAAAAA		AAAAAA	512111	412111
32:	414113	000000	111353	AAAAAA	114151		BBBBBB	212111	121133
33:	000000	112631		112121	AAAAAA		112183	312111	DDDDDD
34:	112173	BBBBBB		BBBBBB	BBBBBB		000000	212111	112111
35:	AAAAAA	BBBBBB		111993	412111		000000	312111	414151
36:	AAAAAA	512213		000000	511313		CCCCCC	212111	114251
37:	111443	BBBBBB		112213	211313		112161	312111	AAAAAA
38:	AAAAAA	BBBBBB		BBBBBB	BBBBBB		000000	212111	BBBBBB
39:	AAAAAA	321133			BBBBBB		114251	312111	BBBBBB
40:	000000	312631			312161		EEEEEE	512111	111333
41:	111523	CCCCCC			000000		311333	512111	BBBBBB
42:	EEEEEE	BBBBBB			CCCCCC			000000	111353
43:	111521	BBBBBB			112521			212111	000000
44:	000000	112631			212111			512111	000000
45:	112161	DDDDDD			312111			312111	121133
46:	112363	000000			CCCCCC			512111	112111
47:		212161			211313			BBBBBB	114251
48:		112111			321133			211353	112111
49:		BBBBBB			211313			000000	114151
50:		BBBBBB			212111			212111	412111
	pastry	1stmor	fashod	2ndmor	bosnia	brtprt	anschl	rhine	munich

The subsequences generally speak for themselves: they are plausible and they are clearly capturing more than random event frequencies. There are clear differences, for example, between the “Threat” subsequences and the two sets of “War” subsequences. Similarly, there are also clear differences between the pre-WWI and post-WWI war subsequences: the extensive communication and negotiation that accompany modern wars is evident in subsequences A and D.

What is surprising—although consistent with the underlying theory—is that many of the subsequences have a degree of internal consistency. For example, in “War1”, subsequence A deals largely with consultation and reaching agreements between the sides; subsequence C is primarily military activity; in War2 subsequence A is extensive international consultation, subsequence B is the main sequence of military action, and subsequence E is international agreements followed by initial hostilities. The only feature within the algorithm that might bias the selection of the subsequences to showing this internal consistency was the sorting of event codes within days, but that process seems unlikely to fully account for the consistency because the sorted codes were the frequency- recoded dyad-prefixed integers, not the original BCOW codes, and sorting applied only to multiple events in a single day. Beyond that, the internal consistency exhibited by the subsequences is purely a product of the data and is evidence that we are actually seeing repeated patterns of events. The plausibility of the subsequences is not perfect—in particular the “Mobilization” event occurs at some rather odd places—but is still striking considering the subsequences were produced by a machine with no preconceived biases for which events should be associated together.

Degree of Fit

Table 5.9 reports the degree of fit, or coverage, of each of the sequences by the set of subsequences. The measure reported is

$$\text{Fit} = \frac{\text{number of events matched by subsequences}}{L}$$

where L = minimum(length of sequence,total length of subsequences). A fit greater than one indicates that some target sequences were matched multiple times by the subsequences.

Table 5.9. Measures of Fit by Sequence

Threat

pastry	1stmor	fashod	2ndmor	bosnia	brtprt	anschl	rhine	munich
0.6452	1.5161	0.3871	0.6774	1.4839	0.4118	0.5161	0.8710	1.3226

Total coverage = 0.428
Random coverage = 0.336

War1

schles	rustrk	spam	centam	balkan	chaco
0.8095	2.6190	1.3333	0.7619	1.4762	1.9048

Total coverage = 0.204
Random coverage = 0.177

War2

italet	kash1	suez	sixday	bangla	kash2	palest
1.9200	0.5800	1.1600	1.1400	1.1200	1.1000	1.7600

Total coverage = 0.368
Random coverage = 0.361

Mixed

pastry	1stmor	fashod	2ndmor	bosnia	schles	spam	centam	balkan	chaco
0.5676	0.7568	0.3125	0.4865	1.4324	0.5676	1.3514	0.4595	1.1892	1.2973

Total coverage = 0.351
Random coverage = 0.272

The Total Coverage measure is the total number of events matched in the data set divided by the total length of the data set. Except for the “War1” set (20%), this figure is in the 35% - 40% range. This is consistent with the WEIS results in Bennett and Schrodt (1987)—which had around 35% coverage—despite the use of a completely different data set and a somewhat different sequence construction method. As in the earlier research, the total length of the subsequences is substantially less than the total length of the target sequences, so the subsequences provide a substantial reduction in the amount of information required to describe the target sequences.

As always, it is useful to gauge the extent to which these results are accounted for by pattern rather than chance. The “null model” for event sequences is less obvious than that found in parametric statistics and one could suggest at least four different null models, of decreasing randomness. In each case, a set of null sequences of the same length as the observed sequences would be constructed; the difference is how the probability of an event occurring in a sequence would be determined:

1. Equal probability
2. Probability equal to the probability of the event occurring in BCOW
3. Probability equal to the probability of the event occurring in the set of sequences
4. Probability equal to the probability of the event occurring in each sequence within the set

These models include progressively more information about the characteristics of the sequences being studied. The strongest test is criterion [4]: it can be simulated by simply shuffling the events in a sequence and applying the subsequences to the shuffled sequences. The total coverage of the shuffled sequences are reported as “Random Coverage” in Table 5.9. The subsequences cover about 30% more of the actual sequences than the random sequences in the “Threat” and “Mixed” sets, but provide only 15% additional coverage in “War1” and almost no additional coverage in “War2”. This last result was surprising and indicates that most of the regularity in “War2” is accounted for by the marginal frequencies of the events rather than the sequencing of events. The “War2” sequences tend to be longer and show a higher amount of repetition (particularly international consultations and agreements) than the other sets, which may account for the difference.

We also did some partial tests of the algorithm against random sequences created from “Threat” and “Mixed” according to criterion [3]—which is equivalent to shuffling events between sequences in a data set as well as within the sequences—and contrary to initial expectations found the random coverage to be somewhat greater than the criterion [4] random sequences—0.352 and 0.349 respectively. This may occur because the algorithm finds subsequences that are *common* to the target sequences and the random sequences produced by shuffling the entire set creates a uniform environment for detecting subsequences.

5.4.3. Using Subsequences to Discriminate Nonwar and War Crises

If common subsequences reflect complex but deliberately planned political activities, one would expect different types of crises to be characterized by different subsequences. If those subsequences are sufficiently distinct, they could then be used to discriminate between crisis types. Using a nearest neighbor approach, one can characterize each target sequence by a vector giving the fit of each of the subsequences to the target sequence; these fit vectors locate each target sequence in an N-dimensional space, where N is the number of subsequences. Ideally, sequences that have characteristics in common will cluster in this space.

Table 5.10 gives fit of the ten sequences in the "Mixed" data set to the six subsequences of that set. Fit in this table is measured as

$$\text{Fit} = \frac{\text{number events matched - number events not matched}}{\text{total length of the subsequence}}$$

The columns give the vector corresponding to each of the sequences in the set. The second half of Table 5.9 reports one measure of the distance between sequences: the Pearson product moment ® of the two vectors. Sequences which have similar fits would be expected to have a high r; dissimilar sequences a low r. This expectation is borne out in general in Table 5.9, although the results are less than spectacular.

Table 5.10. Comparing Sequences by Fit to Subsequences

Subsequence fit for each sequence in Mixed

	past	1stm	fash	2ndm	bosn	schl	spam	cent	balk	chac
A	0.00	-0.33	-0.33	-0.66	1.00	0.00	0.00	-0.33	1.33	1.33
B	0.00	0.44	-0.11	0.00	1.22	-0.33	0.77	-0.33	0.33	1.00
C	0.00	-0.33	-0.66	-0.33	-0.33	1.00	1.33	-0.33	-0.33	0.00
D	-0.40	-0.20	-0.20	-0.60	-0.40	-0.20	0.80	-0.40	0.00	-0.20
E	-0.71	0.28	-0.71	-0.14	0.28	-0.42	-0.71	-0.71	0.00	0.00
F	-0.50	-1.00	-1.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00

Correlations between sequence fits

	past	1stm	fash	2ndm	bosn	schl	spam	cent	balk	chac
pastry	1.00									
1stmor	0.08	1.00								
fashod	0.51	0.62	1.00							
2ndmor	-0.27	0.11	-0.42	1.00						
bosnia	0.40	0.47	0.45	0.16	1.00					
schles	0.49	-0.43	-0.30	-0.19	-0.45	1.00				
spam	0.47	-0.44	-0.01	0.10	-0.38	0.62	1.00			
centam	0.28	-0.77	-0.27	0.18	-0.06	0.30	0.72	1.00		
balkan	0.39	0.04	0.41	-0.46	0.73	-0.30	-0.41	0.04	1.00	
chaco	0.65	0.27	0.51	-0.16	0.92	-0.18	-0.21	0.07	0.87	1.00

Figure 5.1 uses correspondence analysis (Greenacre 1984) to cluster the sequences.[11] All of the five wars (filled dots) cluster in the center of the graph, with the nonwar crises on the periphery. In the attribute space (not shown), the two subsequences most strongly associated with the wars are A and F, which also are the only two subsequences containing “Attack” events. The visual examination of the subsequences in each data set also indicates that the threat subsequences and the war subsequences are quite dissimilar, and one would not expect the threat sequences to fit the war subsequences nearly as well as they fit their own subsequences.[12]

11 Figure 5.1 was produced without subsequence C, whose inclusion distorted the clustering in the two-dimensional map.

12 A quirk in the recoding of sequences precluded a direct test of this without a disproportional amount of effort...

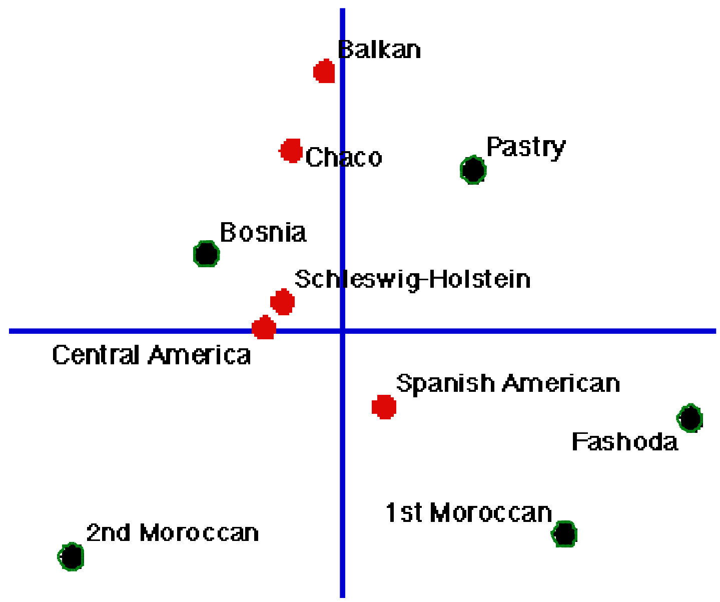

Figure 5.1. Correspondence Map of War and Nonwar Crises

Table 5.10 and Figure 5.1 are a weak test because the subsequences were chosen on the basis of their ability to describe rather than differentiate. It would not be difficult to design a similar algorithm to explicitly search for differentiating sequences, for example modifying the selection criterion in the subsequencing algorithm to maximize the coverage in one set of case while *minimizing* coverage in the other set. To construct a war-identifying subsequence the algorithm would choose, at each stage in assembling a subsequence, the event that occurs in the greatest number of war sequences and smallest number of nonwar sequences using a weighting between such as (# war) minus (# nonwar). This could be extended to the prediction problem—that is, recognizing the "warning signs" that a crisis will result in a war without knowing the entire crisis—by using as training examples the initial phase of the crisis rather than the entire crisis.

5.5. Conclusion

The Levenshtein learning algorithm is clearly only a first step in the larger puzzle of learning to deal with international events as sequences. The strength of the approach lies in its inductive nature. There are clearly simpler rules for distinguishing BCOW war and nonwar crises: looking for codes involving military conflict is the most obvious. But in order to construct those simpler rules, one must first know that distinguishing characteristic; in a sense, one must already know the answer. An inductive learning algorithm does not need to know the answer; it can find the answer. The system did not know, *a priori*, the importance of the BCOW codes designating military conflict: it discovered them. If machine learning systems can discover those distinctions, they may be capable of discovering things which are not so obvious.

If one had a very large set of sequences, it would be useful to find an archetypal centroid for each category. Because of the complexity of the computations involved in determining a Levenshtein distance, this cannot be done analytically, but it probably could be easily done with a genetic algorithm. The GA would start with the population of sequences in a cluster, then evaluate these by their average distance to all of the other sequences in the cluster. Recombination—for example of one crises that is typical in its early phases with another typical in its later phase—and a bit of mutation should work to produce an archetypal sequence near the center of the cluster.

The two methods demonstrated here are only first attempts at dealing with the problem of sequence recognition, and they are relatively simple, relying largely on a linear structuring of the sequences. A more sophisticated approach would be to impose a grammatical structure on the sequences; this would provide a more flexible specification and there are several reasons to think it might work. The basic modelling approach would be similar to that of the syntactic pattern recognition literature, which is discussed in detail in Fu (1974, 1982)

> …[In the syntactic approach] patterns are specified as being built up out of subpatterns in various ways of composition, just as phrases and sentences are built up by concatenating words, and words are built up by concatenating characters. … The rules governing the composition of primitives into patterns are usually specified by the so-called grammar of the pattern description language. After each primitive within the pattern is identified, the recognition process is accomplished by performing a syntax analysis … to determine whether or not it is syntactically correct with respect to the specified grammar. (Fu 1974: 1)

Fu further notes “one of the most attractive aspects is the recursive mature of a grammar. A grammar rule can be applied any number of times, so it is possible to express in a very compact way some basic structural characteristics of an infinite set of sentences.” In this respect, a grammar is functionally similar to a differential equation, which specifies the basic mechanisms of a process while still providing flexibility in the choice of parameters and initial values. The concept of an event grammar is by now fairly common: a useful survey of various “story grammar” concepts is found in Alker (1987).

Two recent studies have demonstrated the potential for this technique. Based on an extensive study of documents of US decision-making during the Korean War, Milliken (1994) constructs a sophisticated formal grammar of state action that can be used to characterize complex political episodes. Working in the domain of institutions rather than actions, Crawford and Ostrom (1995) develop a formal grammar of rules. Both of these projects find that a variety of political behaviors can be systematically modelled using a relatively small lexicon—in the case of Milliken, roughly the size of the BCOW coding scheme—and a set of grammatical structures.

Parallel sequences are a simple form of event grammar in the sense that they can be used to generate a set of well-formed sequences. International politics involves a large number of behaviors, but those behaviors are by no means infinite. The possible set of responses to events is further constrained by resource availability, institutional operating procedures, and custom. Predictability in the international environment is necessary for the system to function: if each event could evoke a full range of possible responses, from war to surrender, international affairs would grind to a halt amid chaos.

The common metaphor “the language of diplomacy” may also be accurate as description. Foreign policies can transmit meaning through actions as well as words, and those actions may assume a quasi-linguistic structure dependent upon a sequence of events rather than any single event. Because the explicit content of those events can vary—the USA/PRC ping pong match obviously involved more than young athletes furiously bouncing white balls across a table—the meaning of a sequence must lie to some extent in its structure. If this structure of action can be understood across time, cultures and policy substitutions, it probably has at least a rudimentary grammar.[13] This is not to say the grammar will be precise and unvarying, nor will it apply to all events that occur in the system: the diplomatic grammar of Kissinger differs from that of Khomeini just as the English grammar of William Safire differs from that of Langston Hughes or James Joyce. But most events most of the time can be expected to follow some sort of order. Behavior that deviates significantly from the expected order signals that one is dealing either with an unusual situation or with someone who doesn't know (or won't follow) the rules.

Grammars are difficult to induce using machine learning methods, but, as we note in Chapter 7, very large amounts of machine-readable text describing political events are now available and it might be possible to adapt some of the newer computational linguistic methods to work on the development of political grammars. The specification of some low-level rules and the selection of regular sequences might be sufficient to bring the problem of constructing these into the range of machine-learning—or at least machine-assisted—systems.

13 This parallel between language and action may extend further: If we assume, following Chomsky, that human linguistic abilities are at least partially genetic, then the interpretation of complex social behavior could also have a genetic component. Social interaction in most vertebrates is highly stylized and involves the communication of specific signals (i.e. "Get out of my territory", "Let's mate", "Something dangerous is coming") to invoke specific responses. These actions are frequently quite complex—particularly those involving fighting and mating—and can be at least partially described syntactically. The cognitive ability to interpret physical actions having social significance preceded the ability to process language in evolution, and linguistic abilities may have been adapted from the mental hardware used to interpret physical activity. While organizations are not under the same cognitive constraints as individuals in this respect, there may still be similarities.

Appendix 5

The Behavioral Correlates of War data set (BCOW; Leng 1987) focuses on a limited number of historical crises using a variety of historical sources, so it has a higher events density than WEIS or COPDAB. The BCOW coding scheme is similar to that in WEIS but more detailed and arguably more precise because it is optimized to code the events that occur in international crises. While BCOW codes both physical and verbal activity, we analyzed only the physical actions on the assumption that these would be more regular than verbal actions over time and across cultures. The focus on physical actions considerably shortens the sequences: this was important due to memory constraints and because the number of operations required to compute a Levenshtein distance is proportional to the product of the length of the two sequences.

5.A.1. Recoding

We used the 5-digit event code in columns 25-29 of a BCOW physical action record, then added a prefix indicating which of five types of dyadic relations were involved in the event, based on BCOW's identification of the "sides" of a conflict. Denoting BCOW's "Side A" and "Side B" as the principal actors in the conflict, and all other actors as "others", the five prefixes are:

```
1  Interaction between principals
2  Principal as initiator, other as target
3  Other as initiator, principal as target
4  Interaction within principals (e.g. between actors on same side)
5  Interaction between others
```

For example:

```
112111 = Diplomatic consultation between Side A and Side B

312111 = Diplomatic consultation initiated by an "other" and directed to
         Side A or Side B
```

Events that occur only once in a set of data—and as such cannot be part of a subsequence common to two or more crises—were recoded to zero to save space; these zeroed events are not used in determining subsequences. Events within a single day are sorted by code so that if identical events occur within a single day in two sequences they will be in the same order in both sequences.

In the parallel event sequence test, an experiment using nine codes that distinguished between Side A and Side B produced no unusual results. Side-specific codes are ambiguous since there is no firmly predetermined identity to “Side A” and “Side B”—although BCOW tends to code the victor of a dispute as Side A—and the combinatorics involved in ascertaining whether better subsequences would be found if the identities of Side A and Side B were reversed in some sequences seemed more trouble than was justified.

5.A.2. Filtering

A persistent problem encountered in Bennett and Schrodt (1987) was that common events—consultations, accusations and, in the case of war, acts of violence—constitute a large part of the data. Common events provide a great deal of regularity—for example much of the WEIS subset Bennett and Schrodt examined was taken up with 12-12-12-12-12 (Accuse) and 22-22-22-22-22 (Force)—but they contribute little to understanding.

From an information theory standpoint, these high frequency event are noise. They can be eliminated without loss of information about the underlying sequence because they are occurring with a much higher frequency than the frequency of the underlying signal. The ongoing shouting matches and wars in WEIS (and their BCOW equivalents) mask the slower, more significant processes of military escalation or diplomatic rapprochement; they are the event data equivalent of the static from a lightening storm intruding on a radio broadcast of the “Toccata and Fugue in D”. One needs, therefore, to apply a high-frequency filter to get rid of the junk in the event stream before looking for the lower-frequency regularities.

The sequences were filtered on the basis of novelty: An event of a particular code was included in the filtered sequence only if it had not occurred in the previous N days, where N is an empirically determined parameter. Novelty filtering has some face validity: To a human analyst, the onset of hostilities is important, but after that point, the day-to-day continuation of hostility provides little new information. Since BCOW codes distinguish the cessation of action more clearly than do WEIS codes, the end of a conflict or negotiation will usually be demarcated by the occurrence of a new code, which will pass through the novelty filter.

The novelty filter also deals automatically with the “nonevent” problem—the issue of whether the *absence* of activity between two actors should be coded. The resumption of an activity after a period of inaction will cause the appearance of an event code; continual activity will not. Finally, novelty filtering insures that each event found in the original data will occur at least once in the filtered event sequence.

Experiments with three BCOW data sets—fashoda, suez and cyprus—showed a clear leveling off in the size of the sequences at a filter length of about 14 to 16 days.[14] Filtering usually resulted in a file containing 40% to 60% of the original events; this was higher in short sequences (e.g. pastry, 76%) and lower in very long sequences (e.g. suez, 29%). Parallel subsequence experiments using data processed with only a three-day filter produced no unexpected differences in the results: the shared subsequences tended to be dominated by the high frequency events and the total coverage was higher.

5.A.3. Levenshtein Metric Test

The four subsets of crises listed in Table 5.A.1 were analyzed. The short names (e.g. “pastry”) correspond to the BCOW file identifiers. “Training” sequences were used to establish weights that discriminated between the war and nonwar sequences; the system was validated with the remaining sequences. The BCOW crises not included in the study are generally those whose length in events is very long (e.g. Suez or the Cuban Missile Crisis); or those that could not easily classified into war or nonwar (e.g. Trieste). No deliberate attempt was made to manipulate the results by choice of crises except that the training cases were representative of the validation cases.

14 The 14-day frequency limit, determined empirically for the BCOW data, turned out to be the same "nonevent" time period as determined by guess and intuition in Bennett and Schrodt (1987).

5.A.4. Parallel Event Sequences Test

This study used four subsets listed in Table 5.A.2. The “Threats” set contains crises that did not result in war because one side backed down; the two “War” sets contain crises that involved wars; and the “Mixed” set included five nonwar crises and five wars. The BCOW crises not analyzed are those directly preceding the two world wars and a set of largely post-WWII crisis that involve some military activity but do not escalate to a full-scale war (e.g. the Berlin airlift).

Table 5A.1. Data Sets Analyzed in Levenshtein Metric Test

Crises without war

Training Set

BCOW file	Crisis	Date	Length*
fashod	Fashoda Crisis	1898-1899	32
1stmor	First Moroccan Crisis	1904-1906	79
bosnia	Bosnian Crisis	1908-1909	116
2ndmor	Second Moroccan Crisis (Agadir)	1911	38
rhine	Rhineland Crisis	1936	65
Test Set			
pastry	Pastry War Crisis	1838-1839	41
brtprt	British-Portuguese Crisis	1889-1890	15
anschl	Anschluss Crisis	1937-1938	37
munich	Munich Crisis	1938	114
berair	Berlin Blockade	1948-1949	118

Crises involving war

Training Set

BCOW file	Crisis	Date	Length*
schles	Schleswig-Holstein War	1863-1864	52
spam	Spanish-American War	1897-1898	171
centam	2nd Central American War	1906-1907	71
chaco	Chaco Dispute and War	1927-1930	125
italet	Italo-Ethiopian War	1935-1936	260
Test Set			
balkan	Balkan Wars	1912-1913	115
palest	Palestine War	1947-1948	177
kash1	First Kashmir War	1947-1949	70
kash2	Second Kashmir War	1964-1966	76
bangla	Bangladesh War	1971	108

*Length = number of events in the filtered sequence

Table 5.A.2. Crises Analyzed in Parallel Event Sequence Test

<u>THREATS: Crises without war</u>

pastry	Pastry War Crisis	1838-1839
brtprt	British-Portuguese Crisis	1889-1890
fashod	Fashoda Crisis	1898-1899
1stmor	First Moroccan Crisis	1904-1906
bosnia	Bosnian Crisis	1908-1909
2ndmor	2nd Moroccan Crisis (Agadir)	1911
rhine	Rhineland Crisis	1936
anschl	Anschluss Crisis	1937-1938
munich	Munich Crisis	1938

<u>WAR1: Pre-WWI conflicts</u>

schles	Schleswig-Holstein War	1863-1864
rustrk	Russo-Turkish War	1877-1878
spam	Spanish-American War	1897-1898
centam	Second Central American War	1906-1907
balkan	Balkan Wars	1912-1913
chaco	Chaco Dispute and War	1927-1930

<u>WAR2: Post-WWI conflicts</u>

italet	Italo-Ethiopian War	1935-1936
palest	Palestine War	1947-1948
kash1	First Kashmir War	1947-1949
suez	Suez Crisis and Sinai War	1956-1957
kash2	Second Kashmir War	1964-1966
sixday	1967 Middle East War	1967
bangla	Bangladesh War	1971

<u>MIX: Mixture of threat and conflict cases</u>

pastry, 1stmor, fashoda, 2ndmor, bosnia,
schles, spam, centam, balkan, chaco