

Markov Process Lexicon

April 23, 2017

1 Introduction

We are interested in modeling the generation of a random lexicon as a Markov process. We will need to check

1. the columns sum to 1 (so the transition to the next state is a probability),
2. the largest eigenvalue is equal to 1,
3. the lexicon generated is finite (finite words are generated).

2 Word Generation

In the simplest formation of markov process language generation, we can imagine creating words randomly until, with a random chance γ of ending the word. To avoid single letter words we need to start the chain with two letters (we will solve this problem differently in later sections. The transition matrix for this (left) Markov process may be expressed

$$P_{ij} = \begin{bmatrix} (1-\gamma)/N & \cdots & (1-\gamma)/N & 0 \\ \vdots & & \vdots & \vdots \\ (1-\gamma)/N & \cdots & (1-\gamma)/N & 0 \\ \gamma & \cdots & \gamma & 1 \end{bmatrix}. \quad (1)$$

$$\mathbf{v}_{\text{NW}} = \begin{bmatrix} (1-\gamma)/N \\ \vdots \\ (1-\gamma)/N \\ \gamma \end{bmatrix} \quad \mathbf{v}_{\text{W}} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \quad (2)$$

$$P_{ij}\mathbf{v}_{\mathbf{NW}} = \begin{bmatrix} N((1-\gamma)/N)^2 \\ \vdots \\ N((1-\gamma)/N)^2 \\ N\gamma(1-\gamma)/N + \gamma \end{bmatrix} = \begin{bmatrix} (1-\gamma)(1-\gamma)/N \\ \vdots \\ (1-\gamma)(1-\gamma)/N \\ \gamma(2-\gamma) \end{bmatrix} \quad (3)$$

$$= (1-\gamma)\mathbf{v}_{\mathbf{NW}} + \gamma\mathbf{v}_{\mathbf{W}} \quad (4)$$

$$P_{ij}\mathbf{v}_{\mathbf{W}} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} = \mathbf{v}_{\mathbf{W}} \quad (5)$$

Write

$$P = \begin{bmatrix} Q & 0 \\ R & I \end{bmatrix}, \quad (6)$$

Where Q represents the transient-to-transient transitions, R the transient-to-absorbed transitions, and I is the identity matrix with rank corresponding to the number of absorbing states.

$$Q = \begin{bmatrix} (1-\gamma)/N & \cdots & (1-\gamma)/N \\ \vdots & & \vdots \\ (1-\gamma)/N & \cdots & (1-\gamma)/N \end{bmatrix} \quad (7)$$

The matrix

$$N = \sum_{k=0}^{\infty} Q^k = (I - Q)^{-1} \quad (8)$$

represents the number of times a transient state is expected to be hit on the way to an absorbing state starting from a specific place. If we substitute our Q matrix, we find

$$N = I + \frac{1}{\gamma}Q \quad (9)$$

Note:

$$NR = (I + \frac{1}{\gamma}Q)\gamma\mathbf{1} = \gamma\mathbf{1} + Q\mathbf{1} = (\gamma + (1-\gamma))\mathbf{1} = \mathbf{1} \quad (10)$$

$$N\mathbf{1} = (I + \frac{1}{\gamma}Q)\mathbf{1} = \mathbf{1} + \frac{1}{\gamma}Q\mathbf{1} = (1 + \frac{1-\gamma}{\gamma})\mathbf{1} = \frac{1}{\gamma}\mathbf{1} \quad (11)$$

$$\langle \ell \rangle = 1 + 1 + N\frac{(1-\gamma)}{N\gamma} = \frac{\gamma+1}{\gamma} \quad (12)$$

3 Lexicon Branch Generation

For the process, we imagine stringing characters together until, at some random point, the string is considered a word. At another random point, it's decided that no more words can be made using the last word as a starting point. There are thus three qualitatively different states for the Markov process (and degeneracies corresponding to the number of allowable characters).

The possible states from the first letter come from adding a character that does not finish a word, adding a character that finishes a word, and the bottom node (corresponding to a finished word that has no suffixes). For N characters, the transition matrix P for this process has a shape of $2N + 1 \times 2N + 1$ (N characters that don't finish words, N characters that could finish words, and the bottom node). If we imagine each character as equally likely and completable words being completed with probability α , then the transition matrix can be expressed as

$$P_{ij} = \begin{bmatrix} 1/2N & \cdots & (1 - \alpha)/2N & \cdots & 0 \\ \vdots & & \vdots & & \vdots \\ 1/2N & \cdots & (1 - \alpha)/2N & \cdots & 0 \\ 1/2N & \cdots & (1 - \alpha)/2N & \cdots & 0 \\ \vdots & & \vdots & & \vdots \\ 1/2N & \cdots & (1 - \alpha)/2N & \cdots & 0 \\ 0 & \cdots & \alpha & \cdots & 1 \end{bmatrix}. \quad (13)$$

We've got to unpack this a little. The first N rows correspond to the N characters that don't end words. The next N rows are the N characters that end words. Characters that do and do not end words may transition to each other (e.g., oend might transition to oendi on its way to oending). The latter selects the next character with equal probability ($1/2N$), but the former must retain the possibility of actually ending the branch (as no word can be constructed from $\text{oending}\bullet$). We denote the probability of ending a branch as α . This α factor is removed uniformly from the transition probabilities away from a character that end words, hence $(1 - \alpha)/2N$. Finally, the last row corresponds to the bottom node state, that cannot transition to anywhere.

The columns can similarly be partitioned into those three categories, where we define

the following vectors

$$\mathbf{v}_{\mathbf{NW}} = \begin{bmatrix} 1/2N \\ \vdots \\ 1/2N \\ 1/2N \\ \vdots \\ 1/2N \\ 0 \end{bmatrix} \quad \mathbf{v}_{\mathbf{W}} = \begin{bmatrix} (1-\alpha)/2N \\ \vdots \\ (1-\alpha)/2N \\ (1-\alpha)/2N \\ \vdots \\ (1-\alpha)/2N \\ \alpha \end{bmatrix} \quad \mathbf{v}_{\mathbf{B}} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \quad (14)$$

Clearly there are many linear dependencies between the columns (as there are N each of $\mathbf{v}_{\mathbf{NW}}$ and $\mathbf{v}_{\mathbf{W}}$). An additional linear dependence that is of note is

$$\mathbf{v}_{\mathbf{W}} = \alpha \mathbf{v}_{\mathbf{B}} + (1-\alpha) \mathbf{v}_{\mathbf{NW}}. \quad (15)$$

The existence of this linear relationship and the number of degeneracies in the matrix P suggests that there should only be two nonzero eigenvalues of P .

Before jumping to the eigendecomposition, let's explore some of the properties of P . First, the inner spaces:

$$\mathbf{v}_{\mathbf{B}} \cdot \mathbf{v}_{\mathbf{B}} = 1 \quad (16)$$

$$\mathbf{v}_{\mathbf{NW}} \cdot \mathbf{v}_{\mathbf{NW}} = 1/2N \quad (17)$$

$$\mathbf{v}_{\mathbf{W}} \cdot \mathbf{v}_{\mathbf{W}} = (1 - 2\alpha + (1 + 2N)\alpha^2)/2N \quad (18)$$

$$\mathbf{v}_{\mathbf{B}} \cdot \mathbf{v}_{\mathbf{NW}} = 0 \quad (19)$$

$$\mathbf{v}_{\mathbf{B}} \cdot \mathbf{v}_{\mathbf{W}} = \alpha \quad (20)$$

$$\mathbf{v}_{\mathbf{NW}} \cdot \mathbf{v}_{\mathbf{W}} = (1 - \alpha)/2N. \quad (21)$$

$$(22)$$

Now, evaluate how each of these vectors is acted upon by P .

$$P_{ij} \mathbf{v}_{\mathbf{NW}} = \begin{bmatrix} N(1/2N)^2 + N(1/2N)((1-\alpha)/2N) \\ \vdots \\ N(1/2N)^2 + N(1/2N)((1-\alpha)/2N) \\ N(1/2N)\alpha \end{bmatrix} = \begin{bmatrix} (2-\alpha)/4N \\ \vdots \\ (2-\alpha)/4N \\ \alpha/2 \end{bmatrix} \quad (23)$$

$$= \frac{2-\alpha}{2} \mathbf{v}_{\mathbf{NW}} + \frac{\alpha}{2} \mathbf{v}_{\mathbf{B}} \quad (24)$$

$$P_{ij}\mathbf{v}_W = \begin{bmatrix} N(1/2N)((1-\alpha)/2N) + N((1-\alpha)/2N)^2 \\ \vdots \\ N(1/2N)((1-\alpha)/2N) + N((1-\alpha)/2N)^2 \\ N\alpha(1-\alpha)/2N + \alpha \end{bmatrix} = \begin{bmatrix} (1-\alpha)(2-\alpha)/4N \\ \vdots \\ (1-\alpha)(2-\alpha)/4N \\ \alpha(3-\alpha)/2 \end{bmatrix} \quad (25)$$

$$= \frac{2-\alpha}{2}\mathbf{v}_W + \frac{\alpha}{2}\mathbf{v}_B \quad (26)$$

$$P_{ij}\mathbf{v}_B = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} = \mathbf{v}_B \quad (27)$$

Observe that the bottom node state transitions only to itself, and is therefore our first eigenvector, and has an eigenvalue of one.

Now we need to find the second unique eigenvalue, and check that it isn't greater than 1. Try

$$\mathbf{v}_2 = \mathbf{v}_B - \mathbf{v}_{NW} = \begin{bmatrix} -1/2N \\ \vdots \\ -1/2N \\ -1/2N \\ \vdots \\ -1/2N \\ 1 \end{bmatrix}, \quad (28)$$

with norm

$$\mathbf{v}_2 \cdot \mathbf{v}_2 = 2N \frac{1}{4N^2} + 1 = \frac{1+2N}{2N}. \quad (29)$$

When acted upon by P , we see

$$P_{ij}\mathbf{v}_2 = \begin{bmatrix} N(-1/2N)(1/2N) + N(-1/2N)((1-\alpha)/2N) \\ \vdots \\ N(-1/2N)(1/2N) + N(-1/2N)((1-\alpha)/2N) \\ N(-1/2N)\alpha + 1 \end{bmatrix} \quad (30)$$

$$= \begin{bmatrix} ((2-\alpha)/2)(-1/2N) \\ \vdots \\ ((2-\alpha)/2)(-1/2N) \\ (2-\alpha)/2 \end{bmatrix} = \frac{2-\alpha}{2} \begin{bmatrix} -1/2N \\ \vdots \\ -1/2N \\ 1 \end{bmatrix} = \frac{2-\alpha}{2}\mathbf{v}_2 \quad (31)$$

Alternatively,

$$P\mathbf{v}_2 = P(\mathbf{v}_B - \mathbf{v}_{NW}) = \mathbf{v}_B - \left(\frac{2-\alpha}{2}\mathbf{v}_{NW} + \frac{\alpha}{2}\mathbf{v}_B\right) \quad (32)$$

$$= \frac{1-\alpha}{2}(\mathbf{v}_B - \mathbf{v}_{NW}) = 1 - \frac{\alpha}{2}\mathbf{v}_2 \quad (33)$$

Now we may write our starting state \mathbf{v}_{NW} as

$$\mathbf{v}_{NW} = \mathbf{v}_B - \mathbf{v}_2 \quad (34)$$

Now, since

$$P\mathbf{v}_{NW} = \mathbf{v}_B - \frac{2-\alpha}{2}\mathbf{v}_2, \quad (35)$$

repeated application of the transition matrix yields

$$P^n\mathbf{v}_{NW} = \mathbf{v}_B - \left(\frac{2-\alpha}{2}\right)^n \mathbf{v}_2. \quad (36)$$

As long as $0 < \alpha \leq 1$, we know that $(2-\alpha)/2 < 1$, implying that

$$\lim_{n \rightarrow \infty} P^n\mathbf{v}_{NW} = \mathbf{v}_B, \quad (37)$$

which means that all our branches must eventually end and so we may expect finite words.

4 Fundamental Matrix

$$N = \sum_{k=0}^{\infty} Q^k = (I_t - Q)^{-1} \quad (38)$$

Now, our Q is

$$Q = \begin{bmatrix} 1/2N & \cdots & 1/2N \\ \vdots & & \vdots \\ 1/2N & \cdots & 1/2N \\ (1-\alpha)/2N & \cdots & (1-\alpha)/2N \\ \vdots & & \vdots \\ (1-\alpha)/2N & \cdots & (1-\alpha)/2N \end{bmatrix} \quad (39)$$

$$Q^* = \begin{bmatrix} 1/N\alpha & \cdots & 1/N\alpha \\ \vdots & & \vdots \\ 1/N\alpha & \cdots & 1/N\alpha \\ (1-\alpha)/N\alpha & \cdots & (1-\alpha)/N\alpha \\ \vdots & & \vdots \\ (1-\alpha)/N\alpha & \cdots & (1-\alpha)/N\alpha \end{bmatrix} \quad (40)$$

$$N = (I - Q)^{-1} = I + Q^* \quad (41)$$

The expected length of a branch (the expected number of steps before hitting the absorbing bottom node state) is $t_{\text{branch}} = 1 + 2N(1/N\alpha) = 1 + 2/\alpha$. As the probability of ending the branch decreases, this increases.

The number of words in a branch corresponds to the number of times the chain visits a word-ending character. The entries N_{ij} are the expected number of times that transient state j is visited before being absorbed, given that the chain started in state i . We will always be starting in a nonword character state, and we don't care which word-ending character we hit. Summing along one of the rows corresponding to starting at a non-word ending character, we see that the expected number of words per chain is $N/N\alpha = 1/\alpha$.

To get the expected word length, we need to compute the number of times we visit one of the characters that end words (which are transient states). The probability of visiting

$$H = (N - I_t)\text{diag}^{-1}(N) \quad (42)$$

5 Word Branch Generation, tunable word probability

$$P_{ij} = \begin{bmatrix} (1-\beta)/N & \cdots & (1-\alpha)(1-\beta)/N & \cdots & 0 \\ \vdots & & \vdots & & \vdots \\ (1-\beta)/N & \cdots & (1-\alpha)(1-\beta)/N & \cdots & 0 \\ \beta/N & \cdots & (1-\alpha)\beta/N & \cdots & 0 \\ \vdots & & \vdots & & \vdots \\ \beta/N & \cdots & (1-\alpha)\beta/N & \cdots & 0 \\ 0 & \cdots & \alpha & \cdots & 1 \end{bmatrix}. \quad (43)$$

$$\mathbf{v}_{\text{NW}} = \begin{bmatrix} (1-\beta)/N \\ \vdots \\ (1-\beta)/N \\ \beta/N \\ \vdots \\ \beta/N \\ 0 \end{bmatrix} \quad \mathbf{v}_{\text{W}} = \begin{bmatrix} (1-\alpha)(1-\beta)/N \\ \vdots \\ (1-\alpha)(1-\beta)/N \\ (1-\alpha)\beta/N \\ \vdots \\ (1-\alpha)\beta/N \\ \alpha \end{bmatrix} \quad \mathbf{v}_{\text{B}} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \quad (44)$$

$$\mathbf{v}_{\text{NW}} \cdot \mathbf{v}_{\text{W}} = \frac{1-\alpha}{N^2}(N(1-\beta)^2 + N\beta^2) = \frac{1-\alpha}{N}(1 - 2\beta + 2\beta^2) \quad (45)$$

$$Q = \begin{bmatrix} (1-\beta)/N & \cdots & (1-\alpha)(1-\beta)/N & \cdots \\ \vdots & & \vdots & \\ (1-\beta)/N & \cdots & (1-\alpha)(1-\beta)/N & \cdots \\ \beta/N & \cdots & (1-\alpha)\beta/N & \cdots \\ \vdots & & \vdots & \\ \beta/N & \cdots & (1-\alpha)\beta/N & \cdots \end{bmatrix}. \quad (46)$$

$$Q^2 = (1-\alpha\beta)Q \quad (47)$$

$$\implies Q^k = (1-\alpha\beta)^{k-1}Q \quad (48)$$

Thus

$$N = \sum_{k=0}^{\infty} Q^k = I + \frac{1}{\alpha\beta}Q. \quad (49)$$

The expected length of a chain starting from the not-word state

$$(N\mathbf{1})_{NW} = (I + \frac{1}{\alpha\beta}Q)\mathbf{1} = \mathbf{1} + \frac{1}{\alpha\beta}Q\mathbf{1} \quad (50)$$

$$= \left(1 + \frac{1}{\alpha\beta} \left(N \frac{1-\beta}{N} + N \frac{(1-\alpha)(1-\beta)}{N} \right) \right) \mathbf{1} \quad (51)$$

$$= \left(1 + \frac{1}{\alpha\beta} (1-\beta)(2-\alpha) \right) \mathbf{1} \quad (52)$$

$$\sum_{k=0}^{\infty} kQ^k = \sum_{k=1}^{\infty} k(1-\alpha\beta)^{k-1} \frac{\beta}{N} \quad (53)$$

$$= \frac{\beta}{N(1-\alpha\beta)} \sum_{k=1}^{\infty} k(1-\alpha\beta)^k \quad (54)$$

$$= \frac{\beta}{N(1-\alpha\beta)} \frac{(1-\alpha\beta)}{(1-(1-\alpha\beta))^2} \quad (55)$$

$$= \frac{\beta}{N\alpha^2\beta^2} \quad (56)$$

A Errata

We are interested in modeling the generation of a random lexicon as a Markov process. For the process, we imagine stringing characters together until, at some random point, the string is considered a word. There are thus four qualitatively different states for the

Markov process (and degeneracies corresponding to the number of allowable characters). We will need to check

1. the columns sum to 1 (so the transition to the next state is a probability),
2. the largest eigenvalue is equal to 1,
3. the lexicon generated is finite (finite words are generated).

The possible states are the top node (before adding the first letter), adding a character that does not finish a word, adding a character that finishes a word, and the bottom node (corresponding to a finished word that has no suffixes). For N characters, the transition matrix P for this process has a shape of $2(N+1) \times 2(N+1)$ (2 edge nodes, N characters that don't finish words, N characters that could finish words). If we imagine each character as equally likely and completable words being completed with probability α , then the transition matrix can be expressed as

$$P_{ij} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1/N & 1/2N & \cdots & (1-\alpha)/2N & \cdots & 0 \\ \vdots & \vdots & & \vdots & & \vdots \\ 1/N & 1/2N & \cdots & (1-\alpha)/2N & \cdots & 0 \\ 0 & 1/2N & \cdots & (1-\alpha)/2N & \cdots & 0 \\ \vdots & \vdots & & \vdots & & \vdots \\ 0 & 1/2N & \cdots & (1-\alpha)/2N & \cdots & 0 \\ 0 & 0 & \cdots & \alpha & \cdots & 1 \end{bmatrix}. \quad (57)$$

We've got to unpack this a little. The first row represents the top node state - no state should be able to return to the top node, so it is filled with zeros. The next N rows correspond to the N characters that don't end words. The top node leads only to these to exclude words that are only one character long, but it can lead to any of the characters with equal probability, hence the $1/N$ populating that segment of the first column. The next N rows are the N characters that end words. Characters that do and do not end words may transition to each other (e.g., `oend` might transition to `oendi` on its way to `oending`). The latter selects the next character with equal probability ($1/2N$), but the former must retain the possibility of actually ending the branch (as no word can be constructed from `oending`). We denote the probability of ending a branch as α . This α factor is removed uniformly from the transition probabilities away from a character that end words, hence $(1-\alpha)/2N$. Finally, the last row corresponds to the bottom node state, that cannot transition to anywhere.

The columns can similarly be partitioned into those four categories, where we define

the following vectors

$$\mathbf{v}_\mathbf{T} = \begin{bmatrix} 0 \\ 1/N \\ \vdots \\ 1/N \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \quad \mathbf{v}_{\mathbf{NW}} = \begin{bmatrix} 0 \\ 1/2N \\ \vdots \\ 1/2N \\ 1/2N \\ \vdots \\ 1/2N \\ 0 \end{bmatrix} \quad \mathbf{v}_\mathbf{W} = \begin{bmatrix} 0 \\ (1-\alpha)/2N \\ \vdots \\ (1-\alpha)/2N \\ (1-\alpha)/2N \\ \vdots \\ (1-\alpha)/2N \\ \alpha \end{bmatrix} \quad \mathbf{v}_\mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \quad (58)$$

Clearly there are many linear dependencies between the columns (as there are N each of $\mathbf{v}_{\mathbf{NW}}$ and $\mathbf{v}_\mathbf{W}$). An additional linear dependence that is of note is

$$\mathbf{v}_\mathbf{W} = \alpha \mathbf{v}_\mathbf{B} + (1-\alpha) \mathbf{v}_{\mathbf{NW}} \quad (59)$$

We proceed now to evaluate how each of these vectors is acted upon by P .

$$P_{ij} \mathbf{v}_\mathbf{T} = \begin{bmatrix} 0 \\ N(1/N)(1/2N) \\ \vdots \\ N(1/N)(1/2N) \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1/2N \\ \vdots \\ 1/2N \\ 0 \end{bmatrix} = \mathbf{v}_{\mathbf{NW}} \quad (60)$$

$$P_{ij} \mathbf{v}_{\mathbf{NW}} = \begin{bmatrix} 0 \\ N(1/2N)^2 + N(1/2N)((1-\alpha)/2N) \\ \vdots \\ N(1/2N)^2 + N(1/2N)((1-\alpha)/2N) \\ N(1/2N)\alpha \end{bmatrix} = \begin{bmatrix} 0 \\ (2-\alpha)/4N \\ \vdots \\ (2-\alpha)/4N \\ \alpha/2 \end{bmatrix} \quad (61)$$

$$= \frac{2-\alpha}{2} \mathbf{v}_{\mathbf{NW}} + \frac{\alpha}{2} \mathbf{v}_\mathbf{B} \quad (62)$$

$$P_{ij} \mathbf{v}_\mathbf{W} = \begin{bmatrix} 0 \\ N(1/2N)((1-\alpha)/2N) + N((1-\alpha)/2N)^2 \\ \vdots \\ N(1/2N)((1-\alpha)/2N) + N((1-\alpha)/2N)^2 \\ N\alpha(1-\alpha)/2N + \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ (1-\alpha)(2-\alpha)/4N \\ \vdots \\ (1-\alpha)(2-\alpha)/4N \\ \alpha(3-\alpha)/2 \end{bmatrix} \quad (63)$$

$$= \frac{2-\alpha}{2} \mathbf{v}_\mathbf{W} + \frac{\alpha}{2} \mathbf{v}_\mathbf{B} \quad (64)$$

$$P_{ij}\mathbf{v}_B = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} = \mathbf{v}_B \quad (65)$$

Note that \mathbf{v}_T transitions into the “not word” state, which is what we wanted. Next, observe that the bottom node state transitions only to itself, and is therefore our first eigenvector, and has an eigenvalue of one.

$$\mathbf{v}_B \cdot \mathbf{v}_B = 1 \quad (66)$$

$$\mathbf{v}_T \cdot \mathbf{v}_T = 1/N \quad (67)$$

$$\mathbf{v}_{NW} \cdot \mathbf{v}_{NW} = 1/2N \quad (68)$$

$$\mathbf{v}_W \cdot \mathbf{v}_W = (1 - 2\alpha + (1 + 2N)\alpha^2)/2N \quad (69)$$

$$\mathbf{v}_B \cdot \mathbf{v}_T = 0 \quad (70)$$

$$\mathbf{v}_B \cdot \mathbf{v}_{NW} = 0 \quad (71)$$

$$\mathbf{v}_B \cdot \mathbf{v}_W = \alpha \quad (72)$$

$$\mathbf{v}_T \cdot \mathbf{v}_{NW} = 1/2N \quad (73)$$

$$\mathbf{v}_T \cdot \mathbf{v}_W = (1 - \alpha)/2N \quad (74)$$

$$\mathbf{v}_{NW} \cdot \mathbf{v}_W = (1 - \alpha)/2N \quad (75)$$

$$(76)$$

$$\mathbf{u}_1 = \mathbf{v}_B \quad (77)$$

$$\mathbf{u}_2 = \mathbf{v}_W - \frac{\mathbf{v}_B \cdot \mathbf{v}_W}{\mathbf{v}_B \cdot \mathbf{v}_B} \mathbf{v}_B = \mathbf{v}_W - \alpha \mathbf{v}_B \quad (78)$$

$$P_{ij}\mathbf{u}_2 = P(\mathbf{v}_W - \alpha \mathbf{v}_B) = P\mathbf{v}_W - \alpha P\mathbf{v}_B \quad (79)$$

Substituting from eqs 64 and 65

$$= \frac{2 - \alpha}{2} \mathbf{v}_W + \frac{\alpha}{2} \mathbf{v}_B - \alpha \mathbf{v}_B \quad (80)$$

$$= \frac{2 - \alpha}{2} \mathbf{v}_W - \frac{\alpha}{2} \mathbf{v}_B \quad (81)$$

Try

$$\mathbf{v}_2 = \mathbf{v}_B - \mathbf{v}_{NW} = \begin{bmatrix} 0 \\ -1/2N \\ \vdots \\ -1/2N \\ -1/2N \\ \vdots \\ -1/2N \\ 1 \end{bmatrix} \quad (82)$$

$$P_{ij}\mathbf{v}_2 = \begin{bmatrix} 0 \\ N(-1/2N)(1/2N) + N(-1/2N)((1-\alpha)/2N) \\ \vdots \\ N(-1/2N)(1/2N) + N(-1/2N)((1-\alpha)/2N) \\ N(-1/2N)\alpha + 1 \end{bmatrix} \quad (83)$$

$$= \begin{bmatrix} 0 \\ ((2-\alpha)/2)(-1/2N) \\ \vdots \\ ((2-\alpha)/2)(-1/2N) \\ (2-\alpha)/2 \end{bmatrix} = \frac{2-\alpha}{2} \begin{bmatrix} 0 \\ -1/2N \\ \vdots \\ -1/2N \\ 1 \end{bmatrix} = \frac{2-\alpha}{2} \mathbf{v}_2 \quad (84)$$

$$\mathbf{v}_2 \cdot \mathbf{v}_2 = 2N \frac{1}{4N^2} + 1 = \frac{1+2N}{2N} \quad (85)$$

Alternatively,

$$P\mathbf{v}_2 = P(\mathbf{v}_B - \mathbf{v}_{NW}) = \mathbf{v}_B - \left(\frac{2-\alpha}{2}\mathbf{v}_{NW} + \frac{\alpha}{2}\mathbf{v}_B\right) \quad (86)$$

$$= \frac{2-\alpha}{2}(\mathbf{v}_B - \mathbf{v}_{NW}) = \frac{2-\alpha}{2}\mathbf{v}_2 \quad (87)$$