Cleavage Site Analysis Using Rule Extraction from Neural Networks

Yeun-Jin Cho and Hyeoncheol Kim

Department of Computer Science Education,
Korea University, Anam-Dong 5-Ga, Seoul, 136-701, Korea
{jx, hkim}@comedu.korea.ac.kr

Abstract. In this paper, we demonstrate that the machine learning approach of *rule extraction from a trained neural network* can be successfully applied to SARS-coronavirus cleavage site analysis. The extracted rules predict cleavage sites better than consensus patterns. Empirical experiments are also shown.

1 Introduction

The first cases of severe acute respiratory syndrome (SARS) were identified in Guangdong Province, China in November, 2002 and have spread to Hong Kong, Singapore, Vietnam, Canada, the USA and several European countries [20]. An outbreak of a life-threatening disease referred to as SARS has spread to many countries around the world. By late June 2003, the World Health Organization (WHO) has recorded more than 8400 cases of SARS and more than 800 SARS-related deaths, and a global alert for the illness was issued due to the severity of the disease [25]. A growing body of evidence has convincingly shown that SARS is caused by a novel coronavirus, called SARS-coronavirus or SARS-CoV [14,19]. A novel SARS associated with coronavirus (SARS-CoV) has been implicated as the causative agent of a worldwide outbreak of SARS during the first 6 months of 2003 [16,24]. Currently, the complete genome sequences of 11 strains of SARS-CoV isolated from some SARS patients have been sequenced, and more complete genome sequences of SARS-CoV are expected to come [13]. It is also known that the process of cleaving the SARS-CoV polyproteins by a special proteinase, the so-called SARS coronavirus main proteinase (CoV Mpro), is a key step for the replication of SARS-CoV [18]. The importance of the 3CL proteinase cleavage sites not only suggests that this proteinase is a culprit of SARS, but also makes it an attractive target for developing drugs directly against the new disease [3,10,23].

Several machine learning approaches including artificial neural networks have been applied to proteinase cleavage site analysis [1,3,4,7,15]. Even though neural network model has been successfully used for the analysis [1,7], one of the major weakness of the neural network is its lack of explanation capability. It is hidden in a black box and can be used to predict, but not to explain domain knowledge in explicit format. In recent years, there have been studies on rule extraction from

L. Wang, K. Chen, and Y.S. Ong (Eds.): ICNC 2005, LNCS 3610, pp. 1002–1008, 2005.

feed-forward neural networks [1,5,6,7,8,12,17,21,22]. The extracted rules provide human users with the capability to explain how the patterns are classified and may provide better insights about the domain. Thus, it is used for various data mining applications.

In this paper, we investigate the SARS-CoV cleavage site analysis using feed-forward neural networks. Also we demonstrate how to extract prediction rules for cleavage sites using the approach of *rule extraction from neural networks.* Experimental results compared to other approaches are also shown.

2 Rule Mining for SARS-CoV

Kiemer, *et al.* used feedforward neural networks for SARS-CoV cleavage site analysis [11]. They showed that the neural network outperforms three consensus patterns in terms of classification performance. The three consensus patterns are 'LQ', 'LQ[S/A]' and '[T/S/A]X[L/F]Q[S/A/G]'. However, they used the neural network for just cleavage site prediction, but not for understanding the sites in explicit knowledge. In this paper, we extract If-Then rules from the neural networks and then compare the generated rules to the consensus patterns. Our experiments includes the followings:

- Training of a feedforward neural network with known SASS-CoV cleavage site data.
- Extraction of If-Then rules from the trained neural network.
- Performance comparison of the extracted rules over consensus rules.

In this paper, we use decompositional approach for rule extraction. Decompositional approaches to rule extraction from a trained neural network (i.e., a feed-forward multi-layered neural network) involves the following phases:

1. Intermediate rules are extracted at the level of individual units within the network. At each non-input unit of a trained network, n incoming connection weights and a threshold are given. Rule extraction at the unit searches a set of incoming binary attribute combinations that are valid and maximally-general (i.e., size of each combination is as small as possible).
2. The intermediate rules from each unit are aggregated to form the composite rule base for the neural network. It rewrites rules to eliminate the symbols which refer to hidden units but are not predefined in the domain. In the process, redundancies, subsumptions, and inconsistencies are removed.

There have been many studies for efficient extraction of valid and general rules. One of the issues is time complexity of the rule extraction procedure. The rule extraction is computationally expensive since the rule search space is increased exponentially with the number of input attributes. If a node has n incoming nodes, there are 3^n possible combinations. Kim [12] introduced a computationally efficient algorithm called OAS(Ordered-Attribute Search). In this paper, the OAS is used for extraction of one or two best rules from each node.

3 Experimental Results

3.1 SARS-CoV Cleavage Sites

Twenty-four genomic sequences of coronavirus and the annotation information were downloaded from the GenBank database [2], of which 12 are SARS-CoVs and 12 are other groups of coronaviruses. The former includes SARS-CoV TOR2, Urbani, HKU-39849, CUHK-W1, BJ01, CUHK-Su10, SIN2500, SIN2748, SIN2679, SIN2774, SIN2677 and TW1, whereas the latter comprises IBV, BCoV, bovine coronavirus strain Mebus (BCoVM), bovine coronavirus isolate BCoV-LUN (BCoVL) , bovine coronavirus strain Quebec (BCoVQ), HCoV-229E (NC002645), MHV, murine hepatitis virus strain ML-10 (MHVM), murine hepatitis virus strain 2 (MHV2), murine hepatitis virus strain Penn 97-1 (MHVP), PEDV and TGEV [9].

The data set includes 8 regions (i.e., 8 positions of P4, P3, P2, P1, P1', P2', P3', P4') and one class attribute. Each region represents one of the 20 amino acids and the class attribute tells if the instance with 8 region values belongs to either cleavage (i.e., 1) or non-cleavage (i.e., 0). Each region value that is one of 20 amino acids is converted into 20 binary values in which only one of them is 1 and the rests are 0s. This binary encoding is illustrated in table 1. Thus, we have 160 (i.e., $8 * 20$) binary input nodes and one output node for our neural network architecture. From each sequence of coronavirus genome, eleven cleavage sites are included and thus, total 264 ($= 24 * 11$) sites are available. We eliminated duplicated ones out of the total 264 results and identified final seventy cleavage sites. For training a neural network classifier, negative examples (presumed non-cleavage sites) are created by defining all other glutamines in the viral polyproteins as non-cleavable sites [11]. Therefore, the data set include 70 positive (i.e., cleavage) and 281 negative (i.e., non-cleavage) examples. Three-fold cross-validation were used for classifier evaluation. That is, every test set contains 117 examples of which 23 or 24 were positive examples.

3.2 Performance of Extracted Rules

We configured neural networks with 160 input nodes, 2 hidden nodes and 1 output nodes and trained them with training sets. The classification performance of the neural networks is shown in table 2.

We used the OAS algorithm to extract rules from trained neural networks [12] and compared classification performance over consensus rules. The consensus patterns of 'LQ', 'LQ[S/A]' and '[T/S/A]X[L/F]Q[S/A/G]' can be converted into the form of rules. All of the consensus patterns have the 'Q' at position p1, and all of 70 cleavage site examples have the 'Q' at position p1. Thus, we created negative examples with 'Q' at position p1 for fair comparison. Then the consensus patterns we use are 'L.', 'L.[S/A]' and '[T/S/A]X[L/F].[S/A/G]' and they are converted into the form of rules in table 3. The '.' represents the position p1. Their performance on examples are also shown in the table. We extracted five best rules from each of the three trained neural networks. The rules and their performance are shown in table 4. A rule is in the form of *"IF condition,*

Table 1. Each amino acid is assigned to a sequence of 20 bits

amino acid	binary code
a	1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
c	0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
d	0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
e	0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
f	0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
g	0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
h	0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0
i	0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0
k	0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
l	0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0
m	0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
n	0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
p	0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
q	0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
r	0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
s	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0
t	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0
v	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0
w	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0
y	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1

Table 2. Performance of three neural networks

data	training accuracy	generalization
dataset1	100%	95.7%
dataset2	100%	97.4%
dataset3	100%	95.7%
average	100%	96.3%

THEN class" where *class* is either of cleavage or non-cleavage. *Coverage* and *accuracy* as defined as follows:

$$Coverage = \frac{\text{Number of examples matched by the } \textit{condition} \text{ part}}{\text{Total number of examples}}$$

$$Accuracy = \frac{\text{Number of true positive examples}}{\text{Number of examples matched by the } \textit{condition} \text{ part}}$$

The five rules extracted generally outperforms the consensus rules. Coverage is reasonably high and accuracy is very high. The rule 'L@p2' in consensus patterns actually subsumes 11 other rules in the table 3. While its coverage is high (i.e. 55.6%), its accuracy is low compared to others. The rules that we extracted also contain the 'L' at position p2, but we excluded the rule 'L@p2' by our 90% of rule extraction threshold.

Table 3. Consensus rules and their performance

Consensus Rules	dataset1		dataset2		dataset3	
	coverage	accuracy	coverage	accuracy	coverage	accuracy
$L@p2$	55.6	83.1	55.6	83.1	55.6	83.1
$L@p2 \wedge S@p1'$	23.1	92.6	23.1	92.6	23.1	92.6
$L@p2 \wedge A@p1'$	16.2	94.7	16.2	94.7	16.2	94.7
$T@p4 \wedge L@p2 \wedge S@p1'$	6.0	100.0	6.0	100.0	6.0	100.0
$T@p4 \wedge L@p2 \wedge A@p1'$	2.6	100.0	2.6	100.0	2.6	100.0
$T@p4 \wedge L@p2 \wedge G@p1'$	0.9	100.0	0.9	100.0	0.9	100.0
$T@p4 \wedge F@p2 \wedge S@p1'$	0.0	0.0	0.0	0.0	0.0	0.0
$T@p4 \wedge F@p2 \wedge A@p1'$	0.0	0.0	0.0	0.0	0.0	0.0
$T@p4 \wedge F@p2 \wedge G@p1'$	0.0	0.0	0.0	0.0	0.0	0.0
$S@p4 \wedge L@p2 \wedge S@p1'$	6.0	85.7	6.0	85.7	6.0	85.7
$S@p4 \wedge L@p2 \wedge A@p1'$	1.7	100.0	1.7	100.0	1.7	100.0
$S@p4 \wedge L@p2 \wedge G@p1'$	0.0	0.0	0.0	0.0	0.0	0.0
$S@p4 \wedge F@p2 \wedge S@p1'$	1.7	100.0	1.7	100.0	1.7	100.0
$S@p4 \wedge F@p2 \wedge A@p1'$	0.0	0.0	0.0	0.0	0.0	0.0
$S@p4 \wedge F@p2 \wedge G@p1'$	0.0	0.0	0.0	0.0	0.0	0.0
$A@p4 \wedge L@p2 \wedge S@p1'$	3.4	75.0	3.4	75.0	3.4	75.0
$A@p4 \wedge L@p2 \wedge A@p1'$	1.7	100.0	1.7	100.0	1.7	100.0
$A@p4 \wedge L@p2 \wedge G@p1'$	0.0	0.0	0.0	0.0	0.0	0.0
$A@p4 \wedge F@p2 \wedge S@p1'$	0.9	0.0	0.9	0.0	0.9	0.0
$A@p4 \wedge F@p2 \wedge A@p1'$	0.0	0.0	0.0	0.0	0.0	0.0
$A@p4 \wedge F@p2 \wedge G@p1'$	0.0	0.0	0.0	0.0	0.0	0.0

Table 4. Extracted rules from neural networks trained on three data sets, and their performance

Data	Extracted Rules	Coverage	Accuracy
dataset 1	$L@p2 \wedge S@p1'$	23.1	92.6
	$L@p2 \wedge A@p1'$	16.2	94.7
	$L@p2 \wedge T@p4$	12.0	100.0
	$L@p2 \wedge R@p3$	10.3	100.0
	$L@p2 \wedge S@p4$	7.7	88.9
dataset 2	$L@p2 \wedge E@p2'$	12.8	100.0
	$L@p2 \wedge T@p4$	12.0	100.0
	$L@p2 \wedge V@p4$	12.0	100.0
	$L@p2 \wedge P@p4$	12.0	100.0
	$L@p2 \wedge K@p2'$	3.4	100.0
dataset 3	$L@p2 \wedge V@p4$	12.0	100.0
	$L@p2 \wedge T@p4$	12.0	100.0
	$L@p2 \wedge P@p4$	6.0	100.0
	$L@p2 \wedge T@p3$	4.3	100.0
	$L@p2 \wedge K@p2'$	3.4	100.0

4 Conclusions

For SARS-CoV cleavage site analysis, we used the approach of rule extraction from neural networks. We trained 3-layered feedforward neural networks on genomic sequences of coronaviruses, and then extracted IF-THEN rules from the neural networks. Their performances are compared to consensus patterns. The results are promising. Rule mining using neural network classifier can be a useful tool for cleavage site analysis.

References

1. Andrews, Robert, Diederich, Joachim, Tickle, Alam B.: Survey and critique of techniques for extracting rules from trained artificial neural networks. Knowledge-Based Systems **8(6)** (1995) 373-389
2. Benson DA, Karsch-Mizrachi I, Lipman DJ, Ostell J, Wheeler DL.: GenBank: update. Nucleic Acids Res, 32 Database issue: (2004)D23-26
3. Blom N, Hansen J, Blaas D, Brunak S.: Cleavage site analysis in picornaviral polyproteins: discovering cellular targets by neural networks. Protein Sci (1996) 5:2203-2216.
4. Chen LL, Ou HY, Zhang R, Zhang CT.: ZCURVE-CoV: a new system to recognize protein coding genes in coronavirus genomes, and its applications in analyzing SARS-CoV genomes. SCIENCE DIRECT, BBRC (2003) 382-388
5. Fu, LiMin.: Neural Networks in Computer Intelligence. McGraw Hill, Inc., (1994)
6. Fu, LiMin.: Rule generation from neural networks. IEEE Transactions on Systems, Man, and Cybernetics **24(8)** (1994) 1114–1124
7. Fu, LiMin.: Introduction to knowledge-based neural networks. Knowledge-Based Systems **8(6)** (1995) 299–300
8. Fu, LiMin and Kim, Hyeoncheol.: Abstraction and Representation of Hidden Knowledge in an Adapted Neural Network. unpublished, CISE, University of Florida (1994)
9. Gaoa F, Oua HY, Chena LL, Zhenga WX, Zhanga CT.: Prediction of proteinase cleavage sites in polyproteins of coronaviruses and its applications in analyzing SARS-CoV genomes. FEBS Letters 553 (2003) 451-456
10. Hu LD, Zheng GY, Jiang HS, Xia Y, Zhang Y, Kong XY.: Mutation analysis of 20 SARS virus genome sequences: evidence for negative selection in replicase ORF1b and spike gene. Acta Pharmacol Sin (2003) 741-745
11. Kiemer L, Lund O, Brunak S, Blom N.: Coronavirus 3CL-pro proteinase cleavage sites: Possible relevance to SARS virus pathology. BMC Bioinformatics (2004)
12. Kim, Hyeoncheol.: Computationally Efficient Heuristics for If-Then Rule Extraction from Feed-Forward Neural Networks. Lecture Notes in Artificial Intelligence, Vol. 1967 (2000) 170–182
13. Luo H, Luo J.: Initial SARS Coronavirus Genome Sequence Analysis Using a Bioinformatics Platform. APBC2004, Vol. 29 (2004)

14. Marra MA, Jones SJM, Astell CR, Holt RA, Brooks-Wilson A, Butterfield YSN, Khattra J, Asano JK, Barber SA, Chan SY, CloutierA, Coughlin SM, Freeman D, Girn N, Griffith OL, Leach SR, Mayo M, McDonald H, Montgomery SB, Pandoh PK, Petrescu AS, Robertson AG, Schein JE, Siddiqui A, Smailus DE, Stott JM, Yang GS, Plummer F, Andonov A, Artsob H, Bastien N, Bernard K, Booth TF, Bowness D, Czub M, Drebot M, Fernando L, Flick R, Garbutt M, Gray M, Grolla A, Jones S, Feldmann H, Meyers A, Kabani A, Li Y, Normand S, Stroher U, Tipples GA, Tyler S, Vogrig R, Ward D, Watson B, Brunham RC, Krajden M, Petric M, Skowronski DM, Upton C, Roper RL.: The Genome Sequence of the SARS-Associated Coronavirus. SCIENCE VOL 300 (2003) 1399-1404
15. Narayanan, A., Wu, X., Yang, Z.R.: Mining viral protease data to extract cleavage knowledge. bioinformatics, **18(1)** (2002) s5–s13.
16. Ruan Y, Wei CL, Ee LA, Vega VB, Thoreau H, Yun STS, Chia JM, Ng P, Chiu KP, Lim L, Tao Z, Peng CK, Ean LOL, Lee NM, Sin LY, Ng LFP, Chee RE, Stanton LW, Long PM, Liu ET.: Comparative full-length genome sequence analysis of 14 SARS coronavirus isolates and common mutations associated with putative origins of infection. THE LANCET o Published online (2003)
17. Setino, Rudy, Liu, Huan: Understanding neural networks via rule extraction. Proceedings of the 14th International Conference on Neural Networks. **(1)** Montreal, Canada (1995) 480–485
18. Shi J, Wei Z, Song J.: Dissection Study on the Severe Acute Respiratory Syndrome 3C-like Protease Reveals the Critical Role of the Extra Domain in Dimerization of the Enzyme. THE JOURNAL OF BIOLOGICAL CHEMISTRY Vol. 279, No. 23 (2004) 24765-24773
19. Shi Y, Yi Y, Li P, Kuang T, Li L, Dong M, Ma Q, Cao C.: Diagnosis of Severe Acute Respiratory Syndrome (SARS) by Detection of SARS Coronavirus Nucleocapsid Antibodies in an Antigen-Capturing Enzyme-Linked Immunosorbent Assay. JOURNAL OF CLINICAL MICROBIOLOGY (2003) 5781-5782
20. Stadler K, Masignani V, Eickmann M, Becker S, Abrignani S, Klenk HD, Rappuoli R.: SARS - BEGINNING TO UNDERSTAND A NEW VIRUS. NATURE REVIEWS, MICROBIOLOGY VOLUME 1 (2003) 209-218
21. Taha, Ismali A. and Ghosh, Joydeep: Symbolic interpretation of artificial neural networks. IEEE Transactions on Knowledge and Data Engineering **11(3)** (1999) 443–463
22. Towell, Geoffrey G. and Shavlik, Jude W.: Extracting refined rules from knowledge-based neural networks. Machine Learning **13(1)** (1993)
23. Tsur S.: Data Mining in the Bioinformatics Domain. Proceedings of the 26th VLDB Conference, Cairo, Egypt (2000)
24. Xu D, Zhang Z, Chu F, Li Y, Jin L, Zhang L, Gao GF, Wang FS.: Genetic Variation of SARS Coronavirus in Beijing Hospital. Emerging Infectious Diseases (www.cdc.gov/eid) Vol. 10, No. 5 (2004)
25. Yap YL, Zhang XW, Danchin A.: Relationship of SARS-CoV to other pathogenic RNA viruses explored by tetranucleotide usage profiling. BMC Bioinformatics (2003)