Capítulo 6. Resultados comparativos

En este capítulo se presentan los resultados obtenidos al llevar a cabo la simulación del IDS con los distintos modelos contemplados: el primero utiliza una red tipo Elman entrenada con el algoritmo de retro propagación con tasa de aprendizaje adaptativa y *momentum*, la segunda arquitectura es la recurrente totalmente conectada entrenada con el algoritmo de RTRL, la tercera arquitectura implementada fue la auto recurrente con wavelets entrenada con el algoritmo de gradiente descendente deducido en la sección 3.7, finalmente se presenta la arquitectura propuesta llamada auto recurrente con unidades radiales multidimensionales entrenada con el algoritmo descrito en el capítulo 4. Mediante estos resultados podremos evaluar distintos indicadores como velocidad de convergencia, error obtenido, porcentaje de clasificación e identificación así como tasas de falsos positivos y falsos negativos que nos permitan generar conclusiones sobre el desempeño de cada uno de los modelos.

6.1. Introducción

Durante la etapa de entrenamiento, se tomaron un total del 70% del total de muestras disponibles de cada categoría y el otro 30% restante para llevar a cabo las pruebas del modelo. En cada uno de los modelos se consideraron el mismo número de entradas y salidas, variando solamente el número de unidades en las capas intermedias Las unidades de salida especifican cada una de las categorías a las que puede pertenecer un patrón, estableciendo un valor cercano a uno si se clasifica dentro de esa categoría o cero en caso contrario.

El funcionamiento correcto del IDS consiste en distinguir entre patrones que representan ataques y aquellos que describen tráfico de red común, a esta medida se le conoce como porcentaje de clasificación. Además, el IDS puede determinar el tipo de ataque que representan los patrones malignos, lo que se conoce como porcentaje de identificación. Como se señala en [ALA08], se espera que para cada uno de los modelos contemplados el porcentaje de clasificación sea mayor que el porcentaje de identificación. Debido a que distintos tipos de ataques pueden compartir características similares, durante la etapa de pruebas, en algunos casos un patrón que represente un ataque será clasificado en una categoría que no le corresponde aunque si sea detectado como patrón dañino.

6.2. Evaluación

Para llevar a cabo una evaluación del sistema de detección de intrusos debemos establecer indicadores que nos permitan conocer el porcentaje de ataques o conductas intrusivas detectadas y también el porcentaje de falsas alarmas emitidas. De la misma forma que los IDS comerciales, nuestra implementación puede verse afectada al variar el umbral de alerta. Es decir que se debe buscar un equilibrio para tener una probabilidad de detección alta manteniendo la emisión de falsas alarmas en porcentajes bajos. En resumen, se podrían ajustar los parámetros de cada topología de la red neuronal para detectar más ataques pero esto ocasionaría también que la tasa de falsas alarmas también se incrementara [ORT04].

La evaluación de los distintos modelos se concentra principalmente en dos indicadores. El primero es el de falsos positivos, mediante este, se indican las falsas alarmas y corresponden a patrones que son etiquetados como dañinos cuando en realidad se traten de elementos inofensivos. El segundo indicador es el de falsos negativos, se trata de ataques o conductas intrusivas con características similares a eventos normales que no son detectados por el IDS. Además de los falsos positivos y falsos negativos podemos también definir el indicador de verdaderos positivos y el de verdaderos negativos. El primero, describe la clasificación correcta de patrones dañinos y el segundo, la clasificación correcta de tráfico normal.

Además del cálculo de los indicadores anteriores es útil definir dos medidas adicionales. La primera se relaciona con el grado de especificación del modelo a evaluar, es decir, el porcentaje de reconocimiento para los patrones de tráfico de red que no son intrusivos; esta medida se llama índice de especificación y está relacionada con la emisión de falsos positivos. La segunda medida es la de sensibilidad y describe el porcentaje de detección de conductas intrusivas llevadas a cabo correctamente. El cálculo se describe en las siguientes fórmulas [FAW04]:

$$especificación = \frac{número\ de\ verdaderos\ \ negativos}{número\ de\ verdaderos\ \ negativos + número\ de\ falsos\ positivos} \qquad (6.\ 1)$$

$$sensibilidad = \frac{número\ de\ verdaderos\ \ positivos}{número\ de\ verdaderos\ \ positivos + número\ de\ falsos\ negativos} \qquad (6.\ 2)$$

6.2.1. Indicadores

La tasa de falsos positivos se calcula por medio de la fórmula [FAW04]:

$$tasa\ falsos\ positivos = \frac{FP}{(FP+TN)} = 1 - IE \quad (6.3)$$

La tasa de falsos negativos está dada por [FAW04]:

$$tasa\ falsos\ negativos = \frac{FN}{(TP+FN)} = 1 - IS \quad (6.4)$$

donde:

Tabla 6. 1: Abreviaturas de indicadores

FP	Es el número de falsos positivos
FN	Es el número de falsos negativos
TN	Es el número de verdaderos negativos
TP	Es el número de verdaderos positivos
IE	Es el índice de especificación
IS	Es el índice de sensibilidad

6.3. Arquitectura Elman

Para la implementación del modelo de red tipo Elman, se utilizó la caja de herramientas de redes neuronales de Matlab a través de su función "newelm". Esta arquitectura consiste en dos capas de procesamiento con 30 unidades cada una y una capa de salida con 5 unidades. Para las unidades de procesamiento se utilizó como función de transferencia la tangente hiperbólica sigmoide, mientras que para las unidades de salida la función de transferencia fue la lineal.

En el proceso de entrenamiento se empleó la función "traingdx", la cual corresponde a un entrenamiento que usa el algoritmo de retro propagación con *momentum* y tasa de aprendizaje adaptativa. El comportamiento del error por época se muestra en la Figura 6.1.

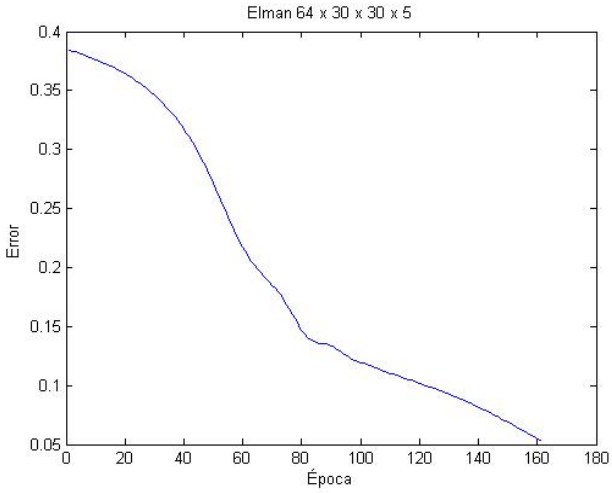

Figura 6. 1: Comportamiento del error por época en red Elman

6.4. Arquitectura FRNN

Para la red recurrente totalmente conectada se utilizó una capa de procesamiento con 30 unidades y una capa de salida con 5 unidades. Las unidades de procesamiento utilizaron como función de transferencia la logarítmica sigmoide y las unidades de salida la lineal.

En la etapa de entrenamiento se utilizó el algoritmo descrito en [WIL89], el comportamiento del error a través de las distintas épocas se muestra en la Figura 6.2.

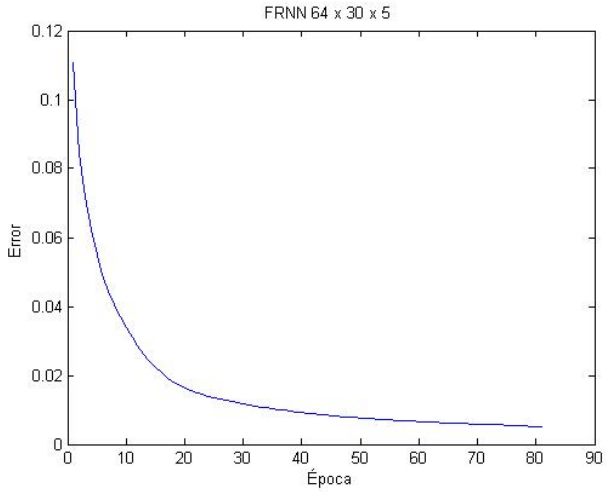

Figura 6. 2: Comportamiento del error por época en red recurrente totalmente conectada

6.5. Arquitectura SRWNN

El modelo auto recurrente basado en wavelets contempla 256 unidades en la capa *wavelon*, 4 unidades *wavelon* multidimensionales en su capa de producto y 5 unidades en su capa de salida. La función de transferencia para las unidades en la capa *wavelon* fue calculada a partir de la wavelet madre conocida como primera derivada Gaussiana modificada por los parámetros de traslación y escalamiento. El cálculo de la salida de las unidades de la capa de producto es el resultado de multiplicar las salidas de la capa anterior. Finalmente las unidades de salida representan combinadores lineales.

El comportamiento del algoritmo durante la etapa de entrenamiento se muestra en la Figura 6.3.

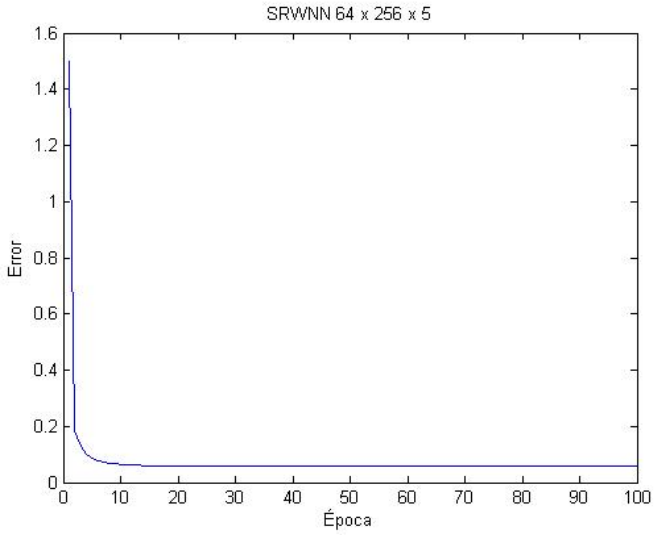

Figura 6. 3: Comportamiento del error por época en red auto recurrente con wavelets

6.6. Arquitectura SRWNN – MRW

Se trata del modelo propuesto y representa una arquitectura de 3 capas donde las unidades de procesamiento son unidades radiales multidimensionales. A diferencia de la arquitectura anterior, esta unidad implementa un número menor de unidades en su capa oculta debido al procesamiento realizado en los nodos *R*. La principal diferencia entre las dos arquitecturas auto recurrentes es la forma de llevar a cabo el procesamiento de múltiples entradas en sus nodos de procesamiento, dando lugar a un nuevo tipo de red neuronal y su algoritmo de entrenamiento.

El comportamiento del algoritmo durante la etapa de entrenamiento se muestra en la Figura 6.4.

6.7. Resultados Comparativos

La Tabla 6.2 muestra los resultados comparativos entre los distintos modelos utilizados en el IDS, la primera columna describe el tipo de arquitectura utilizada así como el número de unidades en cada capa, la segunda columna especifica el número de épocas utilizadas en el entrenamiento, la tercera el error obtenido y las 4 columnas restantes el número de falsos positivos, falsos negativos, verdaderos positivos y verdaderos negativos. Como puede observarse en la Tabla 6.2, la arquitectura con menos falsas alarmas emitidas fue la red con alimentación hacia adelante con solo 8 falsas alarmas, seguida del modelo Elman con 10 y SRWNN – MRW con 15. En términos de falsos negativos la arquitectura con menor número fue la FRNN con 23, seguida de la SRWNN y SRWNN – MRW con 30 y 34 respectivamente. Para la obtención de verdaderos positivos, observamos a la arquitectura FRNN obtener el mejor número con 329 ataques detectados seguida de cerca por las arquitecturas SRWNN y SRWNN – MRW con 322 y 318. Finalmente en la detección de patrones normales podemos establecer que las arquitecturas con mejores números fueron la de alimentación hacia adelante y Elman con 268 y 266, seguida del modelo SRWNN – MRW con 261 patrones.

En la Figura 6.5 se muestran las cantidades FP, FN, TP, TN, obtenidas para cada arquitectura.

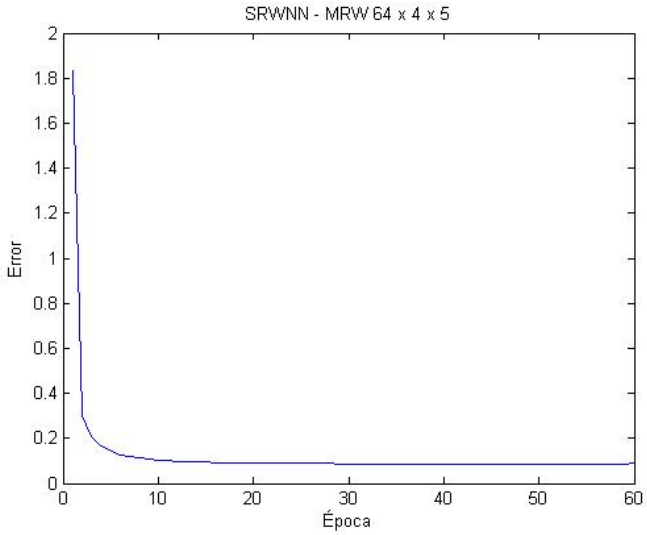

Figura 6. 4: Comportamiento del error por época en red auto recurrente con wavelets - MRW

Tabla 6. 2: Resultados comparativos

Arquitectura	Épocas	MSE	FP	FN	TP	TN
FeedForward 64 x 15 x 15 x5	493	0.0839	8	92	260	268
Elman 64 x 30 x 30 x 5	160	0.0537	10	77	275	266
FRNN 64 x 40 x 5	81	0.0044	20	23	329	256
SRWNN 64 x 256 x 4 x 5	100	0.0589	20	30	322	256
SRWNN – MRW 64 x 4 x5	60	0.0899	15	34	318	261

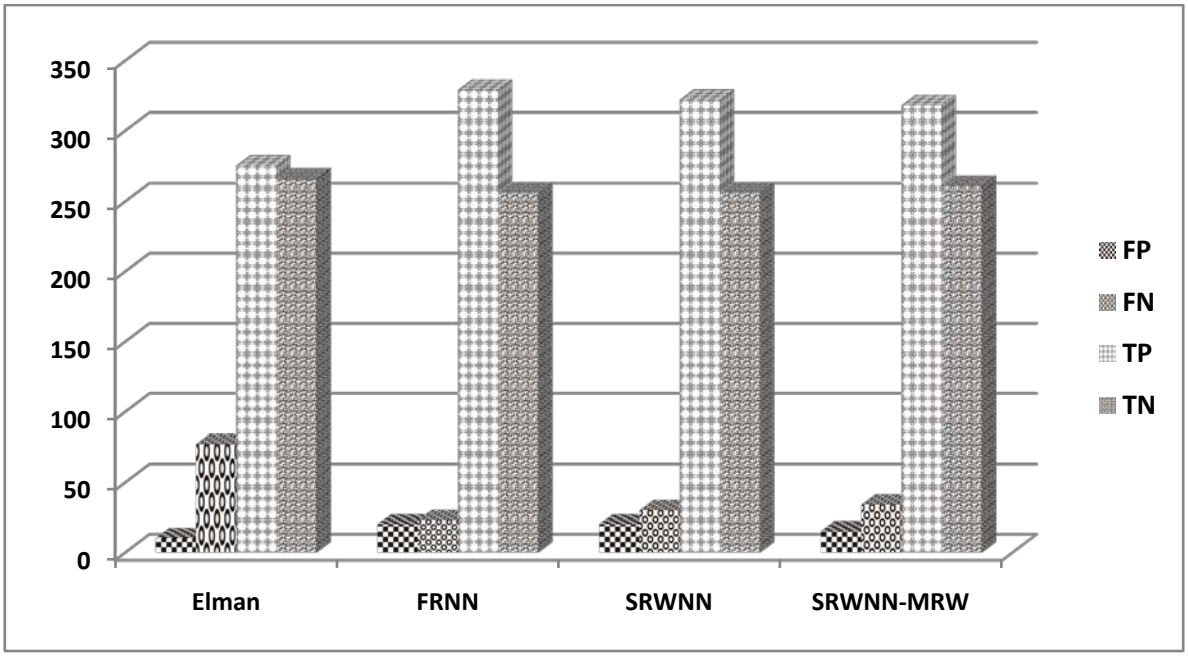

Figura 6. 5: Número de FP, FN, TP, TN

En la Tabla 6.3 se concentran los resultados de las tasas de falsos positivos y falsos negativos así como las medidas estadísticas de desempeño especificación y sensibilidad. Como se puede observar el modelo con menor tasa de FP corresponde a la red con alimentación hacia adelante seguida de los modelos Elman y SRWNN – MRW. Respecto a la tasa FN el mejor resultado lo obtiene la arquitectura FRNN con 6.53%. Los índices de especificación y sensibilidad mejor equilibrados son obtenidos por la arquitectura FRNN, seguida de los modelos SRWNN y SRWNN – MRW.

Tabla 6. 3: Especificación y sensibilidad

Arquitectura	Tasa FP	Tasa FN	Especificación	Sensibilidad
Feed Forward	2.89%	26.13%	97.10%	73.86%
Elman	3.62%	21.87%	96.37%	78.12%
FRNN	7.24%	6.53%	92.75%	93.46%
SRWNN	7.24%	8.52%	92.75%	91.47%
SRWNN – MRW	5.43%	9.65%	94.56%	90.34%

La Figura 6.6 presenta las diferencias entre las medidas de especificación y sensibilidad para cada uno de los modelos estudiados.

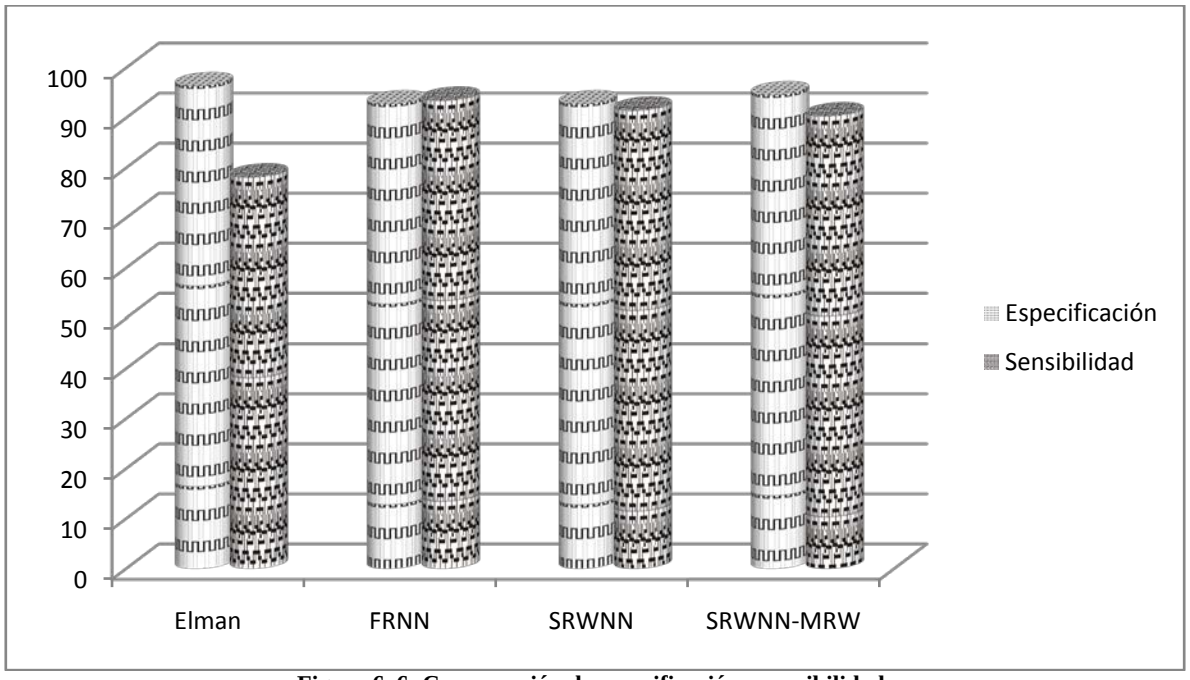

Figura 6. 6: Comparación de especificación y sensibilidad

La Tabla 6.4 muestra los errores por categoría que tuvo cada arquitectura, se puede apreciar que para todas las arquitecturas se obtiene un índice mayor de errores en la categoría de inyección y un menor índice de errores en la categoría de *path*. Si se observa la categoría de tráfico normal, la tabla muestra resultados similares para las arquitecturas recurrente totalmente conectada y la auto recurrente con wavelets, lo que se traduce en un porcentaje similar en la tasa de falsos positivos. De la misma manera se aprecia que el modelo que utiliza la arquitectura tipo Elman es el que produce el menor porcentaje de falsos positivos con la desventaja de tener un porcentaje alto de falsos negativos, es decir un modelo con medidas alta en especificación pero baja en sensibilidad.

Tabla 6. 4: Errores por categoría

Arquitectura	Normal	Inyección	Path	SQL	XSS
FeedForward	8	84	8	7	29
Elman	10	78	14	4	8
FRNN	20	36	1	5	37
SRWNN	20	58	1	3	22
SRWNN - MRW	15	58	3	3	20

La Tabla 6.5 muestra los porcentajes de aciertos por categoría para cada modelo. Al analizar estos valores encontramos que la categoría *path* fue correctamente detectada en todos los modelos, y que las categorías con menor índice de detección fueron SQL e Inyección.

Tabla 6. 5: Porcentajes de detección por categoría

Arquitectura	Normal	Inyección	Path	SQL	XSS
FeedForward	97.10%	3.44%	95.89%	0%	53.96%
Elman	96.37%	10.34%	92.82%	42.85%	87.30%
FRNN	92.75%	58.62%	99.48%	28.57%	41.26%
SRWNN	92.75%	33.33%	99.48%	57.14%	65.07%
SRWNN - MRW	94.56%	33.33%	98.46%	57.14%	68.25%

En la Figura 6.7 se grafican los resultados de porcentajes de aciertos por categoría para cada modelo.

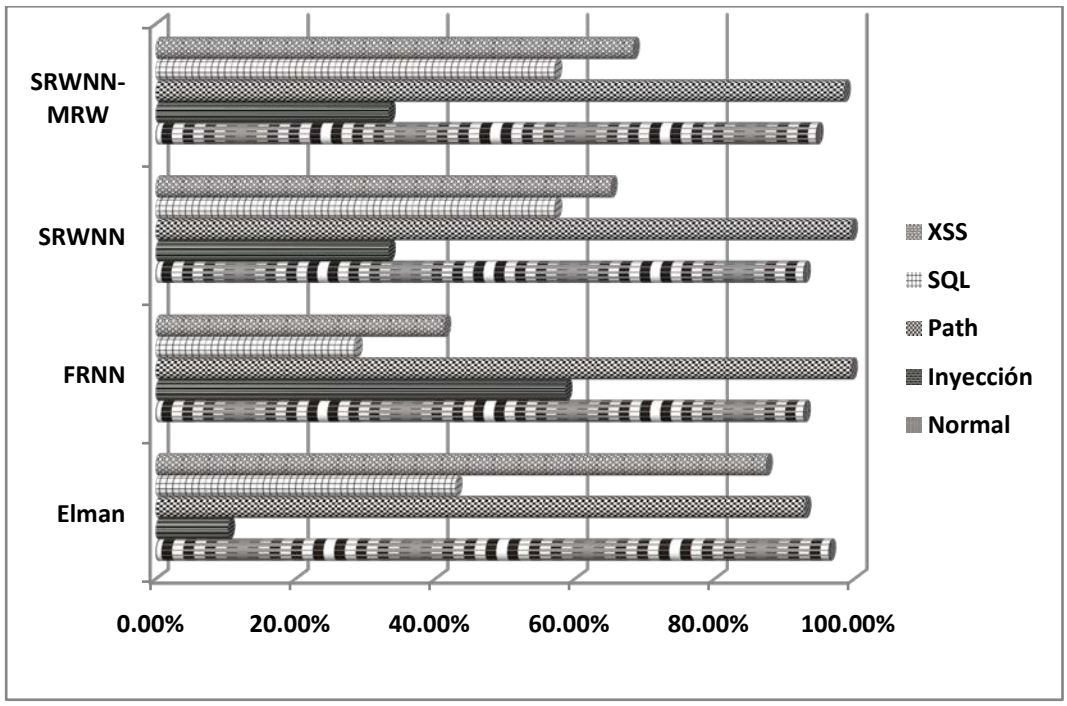

Figura 6. 7: Porcentajes de aciertos por categoría

Como se explicó al principio del capítulo, existen dos cantidades importantes para medir el desempeño de los IDS, llamados porcentajes de clasificación e identificación. El porcentaje de clasificación describe la medida en la que el IDS es capaz de clasificar los patrones de entrada como tráfico normal o intrusivo mientras que el porcentaje de identificación se refiere a la medida en la que el modelo clasifica cada patrón como perteneciente a cada una de las cinco categorías posibles. La Tabla 6.6 resume los porcentajes de clasificación e identificación para cada modelo.

Tabla 6. 6: Porcentajes de clasificación e identificación

	Clasificación	Identificación	Complejidad
FeedForward	84.07%	78.34%	N^2
Elman	86.14%	81.84%	N^2
FRNN	93.15%	84.23%	N^4
SRWNN	92.04%	83.43%	N^2
SRWNN- MRW	92.19%	84.23%	N^2

Tomando en cuenta los resultados de clasificación e identificación, el modelo con mejores valores fue el FRNN, teniendo la misma tasa de identificación que el modelo SRWNN – MRW. En segundo lugar se ubica el modelo SRWNN – MRW con una ligera diferencia de 0.96% en la tasa de clasificación.

Las gráficas ROC (Receiver Operating Characteristics) son una herramienta útil para medir el desempeño de clasificadores, las gráficas ROC comprenden dos dimensiones de las cuales la tasa de verdaderos positivos (TP) se grafica en el eje de las ordenadas y la tasa de falsos positivos (FP) se grafica en el eje de las abscisas. Un grafica ROC muestra la medida entre beneficios (TP) y su costo (FP) [FAW04]. Existen varios puntos dentro del espacio definido en una gráfica ROC, el punto inferior izquierdo con coordenadas (0, 0) describe un modelo en el que no se emiten falsos positivos pero tampoco se obtiene ninguna ganancia (TP). Por otra parte el punto (0, 1) representa el modelo ideal en el que no se presentan falsas alarmas y se identifican de manera positiva todos los ataques o comandos intrusivos. De manera informal se dice que un clasificador es mejor que otro si su ubicación en el espacio ROC se encuentra más al noroeste que la ubicación del segundo.

En la Figura 6.8 se muestra el gráfico ROC donde se ubican los 4 modelos contemplados en este trabajo. Al observar los distintos modelos podemos notar su separación del comportamiento aleatorio denotado por la línea y = x, situándolos en la región triangular superior. Si tomamos en cuenta el eje de falsos positivos ubicamos al modelo Elman con la más baja emisión de falsas alarmas, pero en su coordenada de verdaderos positivos es baja a comparación de los modelos restantes, teniendo un valor de 78.12%. El modelo SRWNN – MRW se ubica a una distancia mayor de Elman sobre el eje

de falsos positivos, indicando una mayor emisión de falsos positivos, sin embargo, obtiene mejores resultados en la detección de verdaderos positivos, 90.34%. Los modelos SRWNN y FRNN parecieran empalmarse sobre el eje de falsos positivos indicando un mismo índice de falsas alarmas, pero el modelo FRNN se ubica un poco más hacia arriba en el eje de verdaderos positivos estableciendo una ventaja de 1.99% en esta tasa.

En la Tabla 6.7 se concentran las tasas de falsos positivos y verdaderos positivos. Idealmente la tasa de falsos positivos debe mantenerse cercana a cero dado que representa la emisión de falsas alarmas, por otra parte la tasa de verdaderos positivos debe ser cercana al 100% pues este valor representa la detección de patrones intrusivos. Como puede observarse la menor tasa de falsos positivos es obtenida por el modelo de alimentación hacia adelante, seguida de Elman y SRWNN – MRW. El mejor resultado obtenido tomando en cuenta ambos indicadores corresponde a la arquitectura SRWNN – MRW con los valores 5.43% y 90.34% pues mantiene el índice de falsos positivos en un valor bajo y la detección de ataques por arriba del 90%.

Tabla 6. 7: Falsos positivos V.S. verdaderos positivos

	Tasa de Falsos Positivos	Tasa de verdaderos positivos
Feed Forward	2.89%	73.86%
Elman	3.62%	78.12%
FRNN	7.24%	93.46%
SRWNN	7.24%	91.47%
SRWNN - MRW	5.43%	90.34%

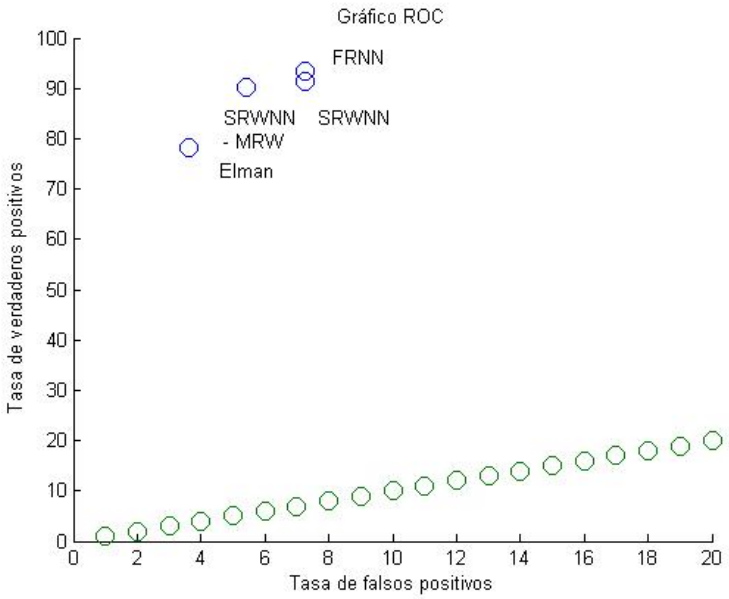

Figura 6. 8. Gráfico ROC

Finalmente la Tabla 6.8 presenta un resumen de las principales características estudiadas para cada arquitectura, como lo son: épocas de entrenamiento necesarias, error obtenido, tasa de falsos positivos y falsos negativos, índice de especificación, índice de sensibilidad, porcentaje de clasificación, porcentaje de identificación y complejidad computacional. A partir de estos resultados podemos determinar que no existe arquitectura alguna que posea todas las medidas de desempeño ideal, por ejemplo, la arquitectura FRNN posee buenas medidas de desempeño pero a un costo computacional alto. La elección de una arquitectura u otra obedece a encontrar un equilibrio entre todas las medidas de desempeño, entonces podemos ubicar a los modelos FRNN, SRWNN y SRWNN-MRW como los más viables en la implementación de IDS con redes neuronales.

Tabla 6. 8. Resumen de características

Arquitectura	Épocas	NMSE	Tasa FP	Tasa FN	Especificación	Sensibilidad	Clasificación	Identificación	Complejidad
Feed Forward	493	7.33e-5	2.89%	26.13%	97.10%	73.86%	84.07%	78.34%	N^2
Elman	160	4.69e-5	3.62%	21.87%	96.37%	78.12%	86.14%	81.84%	N^2
FRNN	81	3.92e-6	7.24%	6.53%	92.75%	93.46%	93.15%	84.23%	N^4
SRWNN	100	5.14e-5	7.24%	8.52%	92.75%	91.47%	92.04%	83.43%	N^2
SRWNN – MRW	60	7.01e-5	5.43%	9.65%	94.56%	90.34%	92.19%	84.23%	N^2

6.8. Discusión

En este capítulo se presentaron los resultados comparativos entre los distintos modelos de redes recurrentes. Para fines comparativos se analizaron distintos indicadores que permiten conocer las características de desempeño de cada uno de los modelos, entre los modelos estudiados sobresalen el recurrente totalmente conectado (FRNN) y la auto recurrente con wavelets (SRWNN - MRW). Como medidas primarias de desempeño se consideraron las tasas de falsos positivos, falsos negativos, verdaderos positivos y verdaderos negativos.

Para cada uno de los modelos se incluyeron distintas tablas y gráficas que muestran las medidas y porcentajes obtenidos en el proceso de simulación. Entre los indicadores más importantes se pone un énfasis especial en las tasas de falsos positivos y falsos negativos

además de los porcentajes de clasificación e identificación para determinar la eficiencia de un modelo sobre otro. Un aspecto complementario a tomar en cuenta es el costo computacional que tiene cada modelo, se puede observar que aunque el modelo recurrente totalmente conectado (FRNN) tiene la mejor tasa de clasificación su costo computacional es superior a todos los demás modelos requiriendo un alto porcentaje de capacidad de almacenamiento y un tiempo de cómputo en cada época de entrenamiento del orden de $O(n^4)$. Por otra parte el modelo auto recurrente con wavelets propuesto (SRWNN – MRW) aproxima con un buen porcentaje al modelo recurrente totalmente conectado sin la necesidad de una alta demanda computacional, $O(n^2)$, haciéndolo un candidato idóneo cuando se tienen recursos limitados o se requiere un menor tiempo para la etapa de aprendizaje.

Para ayudar a la visualización de las características de cada modelo, se ubicó cada arquitectura en un espacio ROC dependiendo a sus tasas de falsos positivos y verdaderos positivos obtenidas. Esto permite conocer la relación costo beneficio entre los distintos modelos estudiados y de la misma manera generar conclusiones sobre el desempeño de cada modelo.