

## تشخیص اعداد و علامات دستنویس فارسی به کمک شبکه‌های عصبی

فریبرز سبحان منش\* - حسین علوی\*\*

چکیده:

در این مقاله، پس از مروری بر مسئله تشخیص الگو و نقش شبکه‌های عصبی در این مورد، از این شبکه‌ها به عنوان یک ابزار محاسباتی قوی در تشخیص اعداد و علامات دستنویس فارسی استفاده می‌کنیم. بدین منظور شبکه‌های چند لایه تغذیه مستقیم را مورد بررسی قرار می‌دهیم و سپس از این ساختار برای رسیدن به مقاصد مورد نظر استفاده می‌کنیم. ابتدا پارامترهای مختلف ساختار فوق از قبیل تعداد نرونهای لایه‌های ورودی، پنهان و خروجی را بهینه‌سازی نموده و سپس به ساده کردن و هرس کردن<sup>۱</sup> شبکه<sup>۱</sup> بهینه می‌پردازیم.

مقدمه:

تشخیص حروف و کلمات تایپ شده و دستنویس به کمک کامپیوتر یا OCR<sup>۲</sup> مدتهاست که ذهن محققین هوش مصنوعی را به خود مشغول ساخته است. در زمینه تشخیص حروف و کلمات لاتین، پیشرفت زیادی حاصل شده و نرم‌افزارهای گوناگون معرفی و به بازار عرضه شده است (برای نمونه به مراجع [۱] تا [۳] رجوع شود). تشخیص حروف و کلمات عربی و فارسی نیز فعالیت‌هایی را در سالهای اخیر به خود اختصاص داده است (مثلاً مراجع [۴] تا [۶]).

\* مربی بخش مهندسی و علوم کامپیوتر - دانشکده مهندسی دانشگاه شیراز

\*\* استادیار دانشکده برق و کامپیوتر - دانشگاه صنعتی اصفهان

روشهای معمول در تشخیص الگوها به طور اعم و نوشته‌ها به طور اخص را می‌توان به دو دسته روشهای ساختاری<sup>۱</sup> و روشهای آماری<sup>۲</sup> تقسیم‌بندی کرد [۷]. در روشهای ساختاری برای تشخیص الگو از خواص هندسی و توپولوژیکی آن نظیر قوسها، حلقه‌ها، زوایا و امتدادهای موجود در شکل الگو استفاده می‌شود. در روشهای آماری به هر الگو یک بردار در یک فضای برداری نسبت داده می‌شود. این فضای برداری را فضای ویژگی<sup>۳</sup> الگو می‌نامند و مؤلفه‌های بردار ویژگی الگو با استفاده از اندازه‌گیری‌های متعددی که روی الگو انجام می‌شود به دست می‌آیند. فضای ویژگی الگو با استفاده از اطلاعاتی که قبلاً در اختیار سیستم تشخیص الگو قرار داده شده، به طبقات و نواحی متعددی تقسیم شده‌است. هدف از تشخیص یک الگوی بخصوص تعیین طبقه یا ناحیه‌ای است که بردار الگو در آن قرار دارد. به این عمل، طبقه‌بندی<sup>۴</sup> و به سیستم تشخیص الگو، طبقه‌بندی‌کننده<sup>۵</sup> می‌گویند. اطلاعات لازم جهت تقسیم‌بندی فضای ویژگی الگو شامل توابع چگالی احتمال شرطی و غیرشرطی و مشترک طبقات و الگوهاست. اما این توابع همواره موجود نیستند و تنها اطلاعاتی که از طبقات و الگوها در دسترس است تعدادی زوج متناظر به صورت طبقه یا الگو است. در روشهای آماری سعی می‌شود با استفاده از این اطلاعات توابع چگالی احتمال لازم تقریب‌زده شود، اما تقریب توابع چگالی شرطی کار مشکلی است؛ خصوصاً اگر ابعاد فضای ویژگی بزرگ باشد.

روش دیگری که به لحاظ سادگی و دقت، توجه زیادی را در سالهای اخیر به خود جلب کرده، استفاده از شبکه‌های عصبی مصنوعی است. هر شبکه عصبی مصنوعی از تعداد زیادی پردازنده ساده تشکیل شده‌است که به دلیل تشابه با شبکه عصبی بیولوژیکی، نرون نامیده می‌شوند. برحسب نوع ساختاری که برای این شبکه‌ها انتخاب می‌شود، هر پردازنده می‌تواند به تعدادی یا به تمام پردازنده‌های شبکه متصل شود. ارتباط بین هر دو پردازنده با یک وزن خاص صورت می‌گیرد، به طوری که  $w_{ij}$  وزن اتصال بین خروجی نرون  $j$ ام و ورودی نرون  $i$ ام است. بسته به اینکه نرون  $j$ ام در تحریک نرون  $i$ ام اثر محرک یا بازدارنده داشته باشد،  $w_{ij}$  مثبت یا منفی خواهد بود و اگر نرون  $i$ ام از نرون  $j$ ام هیچگونه سیگنالی دریافت نکند،  $w_{ij}$  برابر صفر خواهد شد. با توجه به توضیحات فوق مدل مصنوعی یک نرون به صورت نشان داده شده در شکل ۱ ارائه شده است ([۸] و [۹]).

1. Structural

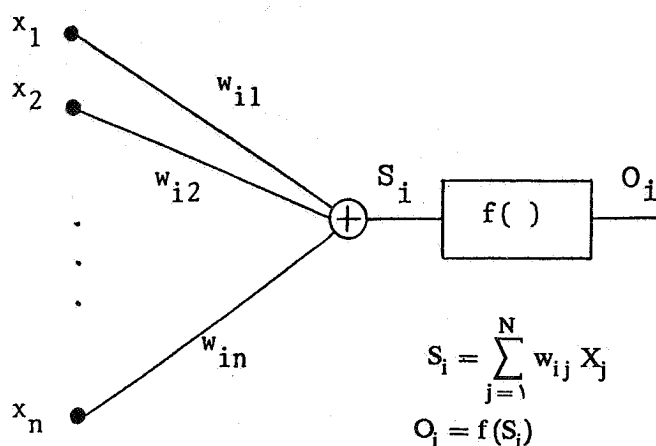
2. Statistical

3. Feature space

4. Classification

5. Classifier

برای تابع  $f$  فرمهای مختلفی را می توان فرض کرد. اگر  $f$  را تابع محدودکننده سخت انتخاب کنیم،



شکل ۱- مدل یک نرون مصنوعی

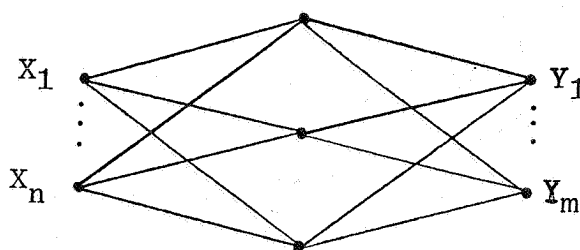
مدل مصنوعی نرون با مدل بیولوژیکی آن کاملاً منطبق می شود، اما از لحاظ محاسباتی استفاده از تابع محدودکننده سخت باعث از دست رفتن اطلاعات مفیدی درباره مجموع وزن دار ورودیها می شود و بنابراین در شبکه های عصبی مصنوعی فرمهای دیگر تابع  $f$  نیز به کار گرفته می شود که معمولترین آنها تابع غیر خطی سیگموئید با معادله  $f(s_i) = \tanh(s_i)$  است.

شبکه های چند لایه تغذیه مستقیم و نقش آنها در تشخیص الگو [۱۰]

یکی از ساختارهای معروف در شبکه های عصبی، شبکه های چند لایه تغذیه مستقیم است (شکل ۲). در این ساختار پردازنده های عصبی در چند لایه مجزا و پشت سرهم قرار داده می شوند و پردازنده های هر لایه فقط مجاز به دریافت سیگنال از پردازنده های لایه ما قبل و ارسال سیگنال به پردازنده های لایه بعد هستند. نرونهای لایه سمت چپ را که با  $x_i$  نشان داده شده اند، نرونهای ورودی

## 1. Hard Limiter

و نرونهای لایه سمت راست را که با  $y_j$  نشان داده شده‌اند، نرونهای خروجی می‌گویند. این نرونها



شکل ۲- یک شبکه چند لایه تغذیه مستقیم

ارتباط شبکه را با دنیای خارج برقرار می‌کنند. نرونهای لایه میانی را که ورودی و خروجی آنها منحصر به نرونهای شبکه می‌شود، نرونهای پنهان می‌نامند. برای تربیت یک شبکه عصبی از تعدادی زوج تربیتی (طبقه مطلوب، الگو) استفاده می‌شود. هدف از تربیت شبکه، اصلاح و فقی و زندهای شبکه به قسمی است که تابع متوسط مربعات خطای شبکه روی مجموعه زوجهای تربیتی می‌نیمد شود. الگوریتم تربیتی شبکه‌های چند لایه تغذیه مستقیم به الگوریتم انتشار خطا به عقب موسوم است که به اختصار به آن اشاره می‌شود. بر طبق این قانون الگوهای تربیتی بکرات با ترتیب تصادفی به ورودی اعمال شده و پس از هر بار اعمال، وزنها تغییر داده می‌شوند. تغییر وزن  $w_{ij}$  به صورت زیر انجام می‌پذیرد:

$$w_{ij}^{\text{new}} = w_{ij}^{\text{old}} + a \delta_i y_j \quad \delta_i = \begin{cases} f'_i(s_i)(d_i - y_i) & \text{نرون } i \text{ در لایه خروجی} \\ f'_i(s_i) \sum_n \delta_n W_{ni} & \text{نرون } i \text{ در لایه پنهان} \end{cases}$$

در این رابطه،  $y_j$  خروجی واقعی نرون  $i$ ام،  $d_i$  و  $y_i$  به ترتیب خروجیهای واقعی و مطلوب نرون  $i$ ام و  $s_i$  مجموع وزن دار ورودیهای نرون  $i$ ام هستند.  $a$  نرخ یادگیری است که سرعت همگرایی وزنها

## 1. Back Propagation (BP)

و پایداری الگوریتم را کنترل می‌کند. هرچه  $a$  بزرگتر باشد سرعت همگرایی و احتمال ناپایداری و واگراشدن وزنها بیشتر می‌شود. معمولاً  $0.01 < a < 1$  انتخاب می‌شود و از  $a = 0.1$  به عنوان حدس اولیه استفاده می‌شود. یکی از مشکلات الگوریتم‌های تربیتی شبکه‌های چند لایه، به تله افتادن وزنها در یک می‌نیمم محلی<sup>۱</sup> و در نتیجه جلوگیری از رسیدن به می‌نیمم قطعی تابع متوسط مربعات خطاست. برای جلوگیری از این امر در الگوریتم BP از یک جمله مومنتوم با ثابت مومنتوم  $\eta$ ، که معمولاً حدود 0.9 اختیار می‌شود، به یکی از دو صورت زیر استفاده می‌شود:

$$\Delta w_{ij}(n) = a \delta_i y_j + \eta \Delta w_{ij}(n-1)$$

$$\Delta w_{ij}(n) = a(1-\eta) \delta_i y_j + \eta \Delta w_{ij}(n-1)$$

در این دو رابطه  $\Delta w_{ij}(n)$ ، تغییر وزن  $w_{ij}$  به ازای الگوی تربیتی فعلی و  $\Delta w_{ij}(n-1)$  تغییر وزن  $w_{ij}$  به ازای الگوی تربیتی قبلی است.

استفاده از شبکه‌های عصبی جهت تشخیص اعداد دست‌نویس فارسی [۱۱]

بعد از معرفی شبکه‌های عصبی و نقش آنها در تشخیص الگو، شبکه‌های چند لایه مستقیم را جهت تشخیص اعداد دست‌نویس فارسی انتخاب می‌کنیم و به بهینه‌سازی پارامترهای مختلف این شبکه از قبیل تعداد نرونهای ورودی، پنهان و خروجی می‌پردازیم. طبق تعریف، شبکه بهینه شبکه‌ای است که با کمترین پیچیدگی، بیشترین دقت را در تشخیص اعداد داشته باشد. اگر تعداد نرونهای لایه پنهان یک شبکه بسیار کم انتخاب شوند، با توجه به تعداد کم درجات آزادی در یادگیری، شبکه قادر به یادگیری نخواهد بود. از طرف دیگر اگر تعداد نرونهای پنهان شبکه خیلی زیاد باشد، شبکه سعی در حفظ و ذخیره کردن الگوهای تربیتی خود خواهد داشت و از قدرت تعمیم<sup>۲</sup> و درون‌یابی<sup>۳</sup> آن کاسته می‌گردد (منظور از تعمیم، توانایی شبکه در تشخیص الگوهای جدیدی است که در مجموعه تربیتی اش نبوده‌اند اما شباهتی با آنها دارند، این پدیده، به تنظیم بر نویز<sup>۴</sup> و یا تربیت

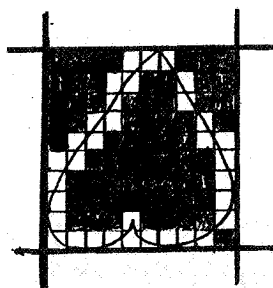
1. Local Minimum

2. Generalization

3. Interpolation

4. Tuning to the Noise

بیش از حد<sup>۱</sup> معروف است). بنابراین برای هر منظور و کاربرد خاص، تعداد بهینه‌ای از نرونهای پنهان وجود دارد که باید به کمک شبیه‌سازی شبکه‌های مختلف و اندازه‌گیری میزان تعمیم آنها تعیین شود. تعداد نرونهای ورودی با نوع کدگذاری<sup>۲</sup> مورد استفاده برای الگوهای ورودی و استخراج ویژگی الگو ارتباط دارد. در این طرح، جهت انتقال الگوهای دستنویس به کامپیوتر از دیجیتالایزر به عنوان سنسور و نرم‌افزار Auto Cad به عنوان محرک<sup>۳</sup> دیجیتالایزر استفاده شد. عمل استخراج ویژگی باید به گونه‌ای انجام شود که بردار ویژگی الگو از اندازه و موقعیت الگو روی دیجیتالایزر مستقل باشد. برای این منظور دو خط افقی بالا و پایین الگو و دو خط عمودی از چپ و راست الگو به سمت الگو حرکت داده می‌شوند تا بر الگو مماس شوند و بدین ترتیب یک پنجره بر الگو محیط‌گردد. سپس گوشه‌چپ و پایین پنجره را به مبدأ مختصات منتقل کرده و مساحت پنجره را به  $M \times N$  مستطیل کوچک تقسیم می‌کنیم و عبور یا عدم عبور الگو از هر مستطیل را به ترتیب با دو عدد باینری ۱ یا ۰ نشان می‌دهیم (شکل ۳). در پایان یک بردار  $M \times N$  بعدی باینری از هر الگو خواهیم داشت که به عنوان بردار ویژگی<sup>۴</sup> آن الگو مورد استفاده قرار می‌گیرد. این نوع کدگذاری را کدگذاری شبکه‌ای می‌نامند. این روش فقط برای دو رقم ۱ و ۰ و علامت منها (-)، که دارای شکل یک بعدی هستند، نتیجه مطلوبی



شکل ۳- کدگذاری شبکه‌ای (عدد ۵)

- 
- |                  |                   |
|------------------|-------------------|
| 1. Over-Training | 2. Coding         |
| 3. Driver        | 4. Feature Vector |

به دست نمی‌دهد. برای رفع این اشکال طول یا عرض پنجره محیط بر این الگوها را افزایش می‌دهیم. نمونه‌ای از کدهای تولیدی توسط این روش برای الگوهای دستنویس ۰ تا ۹ و پنج علامت +، -، ×، ÷ و / (ممیز) در شکل ۴ نشان داده شده است.

با استفاده از روش فوق تعداد ۷۲۰ الگوی دستنویس (۴۸ الگو از هر عدد و علامت) از ۱۵ نفر با ۷ فرمت مختلف ۵×۵، ۶×۶، ۶×۵، ۷×۶، ۷×۷، ۸×۷، ۸×۸ برداشته شد که از این تعداد ۵۲۰ الگو برای تربیت شبکه و ۲۰۰ الگو برای تست شبکه (اندازه‌گیری میزان تعمیم شبکه) در نظر گرفته شد. کدهای ورودی ۵×۵، ۶×۵ به علت فقدان قدرت تفکیک کافی برای دو الگوی دستنویس ۲ و ۴، کدهای یکسانی را نتیجه دادند و بنابراین فرمت‌های مزبور حذف شدند و انتخاب فرمت بهینه ورودی از میان ۵ فرمت دیگر صورت گرفت.

تعیین تعداد بهینه نرونهاى لایه خروجی نیز با نوع کد مورد استفاده برای طبقات خروجی ارتباط دارد. با توجه به تعداد اعداد و علامتهایی که باید مورد تشخیص قرار گیرند. تعداد طبقات خروجی برابر ۱۵ است. برای کدگذاری این طبقات از ۱۵ بردار مقدماتی ۱۵ بعدی استفاده می‌کنیم. در این صورت کد مربوط به طبقه  $i$  ام، یک بردار ۱۵ بعدی است که مولفه  $i$  ام آن ۱ و بقیه مؤلفه‌هایش صفر است (یعنی  $(c_i = (0,0,\dots,0,1,0,\dots,0,0))$ ). دلیل استفاده از این نوع کد، ماکزیم کردن فاصله همینگ<sup>۱</sup> بین کدهای طبقات مختلف است. ثابت می‌شود که تنها در صورت استفاده از این نوع کد، طبقه‌بندی کننده عصبی به سمت طبقه‌بندی کننده بهینه<sup>۲</sup> بیز می‌کند [۸]. در این حالت هر نرون خروجی نماینده یک طبقه خواهد بود و خروجی هر نرون به عنوان تقریبی از امکان<sup>۳</sup> قرار گرفتن الگوی ورودی در آن طبقه تفسیر خواهد شد. اگر از کدهای ۴ بیتی باینری یا ۸ بیتی ASCII استفاده شود، بدلیل فاصله همینگ کوچک بین کدهای طبقات مختلف خطای تشخیص الگوها زیاد می‌شود. با توجه به توضیحات فوق برای پیاده‌سازی کد مربوط به طبقات خروجی روی شبکه عصبی، به ۱۵ نرون در لایه خروجی احتیاج خواهیم داشت.

برای تعیین تعداد بهینه نرونهاى لایه پنهان و فرمت بهینه ورودی از میان ۵ فرمت ۶×۷، ۶×۶، ۷×۷، ۷×۸، ۸×۸، از شبیه‌سازی شبکه‌های مختلف و اندازه‌گیری میزان تعمیم و درونیابی شبکه‌های تربیت شده فوق روی یک مجموعه واحد تست (شامل ۲۰۰ الگو) استفاده می‌کنیم و به

### 1. Hamming Distance

### 2. Bayes

### 3. Likelihood





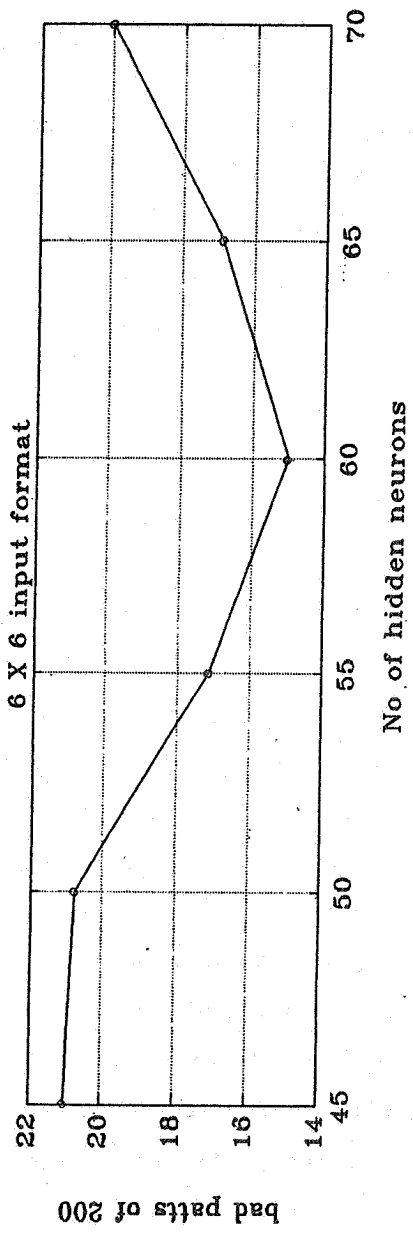
ازای هر کدام از ۵ فرمت ورودی، شبکه‌ای را که دارای کمترین خطا در تشخیص الگوهای مجموعه تست باشد انتخاب می‌کنیم.

#### شبیه‌سازی شبکه‌های مختلف و نتایج بدست‌آمده

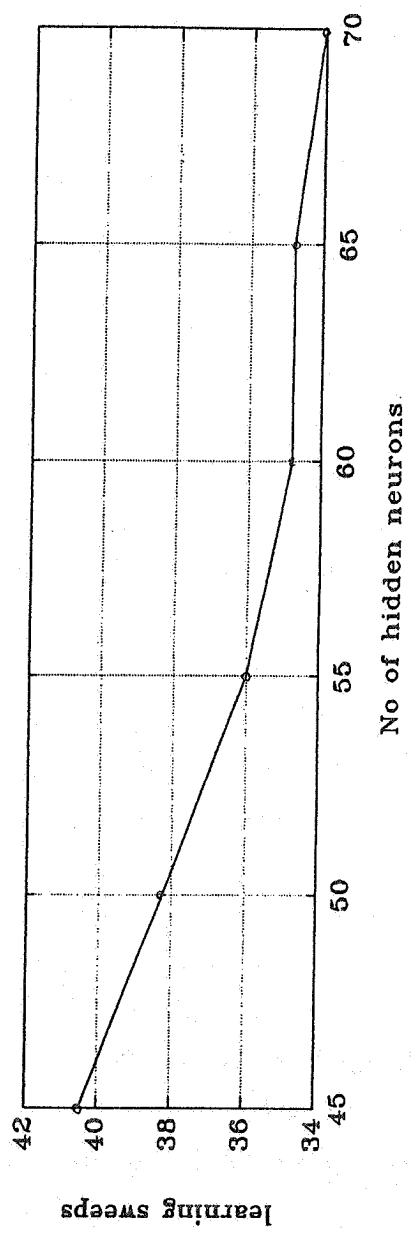
در شبیه‌سازی شبکه‌ها، تابع انتقال نرونها را تابع سیگموئید با معادله  $f(x) = (1 + e^{-x})^{-1}$  در نظر می‌گیریم و با استفاده از الگوریتم تربیتی BP با نرخ یادگیری ۰/۱ و ثابت مومنتوم ۰/۹ و تلورانس تربیتی ۰/۱، به تربیت شبکه‌های مختلف می‌پردازیم. لازم به تذکر است که با انتخاب تلورانس تربیتی ۰/۱، مقادیر قابل قبول خروجی به جای ۱، فاصله ۰/۹ تا ۱ و بجای ۰، فاصله ۰ تا ۰/۱ خواهد بود. با توجه به میل کردن مجانبی تابع سیگموئید به سمت ۱، انتخاب تلورانس تربیتی غیر صفر ضروری است.

به ازای هر یک از ۵ فرمت ورودی مختلف، تعداد نرونهاى پنهان را تغییر داده و به تربیت شبکه‌های حاصل می‌پردازیم. منحنی تغییرات خطای شبکه‌های مختلف در تشخیص ۲۰۰ الگوی مجموعه تست و منحنی تغییرات زمان همگرایی این شبکه‌ها بر حسب تغییرات تعداد نرونهاى لایه پنهان در اشکال ۵ تا ۹ آمده‌اند.

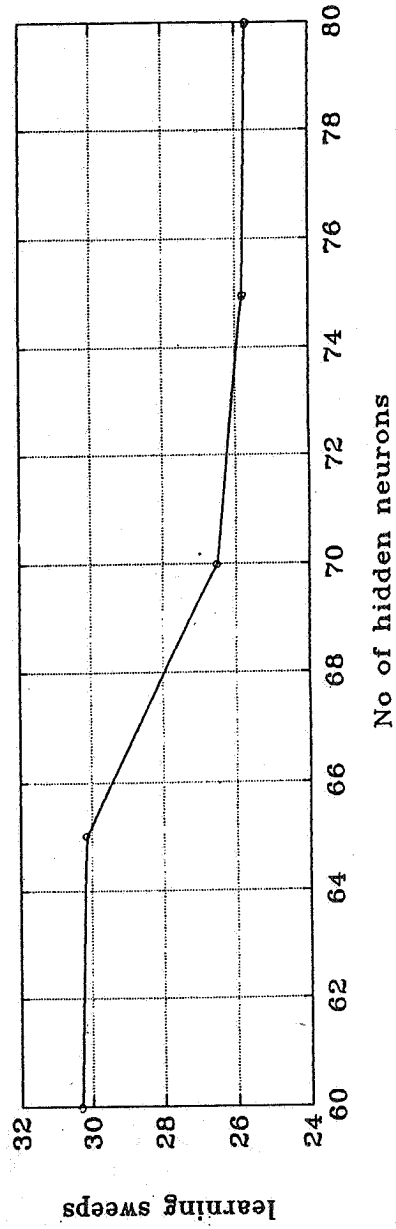
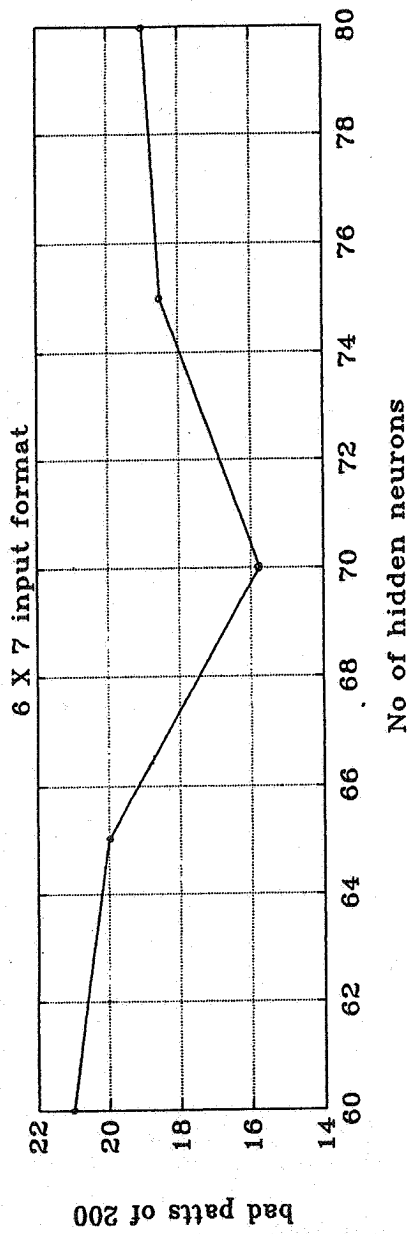
لازم به تذکر است که تربیت هر شبکه با ۷ وزن اولیه مختلف انجام گرفته است و نقاط نشان داده شده روی منحنی متوسط آماری نتایج حاصل از ۷ بار تربیت هستند. نکته قابل توجه دیگر در تربیت شبکه‌ها، لزوم ارائه تصادفی الگوهای تربیتی به شبکه در حال تربیت است. اگر در ارائه الگوهای تربیتی به شبکه، نظم و تربیت خاصی برقرار باشد، توانایی یادگیری شبکه پایین می‌آید و حتی ممکن است به صفر برسد. مثلاً اگر تمام الگوهای مربوط به یک طبقه خاص را به طور متوالی به شبکه بدهیم، آنگاه شبکه در هنگام یادگیری این الگوها هر آنچه را که قبلاً در مورد الگوهای طبقات دیگر یاد گرفته است، از یاد می‌برد. همان‌طور که از شکل‌های ۵ تا ۹ پیداست به ازای هر فرمت ورودی، منحنی خطای شبکه‌ها، به ازای تعداد بهینه‌ای از نرونهاى پنهان می‌نیم می‌شود. هم‌چنین مشاهده می‌کنیم که با افزایش تعداد نرونهاى لایه پنهان، زمان همگرایی شبکه کاهش می‌یابد و این به علت افزایش درجات آزادی شبکه (وزنهاى شبکه) است. مشخصات شبکه‌های انتخاب شده به ازای هر یک از ۵ فرمت مختلف ورودی در جدول ۱ آمده است. این شبکه‌ها متناظر با می‌نیم منحنی تغییرات خطا هستند.



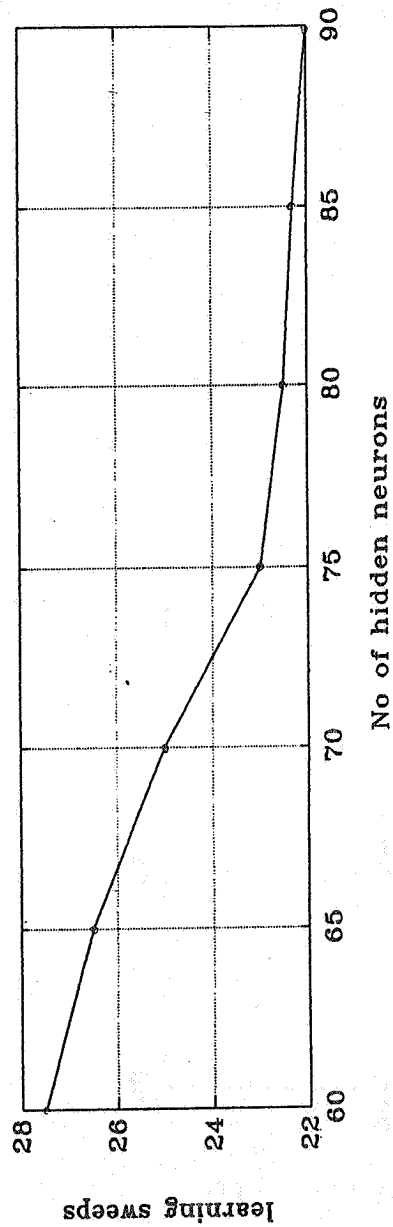
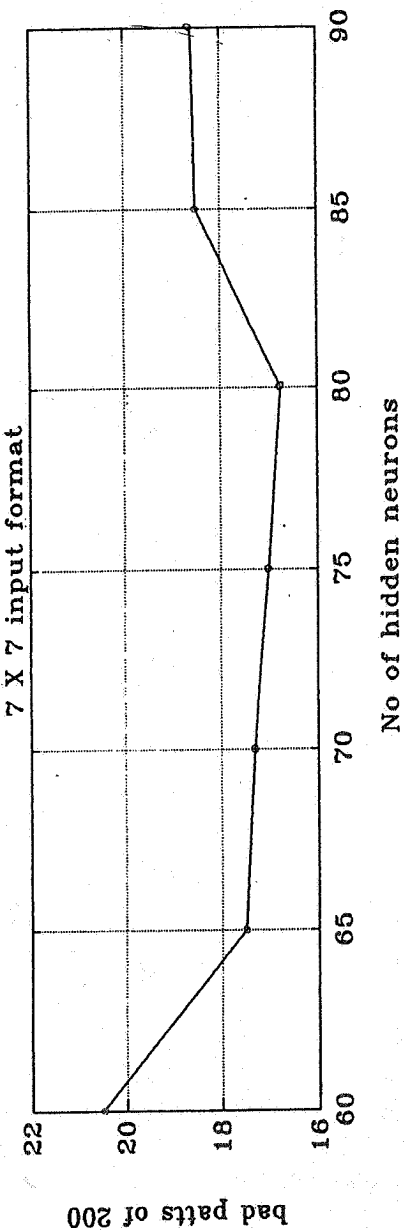
استقلال



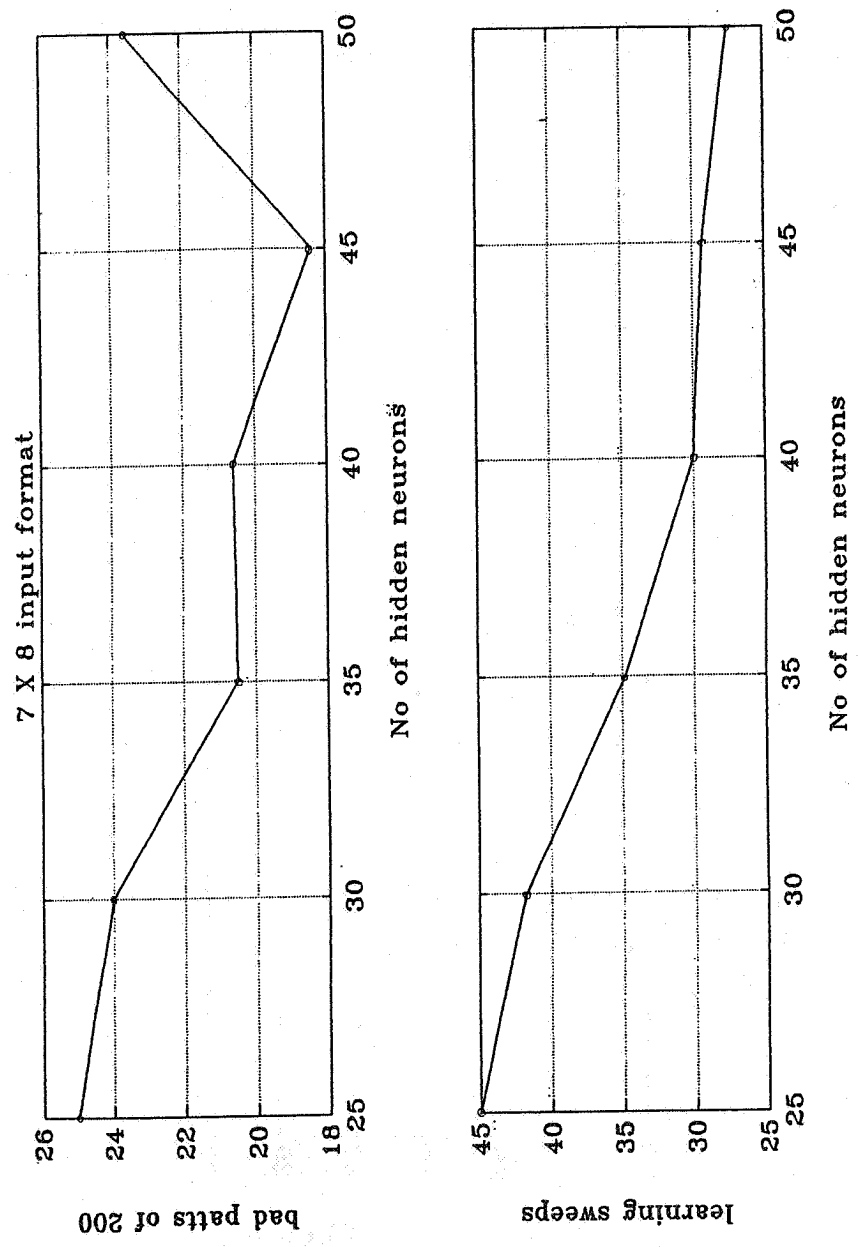
شکل ۵- منحنی تغییرات خطا و زمان همگرایی شبکه با فرمت ورودی ۶×۶، با تغییر تعداد نورونهای پنهان



شکل ۶- منحنی تغییرات خطا و زمان همگرایی شبکه با فرمت ورودی ۶×۷، با تغییر تعداد نورونهای پنهان

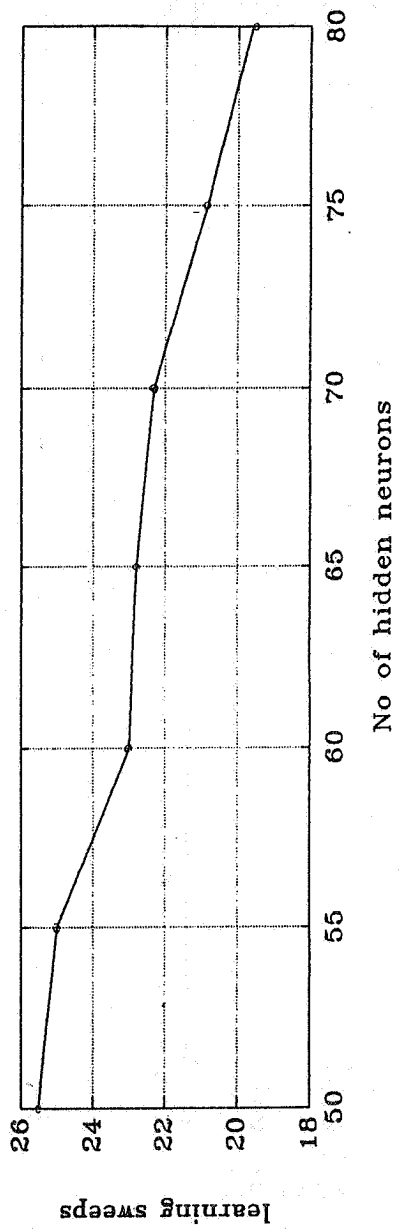
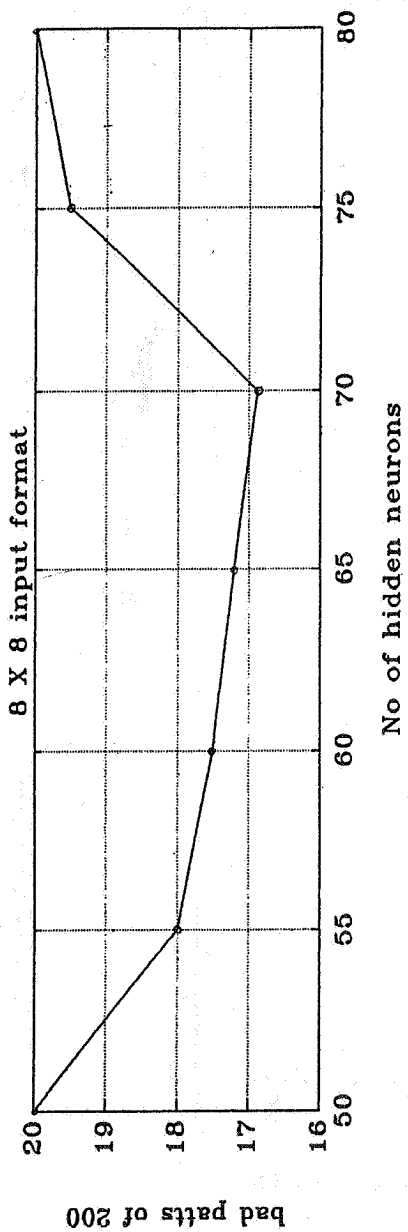


شکل ۷- منحنی تغییرات خطا و زمان همگرایی شبکه با فرمت ورودی ۷×۷، با تغییر تعداد نودهای پنهان



شکل ۸- منحنی تغییرات خطا و زمان همگرایی شبکه با فرمت ورودی ۷×۸، با تغییر تعداد نورونهای پنهان

استقلال



شکل ۹- منحنی تغییرات خطا و زمان همگرایی شبکه با فرمت ورودی ۸×۸ با تغییر تعداد نرونهای پنهان

برای انتخاب فرمت بهینه ورودی، قدرت تعمیم و درونیابی شبکه‌های فوق را در تشخیص الگوهای طبقات مختلف اندازه‌گیری کردیم. بدین منظور یک مجموعه تست شامل ۷۵۰ الگو (از هر طبقه ۵۰ الگوی دستنویس) تهیه شد و درصد خطای شبکه‌ها در تشخیص الگوهای طبقات مختلف به دست آمد. نتایج حاصل از این اندازه‌گیری در جدول ۲ آمده است. همان‌طور که از اعداد مندرج در این جدول پیداست، شبکه با فرمت ورودی ۶×۶ به ازای الگوهای سه طبقه (۳، ۰، -)، شبکه با فرمت ورودی ۶×۷ به ازای الگوی طبقه (-)، شبکه با فرمت ورودی ۷×۷ به ازای الگوی طبقه (۳) و شبکه با فرمت ورودی ۷×۸ به ازای الگوهای طبقات (۳، ۰) خطای زیادی دارند. اما تغییرات خطا به ازای الگوهای طبقات مختلف در شبکه با فرمت ورودی ۸×۸ کم‌تر است، هم‌چنین درصد خطای کل شبکه با فرمت ورودی ۸×۸ نیز از سایر شبکه‌ها کم‌تر است بنابراین فرمت ورودی ۸×۸ را به عنوان فرمت مناسب ورودی انتخاب می‌کنیم.

جدول ۱- شبکه‌های انتخاب شده به ازای فرمتهای ورودی مختلف

فرمت ورودی	لایه پنهان بهینه	تعداد خطا از ۲۰۰	زمان همگرایی
۶×۶	۶۰ نرون	۱۵	۳۴/۸
۶×۷	۷۰ نرون	۱۵/۷۵	۲۶/۵
۷×۷	۸۰ نرون	۱۶/۷۵	۲۲/۵
۷×۸	۴۵ نرون	۱۸/۴۲	۲۹/۲۸
۸×۸	۷۰ نرون	۱۶/۸۷	۲۲/۳

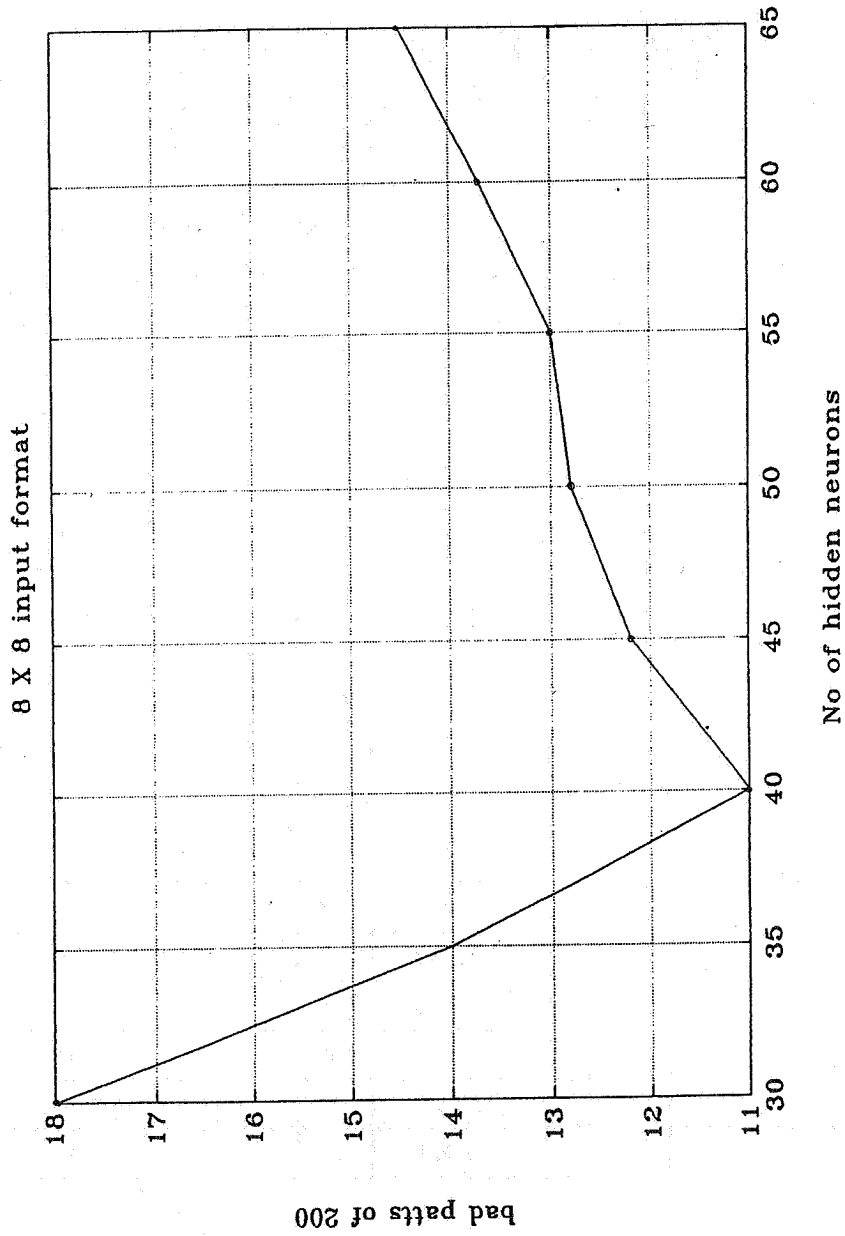
به منظور تعیین تعداد بهینه نرونهای لایه پنهان و هم‌چنین کاهش خطای شبکه بهینه، تعداد الگوهای تربیتی را از ۵۲۰ الگو به ۸۰۰ الگو افزایش می‌دهیم و مجدداً با متغیرگرفتن تعداد نرونهای لایه پنهان، منحنی تغییرات خطای شبکه‌های مختلف را به دست می‌آوریم (شکل ۱۰). همان‌طور که از این شکل پیداست می‌نیم منحنی خطا به ازای ۴۰ نرون پنهان اتفاق می‌افتد و بنابراین تعداد بهینه نرونهای لایه پنهان برابر ۴۰ انتخاب می‌شود. بعد از تعیین تعداد بهینه نرونهای لایه پنهان، دقت شبکه را در تشخیص الگوهای مجموعه تست شامل ۷۵۰ الگو (۱۵ الگو از هر طبقه) اندازه‌گیری کردیم که نتایج حاصل از این اندازه‌گیری بر حسب درصد در جدول ۳ آمده است. همان‌طور که از اعداد مندرج در جدول ۳ پیداست، متوسط میزان دقت شبکه بهینه در تشخیص

جدول ۲- درصد خطای شبکه‌های انتخاب شده روی الگوهای طبقات مختلف

کلاس	فرمت ۶×۶	فرمت ۶×۷	فرمت ۷×۷	فرمت ۷×۸	فرمت ۸×۸
۰	۵۲	۳۰	۲۶	۴۸	۱۶
۱	۱۰	۱۲	۱۰	۸	۱۶
۲	۲۰	۳۰	۲۸	۳۲	۲۰
۳	۴۸	۳۰	۴۲	۳۸	۳۰
۴	۸	۱۰	۱۶	۱۸	۱۶
۵	۸	۱۰	۲۶	۱۶	۱۶
۶	۱۶	۱۴	۱۴	۱۴	۲۴
۷	۰	۲	۱۰	۴	۰
۸	۰	۱۸	۱۴	۱۸	۱۸
۹	۲	۲	۴	۴	۶
+	۱۴	۱۸	۱۴	۱۶	۲۴
-	۵۴	۵۸	۰	۰	۰
×	۲۲	۱۲	۲۲	۱۴	۲۶
÷	۱۶	۲۲	۱۲	۲۸	۱۰
/	۰	۲	۰	۰	۰
کل خطا	%۱۸	%۱۸	%۱۵/۹	%۱۷/۲	%۱۴/۵

اعداد دستنویس فارسی در حدود ۹۵٪ است که نشان‌دهنده توانایی بالای شبکه در تعمیم و تشخیص الگوهای دستنویس فارسی است.





شکل ۱۰- منحنی تغییرات خطای شبکه با فرمت ورودی ۸×۸

جدول ۳- دقت شبکه با فرمت ورودی ۸×۸ و ۴۰ نرون پنهان

متوسط دقت /	/	÷	×	-	+	۹	۸	۷	۶	۵	۴	۳	۲	۱	۰	الگو
۹۴/۸	۱۰۰	۹۸	۹۰	۱۰۰	۹۸	۹۸	۹۰	۹۶	۹۶	۹۲	۹۰	۹۲	۹۲	۹۰	۹۰	%دقت

## ساده کردن و هرس کردن شبکه بهینه

باتوجه به اینکه وزن اتصال بین دو نرون در یک شبکه عصبی معیاری از میزان همکاری و همبستگی بین دو نرون است، به نظر می رسد که حذف وزنه‌های نزدیک صفر تأثیر چندانی بر عملکرد شبکه نگذارد. برای تحقیق در این امر، ابتدا وزنه‌های با قدر مطلق کوچکتر از ۰/۱، ۰/۲، ۰/۳، ... از شبکه بهینه حذف شدند و در هر مرحله خطای شبکه‌های حاصل روی یک مجموعه تست شامل ۶۰۰ الگوی دستنویس اندازه گیری شد. نتایج حاصل از این اندازه گیریها در جدول ۴ آمده است.

جدول ۴- خطای حاصل از حذف وزنه‌های روی ۶۰۰ الگوی تست

خطا با تلورانس ۰/۵	خطا	تعداد وزنه‌های حذف شده	سطح آستانه حذف وزنه‌ها
۶۵	۴۰	۰	۰
۶۵	۴۰	۱۷۷	۰/۱
۶۹	۴۲	۳۶۵	۰/۲
۶۶	۴۲	۵۲۷	۰/۳
۷۱	۴۱	۷۱۵	۰/۴
۷۸	۴۱	۸۹۸	۰/۵
۷۸	۴۵	۱۰۷۲	۰/۶
۹۳	۴۷	۱۲۴۷	۰/۷
۱۱۰	۵۲	۱۳۹۲	۰/۸
۱۲۱	۵۱	۱۵۶۲	۰/۹
۱۳۶	۵۳	۱۶۸۶	۱

ستون دوم این جدول تعداد وزنه‌های حذف شده از شبکه بهینه را نشان می دهد. در ستون چهارم نیز

تعداد تشخیص‌های غلط شبکه با در نظر گرفتن تلورانس ۰/۵ آمده است. در این صورت هنگامی یک الگوی دستنویس به طور صحیح تشخیص داده می‌شود که خروجی نرون مربوط به طبقه آن الگو از ۰/۵ بیشتر و خروجی نرونهای مربوط به طبقات دیگر از ۰/۵ کمتر باشد. در غیر این صورت شبکه در تشخیص الگو دچار خطا خواهد شد. لازم به تذکر است که معیار تشخیص صحیح شبکه در تعلق الگوی  $x$  به طبقه  $i$ ، که تا این مرحله مورد استفاده قرار می‌گرفت ماکزیمم بودن خروجی نرون  $i$ ام در لایه خروجی نسبت به دیگر نرونهای لایه خروجی بود و اعداد مندرج در ستون سوم جدول ۴ نیز بر این اساس به دست آمده‌اند. از اعداد مندرج در این جدول درمی‌یابیم که حذف وزنهای با قدر مطلق کوچکتر از ۰/۵ تأثیری بر عملکرد شبکه نمی‌گذارد (اعداد ستون سوم) اما میزان اعتماد به عملکرد شبکه کاهش می‌یابد (باتوجه به اعداد ستون چهارم)، بنابراین نتیجه می‌گیریم که تعداد ۸۹۸ وزن از کل ۳۲۱۵ وزن شبکه بهینه قابل حذف است.

باتوجه به اینکه حذف تعدادی از وزنهای شبکه، بردار وزنها را از می‌نیم تابع متوسط مربعات خطای شبکه روی مجموعه تربیتی منحرف می‌کند، شبکه‌های حاصل از حذف وزنها مجدداً روی مجموعه تربیتی شامل ۸۰۰ الگو تحت تربیت قرار گرفتند و پس از تربیت مجدد، خطای این شبکه‌ها در تشخیص ۶۰۰ الگوی مجموعه تست اندازه‌گیری شد. نتایج حاصل از این اندازه‌گیری در جدول ۵ آمده است. همان‌طور که از اعداد مندرج در جدول پیداست، برای شبکه‌های با سطح آستانه حذف وزن بیشتر از ۰/۵ تربیت مجدد باعث کاهش خطا گردیده اما برای شبکه‌های با سطح آستانه کمتر از ۰/۵ تربیت مجدد تأثیری نداشته است.

اقدام دیگری که در جهت ساده‌سازی شبکه بهینه انجام گرفت، کوانتیزه کردن وزنهای شبکه بود. بدین صورت که وزنهای شبکه به ترتیب با سه رقم اعشار، ۲ رقم اعشار، ۱ رقم اعشار و بدون رقم اعشار در نظر گرفته شد و خطای شبکه‌های حاصل از کوانتیزه کردن وزنها روی مجموعه تست ۶۰۰ الگویی اندازه‌گیری شد که نتایج حاصل در جدول ۶ آمده است. همان‌طور که از اعداد مندرج در این جدول پیداست، با ۱ رقم اعشار نیز شبکه دارای خطای قابل قبول است که از این نکته می‌توان در پیاده‌سازی سخت‌افزاری شبکه استفاده کرد.

#### اندازه‌گیری قابلیت اعتماد تشخیص

برای جداسازی تشخیص‌های غیر قابل اعتماد شبکه، از آنتروپی خروجی می‌توان بهره جست [۹]. بدین ترتیب که ابتدا اندازه خروجی نرونهای خروجی نسبت به مجموع آنها تراز می‌شود به

قسمی که بتوان خروجی نرونها را به عنوان احتمال وقوع طبقات مختلف ( $P_i$ ) تفسیر کرد. سپس آنتروپی نرونها را به خروجی طبق فرمول  $H = - \sum_{i=1}^{16} P_i \ln(P_i)$  برای هر الگو محاسبه می شود. اگر مقدار آنتروپی از ۰/۷ بیشتر شد تصمیم طبقه بندی کننده عصبی قابل اعتماد نیست. اما اگر آنتروپی کوچکتر از ۰/۷ بود، تصمیم شبکه عصبی قابل قبول است. سطح آستانه فوق به روش سعی و خطا و با مطالعه تعداد زیادی الگوی دستنویس و تصمیم شبکه در برابر آنها تعیین شد که با

جدول ۵- خطای شبکه های دوباره تربیت یافته روی ۶۰۰ الگوی تست

خطا	سطح آستانه حذف وزنها	خطا با تلو رانس ۰/۵
۴۰	۰/۱	۶۵
۴۰	۰/۲	۶۹
۴۰	۰/۳	۶۶
۴۱	۰/۴	۶۸
۳۹	۰/۵	۷۱
۳۷	۰/۶	۷۶
۳۶	۰/۷	۸۰
۳۹	۰/۸	۸۰
۴۰	۰/۹	۸۳
۵۷	۱	۹۰

جدول ۶- خطای شبکه های حاصل از کواتیزه کردن وزنها روی ۶۰۰ الگوی تست

تلورانس ۰/۵	خطا	ارقام اعشار
۶۵	۴۰	۳
۶۶	۴۱	۲
۶۵	۴۳	۱
۱۱۱	۵۳	۰

مقدار به دست آمده از تئوری نیز مطابقت دارد؛ زیرا حداکثر ابهام در تصمیم گیری طبقه بندی کننده

هنگامی اتفاق می افتد که مقادیر دو نرون خروجی نزدیک هم باشند. اگر در این حالت خروجی دیگر نرونها در حدود صفر فرض شود، خروجی تراز شده آن دو نرون در حدود ۰/۵ خواهد بود. در این حالت مقدار آنتروپی برابر خواهد شد با:

$$H = -2 \times 0.5 \ln 0.5 = 0.693$$

### نتیجه گیری

در این مقاله پس از مروری بر مسئله تشخیص الگو و شبکه‌های عصبی مصنوعی، ساختار شبکه‌های چند لایه تغذیه مستقیم انتخابی برای تشخیص اعداد دستنویس فارسی معرفی شد. آنگاه بهینه‌سازی ساختار فوق برای کاربرد مورد نظر انجام گرفت که در نهایت یک شبکه عصبی بهینه با دقت تشخیص ۹۵٪ به دست آمد. سپس ساده کردن شبکه و حذف تعدادی از وزنه‌های آن مورد توجه قرار گرفت. نتایج به دست آمده نشان می‌دهد که می‌توان تعداد قابل توجهی از وزنه‌های شبکه بهینه را حذف کرد بدون اینکه در عملکرد شبکه بهینه تأثیر چندانی داشته باشد. عملکرد شبکه در مقابله با نویز کوانتیزاسیون وزنه‌ها نیز مورد بررسی قرار گرفت. نتایج حاصله حاکی از اثر ناچیز این نویز بر عملکرد سیستم است.

مراجع:

1. Marzoco, F. N., "Computer Recognition of Handwritten First Names", *IEEE Trans. Electron. Computer*, Vol. 14, pp.210-217, Apr. 1965.
2. Suen, C. Y., Berthold, M. & Mori, S., "Automatic Recognition of Handprinted Characters, the State of the Art", *Proc. of IEEE*, Vol. 68, No. 4, pp. 469-487, 1980.
3. Tappert, C. C., Suen, C. Y. & Wakahara, T., "The State of the Art in on-line Handwriting Recognition", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 12, no. 8, pp. 787-804, Aug. 1990.
4. Mahmoud, M. Y., El-Hamalaway, M. A. & Flamy, A. A., "A Statistical Approach For Arabic Character Recognition", *Proc. 9th National Con. & Exhib.*, pp. 1. 301-1. 307, Riyadh, Saudi Arabia, Oct. 1987.
5. Parhami, B. & Taraghi, M., "Automatic Recognition Of Printed FarsiText", *Pattern Recognition*, Vol. 14, No. 14, pp. 395-403, 1981.
6. Tolba, M. F., Wahab, S. M. & Salem, A., "A Recognition Algorithm For Arabic Printed Characters", *Proc. ISTED Int. Symp on Appliad Information*, Grindeweld, Switzerland, pp. 128-131, Feb. 1987.
7. Duda, R. O. & Hart, P. E., *Pattern Classification and Scene Analysis*, John Wiley, New York, 1973.
8. Nielsen, H. R., *Neuro Computing*, Addison Wseley, New York, 1990.

9. Beal, R. & Jackson, T., *Neural Computing, an Introduction*, Adam Hilger, Bristol, 1990.

10. Pao, Y. H., *Adaptive Pattern Recognition and Neural Networks*, Addison Wesley, New York, 1989.

۱۱. سبحان منش، فربرز "تشخیص اعداد دستنویس فارسی به کمک شبکه‌های عصبی"، پایان‌نامه کارشناسی ارشد، دانشکده برق و کامپیوتر دانشگاه صنعتی اصفهان، بهمن‌ماه ۱۳۷۰.