سیستمی بر پایه دانش جهت تشخیص عیوب موتور تراکتور MF-285 با استفاده از شبکه عصبی مصنوعی (۲۵۷)

محمد طهماسبی[۱] ، سید ناصر علوی نائینی[۲]

چکیده

استفاده از فناوری و رایانه برای کمک به تصمیم‌گیری در زمینه‌های تخصصی از مقولاتی است که امروزه در تصمیم‌گیری های مدیران صنایع و مهندسین نگهداری و تعمیرات بسیار مورد توجه قرار گرفته است. از آنجا که در بحث نگهداری و تعمیرات اعمال و تصمیمات انسانی نقش بسزایی دارد و در این میان شرایط محیطی و روحی میتواند بر این تصمیمات تاثیرگذار باشد، بنابراین افزایش خطا و در نتیجه آن افزایش هزینه‌های ناشی از تعمیرات، تعویض قطعات و نگهداری امری اجتناب ناپذیر است. لذا استفاده از تکنیک‌هایی که بتواند این خطاها را کاهش داده یا از بین ببرد ضروری می‌باشد. در این مقاله سعی شده است سیستم خبره‌ای جهت تشخیص اتوماتیک خ ابی‌ها و عیوب موتور تراکتور MF-285 به عنوان ابزاری از نگهداری و تعمیرات بررسی، طراحی و پیاده‌سازی شود. از دانش خبرگان، کاتالوگ‌های مربوطه و کتاب‌های تعمیرات و نگهداری جهت ایجاد پایگاه دانش و از شبکه عصبی مصنوعی، به عنوان موتور استنتاجی سیستم خبره استفاده شده است. در مقایسه نتایج حاصل از سیستم خبره و خبرگان، مشاهده می‌شود که پاسخ‌های سیستم ۹۳/۳ درصد با واقعیت خرابی‌ها، آنچه که در عمل اتفاق افتاده است، تطابق دارد، در حالی که پاسخ افراد خبره ۸۴/۹۹ درصد با واقعیت خرابی‌ها تطابق دارد. همچنین مشاهده می‌شود که جواب‌های سیستم خبره ۹۱/۶۹ درصد با دانش خبرگان تطابق دارد.

کلیدواژه: نگهداری و تعمیرات، سیستم خبره، خرابی‌های موتور تراکتور، شبکه عصبی مصنوعی

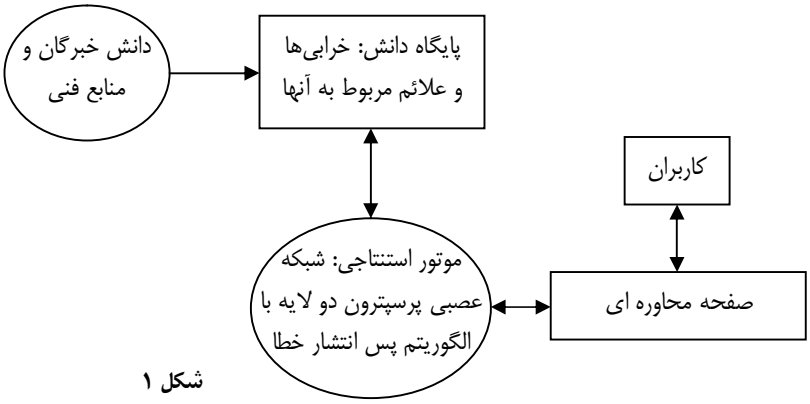
۱- دانشجوی کارشناسی ارشد، مکانیک ماشین‌های کشاورزی، دانشگاه شهید باهنر کرمان، پست الکترونیک: mtahmasbi2000@yahoo.com
۲- استادیار گروه ماشین‌های کشاورزی، دانشگاه شهید باهنر کرمان

مقدمه

از روزی که بشر اولین ابزار خود را ساخت، بحث نگهداری و تعمیرات را در مورد ابزار خود مد نظر داشت. امروزه با پیشرفت تکنولوژی، حرکت به سوی اتوماسیون و استفاده ماشین به جای انسان، سرعت روز افزونی یافته است. لذا اهمیت نگهداری و تعمیر این ماشین‌آلات نیز بیشتر گردیده است. نگهداری و تعمیرات به مجموعه برنامه‌ها و اقدامات به منظور نگه داشتن تجهیزات درسطح قابل قبول از نظر کارکردی (نگهداری) و یا باز گرداندن تجهیزات و دستگاه‌های معیوب به چرخه استفاده و بهره‌برداری می‌باشد و نتیجه مورد انتظار از این اقدامات ایجاد آمادگی، حفظ قابلیت کارکردی، تداوم و استمرار کارکردی تجهیزات برای شرایط تعریف شده خواهد بود. در این ارتباط طیف گسترده‌ای از مسائل مهندسی از جمله مسائل مهندسی نگهداری و تعمیرات را می توان به کمک سیستم‌های خبره طراحی و مدل سازی نمود [۱،۴،۵،۶،۱۱].

سیستم‌های خبره، برنامه‌هایی هستند که رفتار یک انسان متخصص در یک زمینه بخصوص را تقلید می کنند. این نرم افزار هوشمند از دانشی که مهندس دانش در آن ذخیره می کند جهت اعلام عقیده در یک موضوع خاص استفاده می کند[۷]. شکل ۱ قسمت های مختلف سیستم خبره را نشان می دهد.

مهمترین قسمت‌های هر سیستم خبره پایگاه دانش[۱] ، و موتور استنتاج[۲] آن است. پایگاه دانش مخزنی از حقایق و قواعد دستوری، تجربی و خودآموز است [۸]. در اینجا تجربه و دانش خبرگان (تعمیرکاران تراکتور)، کتاب وکاتولوگ‌های تعمیرات و نگهداری و اطلاعات کاربران تراکتور در شناخت عیوب موتور، تشکیل دهنده این پایگاه می باشند[۳]. موتور استنتاج سیستم، نرم افزاری است که شناخت (دانش) را جایگزین می کند و دانش جدید را از پایگاه دانش استنتاج می کند[۷]. در این قسمت سیستم از شبکه عصبی پرسپترون چندلایه با الگوریتم پس انتشار خطا استفاده شده است.



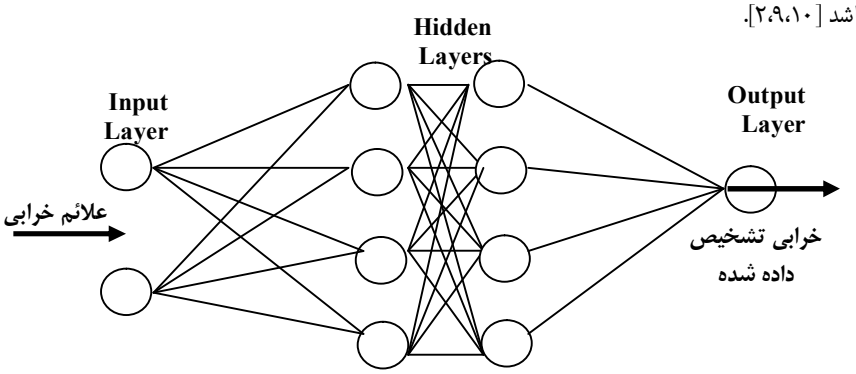
شکل ۱

شبکه‌های عصبی مصنوعی الهام گرفته از عملکرد و نحوه کار شبکه‌های عصبی بیولوژیکی می باشند. با پردازش روی داده‌های تجربی، دانش یا قانون نهفته در ورای داده‌ها به ساختار شبکه منتقل می‌شود که به آن یادگیری می گویند. اصولاً توانایی یادگیری مهمترین ویژگی یک سیستم هوشمند است. سیستمی که بتواند یاد بگیرد منعطف‌تر است و ساده تر برنامه‌ریزی می شود. شبکه های عصبی از یک سری لایه‌هایی شامل اجزای ساده پردازشگری بنام نرون تشکیل شده‌اند که به صورت موازی با هم عمل می کنند. هر لایه ورودی به یک یا تعداد بیشتری لایه میانی[۳] مرتبط است و لایه‌های میانی نیز به لایه خروجی مرتبط می شوند، جایی که جواب شبکه نقش خروجی یک سیستم را ایفا می کند [۲،۹].

یکی از انواع شبکه‌های عصبی ، پرسپترون[۴] می‌باشد و به صورت پرسپترون تک لایه و چند لایه موجود می‌باشد. پرسپترون تک لایه تنها می‌تواند مسائل مجزای خطی را دسته بندی کند و برای مسائل پیچیده تر لازم است که از تعداد بیشتری لایه استفاده گردد. شکل ۲ یک شبکه پرسپترون چند لایه را نشان می‌دهد. این شبکه ، یک شبکه کاملاً به هم مرتبط می باشد چرا که هر

1 - Knowledge Base
2 - Inference Engine
3 - Hidden Layers
4 - Perceptron

نرون در یک لایه به تمامی نرون‌های لایه بعدی مرتبط می‌باشد. اگر بعضی از این ارتباطات وجود نداشته باشد شبکه، یک شبکه مرتبط ناقص است. زمانیکه گفته می‌شود شبکه از n لایه تشکیل شده است، تنها لایه‌های میانی و خارجی به شمار می‌آید و لایه ورودی شمارش نمی‌شود چرا که نرون‌های این لایه محاسبه‌ای را انجام نمی‌دهند. بنابراین شبکه تک لایه شبکه‌ای با تنها یک لایه خارجی می‌باشد [۲،۹،۱۰].



شکل شماره ۲

الگوریتم یادگیری پس انتشار خطا[1] (BP) از نوع یادگیری با ناظر است. در یادگیری با ناظر ، هنگامی که ورودی به شبکه اعمال می شود جواب شبکه با جواب هدفی که برای شبکه تعیین شده است مقایسه می‌شود و سپس خطای یادگیری محاسبه شده و از آن برای تنظیم پارامترهای شبکه استفاده می‌شود، به گونه‌ای که اگر دفعه بعد به شبکه همان ورودی اعمال شود خروجی شبکه به جواب هدف نزدیکتر گردد [۲،۱۰].

این مقاله بر اساس تحقیق انجام شده بر روی خرابی‌های موتور تراکتور MF-285 فرآیند بررسی، طراحی و اجرای نرم افزاری بر پایه دانش، جهت تشخیص عیوب و خ ابی‌های موتور تراکتور با استفاده از شبکه عصبی پرسپترون را ارائه می نماید. هدف کلی از این مدل سازی طراحی و ساخت سیستمی است که به کمک آن شناسایی خرابی‌ها و گزارش آن به صورت اتوماتیک صورت گیرد. از دیگر اهداف، کاهش هزینه تعمیرات با استفاده از تعمیرات پیشگیرانه و همچنین کاهش زمان از کار افتادگی وسیله می باشد.

مواد و روش‌ها

۱- خرابی‌های موتور و علائم مربوط به آن (ایجاد پایگاه دانش)

برای تشخیص خرابی‌های به وجود آمده در قسمت موتور نیاز به یک سری اطلاعات اولیه یا علائمی است که با ملاحظه این علائم بتوان وقوع یک عیب یا خرابی را تشخیص داد، این علائم باید به صورتی باشد که کاربر تراکتور به راحتی و با دانستن اطلاعات کمی از موتور بتواند آنها را تشخیص دهد. در این راستا پرسشنامه‌هایی جهت بررسی علائم خرابی و عیوب مرتبط با این علائم تهیه و توسط خبرگان و کاربران تکمیل گردید. همچنین خ ابی‌های موتور و علائم آنها ، در چندین تعمیرگاه تراکتور از نزدیک مورد تحقیق و بررسی قرار گرفت. با بررسی منابع فنی و مشاوره با کارشناسان فنی وتعمیرکاران خبره و کاربران تراکتور مجموعه علائم خرابی یا عیوبی که با استفاده از پرسشنامه‌ها جمع‌آوری گردید و عموما" در حین استفاده از تراکتور ممکن است کاربر با آنها رو به‌رو شود، شناسایی شد، که در جدول شماره ۱ ارتباط بین علائم و خرابی‌های موجود مشاهده می شود:

1 - Back Propagation

جدول شماره ۱ – ارتباط بین علائم خرابی و خرابی یا عیب موجود

ردیف	علامت خرابی	خرابی و عیب موجود
۱	موتور روشن نمی شود	۱– سوخت در باک موجود نیست ۲– وجود هوا در لوله‌های فشار ضعیف سوخت رسانی ۳–گرفتگی لوله‌های سوخت رسانی ۴– پمپ اولیه سوخت کار نمی کند ۵– کثیفی و گرفتگی فیلتر سوخت ۶– المنت الکتریکی کار نمی کند
۲	موتور روشن شده ولی دارای دور غیر یکنواخت است	۱– اشکال در سیستم سوخت رسانی ۲– فاصله زیاد بین اسبک‌ها و ته ساق سوپاپ ۳– سوختن سوپاپ ۴– سوختن واشر سر سیلندر
.	.	.
.	.	.
.	.	.

همانطور که از جدول مشخص است برای هر خرابی یا عیب فقط به تعداد محدودی علامت نیاز است. گاهی اوقات حتی وقوع یک علامت هم برای تشخیص خرابی کافی است. گاهی هم وجود چند علامت احتمال نوع خاصی از عیب یا خرابی را تقویت می کند بدون اینکه قطعیتی در کار باشد ، با این حساب لازم است رابطه بین علائم و عیوب ممکن بررسی شود. اندیشه موجود آن است که نرم افزار با توجه به شرایط خاص تعریف شده (توسط کاربر) از طریق صفحه کاربران، که همان بیان علائم خرابی است، به جستجو می‌پردازد و چنانچه خرابی و عیبی مناسب شرایط یافت شود، به ترتیب اولویت و بر حسب درصد احتمال رخ دادن فهرستی رتبه بندی شده به عنوان پاسخ به کاربر ارائه خواهد داد. مزیت این کار آن است که عیب‌یابی مرحله به مرحله صورت می‌گیرد و در وقت و هزینه صرفه‌جویی خواهد شد.

جداول۱ و ۲ چگونگی شکل گیری پایگاه دانش نرم افزار را مشخص می‌نماید.

جدول شماره ۲ – روش کد گذاری در بانک اطلاعاتی (پایگاه دانش) نرم افزار

مجموعه کد	جزئیات کد	مورد کاربرد
A	۱	موتور روشن نمی شود
B	۲	موتور به سختی روشن می شود
C	۳	موتور روشن شده ولی دارای دور غیر یکنواخت است
.	.	.
.	.	.
.	.	.

مثال : برای بیان خرابی نظیر : وجود هوا در لوله‌های فشار ضعیف سوخت رسانی، می‌توان نوشت:

A(1)-B(1)-C(1)-D(1)-E(0)-F(0)-G(0)-H(0)-I(1)-J(0)-K(0)-L(0)-M(0)-N(0)-O(0)-P(0)-Q(0)-R(0)-S(0)-T(0)-U(0)

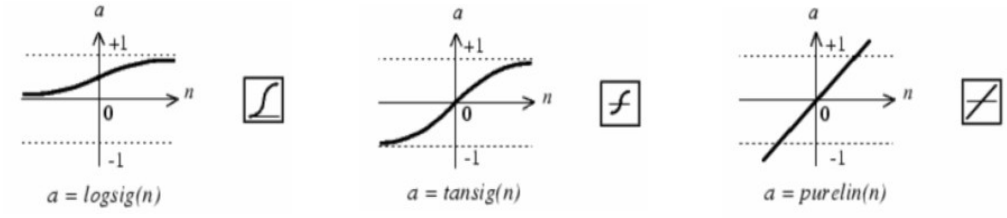
چنانچه فرض کنیم تمام احتمال‌های موجود برای تشخیص عیب و خرابی واقعی باشند، در این صورت تعداد کل خرابی‌هایی که شبکه (موتور استنتاجی) قادر به تولید نام آنهاست بالغ خواهد بود بر :

$N_A \cdot N_B \cdot N_C \cdot N_D \cdot N_E \cdot N_F \cdot N_G \cdot N_H \cdot N_I \cdot N_J \cdot N_K \cdot N_L \cdot N_M \cdot N_N \cdot N_O \cdot N_P \cdot N_Q \cdot N_R \cdot N_S \cdot N_T \cdot N_U$

که در آن منظور از N_A تعداد زیر خرابی‌های گروه A می باشد.

۲- ایجاد موتور استنتاجی نرم افزار خبره با استفاده از شبکه پرسپترون دولایه با الگوریتم BP

در شبکه چند لایه از توابع محرکه log-sigmoid، tan-sigmoid ، purelin استفاده می شود که در زیر (شکل شماره ۳) نمودار این توابع مشاهده می شود:



شکل شماره ۳

اگر در لایه آخر شبکه‌های چند لایه از تابع سیگموئید استفاده شود همانگونه که از شکل ۳ پیداست خروجی‌های شبکه بین محدوده کوچکی قرار می گیرند، اما اگر از تابع محرک خطی استفاده شود خروجی‌های شبکه هر مقداری را می‌تواند اختیار کند. اجرای الگوریتم BP را می‌توان به صورت مراحل زیر تقسیم نمود :

۱- وزن‌ها مقدار اولیه تصادفی می‌گیرند.

۲- هنگامی که شرط توقف میسر نبود مراحل بعدی (۳ تا ۷) انجام می‌گیرد.

۳- هر نرون ورودی یک سیگنال ورودی دریافت می کند و این سیگنال به همه واحدهای لایه‌های بالاتر (لایه‌های میانی) پخش می شود. بردارهای ورودی u و بردارهای خروجی y تعیین می شود.

۴- هر نرون میانی همه وزن‌های سیگنال‌های ورودی را با هم جمع می کند و با تابع محرکه مربوطه‌اش سیگنال خروجی را محاسبه می کند و این سیگنال را به همه واحدهای بالاتر (نرون‌های خروجی) می‌فرستد.

۵- هر نرون خارجی سیگنال‌های ورودی را با هم جمع می‌زند و توسط تابع محرکه سیگنال خروجی را محاسبه می کند $y = f\left(w^t u\right)$.

۶- هر نرون خارجی یک جمله هدفی را دریافت می‌کند که به جمله ورودی آموزشی مرتبط است. حدود خطایش را محاسبه می‌کند و وزن‌ها و جمله بایاس را محاسبه و تصحیح کرده و این مقدار خطا را به واحدهای پایین تر ارجاع می‌دهد. ($w_{j,i} = w_{j,i}(\text{old}) + \Delta w_{j,i}$ و $\Delta w_{j,i} = \eta \delta_j y_i$ در این معادله $w_{j,i}$ وزن نرون i ام لایه میانی به نرون j ام و y_i خروجی نرون i ام و η پارامتر نرخ یادگیری و δ_j گرادیان است).

۷- هر نرون میانی مقادیر خطایش را با هم جمع می‌کند و وزن‌ها و جمله بایاس تصحیح و محاسبه می‌گردد. (طبق رابطه $w_{j,i} = w_{j,i}(\text{old}) + \Delta w_{j,i}$).

۸- هر نرون خارجی و میانی جمله بایاس و وزن‌هایش تصحیح شده است.

۹- معیار توقف آزمایش می شود :

۱)میانگین مربعات خطا در هر سیکل کمتر از مقدار از پیش تعیین شده باشد.

۲) گرادیان خطا از یک مقدار از پیش تعیین شده‌ای کوچکتر گردد.

در اینجا ذکر این نکته الزامی است که روش محاسبه گرادیان برای لایه‌های خارجی و میانی متفاوت است.

پس از بررسی چگونگی شکل‌گیری پایگاه دانش و موتور استنتاجی سیستم، مهمترین کاری که باید انجام شود، ایجاد پایگاه دانش و موتور استنتاجی (شبکه پرسپترون) در محیط نرم افزار MATLAB 7.1 به صورت کد نویسی می باشد. در مدل شبکه عصبی مورد نظر علائم خرابی و عیوب که در پایگاه دانش نرم افزار موجود است، ورودی‌های شبکه و خرابی و عیوب مرتبط به این

علائم، خروجی‌های شبکه را تشکیل می دهند. اما نکته مهم در راستای داشتن واحد استنتاجی قوی انتخاب تابع محرک آموزشی شبکه و به صفر رساندن خطا در خروجی است.

۳- پیاده سازی سیستم خبره به کمک نرم افزار MATLAB 7.1

در این قسمت علائم و خرابی‌های مرتبط که در پایگاه دانش کد بندی شده‌اند، به صورت ماتریس‌های ورودی و خروجی کد نویسی شده و به موتور استنتاجی که به صورت زیر ایجاد و آموزش داده می شود، مرتبط می گردد.

تابع newff، یک شبکه پیشخور (پرسپترون چندلایه با الگوریتم BP) را ایجاد می کند که شامل ۴ ورودی می باشد. اولین ورودی R است که شامل دو ماتریس min وmax مقادیر برای هر عنصر R از بردار ورودی p است. دومین ورودی آرایه‌ای شامل سایزهای هر لایه است. سومین ورودی آرایه‌ای است شامل توابع محرکه‌ای که در در هر لایه استفاده می شود. آخرین ورودی ، شامل نام تابع آموزش مورد استفاده است.

net = newff (minmax(p),[سایز هر لایه],{توابع محرکه در هر لایه},{تابع آموزشی(

برای دادن اطلاعات به دلیل وسعت ماتریس‌های ورودی از برنامه Excel استفاده شده و وجود یا عدم وجود عاملی را با ۰ یا ۱ نشان داده می‌شود، بدین صورت که اگر علائمی به خرابی مورد نظر مرتبط باشد ۱ می شود و در غیر اینصورت عدد ۰ بدان تعلق می گیرد، سپس ماتریس‌های مورد نظر در نرم افزار مطلب خوانده می‌شود.

نتایج و بحث

جهت داشتن موتور استنتاجی قوی در نرم افزار خبره شبکه‌هایی با تعداد لایه‌های مختلف، تعداد نرون‌های مختلف و توابع تحریک مختلف مورد بررسی قرار داده شد که نتایج در جدول شماره ۳ مشاهده می‌شود. در این آزمایشات از تابع آموزش traingdm بدلیل سرعت آن نسبت به توابع دیگر استفاده شده است. در ضمن برای تعیین تابع محرک و تعداد نرون‌ها به روش سعی و خطا عمل شده و هر حالتی را که خطای کمتری دارد انتخاب می‌شود که در اینجا تابع محرک purelin و تعداد نرون ۱ در خروجی و تابع محرک tansig و تعداد نرون ۵۰ در لایه میانی مورد استفاده قرار گرفته است. در نتیجه اعضای شبکه به شکل زیر می‌باشند :

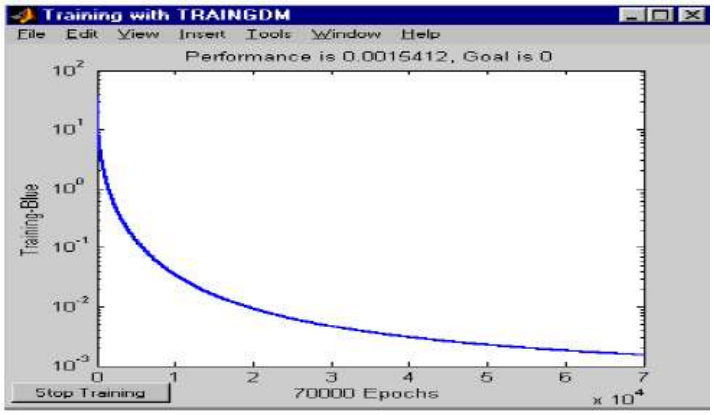
$$net = newff(minmax(p),[50,1],\{'tansig','purelin'\},'traingdm')$$

ضریب مومنتم و نرخ یادگیری نیز با توجه به کاهش خطا تعیین شده و تعداد دفعات آموزش ۷۰۰۰۰ دور انتخاب شده است، لازم به ذکر است که هر چه تعداد دفعات آموزش بیشتر باشد رفتار شبکه دقیقتر خواهد بود ولی سرعت پاسخگویی کمتر است.

جدول شماره ۳

خطا	توابع محرک	تعداد نرون
./۰۴۵۰	satlins-satlins	۳۰-۱۰
./۰۱۲۰	logsig-logsig	۳۵-۱۰
./۰۱۳۰	logsig-tansig	۳۵-۱
./۰۱۸۰	logsig-purelin	۴۰-۱
./۰۱۹۰	purelin –purelin	۴۰-۱
./۰۲۰۰	purelin-tansig	۴۰-۱
./۰۱۸۰	tansig-tansig	۴۰-۱
./۰۰۵۴	satlins-purelin	۴۵-۱
./۰۰۲۸	tansig-purelin	۴۵-۱
./۰۰۱۵	tansig-purelin	۵۰-۱
./۰۰۲۲	tansig-purelin	۵۵-۱

در اینجا با توجه به خطای ناچیزی که شبکه دارد (شکل ۴) ، شبکه (موتور استنتاج) به تمامی عیوب پاسخ مناسب داده است و در صد خطا صفر می‌باشد که این نشانگر آموزش صحیح شبکه و در نتیجه داشتن موتور استنتاجی قوی می‌باشد.



شکل ٤- نمودار مربوط به خطای شبکه

پس از آموزش کامل شبکه (موتور استنتاجی)، نرم افزار خبره آماده پاسخگویی و تبادل اطلاعات با کاربران از طریق صفحه محاوره ای سیستم می باشد. بنابراین به آزمایش و ارزیابی نرم افزار پرداخته شد.

جهت آزمایش سیستم، پرسشنامه‌ای با ۱۵ سوال که در بر گیرنده علائم خرابی بود، به ۴ خبره ارائه و جواب‌های هر یک در مورد سوال مطرح شده، بررسی گردید. همین سوالات (علائم خرابی) نیز در صفحه محاوره ای سیستم خبره ایجاد شده (البته به صورت کدهایی که در بانک اطلاعاتی نرم افزار تعریف شده است) وارد گردید و جواب‌های نرم افزار خبره(تشخیص عیوب در ازای دادن علامت خرابی) نیز مورد بررسی قرار گرفت. نتایج حاصل از نرم افزار خبره، تشخیص عیوب در ازای دادن ورودی (علائم خرابی)، در مقایسه با خبرگان (تعمیرکاران) که در مقابل ۱۵ سوال مرتبط (۱۵ علامت خرابی) پاسخ‌های تخصصی خود را بیان کرده بودند، با آنچه در واقعیت اتفاق می افتد (منظور مشاهده همان خرابی‌ها در عمل است) در جدول شماره ۴ مشاهده می گردد.

جدول شماره ٤ – مقایسه نظرات ٤ فرد خبره تعمیرکار با روجی سیستم خبره

شماره سوال	پاسخ خبره ۱	پاسخ خبره ۲	پاسخ خبره ۳	پاسخ خبره ۴	پاسخ سیستم
۱	۱	۱	۱	۱	۱
۲	۱	۱	۱	۱	۱
۳	۱	۰	۰	۱	۱
۴	۱	۱	۱	۰	۱
۵	۰	۱	۱	۱	۱
۶	۱	۱	۱	۱	۰
۷	۱	۱	۱	۱	۱
۸	۱	۱	۱	۱	۱
۹	۰	۱	۱	۱	۱
۱۰	۱	۱	۱	۰	۱
۱۱	۱	۱	۰	۱	۱
۱۲	۱	۱	۰	۱	۱
۱۳	۱	۱	۱	۱	۱
۱۴	۱	۱	۱	۰	۱
۱۵	۱	۱	۱	۱	۱

همانطور که از جدول شماره ۴ مشاهده می شود پاسخ‌های صحیح با عدد ۱ و پاسخ‌های غلط با عدد ۰ مشخص شده است (واقعیت بر این اساس است که تمامی جواب‌ها باید ۱ باشد). در یک تحلیل آماری ، نتایج زیر بدست می آید:

جدول شماره ۵ – مقایسه پاسخ های نرم افزار خبره با خبرگان و در صد احتمال پاسخگویی صحیح

	تعداد پاسخ صحیح	تعداد پاسخ غلط	درصد احتمال پاسخگوی صحیح
نرم افزار خبره			
	۱۴	۱	۹۳/۳
خبره ۱			
	۱۳	۲	۸۶/۶
خبره ۲			
	۱۴	۱	۹۳/۳
خبره ۳			
	۱۲	۳	۸۰
خبره ۴			
	۱۲	۳	۸۰
متوسط درصد احتمال پاسخگویی صحیح خبرگان = ۸۴/۹۹			

– به احتمال ۹۳/۳ پاسخ‌های نرم افزار خبره با واقعیت خرابی رخ داده تطابق دارند (جدول شماره ۵).
– به احتمال ۸۴/۹۹ پاسخ‌های افراد خبره (تعمیرکاران) با واقعیت تطابق دارند (جدول شماره ۵).

– به احتمال ۹۱/۶۹ پاسخ‌های نرم افزار خبره و افراد خبره با هم تطابق دارند.

حال با توجه به جدول شماره ۵ که به بررسی درصد احتمال پاسخگویی صحیح نرم‌افزار و خبرگان در تشخیص صحیح عیوب موتور تراکتور می‌پردازد، می‌توان به ارزیابی سیستم طراحی شده پرداخت. با توجه به نتایج حاصله پاسخ‌های نرم افزار خبره در مقایسه با افراد خبره، تطابق بیشتری با واقعیت خرابی‌ها دارد و با اطمینان بیشتری عیوب و خ ابی‌ها را مشخص می‌سازد. بدین ترتیب با توجه به عملکرد مناسب سیستم خبره ایجاد شده در عیب یابی موتر می‌توان بدون اینکه نیازی به روش‌های مشکل عیب‌یابی مانند باز کردن قطعات موتور و ... باشد، به عیب مورد نظر در موتور تراکتور پی برد.

نتیجه گیری

به کار گیری سیستم خبره (سیستمی بر پایه دانش)، جهت تشخیص عیوب موتور تراکور باعث افزایش دقت در تشخیص خرابی‌ها و در نتیجه آن کاهش زمان تعمیرات و همچنین کاهش هزینه‌های تعمیرات و نگهداری خواهد شد. از آنجا که در یک سیستم مکانیزاسیون تراکتور نقش عمده‌ای را ایفا می کند، کاهش زمان خرابی و کاهش هزینه‌های تعمیراتی ناشی از خطاهای انسانی می تواند بهره‌وری و سود بیشتری را در سیستم در بر داشته باشد. نتایج ارزیابی و مقایسه پاسخ‌های نرم افزار خبره با پاسخ-های خبرگان و متخصصان، مبین این نکته است که نرم افزار پاسخ‌های با درصد اطمینان بالاتری (نزدیک تر به واقعیت) را در تشخیص عیوب موتور به وجود می آورد.

در انتها پیشنهاد می شود که علائم خرابی و خرابی‌های مرتبط، در دیگر قسمت‌های تراکتور نظیر گیربکس، اکسل جلو وعقب، دیفرانسیل و ... نیز بررسی شود به صورتی که بتوان یک نرم افزار جامع جهت عیب یابی و تعمیر تراکتور ارائه نمود. در نتیجه این کار، شناسایی عیوب تجهیزات بدون حضور متخصص امکان پذیر می گردد.

تشکر و قدردانی

این پژوهش با حمایت مالی معاونت پژوهشی و فناوری دانشگاه شهید باهنر کرمان و انجمن پژوهشگران جوان این دانشگاه انجام شده است.

مراجع و منابع مورد استفاده

۱- ریاحی، م. طراحی، تجزیه و تحلیل سیستم مکانیزه مهندسی نگاهداشت پیشگیرانه، سایت انجمن مهندسی تعمیرات و نگهداری ایران.

۲- منهاج، باقر. ۱۳۸۷، شبکه های عصبی مصنوعی، تهران، دانشگاه صنعتی امیر کبیر.

۳- می، اد. ۱۳۸۱، موتورهای دیزل، امینی، مجید. تهران، انتشارات ک نو.

Support vector machine in machine condition 4- Achmad Widodo, Bo-Suk Yang. 2007. fault diagnosis. Journal of Mechanical Systems and Signal monitoring and Processing 21 2560–2574.

5- D. Bansal, D. J. Evans, B. Jones.2004. A real-time predictive maintenance system for machines systems. International Journal of Machine Tools & Manufacture 44 759–766.

6- E. Gilabert, A.Arnaiz.2006.Intelligent automation systems for predictive maintenance: A case study. Robotics and Computer-Integrated Manufacturing 22 543–549.

7- M.M.Gupta and J.Qi. 1991. Theory of T-norms and fuzzy inference methods, Fuzzy sets and Systems. 40, 431-450.

8- D.W rolston. Principles of Artificial Intelligence and Expert Systems Development. Pages 1-11.

The exploitation of neural networks in .9- P.J.Shayler, M. Goodman, T. Ma. 1999 Journal of Artificial Intelligence, 13: 147-.automotive engine management systems 157.

10- A. Z. Al-Garni, A. Jamal, A. M. Ahmad.2006.Neural network-based failure rate prediction for De Havilland Dash-8 tires. Journal of Engineering Applications of Artificial Intelligence 19 681–691.

11- Jian-Da Wu , Peng-Hsin Chiang. 2007. An expert system for fault diagnosis in internal combustion engines using probability neural network. Journal of Expert System with Applications.

12- Matlab Neural Network, Version 7.1, mat work.