یک الگوی کاوش راهبری وب با استفاده از گراف پیمایش مسیر ، جهت بهبود ساختار گراف و پیش بینی کاربران

غزاله فرخ پور [۱]، سیده فاطمه آقامیری [۲]ساسان حسینعلیزاده [۳]

[۱]دانشکده کامپیوتر دانشگاه آزاد اسلامی قزوین، ghazale.farokhpour@gmail.com
[۲]دانشکده کامپیوتر دانشگاه آزاد قزوین ، Faghamiri@yahoo.com
[۳]دانشکده کامپیوتر دانشگاه آزاد قزوین ، sasan.alizadeh@gmail.com

چکیده

کاوش وب ، فرایند استخراج و کشف دانش مفید از اطلاعات حجیم در وب است. الگوهای پیمایش وب سنتی، برای مدیریت محتوای وب سایت ها موثر نیستند؛ به همین خاطر در اینجا از الگوهای راهبری وب، که الگوهای سراسر گشت و گذار (TSPs) نامیده می شود، استفاده می نماییم. در ابتدا با یک گراف پیمایش مسیر، ساختار گراف وزن دار را ایجاد نموده و سپس با استفاده از ماتریس مجاورت و حل آن از طریق الگوریتم دیکسترا کوتاهترین مسیر و کمترین هزینه را به دست خواهیم آورد. در ادامه مسیرهای راهبری کاربران را با توجه به هدفشان محاسبه و پیش بینی می نماییم که این کار نیاز به اطلاعاتی در مورد راهبری کاربران دارد. در ادامه با جمع آوری مجموعه داده ای از کاربران وب سایت یک دانشگاه و قرار دادن کاربران در خوشه ها با توجه به مسیرهای راهبری، پیشگویی مسیر حرکت کاربران در وب را با استفاده از یک مدل هوشمند شبکه عصبی به دست آورده و با روش های دیگر مقایسه می نماییم که این روش می تواند به درک رساتر از اهداف کاربران کمک نماید و همچنین این روش انعطاف پذیری لازم برای کشف نقاط جدید جهت وزن دهی به گره ها را دارد که غلبه بر مشکلات گیر افتادن در بهینه محلی و باقی ماندن در حالت پایدار می باشد. و دقت پیش بینی با توجه به تعداد جلسات کاربران در این روش بهبود پیدا می کند . از جمله اهداف الگوهای راهبری کاوش وب سایت ها پیش بینی و درک رفتار کاربران است. اپراتورهای وب سایت ها می توانند از درخواست الگوهای راهبری وب سایت به منظور تجزیه و تحلیل و پیش بینی انگیزه کاربر به منظور ارائه توصیه های بهتر و خدمات شخصی برای مشتریان خود استفاده نمایند. همچنین در این پژوهش روش وزن دار کردن گراف را برای پیدا کردن کوتاهترین مسیر با کمترین هزینه (وزن) برای بهبود ساختار گراف و صرفه جویی در زمان و پیش بینی راهبری کاربر ارائه می نماییم.

واژه‌های کلیدی

الگوی راهبری کاوش وب، گراف وزن دار، الگوریتم TSP ، پیش بینی کاربران، الگوریتم ABC-ANN

۱ – مقدمه

روش های وب کاوی بر اساس آن که چه نوع داده ای را مورد کاوش قرار می دهند، به سه دسته تقسیم می شوند:

کاوش محتوای وب : فرآیند استخراج اطلاعات مفید از محتوای مستندات وب است که محتوای یک سند وب متناظر با مفاهیمی است که آن سند در صدد انتقال آن به کاربران است. کاوش ساختار وب : فرآیند استخراج اطلاعات ساختاری از وب است که وب را می توان به صورت گرافی که گره های آن اسناد و یال های آن پیوندهای بین اسناد است، بازنمایی نمود. کاوش استفاده از وب : استخراج الگوهای کاربردی مناسب از داده های ثانویه که خود بر اساس تعامل با کاربران هنگام گشت زنی در وب به منظور درک بهتر نیازهای کاربران مشخص می گردد. باید توجه داشت که مرز مشخصی میان سه گروه وب کاوی وجود ندارد و می توان از ترکیب تکنیک های فوق برای حاصل شدن نتایج بهتر نیز استفاده نمود. روش های متفاوتی به منظور تقابل با مسئله پیشگویی در طول سالیان گذشته پیشنهاد شده است، از جمله این روش ها، روش k-nearest neighbor است که به اختصار kNN نامیده می شود [۱]، شبکه عصبی مصنوعی، استنباط فازی support vector machines که SVM نامیده می شود، قوانین انجمنی یا همان ARM و مدل مارکوف از جمله روشهای مطرح در این حوزه می باشند.

در این مقاله ابتدا تحقیقات گذشته در زمینه راهبری کاوش وب را بررسی نموده و سپس الگوی سراسر گشت و گذار را معرفی می نماییم، در ادامه ساختار گراف را در جهت بهبود یافتن کوتاهترین مسیر در کمترین زمان برای مسیر سراسر گشت و گذار ارائه نموده و روش خود را با دو روش TSP و اپریوری مقایسه می نماییم. هدف اصلی از این کار یافتن کوتاهترین مسیر و کمترین هزینه (وزن) می باشد که رسیدن کاربران به هدف و ساختار گراف را بهبود می بخشد و حجم حافظه مصرفی را نسبت به روش TSP کاهش می دهد.

مزیت اصلی این روش نسبت به روش TSP ، ساختار وزن دار گراف جهت یافتن کوتاهترین مسیر با کمترین هزینه و کاهش حجم پایگاه داده است. در بخش دوم روش پیشنهادی پیش بینی کاربران را با استفاده از الگوریتم پیشنهادی ABC-ANN مبتنی بر یک روش هوشمند شبکه عصبی با خوشه بندی و آموزش گره ها ارائه خواهیم نمود و در انتها نیز نتایج حاصل از الگوریتم جدید خود را بر روی مجموعه داده مورد نظر اعمال خواهیم نمود.

۲ – تحقیقات مرتبط

رفتار راهبری وب به درک اطلاعاتی از تقاضای کاربران آنلاین کمک می نماید که می تواند در پالایش نقشه وب سایت ، سیستم های شخصی

سازی شده مبتنی بر وب، سفارشی سازی و بهبود دقت جستجو و دقت پیش بینی در کاربردهای آنلاین هوشمند استفاده گردد.

مطالعات متعدد پیرامون رفتار راهبری بازدید کنندگان وب سایت وجود دارد که بسیاری از آنها توسط تکنیک های انجام کاوش الگوهای دسترسی به وب مانند بهبود راندمان دسترسی به صفحات وب توسط سیستم وب سایت انطباقی، سازماندهی مجدد یک وب سایت به صورت پویا، شناسایی هدف بازدید کنندگان ، تقویت عملکرد جستجوهای وب و پیش بینی الگوهای رفتار کاربر در سیستم های وب سایت پویا انجام می گردد. (آرایا،سیلواب و وبر ۲۰۰۴، چن و همکاران ۱۹۹۸، چو و کیمب ۲۰۰۴، کازینکو و آدامسکی۲۰۰۷، لی و شیو ۲۰۰۴، تسنگ و لین ۲۰۰۶).

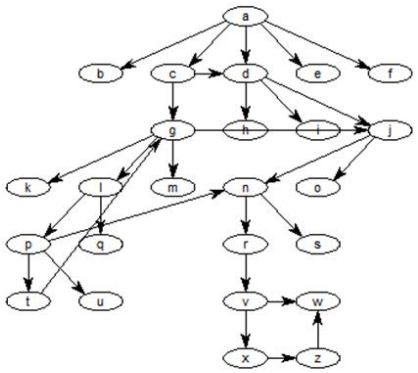
از روش های ساختاری پیمایش وب که تاکنون معرفی شده است مدل های مبتنی بر گراف، مدل های مارکف و پیمایش موضوعی وب را می توان نام برد.

تلاش های تحقیقاتی به منظور کشف الگوهای دسترسی وب سایت در سه بخش اصلی قوانین انجمنی، الگوهای پی در پی و خوشه بندی متمرکز شده است. عمومی ترین الگوریتم برای یافتن قوانین انجمنی، الگوریتم اپریوری است که این الگوریتم در سال۱۹۹۶ توسط چیانج ابداع شد و از مهمترین یافته ها در تاریخ استخراج قوانین وابستگی می باشد.در ادامه ساختارداده درخت WAP (درخت الگوی دسترسی به وب سایت) برای کاوش کارآمد الگوهای دسترسی به وب ابداع شد. (پی و همکاران ۲۰۰۰)

الگوهای راهبری کاوش وب به این معنی می باشد که کاربران با چه توالی و ترتیبی صفحات مختلف وب را بازدید می کنند و در پیش بینی مسیر کاربران، توالی بازدید را در صفحات پیش بینی می نماییم تا محتوای مورد نظر را سریع تر در اختیار کاربر قرار داده و همچنین صفحات را بهینه نماییم. در تمامی سه حوزه کاوش وب می توان از الگوریتم های پیش گویانه جهت حل مسائل استفاده نمود. پیش بینی وب یک مسئله طبقه بندی است که در آن سعی می گردد که مجموعه ای از صفحات وب که کاربر ممکن هست بخواهد در آینده از آن دیدن نماید، پیشگویی نماییم. مسئله پیشگویی وب (WPP) در زمینه های مختلف کاربرد دارد، همانند موتورهای جستجو، سیستم های کش گذاری، سیستمهای پیشنهاد دهنده و نیز کاربردهای وایرلس. علاوه بر کاهش ترافیک شبکه و زمان جستجو برای کاربر، مزیت دیگر این مدل "شخصی سازی " است، که کاربران می توانند بر حسب علایق و سلایقشان طبقه بندی شوند.در پیشگویی وب، دو چالش پیش پردازش و پیشگویی وجود دارد. چالش اول که پیش پردازش است به معنی دستکاری مقدار زیادی داده است که نمی توان یکجا در حافظه کامپیوتر قرار داد و شامل انتخاب اندازه پنجره بهینه، مشخص کردن جلسات، جستجو و استخراج دانش مورد نظر است. پیشگویی چالش دیگری است که شامل زمان آموزش/پیشگویی طولانی، دقت پایین پیشگویی، و محدودیتهای حافظه می باشد.

۳ – روش پیشنهادی

الگوی راهبری وب ، یک الگوی دسترسی به وب سایت است که یک مسیر را از طریق یک یا چند صفحه وب استخراج می نماید. در مراجعه کاربر به وب سایت به مجموعه ای از صفحات بازدید شده به همراه توالی بازدید از صفحات ، جلسه کاربر اطلاق می گردد. در واقع باید بدانیم یک بازدید کننده در چه زمانی از چه صفحاتی بازدید و از چه مسیرهایی عبور نموده است. در این بخش، تجزیه و تحلیل الگوریتم پیمایش گراف برای کاوش الگوهای سراسر گشت و گذار (TSP) انجام می شود. الگوریتم گشت و گذار با انتخاب مناسب، با شروع از لبه ها و پیمایش مکرر مسیر کار خود را شروع می نماید که پیمایش گراف به صورت DFS است. در ادامه روش وزن دار کردن گراف را برای پیدا کردن کوتاهترین مسیر باکمترین هزینه (وزن) را برای بهبود ساختار گراف و صرفه جویی در زمان کاربر ارائه می نماییم.



شکل۱: مثال۱ از ساختار گراف

جدول ۱: مجموعه داده از ۳۰ جلسه مرورگر

Session ID	Web browsing session	Session ID	Web browsing session
S001	(a, b)	S016	(c, g, j)
S002	(a, c, d)	S017	(c, g, l, q)
S003	(a, c, d, j)	S018	(c, g, l, p)
S004	(a, c, g, j, n, s)	S019	(c, g, l, p, u)
S005	(a, c, g, k)	S020	(d, j, n, s)
S006	(a, c, g, l)	S021	(g, j, n, s)
S007	(a, c, g, l, p, t, g, l)	S022	(g, l, p, n, s)
S008	(a, c, g, m)	S023	(j, n, r)
S009	(a, d, h)	S024	(l, p, n, s)
S010	(a, d, i)	S025	(l, p, t, g)
S011	(a, d, j, o)	S026	(t, g, l)
S012	(a, d, j, n, s)	S027	(w, x, y, z)
S013	(a, e)	S028	(y, z, w, x)
S014	(a, f)	S029	(x, y, z, w, x)
S015	(c, d, j)	S030	(w, x, y)

۳-۱- توسعه الگوریتم سراسر گشت و گذار با استفاده از گراف وزن دار

برای اینکه بتوانیم نتیجه قابل لمسی از پیمایش گراف داشته باشیم، پیشنهاد ما این است که گراف مذکور وزن دار گردد. برای وزن دار نمودن آن لازم است که گراف را بصورت ماتریس مجاورت جهت دار تعریف نماییم. ماتریس مجاورت (Adjacency matrix) یک گراف برچسب داراست که حاوی اطلاعات مربوط به تعداد یال‌هایی بین رأس‌های مختلف گراف می‌باشد. بصورت دقیق تر ماتریس مجاورت G یک گراف متناهی (جهت‌دار یا غیرجهت‌دار) با n رأس است، یک ماتریس n × n می‌باشدکه

درایه آن برابر با تعداد یال‌هایی که دو رأس v_i و v_j را به هم متصل نماید است که در نهایت تعداد ۱ های ماتریس مجاورت دو برابر تعداد یال‌ها است .

قوانین کلی به این شرح است: تمام درایه‌های ماتریس مجاورت یک گراف کامل یک است به جز درایه‌های روی قطر اصلی و تمام درایه‌های ماتریس مجاورت یک گراف خالی صفر اند.

۲-۳- کوتاه ترین مسیر بین دو راس در ماتریس مجاورت جهت دار وزن دار

برای اینکه بتوان بین صفحات وبی که در هدف لینک ها قرار می گیرند پیمایش انجام داد ، صفحاتی که در مسیر عبور مرورگر قرار گرفته ولی هدف نهایی نیستند ، بعنوان راس هایی در مسیر که دارای هزینه پیمایش هستند را در نظر می گیریم. از طریق جستجوی عمقی می توان به هر کدام از راسها به کمک یال های وزن دار دسترسی پیدا نمود. از الگوریتم دیکسترا برای پیمایش راسها و یافتن کوتاه ترین مسیر بین راس مقصد و راس هدف استفاده می نماییم.

```
Let G = (V, E) be a weighted, directed graph.
Let T[i, j] give the weight on the edge from vertex i to vertex j, and assume the following:
T[i, j] > 0 (T[i, j] is non-negative);
T[i, i] = 0;
T[i, j] > 0 if there is an edge from vertex i to vertex j in edge set E; and
T[i, j] =  if there is no edge from vertex i to vertex j in E;
void ShortestPath(void)
{
(Let MinDist be a variable which contains edge weights as values)
(and let Minimum(x,y) be a function which returns the lesser value of x and y.)
/*Let v1  V be the vertex at which ShortestPath begins.*/
/*Initialize W and ShortDist[x] as follows:*/
W = {v1};
ShortDist[v1] = 0;
for (each x in V - {v1}) ShortDist[x] = T[v1][x];
/*Begin enlarging W 1 vertex at a time until W contains all vertices in V.*/
while (W != V) {
/*Find the vertex w  V-W which has the shortest distance from v1.*/
MinDist = ;
for (each w  V-W) {
   if(ShortDist[v] < MinDist) {
      MinDist = ShortDist[v];
      w = v;
   } }
/*Add w to W*/
W = W  {w};
/*Update the ShortDist array.*/
for (each x  V-W) {
   ShortDist[x] =
    Minimum(ShortDist[x], ShortDist[w] + T[w][x]);
} } }
```

الگوریتم دیکسترا برای یافتن کوتاهترین مسیر بین دو راس

در ادامه روشی هوشمند با استفاده از شبکه عصبی برای پیش بینی مسیرهای راهبری کاربران ارائه خواهیم نمود.

۳-۳- پیشگویی مسیر حرکت کاربران در وب با استفاده از یک مدل هوشمند شبکه عصبی

در این بخش قصد داریم الگوریتم پیشنهادی خود را برای خوشه بندی داده های مربوط به سابقه جستجوی کاربران به منظور ارائه پیشنهادات بهتر به کاربران جهت میل به بازدید صفحات پیشنهادی ارائه نماییم.

شبکه عصبی در گذشته از روشهایی مانند back-propagation برای یادگیری و مقداردهی به گره ها استفاده می نمود ولی این روش دارای معایبی مانند گیر افتادن در بهینه محلی است. چنانچه در ورودی های شبکه تغییر نسبتا بزرگی رخ دهد، میل به تغییر مقادیر وزن گره ها کم بوده و شبکه تمایل دارد تا در یک حالت پایدار باقی بماند. برای پوشش این ضعف، از الگوریتم کلونی زنبور برای یادگیری و مقداردهی به وزنها استفاده می نماییم که قدرت کافی برای کشف نقاط جدید در فضای جستجو را دارد .

در روش جدید خود که ABC-ANN نام دارد، برای خوشه بندی ما از یک شبکه عصبی دو لایه برای طرح خود استفاده می نماییم. برای مسئله پیشگویی وب، نمونه ای از ورودی یاددهنده به صورت $\langle [k_{t-\tau+1}, \dots, k_{t-1}, k_t]^T, d \rangle$ میباشد که در آن $[k_{t-\tau+1}, \dots, k_{t-1}, k_t]^T$، ورودی شبکه عصبی و d صفحه وب مورد نظر می باشد. همچنین k شماره ID یک صفحه وب است. خروجی شبکه نیز یک عدد بولین است که نحوه انجام این کار در ادامه توضیح داده خواهد شد.

خروجی احتمالی تقریبی به صورت $o' = f(o(I)) = p_{t+1}$ می باشد که I یک ورودی جلسه می باشد و $p_{t+1} = p(d|k_{t-\tau+1}, \dots, k_{t-1}, k_t)$ می باشد. با استفاده از تابع sigmoid به عنوان تابع انتقال دهنده خواهیم داشت:

$$o = \sigma(\omega . I) \qquad \sigma(y) = \frac{1}{1 + e^{-y}}$$

که به این شکل شبکه عصبی می تواند مجموعه داده های مجزا را به صورت غیر خطی دستکاری نماید. در رابطه بالا I ورودی شبکه، o خروجی شبکه، ω ماتریس ضرایب، و σ تابع sigmoid است. برای آموزش گره ها از الگوریتم کلونی زنبور استفاده می نماییم. این الگوریتم سعی می نماید مجذور خطا بین مقادیر خروجی شبکه و مقادیر خروجی مطلوب را کاهش دهد. از این رو، تعدادی راه حل کاندید خواهیم داشت که هر کدام معادل یک منبع غذایی برای زنبورها محسوب می گردد که این راه حل های کاندید هر کدام یک شبکه عصبی مستقل است. بعد از مقداردهی اولیه وزن گره های شبکه که به صورت تصادفی از فضای جستجو انتخاب می شود، نوبت به ورود داده ها از یک مجموعه داده است که به ازای یک ورودی، تمام راه حلهای موجود مقدار تابع شایستگی خود را محاسبه می نمایند. در ادامه نوبت به زنبورهای کارگر می رسد. چون به ازای هر راه حل، یک زنبور کارگر متناظر داریم، پس با یک تناظر یک به یک، تمام راه حلهای موجود یک راه حل جدید بر اساس روشهای گفته شده را تولید می نمایند. سپس مقدار تابع شایستگی بر اساس داده ورودی مورد نظر برای راه حل های جدید محاسبه می گردد. اگر این راه حلهای جدید تولید شده بهتر از راه حلهای فعلی باشند جایگزین آنها خواهند شد. در واقع راه حل جدید دارای مقادیر تغییر یافته ای نسبت به راه حل فعلی برای وزن گره ها می باشد. بر اساس مقادیر شایستگی تولید شده جدید، یک مقدار احتمال برای هر کدام از راه حلها انتخاب می شود و در ادامه بر اساس این مقادیر، زنبورهای تماشاگر که تعدادشان با تعداد راه حلها یکسان است، به طور تصادفی راه حلهایی را انتخاب و راه حل جدیدی را با استفاده از تغییر مقادیر وزن گره ها تولید می نمایند. مجددا، مقدار تابع شایستگی بر اساس داده ورودی مورد نظر برای راه حل های جدید محاسبه می گردد. اگر این راه حلهای جدید تولید شده بهتر از راه حلهای فعلی باشند جایگزین آنها خواهند شد. تعداد تکرار هر کدام از این مراحل نیز خود یک پارامتر کنترلی است که مقدارش در بخش ۴ مشخص خواهد شد.

۴ – آزمایشات

۴-۱- شبیه سازی ساختار گراف

شبیه سازی را با نرم افزار متلب انجام داده و وزن ها را به صورت تصادفی انتخاب نمودیم . گراف شکل ۱ را به عنوان مثال انتخاب نموده ، ورودی زیر که راس ها و وزن های گراف است را وارد می نماییم و کوتاهترین مسیر به همراه هزینه را با اعمال الگوریتم دیکسترا خروجی میگیریم که در جداول زیر ملاحظه می شود. وزن ها بین صفر تا یک است. در ادامه ، w وزن های اختصاص یافته به گراف و DG لینک های خارج شده از گره است. برای مثال گره ۱ به پنج گره دیگر لینک دارد.

```
w = [.41 .99 .51 .32 .15 .45 .38 .32 .36 .29 .21 .41 .99 .51
.32 .15 .45 .38 .32 .36 .29 .21,.04];
```

```
DG = sparse ([1 1 1 1 1 3 3 4 4 4 7 7 7 7 10 10 12 12 16 16
16 20 14 14 18 22 23 26 22],[2 3 4 5 6 7 4 8 9 10 10 11 12
13 14 15 16 17 14 20 21 24 18 19 22 23 26 25 25],w);
```

اکنون برای مثال دو مسیر از گره ۱ به ۲۰ و از گره ۱ به ۷ را انتخاب می نماییم:

```
[cost rute] =dijkstra(DG,1,20)  = (DG,a,t)
[cost rute] =dijkstra(DG,1,7) = (DG,a ,g)
```

کوتاهترین مسیر و کمترین هزینه (وزن) را به عنوان خروجی در جدول زیر ملاحظه می نماییم:

جدول ۲:خروجی الگوریتم دیکسترا برای یافتن کوتاهترین مسیر با کمترین وزن

کوتاهترین مسیر	هزینه	مسیرراهبری کاربران
1→3→7→12→ 16→ 20	۳.۲۴۰۰	از مسیر ۱ تا ۲۰
1→ 3→ 7	۱.۴۴۰۰	از مسیر ۱ تا ۷

در اینجا ملاحظه می نمایید که با وارد کردن وزن ها و راس های گراف کوتاهترین مسیر و کمترین هزینه را خروجی میگیریم. روش گراف وزن دار در مقایسه با روش های TSP و اپریوری مزیت هایی از جمله کاهش حجم حافظه مصرفی به دلیل استفاده از ماتریس مجاورت دارد و تکرار و عمل استفاده از پشته حذف شده است. همچنین در این روش به دلیل وزن دار کردن گراف کاربران با کمترین وزن به مقصد می رسند.

۴-۲- نتایج الگوریتم ABC-ANN

برای نمایش نتایج تولید شده توسط ABC-ANN و مقایسه آنها با نتایج کارهای انجام گرفته در سابق، از معیار دقت استفاده می گردد. این معیار، یک روش استاندارد برای ارزیابی داده هایی است که در مرحله آموزش دیده نشده اند. این معیار با علامت اختصاری pr(hit|mismatch) نیز شناخته می شود. همچنین داده های آموزش دیده نیز با علامت اختصاری pr(hit|match) مورد ارزیابی قرار می گیرند.

۴-۲-۱- مجموعه داده مورد استفاده

برای تست کارایی روش پیشنهادی خود، آنرا بر روی داده های مجموعه داده دانشگاه depaul اجرا می نماییم. این مجموعه داده سابقه وبگردی و جستجوی کاربران در یک بازه یک ماهه را از روی وب سرور دانشگاه گردآوری نموده ایم (http://www.cs.depaul.edu). داده های اصلی (فیلتر نشده) شامل ۲۰,۹۵۰ جلسه از ۵,۴۴۶ کاربر است. فایل داده های فیلتر شده از کنار گذاشتن جلساتی که با اندازه ۱ (به معنای مشاهده شده به تعداد یکبار) بدست آمده است. داده های فیلتر شده شامل ۱۳,۷۴۵ جلسه و ۶۸۳ صفحه مشاهده شده است.

۴-۲-۲- تنظیمات الگوریتم جدید

در الگوریتم جدید پیشنهادی تنظیمات پارامترهای کنترلی و متغیرهای مورد نظر اهمیت بسزایی دارند. به منظور متعادل کردن زمان اجرا و نیز در نظر گرفتن معیار کارایی مورد نظر، راه حلهای مورد استفاده ۱۰ عدد

می باشد. این تعداد مساوی با تعداد زنبورهای کارگر و تماشاگر نیز هست. با توجه به تعداد ورودی های زیاد مسئله، ما به ازای هر ورود داده ۱۰ بار تکرار الگوریتم کلونی زنبور را خواهیم داشت. یعنی به ازای هر داده ورودی از مجموعه داده، هر کدام از زنبورهای کارگر و تماشاگر ۱۰ بار بر روی راه حلهای موجود عمل نموده و سعی در بهینه کردن خروجی مورد نظر با تغییر مقادیر وزن گره ها را دارند .

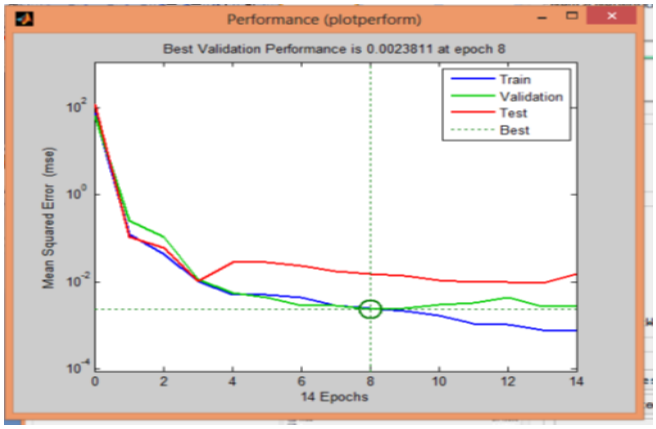
همچنین دو سوم داده های موجود در دیتاست برای کار آموزش به کار می رود و یک سوم باقی مانده نیز برای تست استفاده می گردد.. هر آزمایش نیز ۳۰ بار با دانه های تصادفی مختلف اجرا شده و نتایج بدست آمده در واقع میانگین ۳۰ بار اجرای الگوریتم پیشنهادی بر روی داده های مجموعه داده مورد نظر می باشد.

در این قسمت نتایج بدست آمده از شبیه سازی الگوریتم پیشنهادی ABC-ANN نشان داده می شود. برای پیاده سازی الگوریتم جدید از محیط MATLAB 2013 استفاده شده است. همچنین نتایج بدست آمده از الگوریتم جدید را با نتایج الگوریتم های مراجع [۱] و [۴] مقایسه کرده ایم. هر دو این مراجع روشهایی را برای پیشگویی مسیرهای حرکتی کاربران وب ارائه می دهند.در جدول ۳ نتایج حاصل از الگوریتم جدید پیشنهادی با نتایجی که در مرجع [۴] آمده است مورد مقایسه قرار گرفته اند.

جدول ۳: مقایسه نتایج حاصل از الگوریتم جدید با روش های گذشته [۴]

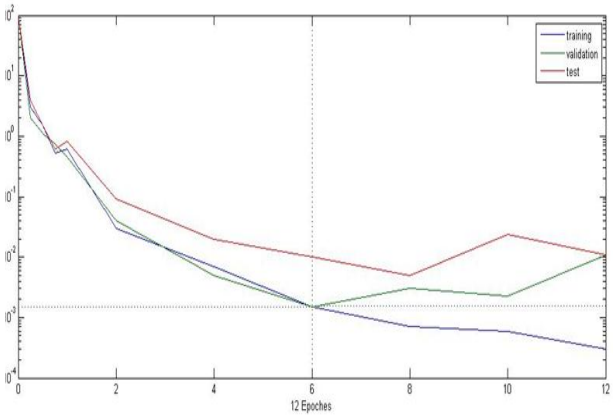
نوع روش پیش بینی	ARM	Markov	ANN	ABC-ANN
Pr(hit|mismatch)	0	0	0.108	0.162
Pr(hit|match)	0.63	0.199	0.152	0.239

نتایج جدول بالا نشان می دهد برای معیار دقت عمومی نتایج الگوریتم جدید ABC-ANN بهتر از نتایج دیگر الگوریتم ها است. هر چند برای این معیار، الگوریتمهای ARM و مارکوف صفر می باشد. علت این امر این است که تمام داده ها در مرحله آموزش دیده می شوند.برای معیار دقت آموزش نیز مجددا کارایی الگوریتم ABC-ANN بهتر از نتایج الگوریتم های دیگر است. برای مرحله آموزش، در بین الگوریتمهای پیشین، کارایی و دقت مارکوف بیشتر است و سپس الگوریتم مبتنی بر شبکه عصبی مصنوعی بهترین کارکرد را دارند. شکل ۲ نتایج شبیه سازی مرجع [۱] را نشان می دهد. ما نتایج الگوریتم پیشنهادی خود را برای معیارهای مورد نظر در شکل ۳ آورده ایم.



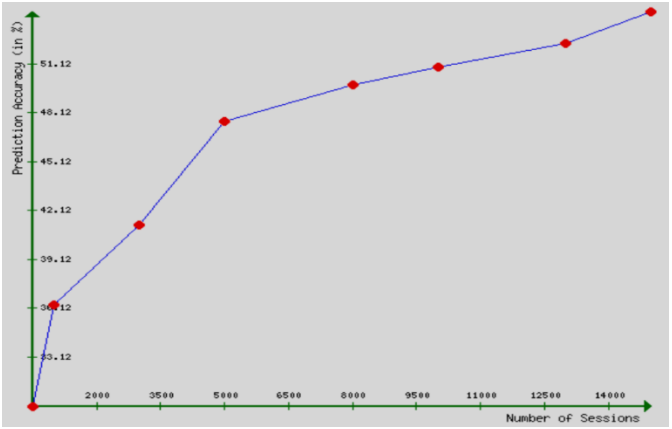
شکل ۲: اعتبارسنجی داده های ورودی در مرجع [۱]

سه رنگ آبی، قرمز و سبز برای نمایش آموزش، اعتبارسنجی، و تست به کار می رود. در مرجع [۱] بهترین کارکرد برای هشتمین epoch حاصل می شود. در حالی که در الگوریتم پیشنهادی ما، در ششمین epoch این تعادل بدست می آید. بهترین مقدار در مرجع [۱] برای اعتبار سنجی مقدار ۰.۰۰۲۳۸۱۱ است در حالیکه در الگوریتم پیشنهادی ما این مقدار ۰.۰۰۱۵ است.

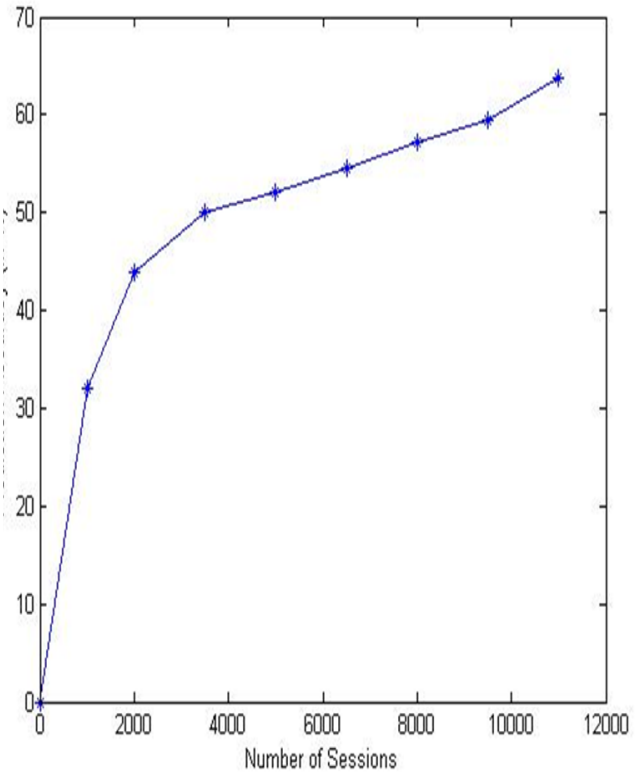


شکل شماره 3: اعتبار سنجی داده های ورودی در الگوریتم پیشنهادی ABC-ANN

همچنین، رابطه بین دقت پیشگویی و تعداد جلسات (اندازه داده های یاددهنده) نیز می تواند کارایی الگوریتمها را نشان دهد. شکل ۴ نمودار حاصل از کارایی الگوریتم معرفی شده در مرجع [۱] را نشان می دهد.



شکل شماره ۴: رابطه بین دقت مدل و تعداد جلسات در مرجع [۱]



شکل شماره ۵: رابطه بین دقت مدل و تعداد جلسات در الگوریتم پیشنهادی ABC-ANN

۵ – نتایج و پیشنهادات

در این مقاله با استفاده از ماتریس مجاورت و از طریق الگوریتم دیکسترا گراف را وزن دار نموده و کوتاهترین مسیر راهبری کاربران و کمترین وزن را پیدا نمودیم که این روش به بهبود ساختار گراف و کاهش حجم حافظه کمک می نماید. همچنین به ارائه روشی جدید برای خوشه بندی سابقه مسیرهای حرکتی کاربران در وب پرداخته ایم. هدف از این خوشه بندی، ارائه پیشنهادات هوشمند جهت راهنمایی کاربران برای جستجوی کارا و بهینه در وب است به طوری که بدون اتلاف وقت خود و ایجاد ترافیک اضافی در شبکه، هر چه بهتر به صفحات مورد نظر خود دسترسی مستقیم داشته باشند. برای خوشه بندی و ایجاد کلاسهایی جهت طبقه بندی جستجوهای پیشین، از شبکه عصبی استفاده نموده ایم. شبکه عصبی قدرت یادگیری و انطباق خود با مسئله خاص را دارد. ولی روشهای موجود برای یادگیری همانند back-propagation دارای ضعف هایی همانند گیر کردن در بهینه های محلی می باشد به همین خاطر جهت کارایی بیشتر مرحله یادگیری، از الگوریتم کلونی زنبور استفاده می نماییم که یکی از روشهای قدرتمند در مسائل بهینه سازی است . امیدواریم در شبکه های عصبی نیز که یادگیری و دادن مقدار به وزن گره ها یک مسئله بهینه سازی به حساب می آید موفق عمل نماید. نتایج حاصل از اجرای الگوریتم پیشنهادی بر روی مجموعه داده موجود، نشان می دهد که روش ارائه شده دارای کارایی مناسبی است و دقت پیش بینی بهبود یافته است.

مراجع

1. O.P. Mandal, H.K. Azad,2014, “Web Access Prediction Model using Clustering and Artificial Neural Network”, International Journal of Engineering Research & Technology (IJERT)
2. Yao-Te Wang, Anthony J.T. Lee,2011, Mining Web navigation patterns with a path traversal graph, Expert Systems with Applications
3. MS DIPA DIXIT, MR JAYANT GADGE, 2010, A New Approach for Clustering of Navigation Patterns of Online Users, International Journal of Engineering Science and Technology
4. M.A. Awad, L.R. Khan, 2007, “Web Navigation Prediction Using Multiple Evidence Combination and Domain Knowledge”, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART A: SYSTEMS AND HUMANS
5. C.J. Carmona ,S. Ramچrez-Gallego , F. Torres, E. Bernal, M.J. del Jesus, S. Garcia, 2012, Web usage mining to improve the design of an e-commerce website: OrOliveSur.com, Expert Systems with Applications
6. Sk.AbdulNabi and Dr. P. Premchand, 2012 , EFFECTIVE PERFORMANCE OF INFORMATION RETRIEVAL ON WEB BY USING WEB CRAWLING, International Journal of Web & Semantic Technology (IJWesT)
7. Hao Ma, Irwin King, and Michael Rung-Tsong Lyu, Fellow, 2012 , Mining Web Graphs for Recommendations, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING
8. .Mohit M.Patel, Assistant Prof.Shailendra K.Mishra,2013, Design and Development of Efficient Algorithm in Web Usage Mining For Web Personalization, International Journal of Engineering Trends and Technology (IJETT)
9. Maryam Jafari1, Farzad SoleymaniSabzchi1, Shahram Jamali, 2013, Extracting Users’Navigational Behavior from Web Log Data: a Survey, Journal of Computer Sciences and Applications
10. Md. Zahid Hasan, Khawja Jakaria Ahmad Chisty and Nur-E-Zaman Ayshik, 2012, Research Challenges in Web Data Mining, International Journal of Computer Science and Telecommunications
11. M. Awad, L. Khan, and B. Thuraisingham,2008, “Predicting WWW surfing using multiple evidence combination,” VLDB J.
12. D. Karaboga, B. Basturk ,2008, "On the performance of Artificial Bee Colony (ABC) algorithm," Applied Soft Computing
13. D. Karaboga, B. Akay,2009, "A comparative study of Artificial Bee Colony algorithm," J Appl Mathe Comput
14. mamoun A.Awad and issa Khalil,2012, prediction of users web browsing behavior: application of markov model, IEEE transactions on systems