

Investigating Operational and Practical Classifications of Algorithmic Perspectives in Computer Science

M. Eslami^{*,1}

¹ Department of Computer Science, Islamic Azad University, Kermanshah Branch, Kermanshah, Iran

ABSTRACT

Received: 20 August 2023

Accepted: 29 November 2023

KEYWORDS:

Algorithm analysis,
Algorithmic perspectives,
Algorithm applications,
Dynamic method,
Artificial intelligence,

¹ Corresponding author

 M.eslami2018@yahoo.com

In today's knowledge-oriented world, there are many problems that need to be solved in a structured and systematic way. In this regard, an effective solution is to use suitable algorithmic approaches and perspectives. Algorithm includes a set of processes and instructions that can be used to solve many computer problems or real world problems in a regular and effective way. The use of algorithms in solving systematic problems saves time and money and provides new and valuable solutions. In fact, the algorithm is considered the beating heart of potentially powerful technologies such as artificial intelligence and today, instead of thinking about the future of algorithms, people consider the future to belong to algorithms. Although algorithms have relatively the same concept in different fields of implementation, but in order to achieve specific applications of the desired algorithm in different fields, we must take advantage of the approaches and perspectives of appropriate algorithms. Therefore, according to the importance of the subject, in this article, we intend to investigate the different dimensions of algorithmic perspectives and describe the common operational and practical classifications of important algorithms in computer science and related technology-oriented fields.



NUMBER OF REFERENCES

21



NUMBER OF FIGURES

0



NUMBER OF TABLES

0

نشریه تخصصی آرمان پردازش، دوره ۴، شماره ۳، پاییز ۱۴۰۲

فصلنامه تخصصی آرمان پردازش (APJ)

Homepage: www.armanprocessjournal.ir

بررسی رده بندی های عملیاتی و کاربردی دیدگاههای الگوریتمیک در علوم کامپیوتری

مریم اسلامی^{۱*}^۱ گروه مهندسی کامپیوتر، دانشگاه آزاد اسلامی، واحد کرمانشاه، کرمانشاه، ایران

چکیده

در دنیای دانش محور امروز مسائل فراوانی وجود دارند که لازم است به شیوه ساخت یافته و قاعده مند حل شوند. یک راهکار موثر بهره گیری از دیدگاههای الگوریتمیک می باشد. الگوریتم شامل مجموعه‌ای از فرایندها و دستورات عمل‌هایی است که به کمک آن‌ها می‌توان بسیاری از مسائل کامپیوتری یا مسائل دنیای واقعی را به شیوه قاعده مند و بهینه حل کرد. استفاده از الگوریتم‌ها در حل مسائل سیستماتیک باعث صرفه جویی در وقت و هزینه شده و راهکارهای نوین و ارزشمندی را پیش رو قرار می‌دهد. الگوریتم در واقع، قلب تپنده فناوری‌های بالقوه قدرتمندی همچون هوش مصنوعی محسوب می‌شود و امروز افراد به‌جای اینکه به آینده الگوریتم‌ها فکر کنند، آینده را متعلق به الگوریتم‌ها می‌دانند. هرچند الگوریتم‌ها مفهوم نسبتاً یکسانی در حوزه‌های اجرائی مختلف دارند، اما برای حصول کاربردهای خاص الگوریتم مورد نظر در حیطه‌های مختلف، باید از دیدگاههای الگوریتم‌های مناسب بهره بگیریم. لذا با توجه به اهمیت موضوع در این مقاله قصد داریم به بررسی ابعاد مختلف دیدگاههای الگوریتمیک پرداخته و رده بندی های عملیاتی و کاربردی رایج الگوریتم‌های مهم در علوم کامپیوتری و حوزه های فناور محور مرتبط را توصیف نمائیم.

واژگان کلیدی:

تحلیل الگوریتم،
دیدگاههای الگوریتمیک،
کاربردهای الگوریتم،
روش پویا،
هوش مصنوعی،


تعداد مراجع
۲۱


تعداد شکل‌ها
۰


تعداد جداول
۰

مقدمه

- همه مراحل مربوط به الگوریتم واضح، مشخص و بدون ابهام باشند.
- در صورت نیاز به ورودی، ورودی ها باید به طور کامل تعریف شوند.
- حداقل یک خروجی داشته باشد و به طور کامل تعریف شود.
- هر مرحله از الگوریتم موثر باشد و وظیفه خاصی داشته باشد.
- الگوریتم باید متناهی باشد یعنی بعد از یک مدت زمان مشخص به پایان برسد.
- الگوریتم باید ساده، عمومی و کاربردی باشد تا با منابع موجود بتوان آن را اجرا کرد.
- الگوریتم های طراحی شده باید مستقل از زبان باشد و دارای دستورالعمل هایی باشد که با هر زبان برنامه نویسی قابل پیاده سازی باشد و خروجی آن در همه زبان ها یکی باشد.
- الگوریتم ها اجزای مختلفی دارند که بسته به کاربردشان می توانند متفاوت باشند ولی در کل هر الگوریتم دارای اجزاء زیر است:

ورودی و خروجی

همانطور که قبلاً هم گفتیم، الگوریتم ها دنباله ای از مراحل هستند که دست به دست هم می دهند تا به یک خروجی مشخص برسند. یک مثال رایج این است که الگوریتم تشخیص چهره گوشی هوشمند برای باز کردن قفل گوشی، به تصویر چهره شما نیاز دارد تا با بررسی ویژگی های چهره شما و مقایسه آن با تصویری که قبلاً ذخیره کرده اید، در مورد باز کردن قفل گوشی تصمیم بگیرد.

دنباله مراحل الگوریتم (شرط و تکرار)

پس از این که الگوریتم ورودی را گرفت، باید از مراحل ترتیبی الگوریتم رد شود تا به خروجی موردنظر برسد. شرط ها، معیاری برای تغییر مسیر ورودی هستند. گاهی اوقات مراحل وجود دارند که باید برای رسیدن به خروجی های موردانتظار تکرار شوند. به عنوان مثال، برای جستجوی یک فایل در یک پوشه، اسم فایل به طور مداوم با بقیه فایل های موجود در پوشه مقایسه می شود تا فایل موردنظر پیدا شود. این مرحله ی جستجو تا زمان رسیدن به خروجی، تکرار می شود [۵]. کامپیوتر، مدام در حال انجام محاسبات ریاضی است و مسائل زیادی برای حل کردن دارد. به همین دلیل است که الگوریتم ها قلب علوم کامپیوتر را تشکیل می دهند. یک الگوریتم ریاضی با گرفتن ورودی و اعمال مراحل محاسباتی و منطقی به خروجی موردنظر می رسد. مراحل یک الگوریتم در برنامه نویسی به صورت زیر است [۶-۷]:

- تعریف مسئله: چه کاری باید انجام دهیم؟
- جمع آوری داده ها: برای حل این مشکل، چه داده ها یا ورودی هایی داریم؟
- پردازش داده ها: تبدیل داده های ورودی به یک فرم قابل استفاده.
- یک رویکرد منطقی: بکار بردن داده های پردازش شده در منطق حل مسئله

سال ها است که علم با یافتن پاسخ سؤالات خود و استفاده از آن ها در پیشامدهایی که الگوی تکراری دارند، اهداف خود را پیش می برد و سریع تر از انتظار رازهای طبیعت را از دل آن بیرون می کشد. راه حل هایی که تست شده و مطمئن هستند و می توانند سوالاتی با مفاهیم یکسان را حل کنند، الگوریتم نامیده می شوند. چنانچه بخواهیم معنی الگوریتم را در زمینه علوم کامپیوتر بررسی کنیم، می توان گفت الگوریتم ها مجموعه فرایندهایی هستند که به کمک آن ها می توان بسیاری از مسائل برنامه نویسی را به راحتی حل کرد. در واقع الگوریتم به توالی مشخصی از دستورالعمل ها گفته می شود که برای حل یک مسئله یا تکمیل وظایف موردنظر اجرا می شوند. در دنیای برنامه نویسی و کامپیوتر، بسیاری از روتین های سخت افزاری و نرم افزاری بر اساس الگوریتم ها پیش می روند چون استفاده درست از الگوریتم ها باعث تسریع و ارتقا کیفیت کارها می شود. الگوریتم ها در دنیای واقعی هم کاربرد زیادی دارند. با استفاده از الگوریتم ها، یک مسئله به چند قسمت قابل مدیریت تقسیم می شود و انجام هر قسمت طی یک توالی مشخص صورت می گیرد [۱]. در حوزه برنامه نویسی اهمیت الگوریتم ها چند برابر می شود زیرا استفاده از یک الگوریتم درست به کشف سریع تر خطاها و سردرگمی های احتمالی مربوط به برنامه کمک می کند. برای پیاده سازی الگوریتم های برنامه نویسی به زبان خاصی محدود نبوده و به راحتی می توان الگوریتم موردنظر را توسعه داد و خروجی موردانتظار را دریافت نمود. پیشنهادات سایت خرید، پیشنهاد ویدیو یوتیوب و رتبه بندی صفحات نتایج جستجو از معروف ترین الگوریتم هایی هستند که به صورت مداوم با آنها سروکار داریم و باعث ارتقا سرعت و کیفیت بعضی از تسک های پیچیده شده اند [۲]. مهمترین مزایا و معایب بهره گیری از دیدگاههای الگوریتمیک در حل مسائل روزمره به شرح زیر است:

مزایا

- درک نحوه کار الگوریتم راحت است.
- نمایش راه حل مسئله به صورت مرحله ای باعث روشن شدن مسیر می شود.
- حل مسئله به زیر مسائل، کار را برای برنامه نویس راحت تر می کند.

معایب

- پروسه نوشتن یک الگوریتم می تواند زمان بر باشد.
- درک یک مسئله پیچیده از طریق الگوریتم می تواند بسیار سخت باشد.
- نمایش شاخه ها و حلقه ها در الگوریتم های مسائل پیچیده، سخت است.
- بمنظور این که توالی دستورات خاص در قالب یک الگوریتم، ارائه شوند باید ویژگی های خاصی داشته باشند [۳-۴]:

- راه حل: ارائه راه حل با استفاده از زبان طبیعی، ابزار گرافیکی، نمودار، فلوجارت، کد یا شبه کد

رده بندی های دیدگاههای الگوریتمیک

الگوریتم‌ها دارای نقش مهمی در پیاده سازی راهکارهای حوزه برنامه‌نویسی و حل مسئله هستند و از لحاظ کارایی و با توجه به نوع مسئله انواع مختلفی دارند که ما در این بخش به بررسی تعدادی از آن‌ها می‌پردازیم. در حوزه علوم کامپیوتری مسائل مختلفی وجود دارند که هر کدام از رده بندی های الگوریتم ها با هدف حل و رفع چالش های مرتبط با مسائل خاصی ایجاد شده اند. از مهم ترین رده بندی های کارکردی الگوریتم ها در حوزه علوم کامپیوتر و برنامه نویسی می توان به موارد زیر اشاره کرد [۱۰-۸و۵]:

دیدگاه Brute Force

Brute Force، ساده ترین الگوریتمی است که می توان برای حل یک مسئله در نظر گرفت. این الگوریتم، همه گزینه های ممکن برای حل یک مسئله را به صورت کورکورانه امتحان می کند تا به یک راه حل بهینه برسد. مانند تکنیک ساده ای که برای باز کردن قفل ۴ رقمی یک گاوصندوق به کار می بریم. ترکیب ۴ رقمی از اعداد ۰ تا ۹ را آنقدر تکرار می کنیم تا در نهایت به یک ترکیب درست برسیم و در گاوصندوق باز شود. Brute Force به راحتی قابل پیاده سازی است و پیچیدگی زمانی و هزینه محاسباتی آن به تعداد راه حل های کاندید بستگی دارد. در کل، Brute Force مناسب مسائلی است که محدود و کوچک هستند و مشکلات پیچیده و بزرگی ندارند. الگوریتم Brute Force زمانی استفاده می شود که هیچ الگوریتم دیگری برای سرعت بخشیدن به فرآیندهای مرتبط وجود نداشته باشد و برنامه نویس باید همه راه حل های ممکن را تست کند [۱۱].

دیدگاه حریصانه Greedy

روش حریصانه یکی از الگوریتم های ساده و موثر است که بیشتر در مسائل مربوط به بهینه سازی استفاده می شود. این روش از هر استراتژی استفاده می کند تا بهترین راه حل برسد. شاید فکر کنید که دو روش Greedy و Brute Force یکی هستند ولی روش حریصانه از لحاظ زمان محاسبه کارآمدتر از Brute Force و به مراتب به نتایج بهتری می رسد. روش کار الگوریتم حریصانه به این صورت است که از بین راه حل های بهینه محلی یکی را به عنوان راه حل بهینه سراسری انتخاب می کند. این الگوریتم برای همه مسائل کار نمی کند ولی وقتی کار می کند، مثل یک طلسم عمل می کند. در کل، روش Greedy بهینه ترین راه حل را تضمین نمی کند. این الگوریتم در حل مسائلی مثل کوله پشتی کسری، مرتب سازی توپولوژیکی و زمان بندی کار به کار می رود [۱۲].

دیدگاه بازگشتی

ایده بازگشتی بودن یکی از ساده ترین روش های حل مسئله است چون نیاز به تفکر خاصی در مورد مشکلات فرعی نیست. این الگوریتم به طور مکرر خودش را فراخوانی می کند تا این که به نتیجه موردنظر برسد. به این صورت که کافی است یک تابع ساده برای حل در نظر بگیریم و با تکرار فراخوانی آن، تمام پیچیدگی های دیگر به طور خودکار حل شوند. در واقع، مسئله به نسخه کوچکتر تبدیل می شوند و با فراخوانی بازگشتی این نسخه های کوچکتر می توان مسئله اصلی را حل کرد. روش بازگشتی در حل برخی از مسائل ریاضی مثل دنباله فیبوناچی، تابع آکرمن و همچنین هوش مصنوعی، بازی های پازل، شطرنج و مرتب سازی ادغامی و سریع کاربرد دارد [۱۳].

دیدگاه Backtracking

Backtracking، نسخه توسعه یافته الگوریتم brute force است و به نحوی، روش بازگشتی و brute force را ترکیب می کند. به این صورت که، ابتدا یک گزینه را انتخاب می کنید و بعد سعی می کنید مسئله را با آن حل کنید. اگر با این گزینه در حل مسئله شکست خوردید به نقطه شکست برمی گردید و با راه حل دیگر ادامه می دهید. روش Backtracking برای حل مسائلی مثل N-Queens، کوله پشتی، رنگ آمیزی گراف و تولید تمام رشته های باینری کاربرد دارد [۱۴و۱۱].

دیدگاه تقسیم و غلبه

این رویکرد، مسئله را به مسائل فرعی کوچکتر تقسیم می کند و بعد، نتیجه این زیر مسائل را با هم ترکیب می کند تا به نتیجه نهایی برسد. پس این روش تقسیم و غلبه دارای ۳ مرحله اصلی است: تقسیم، حل و ترکیب. الگوریتم تقسیم و غلبه در حل بسیاری از مسائل به کار می رود و دلیل آن هم این است که راه حل پایدار و بهینه ای را ارائه می دهد. مسائلی مثل جستجوی باینری، مرتب سازی ادغامی، پیدا کردن میانه و ضرب ماتریس با این روش قابل حل هستند.

دیدگاه برنامه نویسی پویا

عملکرد روش برنامه نویسی پویا شبیه تقسیم و غلبه است ولی با این تفاوت که نتایج حاصل از زیرمسائل ذخیره می شوند تا در آینده استفاده شوند و از محاسبات تکراری جلوگیری شود. این الگوریتم کاربرد بسیار زیادی دارد ولی دلیل آن هم این است که کارآمدترین روش را برای حل مسئله ارائه می دهد و از نظر پیچیدگی زمانی هم بهینه است. این روش دارای دو نسخه است [۱۵]:

- رویکرد پایین به بالا: در این روش، اول زیرمسائل حل می شوند و از نتایج آنها برای حل زیر مسائل بالایی و در نتیجه مسئله اصلی استفاده می شود.

الگوریتم های مرتب سازی

الگوریتم های مرتب سازی عمدتاً برای مرتب کردن مجدد مقادیر زیادی از داده ها به شیوه ای کارآمد استفاده می شوند تا بتوان آن ها را راحت تر جستجو و تغییراتی در آن ایجاد کرد. آنها همچنین برای بهبود کارایی الگوریتم های دیگر مانند جستجو و ادغام، که بر داده های مرتب شده برای عملیات خود متکی هستند، استفاده می شوند. فرض کنید می خواهید تعدادی برگه امتحانی را بر اساس اطلاعات افراد مرتب کنید. می توانید این کار را بر اساس حروف الفبای نام کوچک، نام خانوادگی یا دیگر مشخصات افراد انجام دهید؛ در هر حال رویه ای که بایستی با انجام یک سری مراحل مشخص تا زمان حصول نتیجه در پیش بگیرید، در واقع همان الگوریتم انجام کار است. کاربرد الگوریتمی که از آن صحبت می کنیم، در دنیای کامپیوتر هم فراوان است. این الگوریتم که البته اشکال مختلفی به خود می گیرد، الگوریتم مرتب سازی نام دارد و کاربرد این الگوریتم در برنامه نویسی هم زیاد است.

الگوریتم های تشخیص چهره

شاید کمتر به این موضوع توجه کرده باشید که چطور افراد را از چهره شان تشخیص می دهید. اگر خیلی ساده به این قضیه نگاه کنیم، خواهیم دید که چشم مان تصویری را می بیند، شبکه عصبی پیام مربوطه را به مغز می رساند و مغز تلاش می کند با تجزیه و تحلیل، تصویر را با اطلاعات قبلی اش از جمله ویژگی های ظاهری یا دیگر احساسات و تجربیات فردی انطباق دهد یا مقایسه کند. مسلماً الگوریتم حل مسأله را ما، یعنی انسان ها، باید به ماشین بدهیم تا نتیجه مورد نظرمان حاصل گردد؛ بنابراین کاربرد الگوریتم در این زمینه هم روشن است. البته ناگفته نماند که الگوریتم تشخیص چهره در سیستم های مختلف کامپیوتری، هنوز کاملاً عاری از خطا نیستند و اینجاست که برنامه نویسان و البته کاربرد الگوریتم در برنامه نویسی بایستی نقش خود را به خوبی ایفا کنند تا به الگوریتم تشخیص چهره بهتری دست پیدا کنیم.

حل مسائل تکراری

مسائل زیادی در زندگی ما وجود دارد که قبل از ما پیش روی انسان های دیگری قرار گرفته باشد. وقتی با مسائل تکراری روبرو می شوید، می توانید با روش های مختلف الگوریتم تازه ای برای حل آن ایجاد کنید که ممکن است لزوماً بهینه نباشد یا از الگوریتم هایی بهره بگیرید که قبلاً آزموده و تحلیل شده اند.

مدیریت امور شهروندی

دیدگاه های الگوریتمیک ترتیب مشخصی از گام ها را شامل می شوند که قادر خواهند بود در ابزارها و ادوات مدیریت جریان های کاری و مدیریت امور شهروندی به کار گرفته شوند. به عنوان نمونه پردازشگرهای دستگاه های کنترل ترافیک، چراغ های راهنمایی و رانندگی و زمان بندی حمل و نقل خورکار در امور شهری از این نمونه می باشند. مسلماً اگر الگوریتم های مناسبی برای این اهداف در نظر نگیریم، شاهد مشکلات

- رویکرد بالا به پایین: مسئله اصلی از همان ابتدا حل می شود تا اینکه نوبت به زیر مسئله مورد نظر برسد و با استفاده از نتایج مسائل قبلی، حل شود.

این الگوریتم در حل مسائل زیادی مثل بلمن-فورد، کوله پشتی، جمع زیر مجموعه، طولانی ترین زیر رشته مشترک و ضرب ماتریس زنجیره ای بکار می رود.

دیدگاه تصادفی

این الگوریتم، تصمیم گیری خود را بر اساس عناصر تصادفی انجام می دهد یعنی در منطق خود از این عناصر استفاده می کند. بهترین مثال برای این رویکرد، انتخاب یک عنصر محوری یا همان pivot در مرتب سازی سریع است. با این که این روش، زیاد استفاده نمی شود ولی دلیل اصلی تصادفی بودن، کاهش زمان اجرا و پیچیدگی زمانی است. مرتب سازی سریع تصادفی و Kager's از مهم ترین مسائلی هستند که در روش حل آنها از الگوریتم تصادفی استفاده می شود.

کاربرد های دیدگاه های الگوریتمیک

الگوریتم ها در سرتاسر حوزه ی علوم کامپیوتر به کار گرفته می شوند و ستون فقرات این میدان هستند. بنابراین دارای کاربردهای وسیعی در حوزه علوم مختلف دارند. در علم کامپیوتر الگوریتم به مجموعه ای از دستورالعمل ها گفته می شود که به رایانه اجازه می دهد هر کاری از اجرای ماشین حساب گرفته تا پرتاب موشک فضایی را انجام دهد. پایه و اساس برنامه های کامپیوتری را می توان الگوریتمی دانست که در زبان های برنامه نویسی تعریف شده است و ماشین آن ها را درک می کند. الگوریتم های رایانه ای به شدت روی عملکرد رسانه های اجتماعی تأثیر دارند که از جمله می توان به اینکه کدام پست ها ظاهر شوند و کدام آگهی ها قابل مشاهده باشند، اشاره کرد. مهندسان گوگل از الگوریتم ها برای بهبود جست و جوها، پیش بینی اینکه کاربران وارد چه بخش هایی خواهند شد و غیره استفاده می کنند. اطلاع از روش ایجاد الگوریتم، یکی از مهم ترین اجزای برنامه نویسی کامپیوتر هنگام حل مسئله است. دو جنبه ی اساسی در الگوریتم وجود دارد [۱۶ و ۱۳]:

- اهمیت نظری: شما باید چالش دنیای واقعی را در صورت ارائه، به ماژول های کوچک تر تقسیم کنید. برای ساختار شکنی موضوع ابتدا باید همه ی اجزای نظری آن را درک کنید.

- اهمیت عملی: این نظریه تا زمانی که عملی نشود بی فایده است و در نتیجه ارتباط نظری و عملی الگوریتم ها می تواند ارزیابی شود.

برخی از کاربردهای رایج دیدگاه های الگوریتمیک به شرح زیر می باشند [۱۹-۱۷ و ۱۳ و ۹ و ۲]:

از آن بخش از کاربرد الگوریتم‌هاست که به حصول همین هدف کمک می‌کند. در حوزه رمزنگاری، الگوریتم‌های پیشرفته‌ای وجود دارد که اطلاعات را دستکاری می‌کند تا غیر از مخاطب مورد نظر، برای هیچ شخص سومی قابل فهم نباشند. الگوریتم‌های رمزنگاری باید ویژگی‌های امنیت (عدم ارسال داده‌ها به افراد غیرمجاز)، هویت (تشخیص صحیح افراد مجاز) و غیرجعلی بودن اطلاعات را داشته باشد [۲۰].

حل مسائل منطقی و ریاضی

الگوریتم در ریاضی، مجموعه‌ای از مراحل است که برای محاسبات ریاضی از آن استفاده می‌کنند و در نگاه کلی، هدف از الگوریتم‌های ریاضی افزایش دقت در حل مسائل است. الگوریتم‌های مشهور و پیشرفته زیادی را در دنیای امروز بر پایه محاسبات و معادلات ریاضی بنا کرده‌اند. الگوریتم دایجسترا که از مشهورترین الگوریتم‌های مسیریابی جهان به شمار می‌رود، از قواعد و اصول ریاضی بهره می‌گیرد تا کوتاه‌ترین مسیر را بین مبدأ و مقصد مشخص بیابد. نقشه گوگل (Google Map) از همین الگوریتم بهره می‌گیرد تا مسیری کوتاه و کم‌هزینه پیش روی کاربرانش بگذارد.

یادگیری ماشین

یادگیری ماشین یکی از شاخه‌های هوش مصنوعی است و از الگوریتم‌های متعددی بهره می‌گیرد تا با استفاده از داده‌ها، نتایج دقیقی را پیش بینی کند. علم یادگیری ماشین را به‌طور کلی به دو روش یادگیری با ناظر (Supervised) و یادگیری بدون ناظر (Unsupervised) دسته‌بندی می‌کنند.

امروزه با توسعه صنایع و کسب و کارها و البته پیچیده‌تر شدن روابط انسانی، نیاز به پیش‌بینی خیلی چیزها بیشتر احساس می‌شود. اینجاست که یادگیری ماشین و البته کاربرد الگوریتم خودی نشان می‌دهند و اهمیت الگوریتم را بیش از پیش به ما یادآور می‌شوند.

علوم پزشکی

اهمیت علم پزشکی در زندگی انسان بر کسی پوشیده نیست. قطعاً پیشرفت علم پزشکی را نمی‌توان فارغ از توسعه کامپیوترها و استفاده از ابزارهایی نظیر اینترنت دانست. بنابراین جای شکی باقی نمی‌ماند که ردپای الگوریتم را در این حوزه هم خواهیم دید و کاربرد الگوریتم در پزشکی نیز زیاد است. ساده‌ترین حالت مثل ویزیت بیمار را در نظر بگیرید، پزشک در چنین موقعیتی هم گام‌های نسبتاً مشخصی (یا بهتر است بگوییم الگوریتمی) را طی می‌کند تا بتواند مشکل بیمار را تشخیص دهد. الگوریتم‌های پزشکی با اهداف انفورماتیک پزشکی و تصمیم‌گیری پزشکی تناسب دارند. تصمیمات پزشکی در چندین زمینه از فعالیت‌های پزشکی از جمله انتخاب تست پزشکی، تشخیص، درمان و پیش‌آگهی و کنترل خودکار تجهیزات پزشکی رخ می‌دهد.

عده‌ای خواهیم بود و باید وقت و انرژی بسیار بیشتری در راستای مرتفع‌سازی‌شان صرف کنیم. بی‌شک چنین حالتی مطلوب هیچ‌کس نیست.

کاربرد های اینترنتی و موتورهای جستجو

اینترنت ابزار بسیار مهم و حیاتی در دنیای امروز است و فقط در عرض چند سال، زندگی همه انسان‌ها را دستخوش تغییرات عمده‌ای کرده است. دیگر تا این‌جا مطلب باید مشخص شده باشد که تقریباً پشت پرده هر رویه و فرآیندی، یک الگوریتم دست‌اندرکار است. اینترنت هم پیروی همین قاعده و برای داشتن عملکرد صحیح، به الگوریتم‌های مختلف وابسته است. از سرچ در موتورهای جستجو گرفته تا خریدهای اینترنتی، ارسال و دریافت اطلاعات و... همگی وام‌دار کاربرد الگوریتم‌ها در دنیای کامپیوتر هستند. استخوان‌بندی اینترنت وابستگی عمیقی به الگوریتم دارد و در ادامه برخی از کاربردهای الگوریتم در این دنیای گسترده را بررسی خواهیم کرد.

الگوریتم‌های جستجو هر روزه نیازمندی‌های زیادی را در زندگی ما برطرف می‌کنند. نیازی به گفتن ندارد که یکی از مهم‌ترین موتورهای جستجو، گوگل است. مسلماً تجربه کرده‌اید که کافی است چند کلمه از مفهوم مورد نظرتان را در گوگل تایپ کنید تا در کسری از ثانیه، دریایی از اطلاعات را در آن مورد در اختیارتان قرار دهد. گوگل از کاربرد الگوریتم در برنامه نویسی استفاده می‌کند تا این اطلاعات را در اختیار ما بگذارد. بنابراین اهمیت الگوریتم در موتورهای جستجو از جمله گوگل کاملاً روشن است.

مسیریابی در شبکه

در دنیای امروز، شبکه‌های کامپیوتری نقش بسیار مهمی در انتقال اطلاعات ایفا می‌کنند و بدون آن‌ها داشتن بسیاری از امکانات، از جمله اینترنت، میسر نیست. البته برای اینکه بسته‌ها را از مبدأ به مقصد برسانند، باید بهترین مسیر ممکن را بیابند و سپس داده‌ها را از آن مسیر منتقل کنند. شبکه‌های کامپیوتری در این راستا، از الگوریتم‌های مسیریابی استفاده و ترافیک اینترنت را به‌طور مؤثر هدایت می‌کنند؛ منظور از مؤثر در واقع مسیری است که «کم هزینه» باشد. الگوریتم‌های مسیریابی مختلفی وجود دارد، اما برخی از آنها از جمله الگوریتم‌های فازی، اکتشافی و تطبیقی از مشهورترین آن‌ها هستند و امکان ارتباطات مؤثر در شبکه را فراهم می‌آورند.

رمزنگاری

اینترنت در ابعاد زیادی از زندگی ما و فعالیت‌های شخصی، روزمره، شغلی و سازمانی مان‌رخنه کرده و بنابراین مهم است از اطلاعات حساسی که در این میان رد و بدل می‌شود، حفاظت و حراست کنیم تا دستمایه سودجویان نشوند. عدم دسترسی افراد به اطلاعات شخصی و محرمانه یکدیگر از چالش‌های امنیتی مهم در زمینه توزیع و مبادله اطلاعات در بستر اینترنت به شمار می‌رود. رمزنگاری در واقع بهره‌گیری

هایی برنامه نویسی باید متناسب با قدرت پردازنده و ظرفیت حافظه طراحی شوند. یک پردازنده دارای سرعت بی نهایت نیست و حافظه نیز دارای محدودیت هایی است، پس یک الگوریتم باید طوری باشد که از این منابع به صورت بهینه استفاده کند و از لحاظ پیچیدگی زمانی هم کارآمد باشد.

بحث و نتیجه گیری

نقش الگوریتم ها در دنیای فناوری اطلاعات و ارتباطات و علوم کامپیوتری غیر قابل انکار است. الگوریتم از مفاهیم مهمی است که تقریباً در همه ابعاد زندگی ما کاربرد دارد. از ساده ترین کارهای روزمره گرفته تا پیچیده ترین اموری که انجام آنها به لطف فناوری میسر شده است، همه و همه با مفهوم الگوریتم درگیر هستند. الگوریتم در واقع مراحل انجام کارها را به صورت گام به گام در اختیارمان می گذارد و ما را در حل مسائل یاری می کند. اساساً الگوریتم های قاعده مند که از ویژگی های عملیاتی مناسب برخوردار باشند و استفاده بهینه از منابع سیستم داشته باشند، می توانند در زمینه حل مسائل مختلف موثر واقع شود. انواع الگوریتم هایی که در این مقاله بررسی کردیم، هر کدام متناسب با ساختار خود، کاربردهای خاصی دارند که می توانند پیروسه حل مسئله را سریع تر و موثرتر جلو ببرند. آشنایی با ایده های مختلف طراحی الگوریتم می تواند به رشد کارکردی و حرفه ای سازمان ها کمک کند. با توجه به این که سخت افزارها در حال پیشرفت هستند، حوزه طراحی الگوریتم نیز در حال تکامل است. بنا به اهمیت موضوع در این مقاله رده بندی های مختلف عملیاتی و کاربردی را از الگوریتم های رایج حوزه فناوری اطلاعات و ارتباطات ارائه نمودیم.

تعارض منافع

«هیچ گونه تعارض منافع توسط نویسندگان بیان نشده است»

منابع و مأخذ

- [1] Panneerselvam R. Design and Analysis of Algorithms. PHI Learning Pvt. Ltd.; 2016.
- [2] Kabat MR. Design and analysis of algorithms. PHI Learning Pvt. Ltd.; 2013 Aug 21.
- [3] Dimri SC, Malik P, Ram M. Algorithms: Design and Analysis. Walter de Gruyter GmbH & Co KG; 2021 Mar 8.

الگوریتم های ژنتیک

هرچند با دیدن نام الگوریتم ژنتیک ذهنمان به سمت علوم پزشکی و کاربرد الگوریتم در این زمینه می رود، اما این الگوریتم در اصل نوع خاصی از الگوریتم های تکاملی است که از تکنیک های زیست شناسی فرگشتی مانند وراثت، جهش زیست شناسی و اصول انتخابی داروین برای یافتن فرمول بهینه جهت پیش بینی یا تطبیق الگو بهره می برند. الگوریتم های ژنتیک اغلب گزینه خوبی برای تکنیک های پیش بینی بر مبنای رگرسیون هستند.

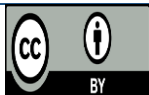
در الگوریتم های ژنتیک ابتدا به طور تصادفی چندین جواب برای مسأله تولید می کنیم. آن را جمعیت اولیه و هر جواب را یک کروموزوم می نامیم. سپس با استفاده از عملگرهای الگوریتم ژنتیک و پس از انتخاب کروموزوم های بهتر، کروموزوم ها را با یکدیگر ترکیب کرده و در آن ها جهش ایجاد می کنیم. در نهایت نیز جمعیت فعلی را با جمعیت جدیدی که از ترکیب و جهش در کروموزوم ها حاصل می شود، ترکیب می کنیم. کاربرد الگوریتم ژنتیک در زمینه های مختلفی است. در این میان می توان به کاربرد الگوریتم ژنتیک در هوش مصنوعی و شاخه یادگیری ماشین (مواردی نظیر دسته بندی، پیش بینی، شبکه های عصبی مصنوعی و...)، کاربرد الگوریتم ژنتیک در کامپیوتر و مواردی نظیر برنامه نویسی خودکار (موارد مرتبط با اتوماتای سلولی و شبکه های مرتب سازی)، کاربرد الگوریتم ژنتیک در زمینه بهینه سازی (مثل بهینه سازی عددی و ترکیبیاتی) و... اشاره کرد [۲۱].

الگوریتم های هوش مصنوعی

الگوریتم های هوش مصنوعی زیرمجموعه ای از الگوریتم هایی هستند که به طور خاص برای پردازش و تفسیر داده ها طراحی شده اند و ماشین ها را قادر می سازند تا رفتار هوشمندانه ای از خود نشان دهند. برخلاف الگوریتم های سنتی که از مجموعه ای از قوانین از پیش تعیین شده پیروی می کنند، الگوریتم های هوش مصنوعی توانایی یادگیری و انطباق از داده ها را دارند و عملکرد خود را در طول زمان بهبود می بخشند. آن ها نیروی محرکه بسیاری از کاربردهای هوش مصنوعی هستند، از تشخیص گفتار و پردازش زبان طبیعی گرفته تا تشخیص تصویر و وسایل نقلیه خودران و بسیاری از موارد دیگر که امروزه با آن سروکار داریم. امروزه غول های تکنولوژی و فناوری از الگوریتم های جدید هوش مصنوعی به وفور در زیرساخت های جود استفاده می کنند که از مهم ترین آن ها می توان الگوریتم های هوش مصنوعی گوگل، مایکروسافت، آمازون و OPEN AI را نام برد.

در پایان لازم به ذکر است، در کل یک دیدگاه های الگوریتمیم به ما کمک می کنند تا تصمیم بگیریم که آیا مسئله مورد نظر بصورت قاعده مند قابل حل است یا خیر. اگر جواب بله بود که چه بهتر، باید به فکر نحوه ارائه یک راه حل سریع و دقیق باشیم. اگر جواب خیر بود، الگوریتم به ما کمک می کند که تصمیم بگیریم آیا می توانیم بخشی از مشکل را حل کنیم یا خیر. البته به این نکته نیز توجه داشته باشید که الگوریتم

- dissertation, State University of New York at Stony Brook).
- [13] Huang CY, Lai CY, Cheng KT. Fundamentals of algorithms. In *Electronic design automation 2009* Jan 1 (pp. 173-234). Morgan Kaufmann.
- [14] Alsuwaiyel MH. *Algorithms: design techniques and analysis*. World Scientific; 2021 Nov 8.
- [15] Erciyes K. *Guide to graph algorithms: sequential, parallel and distributed*. Springer; 2018 Apr 13.
- [16] Erickson J. *Algorithms*. 2023 Jul 21.
- [17] Levitin A, Levitin M. *Algorithmic puzzles*. Oxford University Press; 2011 Oct 12.
- [18] Jena SR, Patro S. *Design and Analysis of Algorithms*. University Science Press, Laxmi Publications, New Delhi; 2018 Jul 21.
- [19] Benoit A, Robert Y, Vivien F. *A guide to algorithm design: paradigms, methods, and complexity analysis*. CRC Press; 2013 Aug 27.
- [20] McConnell J. *Analysis of algorithms*. Jones & Bartlett Publishers; 2007 Nov 2.
- [21] Abdullah AS, Shree TA, Priyadharshini P, Saranya T. *Algorithm and design techniques—a survey*. *Global Journal of Computer Science and Technology*. 2019.
- [4] Gupta P, Agarwal V, Varshney M. *Design and analysis of algorithms*. PHI Learning Pvt. Ltd.; 2012 Dec 9.
- [5] Kimani CJ, Scott J. *Introduction to Algorithms Professional Level*. Finstock Evarcity Publishers; 2023 Aug 7.
- [6] Laaksonen A. *Guide to competitive programming*. Cham: Springer; 2020.
- [7] Basu SK. *Design methods and analysis of algorithms*. PHI Learning Pvt. Ltd.; 2013 Apr 17.
- [8] Alsalmi AA. *A comparative analysis of searching algorithms*. *Journal of University Studies for inclusive Research*. 2020;1(1):1-6.
- [9] Puntambekar AA. *Analysis and Design of Algorithms*. Technical Publications; 2020 Dec 1.
- [10] Horowitz E. *Algorithms, design and classification of*. In *Encyclopedia of Computer Science 2003* Jan 1 (pp. 45-51).
- [11] Borodin A. *Further reflections on a theory for basic algorithms*. In *International Conference on Algorithmic Applications in Management 2006* Jun 20 (pp. 1-9). Berlin, Heidelberg: Springer Berlin Heidelberg.
- [12] Ganapathi P. *Automatic discovery of efficient divide-&-conquer algorithms for dynamic programming problems* (Doctoral



COPYRIGHTS

©2021 The author(s). This is an open access article distributed under the terms of the Creative Commons Attribution (CC BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, as long as the original authors and source are cited. No permission is required from the authors or the publishers.