文章编号：2095-2163(2023)04-0142-05 中图分类号：TP391.9 文献标志码：B

基于 Spark 框架的瀑布型融合旅游推荐系统

杨佳鹏[1]，俎毓伟[1]，纪佳琪[2,3]，陈丽芳[1]
（1 华北理工大学 理学院，河北 唐山 063210；2 河北民族师范学院，河北 承德 067000；
3 河北省文化旅游大数据技术创新中心，河北 承德 067000）

摘 要：目前旅游信息数据量庞大却无法满足用户的特定需求，很多不确定因素导致用户评分出现偏差，使推荐结果不准确且实时性差。鉴于此，本文提出构建基于 Spark 框架的瀑布型融合旅游推荐系统。首先，利用爬虫技术对各大旅游网站景点信息进行爬取和整理，搭建 Spark 框架读取数据并进行数据清洗和预处理，通过 API 将景点地理位置转换为经纬度坐标以便后续可视化；其次，设计 2 个过滤层，第一层采用 SimHash 算法，该算法能够实现海量数据的快速降维处理，有效节约时间。第二层采用余弦相似度算法并利用 TF-IDF 计算词频，进而过滤和更新旅游景点推荐数据库，最终形成反馈给用户的推荐数据库；最后，用户从系统推荐的 Top－N 景点选择自己感兴趣的景点，系统将会对其进行地图可视化，并标注每个省市景点的数量和平均票价，为用户提供智能旅游推荐的完美体验。该系统从用户需求出发，通过分析用户需求文本语义，与旅游数据库进行相似度计算进而获得推荐结果，达到了实时性和准确性的统一，是对旅游推荐系统的补充和完善，具有一定的实用价值。

关键词：瀑布型融合；旅游推荐；局部敏感哈希；余弦相似度计算；特征向量

Waterfall-type integrated tourism recommendation system based on Spark framework

YANG Jiapeng [1], ZU Yuwei [1], JI Jiaqi [2,3], CHEN Lifang[1]
(1 College of Science, North China University of Science and Technology, Tangshan Hebei 063210, China;
2 Hebei Normal University for Nationalities, Chengde Hebei 067000, China;
3 The Technology Innovation Center of Cultural Tourism Big Data of Hebei Province, Chengde Hebei 067000, China)

【Abstract】At present, the large amount of travel information data cannot meet the specific needs of users, and many uncertain factors lead to deviations in user ratings, resulting in inaccurate and poor real-time recommendation results. In view of this, this paper proposes to build a waterfall fusion tourism recommendation system based on Spark framework. Firstly, the paper uses crawler technology to crawl and organize the scenic spots information of major tourism websites, builds a Spark framework to read the data and performs data cleaning and preprocessing, and converts the geographic location of scenic spots into longitude and latitude coordinates through API for subsequent visualization; Secondly, two layers are designed. The first layer uses the SimHash algorithm, which can achieve rapid dimensionality reduction processing of massive data, effectively saving time; the second layer uses the cosine similarity algorithm and uses TF-IDF to calculate word frequency, and then filters and updates tourist attractions recommendations database, thereafter forms a recommendation database that could be fed back to the user; Finally, the user selects the attractions of interest from the Top－N attractions recommended by the system, and the system will visualize them on a map, and mark the number of attractions in each province and city and the average ticket price. It provides users with the perfect experience of intelligent travel recommendation. The system starts from user needs, analyzes the text semantics of user needs, and calculates the similarity with the tourism database to obtain recommendation results, furtherly achieves the unity of real-time and accuracy. It is a supplement and improvement to the tourism recommendation system and has certain practicality value.

【Key words】waterfall fusion; tourism recommendation; local sensitive Hash; Cosine similarity calculation; feature vector

0 引 言

互联网技术的发展引发了信息超额问题，从快速增长数据中找到符合需求的信息需要花费大量的时间和精力，而形成的海量数据也促进了大数据分析以及各种推荐系统的发展。目前随着人们生活水

作者简介：杨佳鹏(2001-)，男，本科生，主要研究方向：机器学习；俎毓伟(1999-)，女，本科生，主要研究方向：机器学习；纪佳琪(1984-)，男，博士，讲师，主要研究方向：机器学习与深度学习、大数据；陈丽芳(1973-)，女，博士，教授，主要研究方向：数据智能处理、大数据、网络应用安全。

收稿日期：2022-05-20

平的提高，外出旅游已成为一种重要的休闲方式。但由于相关知识和经验的不足，游客难以对复杂多样的景点信息做出最优决策，对游客而言，符合用户特征的大数据推荐系统是一种不错的选择[1]。目前，已有研究人员根据用户历史数据及用户评分构建旅游景点推荐系统，但该系统并不能在流数据上运行，并且推荐结果存在很大差异，不符合用户需求，具有局限性。根据用户意向和景点信息的相似度推荐景点，可以更有效地满足用户的特定化需求，对实现智能化生活以及大数据推荐系统具有积极意义。

结合图神经网络和用户情感画像的协同过滤算法是旅游推荐最常用的方法，研究人员结合图神经网络和用户情感画像推动了数据分析与信息挖掘快速发展[2-4]，然而此类算法无法满足用户的特殊需求。图神经网络运用注意力机制[5]获取景点序列中游客的长短期偏好，根据用户历史游玩行为信息推荐景点，而用户的历史游玩信息和长短期偏好容易受到客观环境影响；用户情感画像依据景点评分进行推荐，具有很强的主观性。因此，该方法无法在流数据上使用，并且不能实时应对用户需求的变化。

根据用户评论和评分的协同过滤算法也是一种常用的方法，谭云志等学者[6]提出根据文本评论信息学习项目特征，将不同特征分布及用户偏好同时引入协同过滤推荐系统中。杨家慧学者[7]引入巴氏系数[8]减少共同评分的影响，使用 Jaccard 系数[9]增加协同过滤的共同评分项占比。由于不合理因素可能会造成用户的情感产生一定的偏差，并导致少量用户打出极端的评分，这就会使推荐结果出现明显误差。

针对以上问题，本文提出基于瀑布型融合的旅游推荐系统(WFRA)对用户的想法进行实时分析，因此在面对用户兴趣突然发生更改的情况时，系统可以分析用户最新的想法去重新推荐景点，实时更新模型。在该系统中将构建 2 个过滤层。第一个过滤层采用 SimHash 算法，该算法能够实现大规模文档相似性的精确检测，同时在实际应用中对程序运行速度有所保障，能够对亿万旅游数据进行快速过滤，从而降低后面过滤层的压力；第二个过滤层采用 TF-IDF 和余弦相似度相结合的算法进行过滤，能够进一步把握语义。实验结果表明庞大的旅游数据在经过 2 个过滤层筛选后所推荐得到的结果更具有针对性和精准性。

1　系统设计

1.1　设计思路

在瀑布型融合模型中，构建了 SimHash 算法和余弦相似度算法两个过滤层。经过 SimHash 算法对海量数据进行降维操作，把文本比较次数从最初的上亿次减少到几百万次，大大降低了时间成本，该设计思路为海量数据的精准处理提供了创新性的改善策略。

在首层过滤所得数据的基础上，再选择余弦相似度算法和 TF-IDF 算法进行第二次过滤。通过程序执行窗口，用户可以输入自己想去做的事情，系统会反馈给用户 TOP-100 个推荐结果，其中包含景点所在城市、景点名称、评分、门票价格、销量、以及景点所在的省市区等相关信息。如果用户想要继续了解其中某几个景点的地理位置信息、各省市推荐景点的数量以及平均票价在地图上的分布情况，可以继续执行程序，得到相关数据的可视化结果，设计思路如图 1 所示。

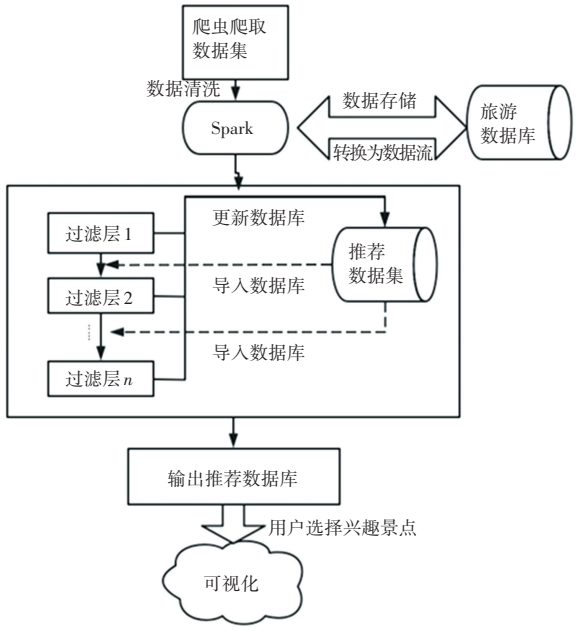

图 1　设计思路图

Fig. 1　Design idea diagram

1.2　系统架构

根据上述设计思路，将上述的系统架构分为 X 层，系统架构如图 2 所示。由图 2 可看到，对系统结构中各层设计，拟展开研究分述如下。

(1)Data 层。Data 层利用 Spark 读取得到的景点数据，存储在原始旅游景点数据库中进行备份，以备后续使用，同时创建一个空白的推荐数据库存储

最终的推荐结果。

(2)处理层。该系统在处理层搭建了 2 个过滤层,根据用户输入的文本,过滤层会对数据流进行筛选,每经过一个过滤层,就把推荐结果存入推荐数据库中,作为下一个过滤层的输入。

(3)输出层。系统在输出层输出最终的推荐结果,用户可以在推荐结果中选取兴趣景点,输出层会对这些兴趣景点做统计分析,并进行地图可视化来展示景点的详细信息与位置。

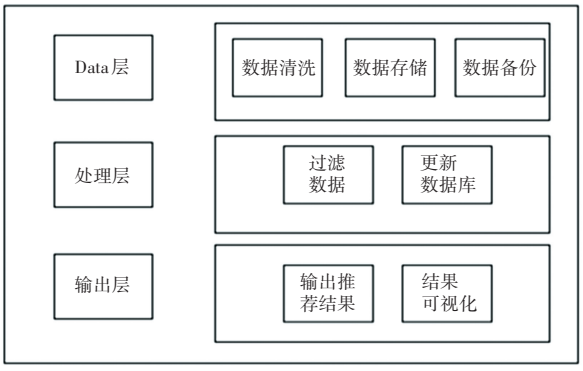

图 2 系统架构

Fig. 2 The architecture of the system

1.3 主要算法

1.3.1 SimHash 算法

Liu 等学者[10]提出 SimHash 算法,其主要思想是对特征向量进行降维,利用 2 个向量的汉明距离来计算相似度。

Yu 等学者[11]证明,SimHash 算法在处理亿万级别的数据时不仅拥有运行速度优势,也具有较高的准确性和鲁棒性。该算法不仅大大降低了运行时间成本,也有效地减少了第二过滤层的工作量,使用户能更快地得到推荐结果。SimHash 流程如图 3 所示。

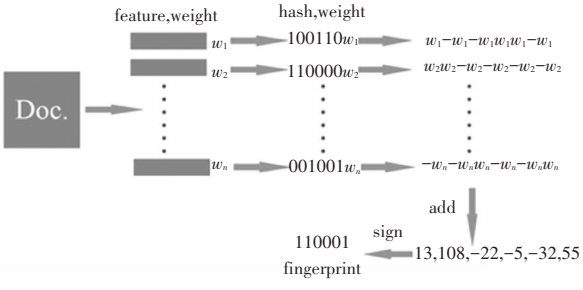

图 3 SimHash 流程

Fig. 3 SimHash process

1.3.2 余弦相似度算法与 IF-IDF

在第一层过滤掉了大量景点后,第二过滤层采用余弦相似度算法,通过提取关键词进一步精简文本信息,从而提升系统的效率。假定用户输入的特征向量 $\boldsymbol{a}$ 为$[x_1, y_1]$,某个景点的介绍、即向量 $\boldsymbol{b}$ 为$[x_2, y_2]$,那么能将余弦定理改写成式(1):

$$\cos\theta = \frac{x_1x_2 + y_1y_2}{\sqrt{x_1^2 + y_1^2} \times \sqrt{x_2^2 + y_2^2}} \tag{1}$$

余弦的这种计算方法对 n 维向量也成立。假定 $\boldsymbol{A}$ 和 $\boldsymbol{B}$ 是2个 n 维向量,这里 $\boldsymbol{A}$ 是 $[A_1, A_2, \cdots, A_n]$,$\boldsymbol{B}$ 是 $[B_1, B_2, \cdots, B_n]$,则 $\boldsymbol{A}$ 与 $\boldsymbol{B}$ 的夹角 θ 的余弦为式(2):

$$\cos\theta = \frac{\sum_{i=1}^{n}(A_i \times B_i)}{\sqrt{\sum_{i=1}^{n}(A_i)^2} \times \sqrt{\sum_{i=1}^{n}(B_i)^2}} = \frac{\boldsymbol{A} \cdot \boldsymbol{B}}{|\boldsymbol{A}| \times |\boldsymbol{B}|} \tag{2}$$

利用式(2),就可得到 $\boldsymbol{A}$ 与 $\boldsymbol{B}$ 的相似度。综上所述,本文在第二层使用的余弦相似度算法和 TF-IDF算法的步骤如下:

(1)用 TF-IDF 算法计算词频向量找出每段文本的关键词。

(2)将每段文本取出的若干个关键词合并成一个文本库,计算每段文本对于该文本库中关键词的词频,并使用相对词频可以避免文本长度的差异。

(3)获得两段文本各自的词频向量。

(4)计算两段文本的余弦相似度。

2 数据获取与预处理

2.1 数据集的获取

本文对原始数据集进行初步整理得到 34 个省市的景点数据。其中包括省市、景点名称、星级、评分、票价、销量、简介以及所在省/市/区,北京市部分景点数据见表 1。

2.2 数据预处理

传统的数据库[12]在应用上存在着一定技术瓶颈,主要表现在 2 个方面:

(1)MySQL 等数据库的数据处置能力有限,数据量的大幅度增加会使 Join、GroupBy、OrderBy 等操作速度受到很大的限制,有可能出现资源成本过高从而导致运行失败的情况;其次,将数据存储转移到分布式系统的代价太大。

(2)无法进行跨数据源的访问。比如,对 Hive Table 和 MySQL 的数据混合进行查询。大多数做法是将数据源进行转移。过程中涉及的技术应用包括 Sqoop[13]、Hive[14]外表等,不过这些技术需要很高的时间成本,且 Sqoop 在对特殊字符的处理中也存在着问题与不足。

Spark 作为开源的大数据处理平台[15],通过将

Resilient Distributed Datasets 以分布式的形式存储在集群的内存中和将计算压力转移到 Hadoop 集群中来提高执行效率。利用 Spark 读取整理好的数据集并进行预处理操作，清洗不符合要求的数值；通过高德地图 API 和 Echarts 将景点的地理位置转换为经纬度坐标，以利于后续进行可视化处理。

表 1　北京市部分景点数据

Tab. 1　Data of some scenic spots in Beijing

省市	景点名称	星级	评分	票价	销量	简介	所在省/市/区
北京	故宫	5A	5.0	58.6	15 277	世界五大宫之首，穿越与您近在咫尺.....	北京·北京·东城区
北京	颐和园	5A	4.1	30.7	9 633	北方也有江南园林.....	北京·北京·海淀区
北京	八达岭长城	5A	4.1	40.0	9 618	一定要看那块“不到长城非好汉”碑.....	北京·北京·延庆县
北京	天坛公园	5A	4.0	16.2	5 300	探寻古代皇帝祭天仪式的奥秘.....	北京·北京·东城区
北京	恭王府	5A	3.7	42.9	5 260	一起去看看和珅家.....	北京·北京·西城区
北京	圆明园	5A	3.8	10.0	3 829	追忆昔日万园之园.....	北京·北京·海淀区
北京	北海公园	4A	3.8	11.9	1 980	让我们荡起双桨，小船儿推开波浪.....	北京·北京·西城区
…	…	…	…	…	…	…	…

3　系统设计与应用

3.1　系统设计

为了能够更具有针对性地向用户推荐旅游景点，开发了基于瀑布型融合的旅游推荐系统。用户输入自己的想法，便可以得到系统推荐的景点，用户还可以从推荐结果中选择兴趣景点，并查看其详细信息和地图上的具体位置。本文主要通过应用不同的算法搭建过滤层对用户期待体验的文本描述和景点信息进行匹配来挖掘用户感兴趣的景点。

3.2　系统应用

(1)打开并运行旅游景点推荐程序，会有文字提示用户输入想要去做的事情，例如：用户输入：我突然想去泡个温泉，最好是专门的中心。用户在输入想做的事情后按下回车键，系统会反馈给用户 100 个推荐结果。根据推荐结果可知，该算法反馈给用户推荐景点的针对性较强，具有较高的精准性。

在得到 TOP-100 个景点的数据后，将其按照省市分类得到各省市兴趣景点的数量及票价的平均值，如图 4、图 5 所示。

(2)用户还可以选择兴趣景点所对应的序号，得到其在地图上具体地理位置的分布情况，以及清晰、直观的 3D 可视化结果。

(3)用户可以选择继续获取各省市兴趣景点的数量以及平均票价的相关信息，最终以地图和柱状图的形式进行可视化。如果想了解某个省市的兴趣景点分布数量以及平均票价，只需将鼠标移动至该省市的柱子上，就会动态展示相关信息。其中，*lng* 和 *lat* 分别表示该省市的经纬度，*alt* 表示该省市所包含兴趣景点的个数，*value* 表示该省市兴趣景点的平均票价。

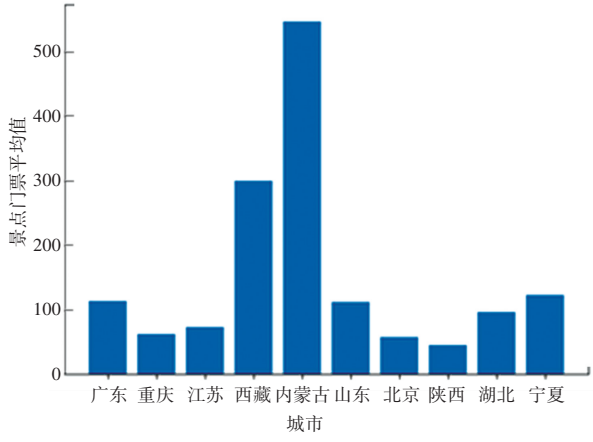

图 4　各省市兴趣景点的平均票价

Fig. 4　Average ticket prices for attractions of interest in various provinces and cities

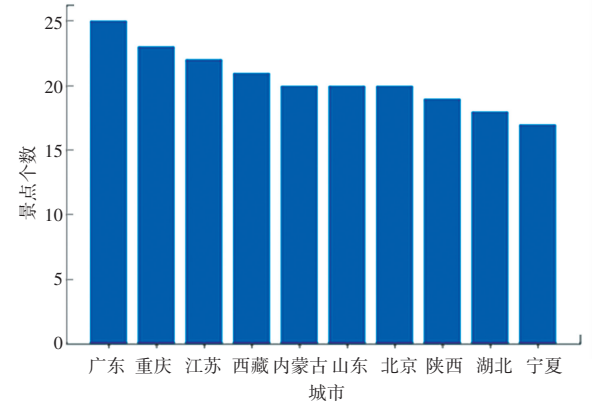

图 5　各省市兴趣景点的数量

Fig. 5　Number of attractions of interest in each province and city

4　结束语

本文提出的基于瀑布型融合的旅游景点推荐系统在测试中具有很强的针对性和较高的精准度。其本质是根据用户的想法，对各旅游景点的信息进行过滤，在每一个过滤层除去不符合条件的景点，筛选

出更符合用户意愿的景点。该系统不仅能够更好地满足用户的个性化需求,还能灵活地应对用户兴趣更改的情况,达到实时分析的效果,在某种程度上实现智能化旅游推荐,使用户能够在任何时候都能得到符合当下内心想法的推荐结果,使旅游推荐更加智能化。由于在处理海量旅游景点数据过程中,第一过滤层采用 SimHash 算法降低了整个算法的时间成本,并使程序的运行速度得到明显提升,整个算法的结构体系具有很强的实用性。

在后续的研究中,可搭建并行过滤层分别针对不同特征属性进行过滤筛选,并根据数据集的大小、算法的时间复杂度、系统运行效率,以及利用景点与景点之间的相似度进行横向推荐等多个角度进行改进,从而获得更加智能的推荐结果。

参考文献

[1] CASTILLO L, ARMENGOL E, ONAINDIA E,et al. Samap : an user-oriented adaptive system for planning tourist visits[J]. Expert Systems with Applications, 2008, 34(2) : 1318-1332.

[2] 陈源鹏,古天龙,宾辰忠,等. 融合图表示学习和序列挖掘的景点推荐方法[J]. 计算机工程与设计,2020,41(12):3563-3569.

[3] 孙振强,罗永龙,郑孝遥,等. 一种融合用户情感与相似度的智能旅游路径推荐方法[J]. 计算机科学,2021,48(S1):226-230.

[4] 史睿瑶. 基于协同过滤算法的旅游推荐系统的设计与实现[J]. 电脑知识与技术,2020,16(35):64-66.

[5] 周末, 宋玉蓉, 宋波, 等. 融合自注意力机制的 D-BGRU 文本分类模型[J/OL]. 微电子学与计算:1-9[2021-09-29].http://kns.cnki.net/kcms/detail/61.1123.TN.20210914.1630.018.html.

[6] 谭云志,张敏,刘奕群,等. 基于用户评分和评论信息的协同推荐框架[J]. 模式识别与人工智能,2016,29(04):359-366.

[7] 杨家慧,刘方爱. 基于巴氏系数和 Jaccard 系数的协同过滤算法[J]. 计算机应用,2016,36(07):2006-2010.

[8] 程苗,陈海龙,孙海娇,等. 基于 BM25 聚类与巴氏系数相似度改进的推荐算法[J]. 黑龙江大学自然科学学报,2020,37(05):610-616.

[9] 谢红. 基于词频比的改进 Jaccard 系数文本相似度计算[J]. 内江科技,2021,42(08):27-28.

[10] LIU Longcheng, ZHANG Jianzhong. Inverse maximum flow problems under the weighted Hamming distance[J]. Journal of Combinatorial Optimization, 2006, 12(4):395-408.

[11] YU Yi, HU Zijian, ZHANG Yuzhu. Rearch on large scale documents deduplication technique based on Simhash algorithm [C]//Proceedings of the First International Conference on Information Sciences, Machinery, Materials and Energy. [S.l.]: Atlantis Press,2015:1225-1228.

[12] 陈漫红. 数据库原理与应用技术(SQL Server2008)[M]. 北京:北京理工大学出版社,2016.

[13] 杨彬. Sqoop 数据收集与入库系统的应用[J]. 电子制作,2017(21):38-39.

[14] 唐燕,刘仁权,王苹. 基于 Hadoop 的高校大数据平台的设计与实现[J] . 信息技术,2017 (12):105-109.

[15] ZAHARIA M, CHOWDHURY M, FRANKLIN M J, et al. Spark: Cluster computing with working sets[C]//Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing. Berkeley, CA, USA:USENIX Association, 2010:10.

(上接第 141 页)

[6] AHMED F, DALIA Y, HEGAZY R, et al. An efficient capuchin search algorithm for allocating the renewable based biomass distributed generators in radial distribution network[J]. Sustainable Energy Technologies and Assessments, 2022,53 (34): 547-554.

[7] KANIPRIYA M, HEMALATHA C, SRIDEVI N, et al. An improved capuchin search algorithm optimized hybrid CNN-LSTM architecture for malignant lung nodule detection[J]. Biomedical Signal Processing and Control,2022,78(63): 460-472.

[8] 焦珊. 基于改进群智能优化的太阳能光伏系统参数辨识方法研究[D]. 温州:温州大学, 2020.

[9] AYANG A, WAMKEUE R, OUHROUCHE M, et al. Maximum likelihood parameters estimation of single - diode model of photovoltaic generator[J]. Renew Energy, 2019,130(19):111-121.

[10] ORTIZ-CONDE A, S′ANCHEZ F J G, MUCI J. New method to extract the model parameters of solar cells from the explicit analytic solutions of their illuminated I-V characteristics[J]. Solar Energy Mater Solar Cells, 2006,90(3):352-361.

[11] 张伟伟, 陶聪, 范岩,等. 改进回溯搜索算法解决光伏模型参数识别问题[J]. 计算机应用, 2021, 41(04):1199-1206.

[12] MIRJALILI S, MIRJALILI S M, LEWIS A. Grey wolf optimizer [J]. Advances in Engineering Software, 2014, 69(73): 46-61.

[13] GHASEMI M, RAHIMNEJAD A, HEMMATI R, et al. Wild geese algorithm: A novel algorithm for large scale optimization based on the natural life and death of wild geese[J]. Advances in Engineering Software, 2021, 383(35): 238-253.

[14] MIRJALILI S. The ant lion optimizer [J]. Advances in Engineering Software, 2015, 83(28): 80-98.

[15] KHISHE M, MOSAVI M R. Chimp optimization algorithm[J]. Expert Systems with Applications, 2020,149(37): 51-67.

[16] MIRJALILI S. Moth - flame optimization algorithm: A novel nature - inspired heuristic paradigm [J]. Knowledge - Based Systems, 2015, 89(54): 228-249.

[17] MIRJALILI S. SCA: A sine cosine algorithm for solving optimization problems[J]. Knowledge-Based Systems, 2016, 96(16): 120-133.

[18] YAO Xin, LIU Yong, LIN Guangming. Evolutionary programming made faster[J]. IEEE Transactions on Evolutionary Computation, 1999, 3(2):82-102.

[19] CHEN Zicong, WU Lijun, LIN Peijie, et al. Cheng S. Parameters identification of photovoltaic models using hybrid adaptive Nelder-Mead simplex algorithm based on eagle strategy [J]. Applied Energy, 2016,182(36): 47-57.

[20] EASWARAKHANTHAN T, BOTTIN J, BOUHOUCH I, et al. Nonlinear minimization algorithm for determining the solar cell parameters with microcomputers[J]. International Journal of Solar Energy, 1986, 4(1):1-12.