Hindawi
Journal of Electrical and Computer Engineering
Volume 2022, Article ID 5190979, 13 pages
https://doi.org/10.1155/2022/5190979

Research Article

Status Analysis and Future Development Planning of Fitness APP Based on Intelligent Word Frequency Analysis

Siqiu Wang and Shengze Wang

College of Mechanical Engineering, Donghua University, Shanghai 201620, China

Correspondence should be addressed to Siqiu Wang; wsq@mail.dhu.edu.cn

Received 21 March 2022; Revised 21 April 2022; Accepted 29 April 2022; Published 25 May 2022

Academic Editor: Wei Liu

In order to analyze the current situation analysis and future development plan of fitness APP, this paper analyzes the current situation of fitness APP combined with the intelligent word frequency analysis algorithm. In order to deal with the problem that the existing qPMS algorithm is very time-consuming to discover motifs in the large DNA sequence dataset *D*, a structural motif discovery algorithm for large DNA sequence datasets, SMS, is proposed. Moreover, this paper performs motif discovery by mining substrings with high frequency in the input sequence. Through data research, it can be seen that the fitness APP based on intelligent word frequency analysis proposed in this paper has good results. On this basis, this paper analyzes the problems existing in the existing fitness APP and provides suggestions for its future development.

1. Introduction

Sports and fitness APP refers to the third-party application software that is based on the mobile phone terminal and can help users record and share their sports and fitness data and can also provide personalized suggestions. This type of software focuses on providing sports and fitness content for different user needs, so it generally does not include sports news and games. According to different content and user needs, sports and fitness APPs can be divided into four types. The first is a fitness APP based on sports such as running and fitness, which mainly monitors sports data in real time and provides functions such as online sharing and friend PK. At the same time, in order to enhance user stickiness, it also provides certain incentive measures. The second is a venue APP that integrates offline venue resources based on mobile Internet technology to provide users with venue booking services. This type of APP can intelligently match the nearest venues that meet their needs for users and provide reservation services. The third is a fitness APP mainly based on online fitness guidance. This type of APP mainly attracts users through personalized customized courses, online fitness interaction with real people, and dynamic sharing of fitness results. The fourth is an O2O fitness APP based on personal trainers. This type of APP mainly attracts upstream coach resources and downstream user resources by means of free and subsidy and provides a platform for upstream and downstream users to directly connect. Then, when the upstream and downstream resources accumulate to a certain level and a large user viscosity is formed, it will realize profits in terms of coach remuneration, user tuition, and value-added services. The characteristics of sports fitness APP are as follows. One is that it is not limited by time and space. As long as the mobile phone is in hand, the exercise data can be recorded at any time, and the data can be uploaded to the cloud storage, which is convenient for users to compare and formulate their own exercise plan. Second, it is more scientific and convenient than traditional exercise. Fitness apps can not only provide users with more targeted fitness advice, but also recommend sports equipment and recipes for users. Therefore, it is more scientific and convenient than traditional self-exploration or personal trainer-style exercise. The third is to greatly expand the social circle.

Generally speaking, according to the different functions of sports and fitness apps, they can be divided into three categories, namely, daily records, training plan formulation, and fitness guidance. However, with the homogenization of more and more sports apps, many sports apps provide similar service functions to meet the service needs of different sports enthusiasts. The second category is the planning category. This type of app provides functions such as exercise reminders and exercise optimization. The last category is fitness guidance apps, which are relatively more popular among sports enthusiasts. This type of app mainly provides sports guidance services. This type of app provides a wealth of sports fitness teaching videos and also provides professional guidance functions. At the same time, it can customize different sports guidance content according to the hobbies and needs of different groups, so that ordinary people can also enjoy professional sports experience.

This paper combines the intelligent word frequency analysis algorithm to analyze the current situation of fitness APP and studies its future development plan to improve the development effect of fitness APP.

2. Related Work

In the Internet age, sports and fitness culture has gradually entered people's attention. Due to the accelerated pace of life, people's work and life pressure is too great, rest time cannot be guaranteed, various physical and psychological diseases are increasing, and physical health has become a problem that the public must face and pay attention to in daily life. In addition, the working environment and methods of modern people have cut people's time. Most people have no way to spare a lot of time for fitness exercise and can only find ways to use the fragmented time for exercise and fitness [1]. When people generally lack time and energy for systematic physical exercise, and the shortness and uncertainty of idle time limit the effect of people's physical exercise, a convenient and efficient fitness method is needed [2]. Under this circumstance, sports and fitness APPs have a broad space for growth. Their advantage is that they can provide people with exercise and fitness guidance, record relevant data, etc. anytime, anywhere. As long as people are in their spare time, they can open the APP to exercise. Once you have something to do, you can click to stop, and the APP will automatically record the stop moment for the user. The next time the user opens the APP, he can directly continue the last training session [3]. This kind of exercise instruction that is not limited by time and space is favored by young people. The APP development team paid attention to the current social status, combined with people's life rhythm and attention to health, and created a sports and fitness APP. They captured the current consumer psychology of the audience and fit people's inner needs and completed the creation of the brand and the accumulation of users within the time [4]. Users with different characteristics have different needs when using sports and fitness APPs. Therefore, the design team will spend a lot of time and energy to design detailed courses, which can meet the various training requirements of users to the greatest extent. A complete lesson plan is included to ensure that users receive the best training guidance in each training session. This design not only ensures the practicability of the software, but also achieves true personal customization [5]. For example, after the "keep" registration and login is successful, the user enters personal information such as his gender, age, height, and weight, thus completing the preliminary targeted service, and then the APP requires the user to make an estimate of his exercise experience, select a better fitness guide for themselves according to your level of exercise, and perform efficient fitness exercises while avoiding or reducing sports injuries. This customized service for different individuals can reduce the time it takes for users to find suitable courses, and for those who lack knowledge of sports and fitness, it also eliminates the trouble of facing dazzling training courses but not knowing how to choose [6]. In addition, the highly targeted exercise intensity and difficulty also ensure the scientific nature of exercise and fitness. As a social group, socializing is the most basic need of everyone. Most of the users who use APP for exercise and fitness want to share their daily exercise situation. Based on this, the sports and fitness APP is specially created for the majority of users. The community sets up corresponding sections for users to share their fitness experience [7]. Every day, various fitness experts post news and share their fitness experience. The content shared includes text, pictures, and videos in various and vivid forms. Users can leave comments below the news to communicate with each other, and they can also follow their favorite people and share them with them. Not only are Sports and fitness APPs used for sports and fitness, but their internal sharing communities often form a small society, and information spreads from interpersonal communication to intergroup communication [8]. Exercise and fitness itself is a boring behavior. When people keep repeating the same action, they will gradually get bored with it. At the same time, doing exercise requires people's physical strength and energy, which is why many people do not like exercise. In the community, there are thousands of users who are insisting on exercising, which will make people feel a sense of group belonging, reduce their resistance to exercise or fitness, and gradually find the fun of exercise and fitness. After a complex action or a goal set by oneself is achieved, one can share their joy and experience with others [9].

In addition to the product itself, sports and fitness APPs persuade others to change their sports attitudes and behaviors, but also affect user behavior through the interpersonal interaction of product users [10]. For example, the APP named "Le Power" has a sharing section in the application, which brings together people with the same health needs. Users can share their daily progress, and they can also view the popular shares and learn their strengths. At the same time, self-motivation is generated [11]. If a user often checks the information posted by popular and other users, it will invisibly affect the individual's attitude towards sports and change their cognitive mechanism of sports and fitness. It will prompt them to spontaneously generate the desire to integrate into the group and therefore urge themselves to exercise, and gradually they will like the changes brought about by exercise [12]. Sports and fitness APPs can grasp the psychology of the audience, use the stimulation of the various senses of the APP itself, add some incentives, cooperate with other users to interact and share content, and carry out effective persuasion. The above types of effective persuasion designs are conducive to user changing one's own sports attitude and living habits, actively participating in sports groups, and using APP to assist sports, producing obvious persuasion effect, and finally achieving the purpose of promoting products [13].

3. Intelligent Word Frequency Analysis

For a fixed (l, d) and single sequence length n, this section analyzes the effect of the number of input sequences t and the proportion of sequences containing motif instances in all sequences q on the time performance of the qPMS algorithm. The analyzed algorithms mainly focus on the suffix tree-based pattern-driven qPMS algorithm and the sample-based pattern-driven qPMS algorithm.

For a pattern-driven qPMS algorithm based on a suffix tree, the algorithm needs to construct a suffix tree index structure of t input sequences of length n, where each edge marks a non-empty substring in the input sequence. For any node v in the tree, str_v represents the string formed by concatenating the substrings on the path from the root to v. If v is a leaf, then str_v represents a suffix of the input sequence; otherwise str_v represents the common prefix of the suffixes corresponding to all leaf nodes under node v. The suffix tree has exactly t n leaf nodes, which means that after t n suffixes of the input sequence the number of the sequence in which str_v occurs exactly is stored at each node v in the tree. Using a vector of t bits, the i-th bit is set to 1 if the string appears in the i-th sequence; otherwise, it is set to 0.

In addition to the suffix tree, the algorithm also uses a pattern tree. It is a complete quadtree of depth l that represents all patterns on the alphabet $\sum$ of lengths ranging from 1 to l. The algorithm performs a depth-first search on the pattern tree. When searching for the pattern p corresponding to a node, use the suffix tree index to obtain the union of the exact occurrence sequence numbers of all d neighbors of p. Thus, a sequence in which p occurs approximately with a mismatch of the largest d positions is obtained. If the number of sequences in which p approximately occurs is greater than or equal to qt and the length of p is less than l, then continue searching for pattern $pb\,(b \in \sum)$. Otherwise, the algorithm performs a pruning strategy on the subtree representing the node of p and finally outputs all patterns of length l that can span at least qt sequences.

The smaller q is, the more computation time is required to drive the qPMS algorithm based on the suffix tree pattern [14]:

$$\mathrm{Pr} = \sum_{i=qt}^{t} \binom{t}{i} \left(1 - \left(1 - P_d\right)^{(n-l+1)}\right)^{i}. \tag{1}$$

For sample-pattern-driven qPMS algorithms, their temporal performance mainly depends on the number of candidate motifs generated. The algorithm uses all h-tuples in t-qt + h reference sequences to generate candidate motifs; that is, this takes into account all possible combinations of h reference sequences in t-qt + h reference sequences, and the number is denoted as N_{com} , and it is calculated by formula (2). For a particular algorithm, the value of $h\,(h \geq 1)$ is generally fixed. Therefore, N_{com} is mainly affected by t and q. Obviously, with the increase of t or the decrease of q, N_{com} will increase; that is, more candidate motifs need to be generated, which requires more computation time.

$$N_{\mathrm{com}} = \binom{t - qt + h}{h} = \frac{\prod_{i=1}^{h} (t - qt + i)}{h!}. \tag{2}$$

The algorithm obtains the k mismatch counts for all w-mers in the input sequence. The algorithm is given a w-mer x, where $\mathrm{count}_k(x)$ represents the k mismatch count of x, which is calculated by (3). Among them, I_y is an indicator variable, which is 1 when $d_H(x, y) \leq k$; otherwise, it is 0.

$$\mathrm{count}_k(x) = \sum_{y \in_w D} I_y. \tag{3}$$

The k mismatch count is based on the counting operation on the FM-Index (that is, counting the exact number of occurrences of a given w – mer in the data set D, which is denoted as $\mathrm{count}(y)$ to solve $\mathrm{count}_k(x)$). $\mathrm{count}_k(x)$ is converted to the sum of the exact number of occurrences of multiple w-mers, which is calculated by .

$$\mathrm{count}_k(x) = \sum_{y \in B_k(x)} \mathrm{count}(y). \tag{4}$$

FM-Index is a self-indexing data structure. We assume that all suffix ranking intervals prefixed by the string φ in the input sequence are denoted as $[L_\varphi, R_\varphi]$. The process of calculating $\mathrm{count}(y)$ using FM-Index is briefly described as follows [15].

The algorithm traverses the w characters of y sequentially from right to left (that is, backward search). When accessing the i-th $(1 \leq i \leq w)$ character $y[i]$, on the basis of the interval $[L_{\varphi'}, R_{\varphi'}]$ of $\varphi' = y[i+1 \cdots w]$, the interval $[L_\varphi, R_\varphi]$ of $\varphi = y[i \ldots w]$ is obtained in $O(\log|\Sigma|)$ time. Finally, $\mathrm{count}(y)$ equals $R_y - L_y + 1$. Thus, the time to compute $\mathrm{count}(y)$ is $O(w \log|\Sigma|)$.

FM-Index can compute the counts of individual w-mers very efficiently. However, if the counts for each w-mer y in $B_k(x)$ are independently computed to obtain $\mathrm{count}_k(x)$, then there will be many common suffixes that are repeatedly searched backwards. For example, we assume to compute a 3-mer $x = ACG$ with a 1 mismatch count; when independently counting the occurrences of these four elements ACG, CCG, GCG, and TCG in $B_1(x)$, their common suffix CG will be searched backward 4 times. Moreover, the goal of this step is to get the k-mismatch counts for all w-mers in the input sequence, so that the number of repetitions can be quite large. Therefore, in order to achieve this goal efficiently, it is necessary to design a method that minimizes the number of repetitions of backward search operations.

Based on these considerations, as shown in Figure 1, the k-mismatch count precomputes the $\mathrm{count}(y)$-values of all w-mer y in the input sequence and stores it in a table T of size 4^w. Among them, $T[i]$ stores the count (y) value of w-mer y of $stn(y) = i$. Then, the k mismatch count $\mathrm{count}_k(x)$ for a given

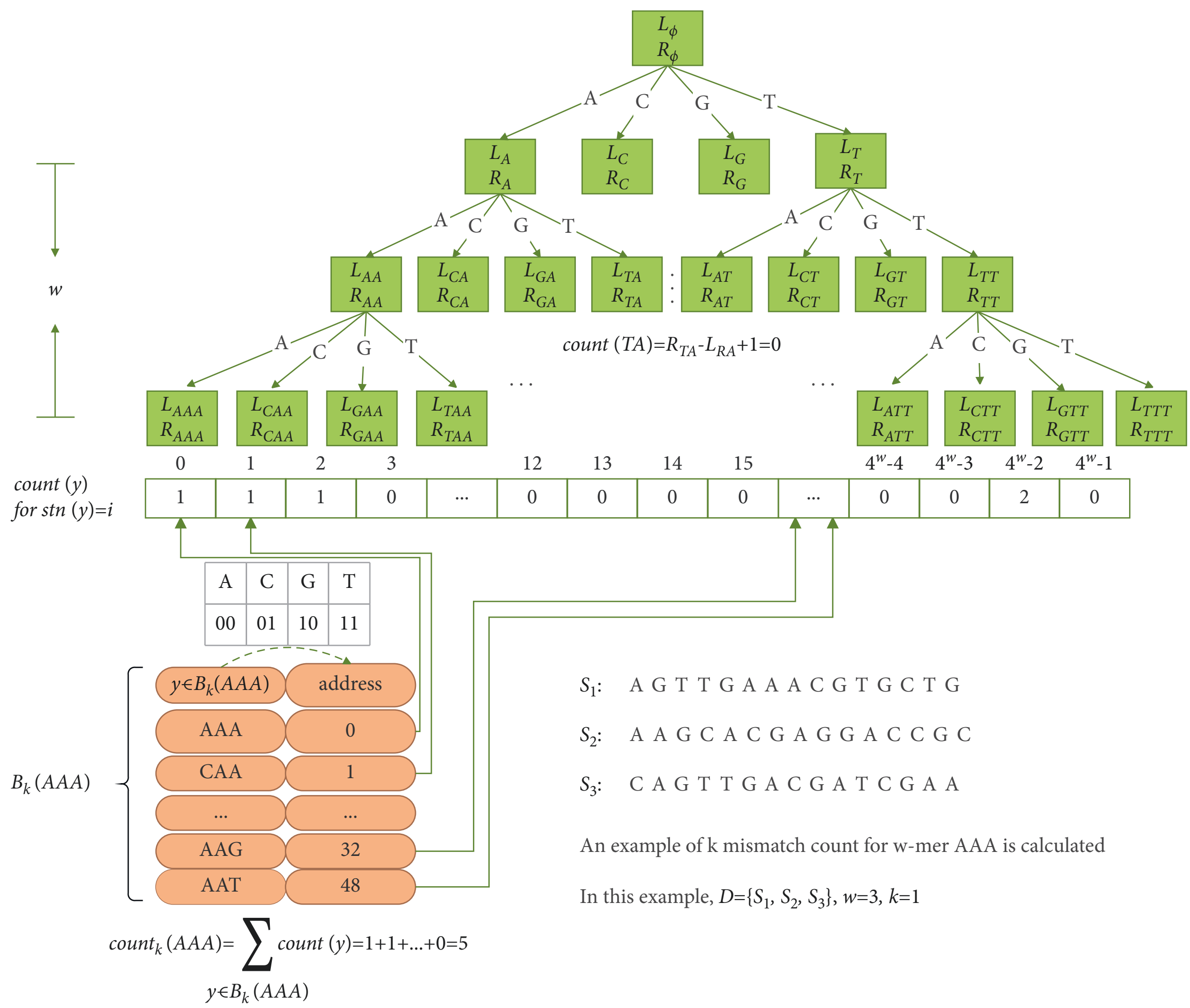

Figure 1: Schematic diagram of k-mismatch count.

w-mer x is obtained by querying table T $|B_k(x)|$ times and accumulating the $T[\text{stn}(y)]$ value for each y in $B_k(x)$ [16].

To ensure good spatial and temporal performance, the k mismatch counts must be chosen appropriately for w and k. Except for constructing FM-Index, it is not affected by w and k; the space complexity of k mismatch count is $O(4^h)$, which is mainly used to store table T. Equation (5) gives the computation time T_{count} for the k-mismatch count, which mainly depends on two parts. One is to establish a table T, and the time is recorded as T_1. In the worst case, each node in the quadtree with depth w is visited in $O(\log|\Sigma|)$ time. The second is the lookup table T; the time is recorded as T_2, that is, the k mismatch counts of all w-mers in the sequence of t bars of length n are calculated through the lookup table T. Among them, the algorithm needs to look up the table $(B_k(w-\text{mer}))$ times to calculate the k mismatch count $\text{count}_k(w-\text{mer})$ of a w-mer.

$$\begin{aligned} O(T_{\text{count}}) &= O(T_1+T_2) = O\left(\sum_{i=0}^{w} 4^w \log|\Sigma| + \text{tn}|B_k(w-\text{mer})|\right) \\ &= O\left(\sum_{i=0}^{w} 4^w \log|\Sigma| + \text{tn}\sum_{i=0}^{k}\binom{w}{i}(|\Sigma|-1)^i\right). \end{aligned} \tag{5}$$

The overall process is shown in Figure 2. The initial threshold f is set as the sum of the frequency N_r of random occurrences in the input sequence when w-mer is the background sequence and the frequency N_m of random occurrences in the input sequence when it is a motif instance, and the computation of N_r and N_m is referenced. For any two w-mers that overlap, if the length of the overlap is greater than $w/2$, the algorithm combines the two w-mers into a substring. Furthermore, some substrings are synthesized by more than two w-mers (for example, the substrings in s_t in Figure 2).

Next, the method of splitting substrings is described as follows.

The algorithm builds an attraction table of size $|\varphi|-l+1$ for each substring φ, which is denoted by $\text{attractTable}_\varphi$. To explain the attraction table, we first define the distance $\text{dis}(\varphi,\varphi')$ of the two substrings φ and φ' as the Hamming distance of the pair of l-mers with the smallest Hamming distance in φ and φ', which is calculated by formula (6). On this basis, the value attract $\text{Table}_\varphi[i]$ of the i-th element of the attraction table is calculated by formula (7), where $\min \text{Pos}_\varphi(\varphi')$ represents the positions of all possible l-mers in φ that produce $\text{dis}(\varphi,\varphi')$.

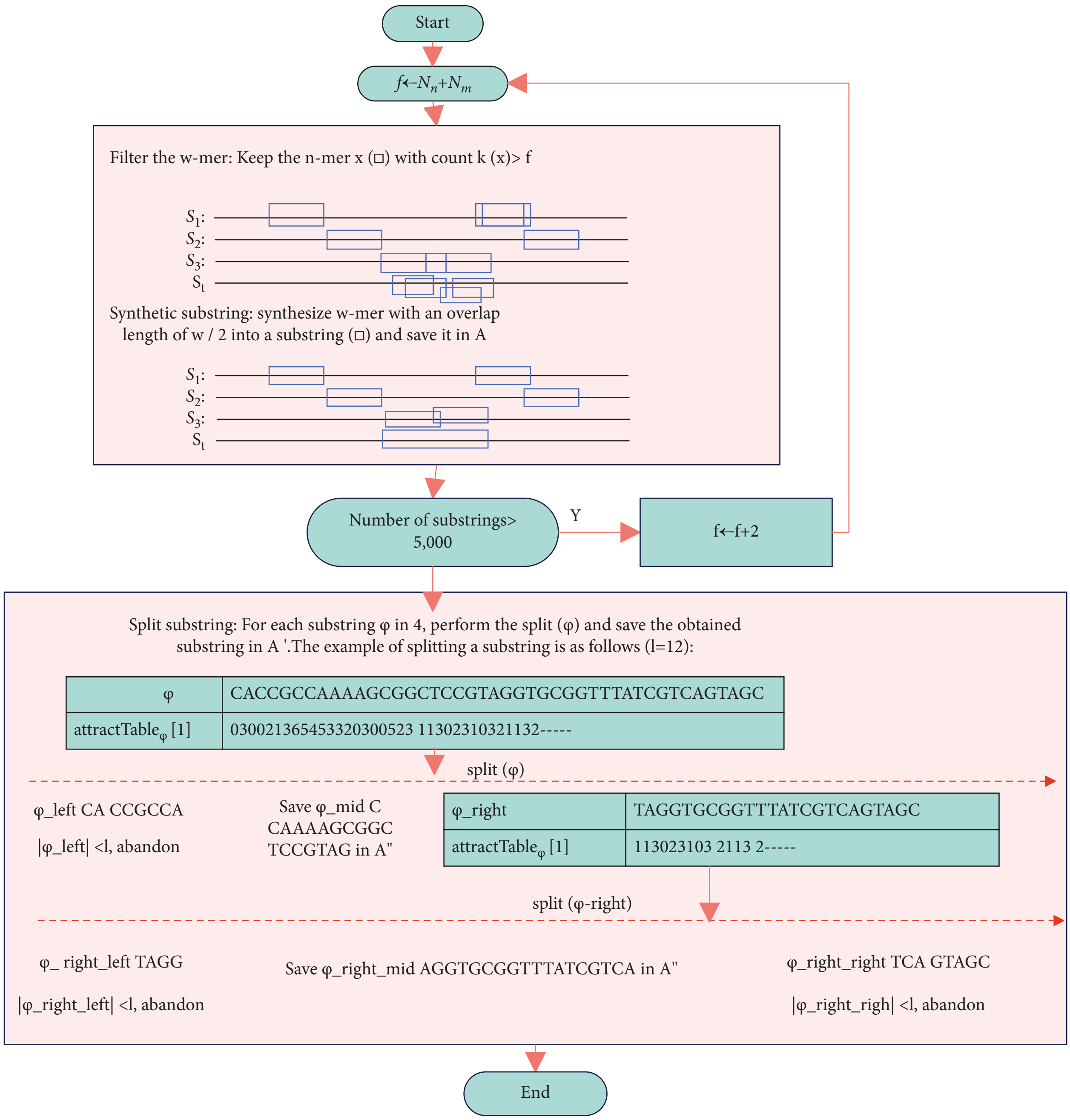

Figure 2: Schematic diagram of high-frequency substring acquisition.

$$\operatorname{dis}(\varphi, \varphi') = \min_{x \in \varphi, x' \in \varphi'} d_H(x, x'), \tag{6}$$

$$\text{attract Table}_\varphi[i] = \left|\left\{\varphi' : \varphi' \in A - \{\varphi\}, i \in \min \operatorname{Pos}_\varphi(\varphi')\right\}\right|, \tag{7}$$

$$\min Pos_\varphi(\varphi') = \arg \min_{1 \le i \le \varphi\varphi - l + 1} \operatorname{dis}(\varphi[i \cdots i + l - 1], \varphi'). \tag{8}$$

The algorithm uses clustering to distinguish substrings corresponding to different motifs. Clustering adopts AP clustering, which has the advantages of fast calculation speed, the ability to automatically determine the number of clusters, and the ability to obtain cluster centers. For each obtained cluster, the cluster center (the centroid of the cluster) is regarded as the substring most similar to the motif, which will help to filter out randomly occurring high-frequency substrings. During clustering, the calculation method of similarity $\operatorname{sim}(\varphi, \varphi')$ of two elements (that is, two substrings φ and φ') is shown in formula(9), where $\operatorname{dis}(\varphi, \varphi')$ is calculated by formula (7) [17].

$$\operatorname{sim}(\varphi, \varphi') = \begin{cases} -\operatorname{dis}(\varphi, \varphi'), & \text{if} \operatorname{dis}(\varphi, \varphi') \le 2d, \\ -\operatorname{dis}(\varphi, \varphi') \times 10, & \text{otherwise.} \end{cases} \tag{9}$$

After clustering, the resulting clusters need to be merged. Because the AP algorithm generally obtains multiple clustering results, there may be multiple clusters corresponding

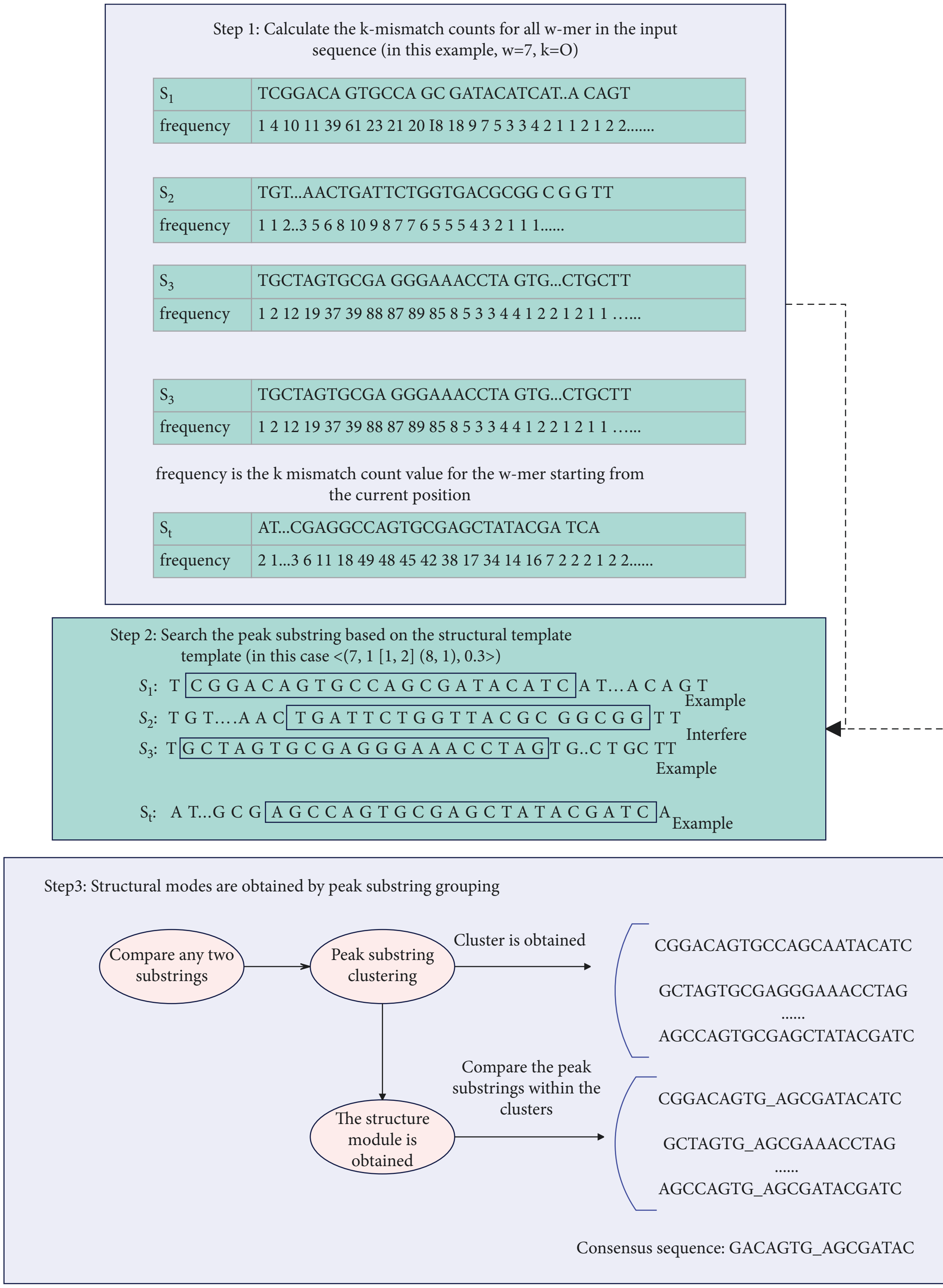

Figure 3: Schematic diagram of SMS algorithm.

to the same motif. The discriminant conditions for merging two clusters c and c' ($|c| \geq |c'|$) are as follows.

The algorithm uses the cluster center φ of c to compare each substring φ' in c'. If the number of φ' in c' satisfying $\mathrm{dis}(\varphi, \varphi'') \leq d$ is significantly larger than the number $P_d|c'|$ in the random case (by default, according to formula (10)), then c and c' are merged. The merging process adopts a greedy strategy.

$$\left|\{\varphi' : \varphi' \in c', \mathrm{dis}(\varphi, \varphi') \le d\}\right| > P_d|c'| + 20\%|c'|. \tag{10}$$

Most of the motif sets and induced sets can be distinguished using statistical methods, and their weights (calculated by ((11) are significantly different. Specifically, they determine two thresholds a_m and a_d. Then, for a given bounded set D', if $w(D') \le a_m$ or $w(D') \ge a_d$, then D' is judged to be a motif set or induced set. If $w(D') \le a_m$, then D' is judged to be a set of motifs. Otherwise, the algorithm iteratively prunes 10 substrings from D' until it is determined to be a motif set. In the experiment, the following method is used to determine the threshold a_m: the algorithm randomly generates 1000 samples containing $|D'|$ motif instances, calculates the weight of each sample, obtains the mean μ and standard deviation σ, and sets $a_m = \mu + \sigma$.

$$w(D) = \sum_{\varphi, \varphi' \in D} \mathrm{dis}(\varphi, \varphi'). \tag{11}$$

For each output sample sequence set $D', t' = |D'|, q'$ selection is discussed as follows: although rule 3 is used to ensure that D' corresponds to a set of motif instances with maximum likelihood, q' cannot be set to 100. On the one hand, the rule 3 test D' is the statistical method used. On the other hand, when calculating the distance between two substrings φ and φ', the value is the Hamming distance of a pair of l-mer x and x' with the smallest Hamming distance in φ and φ'. In this way, when the distance between φ and φ' is calculated, the position of the minimum Hamming distance in φ may not be fixed, which also affects the accuracy of the result.

The high-frequency w-mer is obtained. A single monomorphic instance may correspond to multiple overlapping high-frequency w-mers. An instance of a structural motif might be a sequence fragment enriched in high-frequency w-mers, which can be referred to as a peak substring. Therefore, the algorithm can search for peak substrings in the input sequence set according to the structural motif template to obtain potential structural motif instances.

Because some peak substrings may be structural motif instances, and some peak substrings may also be randomly overrepresented substrings, these substrings need to be clustered, and it is expected that there is a cluster that contains only structural motif instances. Then, the substrings in the clusters are aligned to obtain structural motifs.

Based on this, as shown in Figure 3, SMS consists of the following three steps. First, the algorithm computes the k mismatch counts for all w-mers in the input sequence. Second, the algorithm searches for peak substrings according to the structural motif template. Finally, the algorithm obtains structural motifs by grouping peak substrings.

For a given structural motif template, the length L of the peak substring is set to a fixed value. Structural motif instances are not of fixed length due to variable-length intervals between adjacent monomorphs. L' represents the maximum length of a structural motif instance. In order for a peak substring to cover a complete structural motif instance, L should be greater than or equal to L'. In addition, w-mers in D that overlap with both ends of the structural motif instance may also appear frequently. If L is set directly to L', these substrings may cause the resulting peak substrings to be shifted in position relative to the true structural motif instance. In order to solve this problem, it is necessary to add a safety threshold f to L' to get L, and f is 5 by default in the experiment. Based on this, L is calculated by (12), where l_i is the length of the i-th monomorphic body, and g_i' is the maximum value of the interval between the i-th and i+1 monomorphic bodies [18].

$$L = \sum_{i=1}^{\beta} l_i + \sum_{i=1}^{\beta-1} g_i' + f. \tag{12}$$

For each sequence $s_i\,(1 \le i \le t)$ in D, a peak substring is obtained as follows:

The algorithm traverses all substrings $s_i[j \cdots j + L - 1]$ $(1 \le j \le |s_i| - L + 1)$ of length L in s_i in turn, calculates the score of $s_i[j \cdots j + L - 1]$, and then obtains the substring with the largest score as the peak substring in s_i. At most, qt sequences in D contain instances of structural motifs. Therefore, the top qt peak substrings with the largest scores at the end are selected.

The main idea of calculating the $s_i[j \ldots j + L - 1]$ score is to use a recursive method. The algorithm iterates over all combinations of unimodal positions that satisfy the separation constraint. For each combination, a score was obtained by accumulating the k-mismatch counts of the w-mers located at the respective monotome positions. Finally, the maximum score is taken as the score of $s_i[j \cdots j + L - 1]$.

$$\mathrm{maxPOS} = L - \sum_{i=1}^{\beta} l_i - \sum_{i=1}^{\beta-1} g_i + 1. \tag{13}$$

Some of the symbols used are defined below. $[s_i, e_i]$ represents the interval in which the instance of the i-th monotome may appear in φ and φ', which is calculated by (14). We define a triplet $\langle\langle p_1, \ldots, p_i\rangle, \langle p_1', \ldots, p_i'\rangle, \mathrm{dis}\rangle$: $p_1, \ldots, \rangle, p_i\rangle$ and $\langle p_1', \ldots, p_i'\rangle$ to mark the positions of the first i instances of the monotome in φ and φ', respectively. dis represents the sum of the Hamming distances of the two instances of each of the first i unimorphs, which is calculated by (15). If $i = \beta$, then this triple represents an alignment of φ and φ' and the distance of φ and φ'.

$$[s_i, e_i] = \begin{cases} \left[0, L - \sum_{k=2}^{\beta} l_i - \sum_{k=1}^{\beta} g_i\right], & \text{if } i = 1, \\ \left[\sum_{k=1}^{i-1} l_i + \sum_{k=1}^{i-1} g_i, L - \sum_{k=i+1}^{\beta} l_i - \sum_{k=i}^{\beta} g_i\right], & \text{if } 1 < i < \beta, \\ \left[\sum_{k=1}^{\beta-1} l_i + \sum_{k=1}^{\beta-1} g_i, L\right], & \text{if } i = \beta, \end{cases} \tag{14}$$

$$\mathrm{dis} = \sum_{j=1}^{i} d_H\left(\varphi\left[p_j, p_j + l_j - 1\right], \varphi\left[p_j' \ldots p_j' + l_j - 1\right]\right). \tag{15}$$

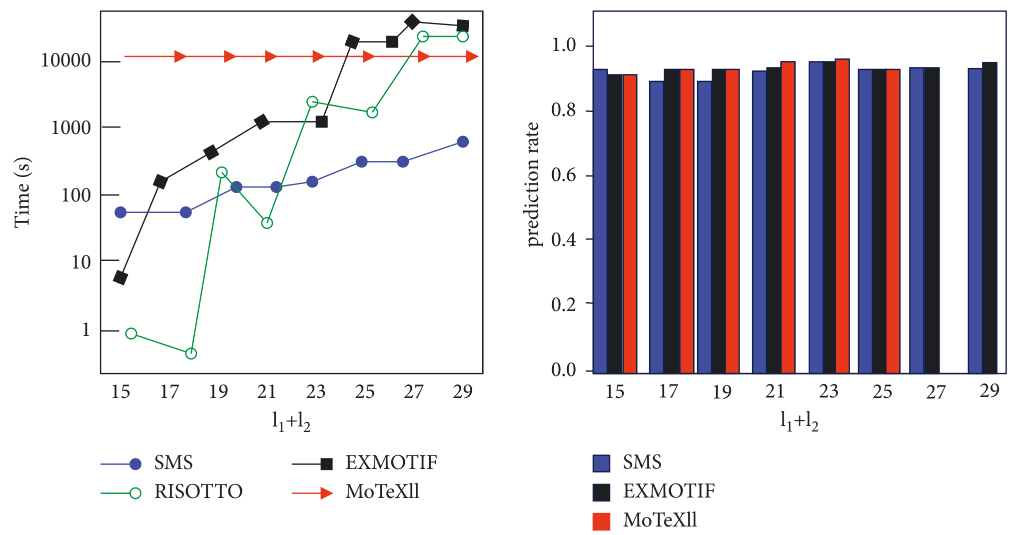

Figure 4: Experimental results at different $l_1 + l_2$.

Clustering uses AP clustering, which has the advantages of fast calculation speed, the ability to automatically determine the number of clusters, and the ability to obtain cluster centers. For each obtained cluster, the cluster center was considered as the substring most similar to the structural motif. We give two peak substrings as φ and φ', and their similarity $\mathrm{sim}(\varphi, \varphi')$ is calculated by formula (16), where $\mathrm{dis}(\varphi, \varphi')$ represents the minimum distance between φ and φ', which is obtained by algorithm 7.

$$\mathrm{sim}(\varphi, \varphi') = \begin{cases} -\mathrm{dis}(\varphi, \varphi'), & \text{if} \, \mathrm{dis}(\varphi, \varphi') \leq \sum_{i=1}^{\beta} d_i, \\ -\mathrm{dis}(\varphi, \varphi') \times 10, & \text{otherwise.} \end{cases} \tag{16}$$

In each cluster, the cluster center φ is used to align with each substring φ' in the class, and the tag value (tag) is added to the position corresponding to the best alignment in φ. The method to obtain the comparison is as follows: for each substring φ' in the class, the best alignments of φ and φ' are queried, and the positions of each monomorphic instance of φ are found, and the same marker values are added to these positions, and the values are calculated by

$$\mathrm{tag} = \sum_{i=1}^{\beta} l_i - \mathrm{dis}(\varphi, \varphi'). \tag{17}$$

Figures 4–8 show the experimental results of testing different algorithms on five datasets, respectively. Through experiments, it can be concluded that compared with other structural motif discovery algorithms the SMS algorithm can find implanted structural motifs faster while maintaining similar accuracy. SMS is not sensitive to input parameters. Faced with different parameters of different test data sets, the algorithm can search for structural motifs within a few minutes (except for the case of $\beta(2)$).

As shown in Figure 4, with the increase of $l_1 + l_2$, the running time of the left figure is shown as follows.

The runtime of SMS grows linearly and insignificantly, processing a use case with $l_1 + l_2$ equal to 29 in just 98 s. RISOTTO's runtime grows intermittently exponentially. There is a small drop in time after each exponential increase (for example, 17 to 15 drops, 21 to 19 drops). This is due to the fact that the constraints of the first unimodal in the structural motif are stronger than the previous one. EXMOTIF is also growing exponentially. However, its growing trend is more pronounced, and a segfault is reported when $l_1 + l_2$ is greater than or equal to 27. The running time of MoTeX-II is basically maintained at about 11000 s, and the time performance is not affected by parameters. The recognition accuracy of the right image is as follows: MoTeX-II does not participate in the comparison, and the other three algorithms have comparable accuracy. SMS finds that the result is shifted by one bit when $l_1 + l_2$ is equal to 17, but the accuracy is not greatly affected.

As shown in Figure 5, as the number of sequences changes, the runtime aspect of the left graph is as follows: SMS always finds the right result in a very short time. RISOTTO's execution time is second only to SMS. This is because RISOTTO is implemented based on a suffix tree, and it is not sensitive to the number of sequences. The execution time of EXMOTIF and MoTeX-II is greatly affected by t, where EXMOTIF reports a segmentation fault when t is greater than or equal to 8000, and MoTeX-II cannot get results within 48 hours when $t \geq 6000$. The recognition accuracy of the right picture is as follow: MoTeX-II does not participate in the comparison, RISOTTO and EXMOTIF are slightly better than SMS when t is equal to 1000, and the accuracy is the same in other cases.

As shown in Figure 6, as q varies from 0.2 to 0.9, the runtime aspect of the left graph is as follows: SMS can get results within 100 s in the worst case, and the time increases linearly. RISOTTO can get results within 300 s in the worst case, and the time also increases linearly. The running time of EXMOTIF decreases with the increase of q, and it takes 4784 s in the worst case when q is equal to 0.2. MoTeX-II

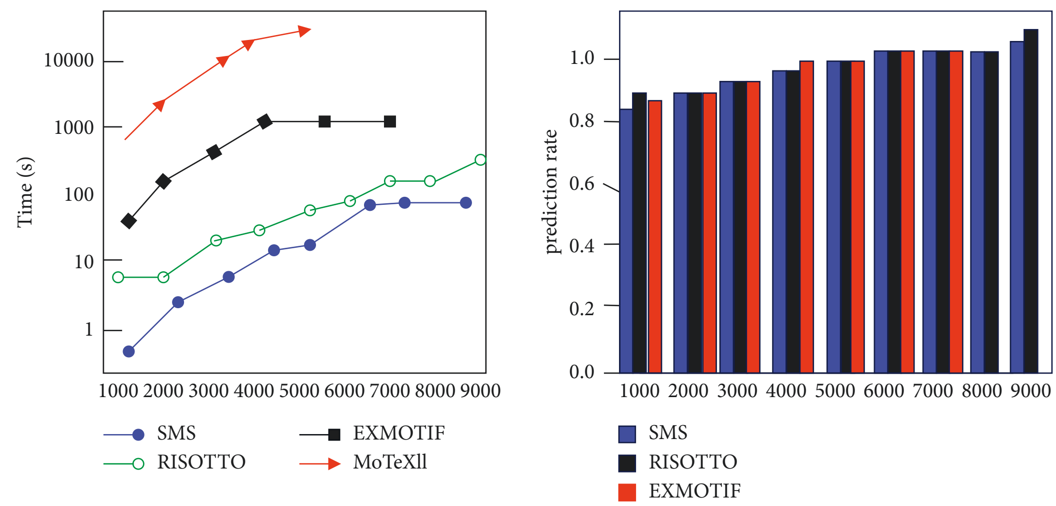

Figure 5: Experimental results at different t.

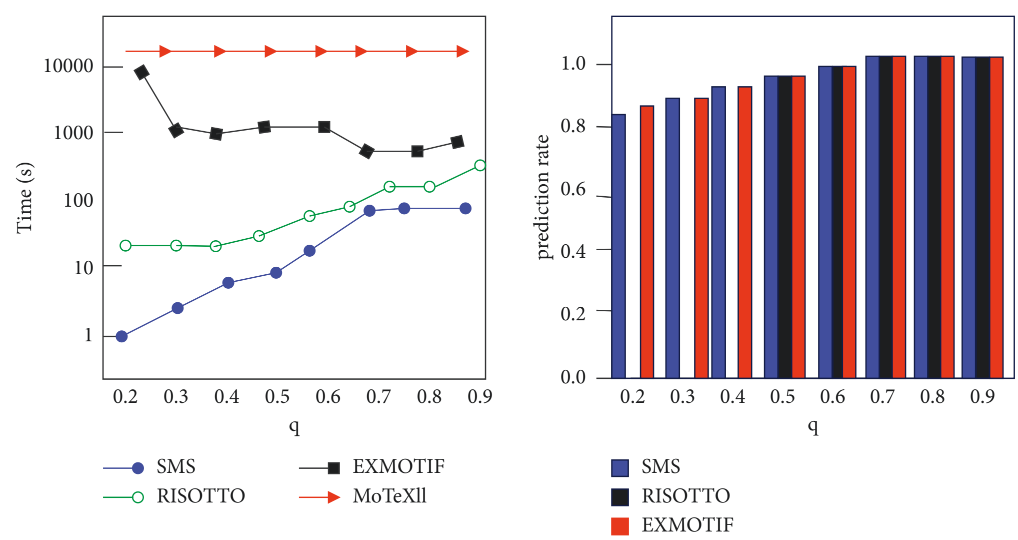

Figure 6: Experimental results on different q.

running time is not affected by q. The recognition accuracy of the right image is as follows: when q is less than 0.5, the accuracy of RISOTTO is zero, and in the rest of the cases, the three algorithms (except MoTeX-II) have the same accuracy.

As shown in Figure 7, as the gap increases, the runtime aspect of the left graph is as follows: EXMOTIF and MoTeX-II are not affected by gap and the running time remains unchanged. SMS and RISOTTO time grow linearly; the latter is slightly stronger than the former. The recognition accuracy of the right image is as follows: except for MoTeX-II, the accuracy of the other three algorithms is basically the same, and the accuracy of SMS is slightly lower when the gap is equal to [1, 30].

As shown in Figure 8, as β goes from 2 to 5, the runtime aspect of the left graph is as follows: SMS can get β equal to 5 at 1562 s. RISOTTO can get β equal to 5 in 2465 s. EXMOTIF and MoTeX-II can only get results when $\beta \leq 3$, EXMOTIF is faster than MoTeX-II. When $\beta > 3$, the former reports a segfault, the latter cannot get results within 48 hours. The recognition accuracy in the right image is as follows: MoTeX-II does not participate in the comparison, EXMOTIF has zero accuracy, and both SMS and RISOTTO can find results with the same accuracy.

4. Status Analysis and Future Development Planning of Fitness APP Based on Intelligent Word Frequency Analysis

The fundamental purpose of "smart sports" is to meet people's needs for physical fitness and leisure and entertainment, and to respond in a timely manner according to the needs. In traditional physical exercise, the exercise needs are solved by finding a sports venue or simply finding a park for running, brisk walking and other sports. However, there will always be situations where the sports venue cannot be

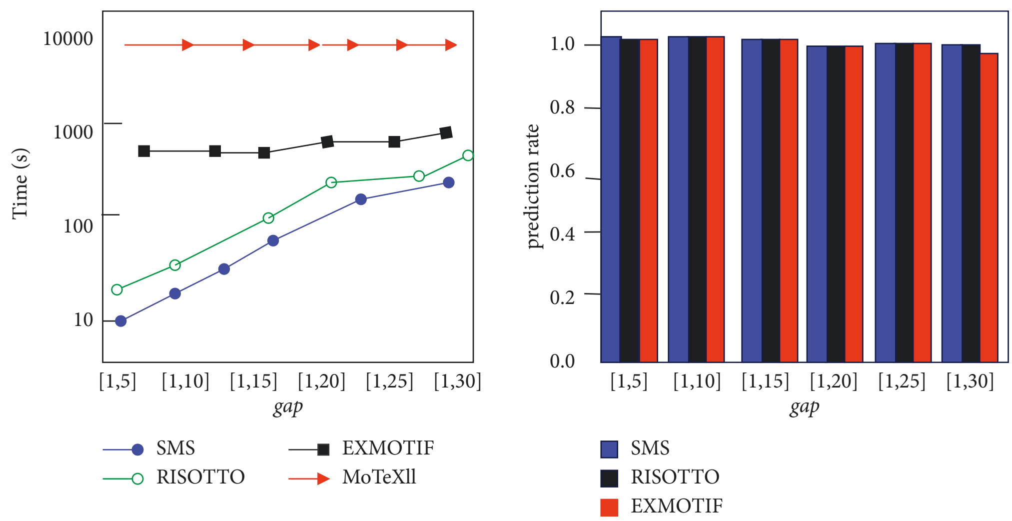

Figure 7: Experimental results on different gaps.

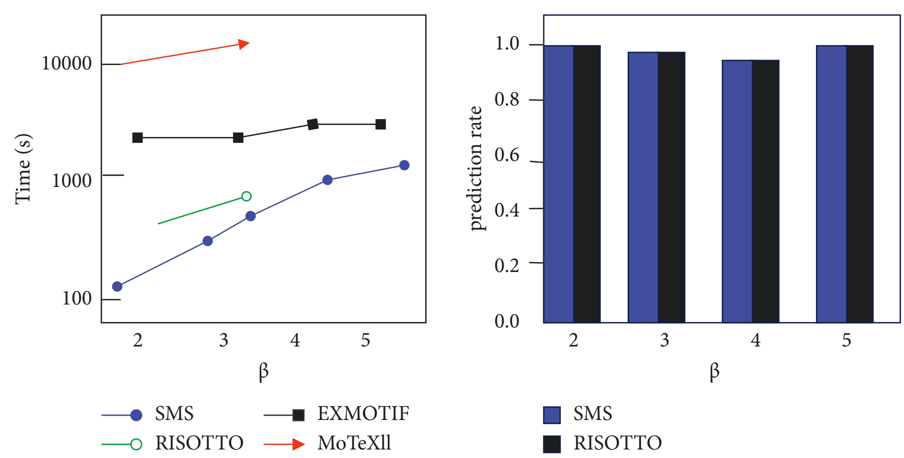

Figure 8: Experimental results on different β.

found, the sports venue is occupied, or the weather is bad and there is no venue. Moreover, not every sport is suitable for different people, some people are more suitable for running, some people are more suitable for cycling, and so on. The development framework of "smart sports" is shown in Figure 9.

Through word frequency analysis, the demands of users after experience are basically manifested in three aspects: instinct layer, behavior layer, and reflection layer. In terms of the instinct level, the audience of fitness apps is mostly the general public. The ultimate goal of users is to enhance physical fitness and master fitness culture. Today's fitness apps blindly pursue gorgeous interfaces and settings, Flow, and volume while ignoring the most basic layers. At the behavioral level, the way and mode of physical exercise are the basis of user behavior. Users pay attention to the practicability, simplicity, and operability of sports products. Although innovation is very important for the sustainable and stable development of products, the practicability of product upgrading needs to be considered, the guidance after product update is not enough, and the connection with user experience is not considered in design innovation. In terms of reflective layer, there are "high cost," "annoying, I get stuck playing games," "junk software, why there is no celebrity fitness interactive message function," "lack of sports injury and sports medical information," and other feedback problems. There is not enough connection between user needs and usability design, perceptual design, interaction, and emotional design.

The smooth use of fitness apps is an important guarantee for user experience. This requires fitness APP to follow-up user feedback in a timely manner and update the software to adapt to the user's individual needs. The lack of correlation between user needs and the functional content of fitness APPs is mainly reflected in the following aspects: some fitness users cannot obtain valuable services such as sports protection and health knowledge, and the application does not take into account the needs of users with low fitness literacy; There is no intuitive feedback, the function does not take into account the user's stage goals, personality characteristics, and environmental conditions; the module is too broad, the lack of payment guidance mechanism leads to

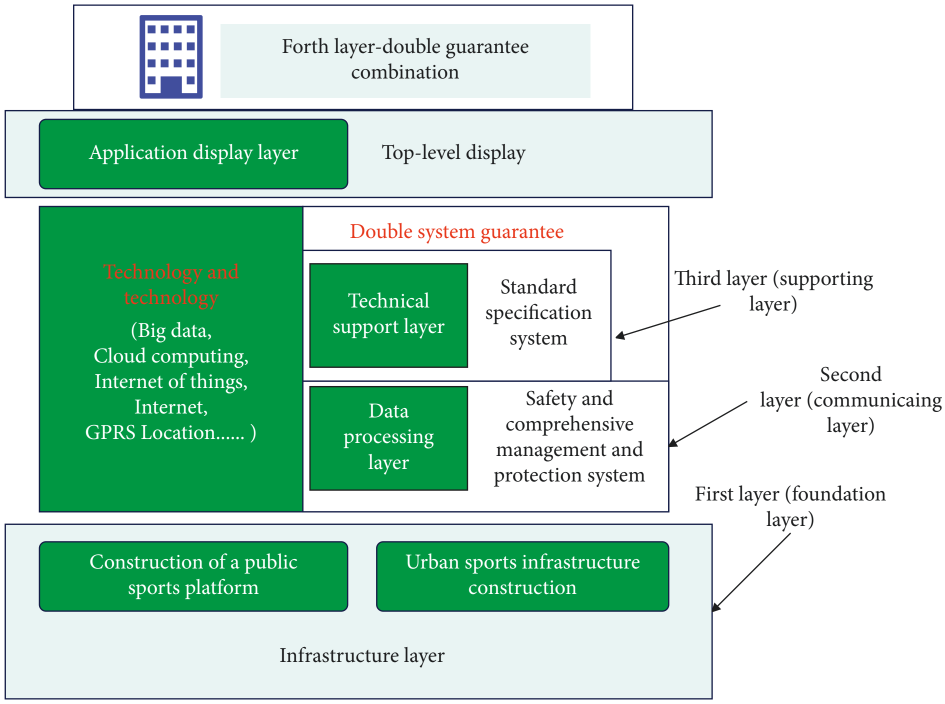

Figure 9: Development framework of “smart sports”.

limited access; the problem word frequency “update will not be used” repeatedly refers to the content of fitness APP. In this regard, it is generally believed that the new version of the APP is only an extension of the old version of the APP. More original content and new fitness models should be created, rather than the mutual use of each APP content. The content update has not reached a practical level. Relevant running data is received in a timely manner, and the fitness plan cannot be adjusted in real time for the individual. In addition, the algorithm recommendation leads to overloading the fitness information of the APP, and it is difficult to customize the personal fitness method by observing the comprehensive situation of the user’s topic selection, favorite collection, fitness problem investigation, and physical condition.

Based on the above research, the following recommendations are made.

Improve the development of fitness functions and enhance users’ “action + sensory” experience. (1) Further improve the APP’s existing experience functions such as target management, interactive display, follow-up courses, customer service, and other experience functions, and strengthen the practical application of sports encyclopedia, training plan, and live video, to expand the scale of activities such as red envelope rewards, offline “encounters,” online communities, and offline running groups. (2) The fitness APP needs to further optimize the AI algorithm to improve the user’s exercise experience and the accuracy of exercise needs and increase the integration of fitness text, images, audio, video, and other forms with the media forms of virtual reality, mixed reality, and augmented reality. Under 5 G, the development of smart detection of motion status of watch or other wearable devices will be further improved to improve convenience. Increase the integration of IQ media + APP, improve the intuitive feedback of personality characteristics, emotional expression, body senses, etc. in sports, so as to consolidate the effectiveness of interaction with learning. (3) Set up cultural digital modules such as health knowledge and health methods to promote cultural publicity functions such as health information, healthy life tips, fitness history, and fitness experts. Sports doctors tailor-made fitness plans for each person provide online consultation and offline service models and improve the APP’s function of combining sports and medical care. (4) The online exercise mode of the APP requires the intelligent connection of offline actions. The APP platform is synchronized with the data of national fitness venues, gyms, fitness parks, and fitness trails to achieve convenient fitness nearby and real-time feedback of fitness information.

Develop smart fitness social networking and improve users’ “relevance + emotion” experience. The word frequency analysis reflects users’ strong willingness to exercise and socialize. Creating a “relevance + emotion” fitness social environment is the key to improving user experience. (1) Fitness APP needs to increase cooperation with social media such as WeChat, Weibo, QQ, Douyin, and Toutiao, comprehensively connect network users, enhance the influence of fitness culture, and make fitness communication more convenient and effective. Increase the reward services for interactive users such as online voting, fitness surveys, running reviews, sports live broadcasts, and brand topics. (2) Build fitness leaders. On the one hand, build grassroots talents, fitness stars, and community opinion leaders, and actively participate in the guidance of community topics and community activities. Li Huizhen)’s social activities to realize the transformation of emotional capital into fitness motivation. (3) Create an O2O fitness social model, improve online discussions and offline fitness activities, offline equipment experience online purchases, and online exercise plans to formulate offline fitness interactions, etc. Fitness APP needs to strengthen its interaction with offline stores and sports running groups, body sculpting hall, fitness plaza, and other cooperation, through the APP to the online communication crowd extended to offline collective fitness activities. At the same time, strengthen the connection between fitness APPs and mass leisure and fitness activities such as square dance and marathon, and finally create exclusive brand tourism and events that belong to fitness APPs, to further enhance the offline experience relevance and participation emotion of fitness APPs. (4) Attach importance to users' social and emotional expressions, improve privacy mechanisms, and protect users' personal information; establish a knowledge exchange area, popularize scientific fitness knowledge and fitness methods, use APP to expand health education, and improve national health literacy; pay attention to fitness users' psychology; create transmedia storytelling and gamification design.

5. Conclusion

Sports and fitness apps can use big data mining technology to provide users with user groups similar to their sports hobbies. Users can exchange their own experiences about venues, equipment, and exercise online anytime and anywhere, which greatly expands their social circles. At present, there are many sports and fitness apps in the mobile app market, and they are rich in functions and provide differentiated services for people with different service needs. In recent years, with the rapid development of sports fitness apps, the functions of sports fitness apps have become more and more abundant. In addition, the functions of major sports and fitness platforms are also developing towards homogeneity, so as to meet the fitness needs of different people and improve market competitiveness. This paper combines the intelligent word frequency analysis algorithm to analyze the current situation of fitness APP and study its future development plan. Through the experimental research results, it can be seen that the fitness APP based on intelligent word frequency analysis proposed in this paper has a good effect. On this basis, this paper analyzes the problems existing in the existing fitness APP and provides suggestions for its future development.

Data Availability

The labeled dataset used to support the findings of this study is available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This work was supported by Donghua University.

References

[1] G. Hua, L. Li, and S. Liu, "Multipath affinage stacked—hourglass networks for human pose estimation," *Frontiers of Computer Science*, vol. 14, no. 4, Article ID 144701, 2020.

[2] K. Aso, D. H. Hwang, and H. Koike, "Portable 3D human pose estimation for human-human interaction using a chest-mounted fisheye camera," in *Proceedings of the Augmented Humans Conference 2021*, pp. 116–120, New York, NY, United States, Februry 2021.

[3] D. Mehta, S. Sridhar, O. Sotnychenko et al., "Vnect: real-time 3d human pose estimation with a single rgb camera," *ACM Transactions on Graphics*, vol. 36, no. 4, pp. 1–14, 2017.

[4] S. Liu, Y. Li, and G. Hua, "Human pose estimation in video via structured space learning and halfway temporal evaluation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 7, pp. 2029–2038, 2019.

[5] S. Ershadi-Nasab, E. Noury, S. Kasaei, and E. Sanaei, "Multiple human 3d pose estimation from multiview images," *Multimedia Tools and Applications*, vol. 77, no. 12, Article ID 15573, 2018.

[6] X. Nie, J. Feng, J. Xing, S. Xiao, and S. Yan, "Hierarchical contextual refinement networks for human pose estimation," *IEEE Transactions on Image Processing*, vol. 28, no. 2, pp. 924–936, 2019.

[7] Y. Nie, J. Lee, S. Yoon, and D. S. Park, "A multi-stage convolution machine with scaling and dilation for human pose estimation," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 13, no. 6, pp. 3182–3198, 2019.

[8] A. Zarkeshev and C. Csiszár, "Rescue method based on V2X communication and human pose estimation," *Periodica Polytechnica: Civil Engineering*, vol. 63, no. 4, pp. 1139–1146, 2019.

[9] W. McNally, A. Wong, and J. McPhee, "Action recognition using deep convolutional neural networks and compressed spatio-temporal pose encodings," *Journal of Computational Vision and Imaging Systems*, vol. 4, no. 1, p. 3, 2018.

[10] A. Bakshi, D. Sheikh, Y. Ansari, C. Sharma, and H. Naik, "Pose estimate based yoga instructor," *International Journal of Recent Advances in Multidisciplinary Topics*, vol. 2, no. 2, pp. 70–73, 2021.

[11] S. L. Colyer, M. Evans, D. P. Cosker, and A. I. T. Salo, "A review of the evolution of vision-based motion analysis and the integration of advanced computer vision methods towards developing a markerless system," *Sports medicine-open*, vol. 4, no. 1, pp. 24–15, 2018.

[12] I. Sárándi, T. Linder, K. O. Arras, and B. Leibe, "Metrabs: metric-scale truncation-robust heatmaps for absolute 3d human pose estimation," *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol. 3, no. 1, pp. 16–30, 2021.

[13] A. Azhand, S. Rabe, S. Müller, I. Sattler, and A. Heimann-Steinert, "Algorithm based on one monocular video delivers highly valid and reliable gait parameters," *Scientific Reports*, vol. 11, no. 1, Article ID 14065, 2021.

[14] J. Xu and K. Tasaka, "[Papers] keep your eye on the ball: detection of kicking motions in multi-view 4K soccer videos," *ITE Transactions on Media Technology and Applications*, vol. 8, no. 2, pp. 81–88, 2020.

[15] Q. Yu, D. Wei, and H. Huo, "Samselect: a sample sequence selection algorithm for quorum planted motif search on large dna datasets," *BMC Bioinformatics*, vol. 19, no. 1, p. 228, 2018.

[16] J. Bhombe, A. Jethwa, A. Singh, and T. Nagarhalli, "Review of pose recognition systems," *VIVA-Tech International Journal for Research and Innovation*, vol. 1, no. 4, pp. 1–8, 2021.
[17] N. Vallabhaneni and P. Prabhavathy, "The analysis of the impact of yoga on healthcare and conventional strategies for human pose recognition," *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, vol. 12, no. 6, pp. 1772–1783, 2021.
[18] J. J. Liu, J. Newman, and D. J. Lee, "Using artificial intelligence to provide visual feedback for golf swing training," *Electronic Imaging*, vol. 33, no. 6, pp. 3211–3216, 2021.