



Method for Re-finding Mobile Phone Documents Based on Feature Knowledge Graph

Jing Zhang¹, Yukun Li^{1,2}(✉), Yan Zhang¹, and Yunbo Ye¹

¹ Tianjin University of Technology, Tianjin 300384, China

² Tianjin Key Laboratory of Intelligence Computing and Novel Software Technology, Tianjin, China

Abstract. People often need to find documents that previously saved on their mobile phones. When searching a document, sometimes people cannot recall the exact name of it, but can remember some objects in it, such as the figures, tables or formulas in the documents, the source of the documents, or related topics. And the existing mobile phone document management software cannot associate the content of the document with itself. Therefore, how to extract document information into attributes tags to associate them with documents and how to re-find the expected documents based on these tags are practical and valuable. This paper proposes a method for re-finding mobile phone documents based on feature knowledge graph. First, we use four features to describe the documents and extract tags to form attributes. Then we link the attributes with the documents to form the document knowledge graph and propose a calculation algorithm of the four tags' weight. Finally, we propose the sorting and re-finding method based on the feature knowledge graph. By using personal mobile phones data, the experiment results verify the effectiveness and efficiency of this method.

Keywords: Re-finding · Mobile phone documents · Feature knowledge graph

1 Introduction

With the rapid development of mobile Internet technology, more and more people have started to use mobile devices for work and study, such as mobile phones and tablets. With the accelerated spread and the exponential growth of information in the data age, more and more personal information is stored in mobile phones. According to the global mobile economy report [1] released by the GSMA in 2019, the total number of independent mobile users will reach 5.1 billion by the end of 2018, accounting for about two thirds of the global population. Relying on the mobile Internet, mobile office also has a breakthrough development in 2016. The international well-known data company IDC [2], their survey shows that 82% of Chinese employees use their mobile phones for work. With the accumulation of time, files stored in the mobile phones are increasing. How to effectively manage mobile files and how to quickly and efficiently find files become a big problem.

Studies have shown that most of the access to desktop information is to re-find the existing and visited information [3]. In the field of mobile devices such as mobile phones, files are saved through application software. Users cannot choose the proper path to store it on phones as easy as what they do on computers. Because the mobile phone files have application isolation, the documents downloaded by different applications will be saved in different folders. Besides, when people use the mobile phone, a large amount of redundant and irrelevant information will be generated. This information makes it difficult for users to quickly find the information they need.

But users often encounter the situations similar to the following: (1) when searching for a document, the user cannot remember the name of this document, but can only remember the topic of a picture in the document. For example, a user want to find a document he/she wrote a year before, but he/she can only remember a picture is about neural network in the document. (2) When users searching for a document, they can only remember where they get. For example, a user want to find a file, he/she can only remember that he/she got this file from WeChat, but as time went, the chat records are deleted by users (Because for Android phones, more and more chat records may make our phone slow, many users may clear their App data after a while) So he/she can't re-find the document with searching contents by App itself. (3) The user wants to find a document related to one topic, and can only remember a vague opened time. For example, when a user wants to find a file about big data, what he/she remembers is that he/she has opened the file a week ago.

It can be seen from the above examples that users often establish a relationship between document attributes and document titles in their minds, which is similar with knowledge graph structure. The knowledge graph can be used to establish the connection between attributes and documents. Knowledge graph is to describe knowledge resources and their carriers with visual technology, mining, analyzing, constructing, drawing and displaying knowledge and their interconnections. It can describe the connection between different entities concisely and intuitively, reduce the influence of invalid information, and integrate seemingly unrelated fragmented information to form a related graph. Knowledge graphs are widely used in management science, military science, medicine, economics and other fields. However, these applications are mainly concentrated in enterprises or academia, and there are relatively few applications in the field of personal information management. This paper applies the knowledge graph to the field of personal information management, and proposes the method for re-finding mobile phone documents based on feature knowledge graph.

The first section of the paper is the introduction, which puts forward the research of mobile phone documents and describes the discovered problems. The second section introduces the related works about re-find of desktops and mobile phones, the application of knowledge graph. The section three introduces the concepts model of feature knowledge graph and theoretical basis of them. The section four proposes feature-based weight assignment and search algorithms. The section five describes the experimental process and analyzes the results. The section six summarizes the contribution of this paper: by comparing the proposed sorting method with traditional methods such as mobile phone file management software, the method proposed in this article can re-locate the

searched documents more efficiently and accurately, and finally introduced the future work direction.

2 Related Work

As far as the author knows, there is less work related to re-find documents in the field of mobile phones information management. Mobile file management applications are often used as classification. But when people can't remember the correct file name, this software loses its advantage.

In the field of computer desktop re-finding, there are two aspects. The first is based on browsing history information. Trien V. Do and Roy A. Ruddle [4] propose a tool which describe the user's interest in web pages by recording residence time, frequency of visits, number of recent visits and so on. The second is based on the user recall context for re-finding. Tangjian, D., Liang, Z. et al. [5] developed a context-based method of information reflection. They proposed to store it in a cluster method. When re-finding, they can access these relationships and positioning to the required information.

Xuan Luo [6] takes personal notes as a starting point and uses natural language processing technology to intelligently analyze the content of the notes. It is proposed to automatically extract attributes and make associations based on the relationships between attributes to construct a knowledge graph about notes. Yan Liang [7] proposed to manage fragmented knowledge based on knowledge graph technology. By constructing the associations between knowledge points, the fragmented learning resources are developed around each knowledge point, forming a learner's personal knowledge system.

Although there is some mobile phone file management applications, few people pay attention to the role of prominent descriptions of file content in mobile phone files re-finding. Re-finding tools for desktop files already have mature studies. However, these studies are not fully applied to mobile phone re-finding. One reason is the structure of mobile phones and computers is different, and the directory of documents is also different. Another reason is the most existing studies in PC need to interact with users to make sure the precision rate. But due to the limitation of the mobile phones screen, our re-finding will be limited without interacting with users. Based on above reasons, this paper uses the knowledge graph to connect the documents and its' attributes. Each document is marked with tags extracted by the algorithm, and these tags represent the relevant attributes of the documents. When a document has a certain tag, a connection between the document and the attributes is established. Based on this relationship, all documents in the mobile phones can be aggregated to form a graph of association relationships between documents. Perform qualitative and quantitative analysis on the document information in the mobile phones, and construct the "entity-attribute-attribute value" triples about the documents. Then the users can re-find the documents quickly with the method proposed in the paper.

3 Concept Model of Feature Knowledge Graph

In order to solve the problem of re-finding without keywords, the paper proposes the concept model of feature knowledge graph. The process of people's search activities is:

first of all, search for keywords with their mind. These keywords are not limited to the name of the document, but may be a word or a sentence appearing in the document, or a picture, table or mathematical formula referenced by the document [8]. The method proposed in reference [5] and [8] is to classify the attributes, which is not automatically tagged, and can only be applied to the computer. Different from reference [5] and [8], the paper proposes a method of quantifying tags. We propose four features. They are resource feature, time feature, user memory feature, location feature, and their corresponding tags are: topics, last opened time, references and sources. After tagged the document, we mainly use knowledge graphs to show the relationship between document content attributes.

3.1 Resource Feature

Resource feature reflect the content of a document as a whole. The resource feature can describe the characteristic of data distribution, such as the words frequency, the part of speech and the topics. In this paper, we use topics to describe the resource feature. Because every document has one or some topics and these usually make us impressive. Another reason is it can fully reflect the information of the documents and can highly summarize the content of the documents.

3.2 Time Feature

The time feature is described by the last opened time of the document. We interview 30 mobile phone users with different careers for interval time of re-finding the documents by phone, all of them use mobile phones to work. The survey results show 80% of the interviewees thought the last time he opened the document is no more than one month. Only 20% of them show they can sometimes re-finding documents beyond three months.

As time goes on, we will forget what we remembered. According to the forgetting curve (Fig. 1) proposed by the German psychologist Ebbinghaus [9]: The law of forgetting is not to forget the content of a fixed length in a fixed time, but to forget quickly and then slowly.

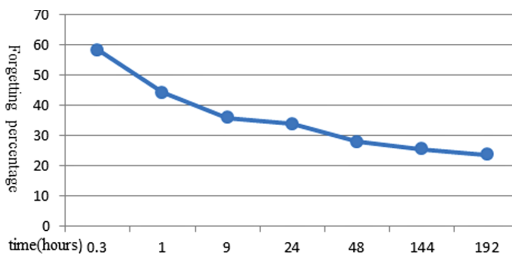


Fig. 1. Ebbinghaus forgetting curve

It can be seen from Fig. 1 that after the 6th day (the 144th hour), the forgetting curve started to be smooth. The user can only remember less than 30% of the documents

opened six days ago. According to the Ebbinghaus Forgetting Curve as the theoretical basis, we took six days as a period named last opened time of documents to tag the time feature.

3.3 User Memory Feature

This feature describes the memory of users. Users are most likely to recall some fragments such as pictures, tables, mathematical formulas, etc. We use the above three factors to describe the user memory feature called references tag. Studies have pointed out in most cases, people always remember pictures better than words, which is called PSE (the picture superiority effect) [10]. It is because of the high compatibility between pictures rendering and visual space template.

3.4 Location Feature

The location feature refers to which applications the documents are downloaded by. Due to the isolation between different applications in the phone, taking WeChat and QQ application as an example, although both are Tencent’s applications, the location of the data file is not the same. The files downloaded by WeChat are stored in the “tencent/MicroMsg/Download” directory under the internal storage file directory of the phone, and the files downloaded by QQ are stored in the “tencent/qqfile_recv” directory. DingTalk is the software which the 80% of mobile office enterprises choose. Its downloaded files are stored in the Dingtalk directory. For users who can-not clearly find the file directory, re-finding the documents from these applications is hard. Based on the above situation, we use source tag to describe the location feature of documents.

We have surveyed 141 enterprises which include varieties of industries to know the usage of DingTalk, WeChat and QQ. The result is shown in Fig. 2.

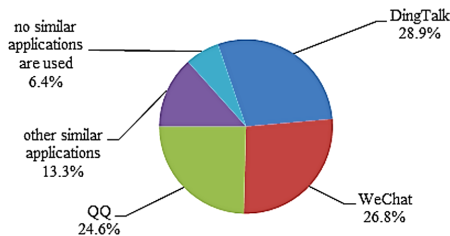


Fig. 2. The usage of DingTalk, WeChat and QQ

From Fig. 2, we can see the total proportion of DingTalk, WeChat and QQ applications is 80.3%. And from the questionnaire, we found more than 50% enterprises use these three applications to share the files. So we use WeChat, QQ, and DingTalk to classify the source tag.

3.5 Feature Knowledge Graph

The corresponding tags extracted from each document describe the relevant attributes of the document, and the knowledge graph is constructed by combining the documents and tags. Two kinds of nodes are provided in the knowledge graph constructed in this paper, namely document nodes and tag nodes. The document node records the relevant information of the document, including information such as title and content. The tag node records the relevant attributes of the document node, such as topic, source, last opened time, and references contained in the document. When a document has a certain tag, a connection between the document node and the tag node is established. Based on this basic relationship, all documents in a user’s mobile phone are aggregated to construct a knowledge graph of the associations between personal mobile phone documents. Figure 3 describes the relationships between several entities. The nodes represent entities, and edges represent relationships between entities.

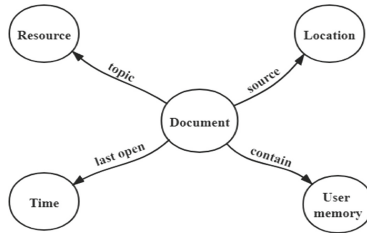


Fig. 3. The relationships between entities

Take the following three documents as an example. Figure 4 shows part of the knowledge graph. The graph diverges around these three documents and shows the connections between the three documents.

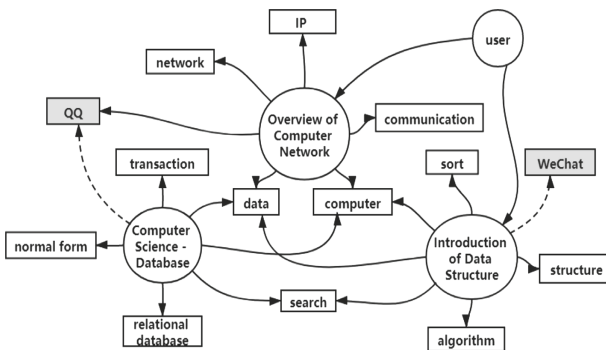


Fig. 4. The part of the knowledge graph

The round node represents the “document” entity, the white rectangular node represents the “resource” entity, the gray rectangular node represents the “location” entity,

the solid arrow represents the relationship of “resource-topic tag-document”, and the dashed arrow represents the relationship of “document- Source-location” relationship.

4 Re-finding Method Based on Feature Tags

We use four-tuple $D_i(L_i, T_i, M_i, R_i)$ to represent the four feature tags. Among them D_i means the i th document, L_i means the topic of D_i , T_i means the last opened time of D_i , M_i means a reference to D_i , R_i means the source of the D_i . For example, D_1 (Big data, 20191005, Table, WeChat)’s meaning is the topic of D_1 is “Big data”, the latest opened time is “October 5, 2019”, the reference contains “Table”, and the document’s source is “WeChat”.

4.1 The Weight Assignment Algorithm of Feature Tags

Different users have different memory points when searching. For example, user A and user B search for the same document, user A remembers a table in the document, and user B remembers the source of the document. Obviously user A and user B have different focus on the tag. So considering the above situation, it is necessary to propose the weight distribution algorithm to adjust the weight of each feature for different user memories.

Resource Feature—Determine the Topic. We first use the TF_IDF to count the frequency of words, and second use LDA based on part of speech to select highly summarized topic.

The method used to count the word frequency is TF_IDF (term frequency–inverse document frequency), TF is the word frequency, IDF is the reverse file frequency, and the TF_IDF is the product of TF and IDF. However, after getting the TF_IDF values, the words with the highest TF_IDF value cannot be a topic of the document, because we cannot determine the distribution of these high-frequency words in the document. This situation may occur: The distribution of a high-frequency word is scattered in the document. And the scattered words are not enough to express the topic of the document. Therefore, we use the LDA topic model based on part of speech to determine the distribution of high-frequency words through TF_IDF statistics.

The LDA (Latent Dirichlet Allocation) topic model selects multiple groups of words. It is a Bag-of-word. It is most widely used in the field of text clustering and classification [11]. The traditional LDA topic model has the disadvantages of polysemy, blindness and semantic degradation. So we use POS_LDA model (a LDA topic model based on Part of Speech) to avoid the shortcomings of traditional model.

After removing “stop words”, we build the POS_LDA model to calculate the probability of the words with parts of speech. We use a standard annotation set-ICTCLAS. The partial ICTCLAS comparison table is shown in Table 1.

By counting the parts of speech before segmentation, as shown in Fig. 5, we found the coverage of nouns and verbs reached 100%, and the coverage of adjectives reached 99.5%. After removing “stop words”, the preserved parts of speech are nouns, verbs adjectives, distinguish words and place name. It was found that the accumulative proportion of nouns, verbs and adjectives accounted for more than 98%, while the remaining 10 parts of speech accounted for less than 2%. Therefore, we use nouns, verbs, and adjectives to represent document topics.

Table 1. Partial ICTCLAS comparison table

Part of speech	Meaning	Part of speech	Meaning
a	Adjective	n	Noun
b	Distinguish word	v	Verb
c	Conjunction	nh	Person name
d	Adverb	ns	Place name

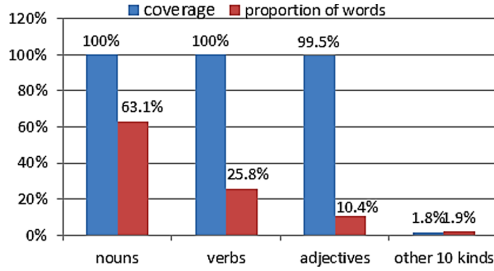


Fig. 5. Distribution of part of speech

The Weight Assignment Algorithm. We propose a dynamic weight algorithm for attributes with user priority based on information gain. Before the feature tags are added, the document has the original information entropy. After adding feature tags, the conditional information entropy is generated [12]. The information entropy calculation is as formula (1).

$$H(x_i) = - \sum_{i=1}^n p(x_i) \log p(x_i) \tag{1}$$

In this paper, we use the difference between the original information entropy and the conditional information entropy named information gain for weight allocating. Each document ensure the attribute tags selected by D_i corresponds to four variables of tags, they are x_1, x_2, x_3, x_4 . The information gain algorithm of the four tags attributes is shown in Algorithm 1. k is the number of attributes contained in each tag and C_{x_i} is attribute variable in each tag.

Algorithm 1 Information Gain Algorithm of the Four Tags Attributes

Input: D_i document set

Output: IG of the four tags attributes

1: **for** each D_i **do**

2: calculate the original information entropy with formula (1):

$$3: H(d_i) = -\sum_{k=1}^n p(x_k) \log p(x_k);$$

4: calculate the conditional information entropy with formula (2) & (3):

$$5: H(d_i|x_i = c_{x_i}) = -\sum p(d|x_i = c_{x_i}) \log p(d|x = c_{x_i}) \quad (2)$$

$$6: H(d|x_i) = \sum p(x_i = c_{x_i}) H(d_i|x_i = c_{x_i}) \quad (3)$$

7: calculate the information gain with formula (4):

$$8: IG(x_i) = H(d_i) - H(d|x_i) \quad (4)$$

9: **return** $IG(x_i)$

10: **end for**

After obtaining the information gain of each tag, formula (5) is used to calculate the weight corresponding to each attribute. W_t is the weight of four tags. The weight obtained here represents the importance of each attribute to the document.

$$W_t = \frac{IG(x_i)}{\sum_{i=1}^n IG(x_i)} \quad (t = 1, 2, 3, 4) \quad (5)$$

4.2 Sorting and Re-finding Method Based on Feature Tags

The documents of most third-party file management applications are sorted by type or file size or time. This sorting method cannot effectively reduce the user's re-finding time, moreover, the traditional search methods did not personally optimize the search for each user's personal characteristics, nor did they show the relationship between the document and the content. Traditional search methods are often out of touch between the external and internal features of the document. The knowledge graph introduced in this paper establishes a relationship between the external characteristics of the document and the internal characteristics including the user characteristics, thereby turning the document and the user into an interconnected whole.

The subjective factor that affects the efficiency of document re-finding is the similarity between the keywords formed in the memory and the contents of the documents. The objective factors affect the re-finding efficiency are the classification and ordering rules of the documents. We propose a sorting re-finding algorithm to reduce the negative influences of subjective factors.

We have two assumes. The first one: the users are not affected by subjective memory when they are re-finding, which means they have a vague memory of some features of documents. When you ask them if they remember a table referenced in a document or where they get the document, they will tell you "I can't remember". So, we need to provide users with a sorting rule that minimizes the scope of the re-finding. Another assumption is users are affected by subjective memory when they are re-finding.

They know what keywords they are looking for. For example, user A needs to find a document. He clearly remembered there is a table in this document, but he has no impression of the document's name. For this situation, when A chose the references tags, the weight of references tag is changed, the ranking results are also changed. In this condition, in order to interact with users, we showed them the contents of four tags and asked them to select the tag closest to their remembered keyword. We dynamically adjust the weight of the tags by adding the user priority value (u_i), u_i is determined according to the connection between the user and the document in the personal knowledge graph. For example, the connection between the document D_1 and the user u_1 is that the user u_1 opened the document D_1 three days ago, and the corresponding tag is last opened time. The knowledge graph establishes a connection between users and attributes through documents, and then u_i can be calculated by formula (6). It means the percentage of the user's memory keyword (p_j) in the total number of tag's attributes it belongs to.

$$u_i = \frac{\sum_{j=1}^m p_j}{\sum_{k=1}^n x_k} \quad (6)$$

In order to ensure the attribute tags selected by users will not be ignored due to the small number of attribute tag values, the following calculation method is proposed to increase the weight of the attribute with a small attribute value.

$$IG'(x_i) = H(d_i) - (1 - u_i) * H(dx_i), \quad WU_t = \frac{IG'(x_i)}{\sum_{i=1}^n IG'(x_i)} \quad (t = 1, 2, 3, 4) \quad (7)$$

When the value of u_1 is larger, the value of $IG'(x_i)$ is larger. And then, we use Algorithm 2 given below to get initial rank of documents for re-finding.

Algorithm 2 The Sorting Algorithm of Re-finding Documents

Input: the weight of documents W_t

Output: the rank of documents

- 1: **for** each re-finding tasks **do**
 - 2: **if** user have keywords in their memory **then:**
 - 3: calculate the weight with formula (7) by adding user priority value;
 - 4: **else**
 - 5: calculate the weight with formula (5);
 - 6: **end if**
 - 7: construct the attribute quantization matrix $A(a_{ij})_{n*m}$;
 - 8: construct weight matrix $W(w_t)_{m*1}$;
 - 9: calculate $S_{n*1} = A(a_{ij})_{n*m} * W(w_t)_{m*1}$;
 - 10: **return** S_{n*1}
 - 11: **end for**
-

In Algorithm 2, a_{ij} in $A(a_{ij})_{n*m}$ is the proportion of a certain attribute to the total attributes under each tag. The document corresponding to the first value in the S_{n*1}

matrix obtained in Algorithm 2 is most similar to the user’s expectations. However, the following situation will occur: some documents similar to the user’s expectations will be ranked in the lower position. In order to prevent the above situation, this paper uses the Common Neighbors algorithm to calculate the document similarity in the order of the matrix $S_{n \times 1}$ by formula (8). $N(x)$ and $N(y)$ are two document nodes, the formula calculates the number of nodes shared between them.

$$CN_{sim(x,y)} = |N(x) \cap N(y)| \quad (8)$$

After obtain the document similarity, then uses the formula (9) to normalize the calculated number of nodes. The max is the maximum value of the sample data, and the min is the minimum value of the sample data.

$$S_i = \frac{CN_{sim(x,y)} - min}{max - min} \quad (9)$$

5 Experiment Evaluation

This paper studies the re-finding of the personal mobile phone documents. A big challenge is there is no public data set, and the data in personal mobile phones has a wide range of fields and diverse contents. Therefore, the data set of this experiment is constructed by collecting data of users’ mobile phones.

5.1 Experiment Data Set and Preprocessing

Due to the privacy of the documents in the personal mobile phones, the experimental data were collected from personal data of mobile phone users. To make the data as comprehensive as possible, the users selected for the experiment cover different industries, including computers, electricity, education, and the economy. The experimental training set is 1680. To evaluate the efficiency of the proposed method, we selected 352 mobile phones data from real users in the last six months as test sets, and the details of the test set are shown in Table 2.

Table 2. Test set

Test set	Involved field	Number of documents
Test1	Economy	74
Test2	Computer	88
Test3	Finance	69
Test4	Mechanical	70
Test5	Electricity	51

5.2 Experiment Evaluation

We use search time to evaluate the experimental results. We find eight volunteers to help complete the experiment. The experiment is performed in two parts and takes two weeks. First, give the mobile phone with the test set to the volunteers, and require them to be as familiar with the content of the test set as possible for a week. Then give some re-finding tasks to the volunteers. Then we use the method proposed in this paper to compare with the file management applications and traditional directory method. Finally, the less time it takes, the higher the re-finding efficiency is. The description of ten tasks is shown in Table 3.

Table 3. Test tasks

Task number	Re-finding mission details	User subjective memory factor
1	A document about computer	Document was last opened five days ago
2	A document about a school recruitment requirements	Contain a table about career
3	A document about the role of party members	Document is from WeChat
4	A document	Topic is about computer networks
5	A document about English translation	None
6	A document about a thermoelectric accident	Contain pictures about machine, document is from WeChat
7	A test question about Internet applications	Last opened a week ago and the document is from QQ
8	A document	Topic is educational and from WeChat
9	A document about the financial crisis	The document was last opened two weeks ago with a table
10	A document	Topic is about computer database and contains mathematical formulas

5.3 Analysis of Results

First, according to the method described in Sect. 3, all documents in the test set are tagged with the four feature tags to form the four-tuple. And the weights are calculated with method proposed in Sect. 4. The weight without user priority matrix is $W(w_t)_{m*1} = (0.361, 0.182, 0.257, 0.188)^T$ (three decimal places). So we have a basic rank of documents based on $W(w_t)_{m*1}$.

From the results of volunteer experiments we observe: (1) in task 2 the method proposed in this paper takes longer than the other two methods. After analysis, we find there are two main reasons: First, because the name of the document corresponding to

task 2 starts with a number, it leads to the front position when using the file management applications. Since there is no need to scroll down when searching, the volunteers quickly found it. Second, volunteers have a deep memory of the document, which also results in shorter time. This situation is where we need to improve in the future. (2) From other tasks, it can be seen that the search time using the method proposed in this paper is shortened compared with the other two methods. Figure 6 shows the average search time of these three methods.

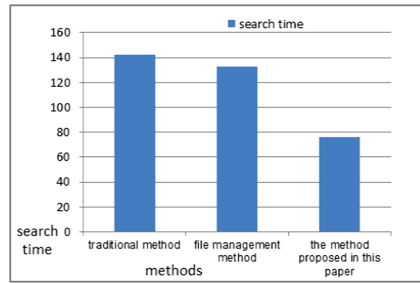


Fig. 6. The average search time of three methods

From Fig. 6, we obtain the following conclusions: (1) compared with the other two methods, the traditional search takes longer. The method proposed in this paper takes the least time and is generally shortened by about one-half compared with the traditional method. (2) When the users use the traditional mobile phone search function, time is mainly wasted between switching the parent and son directory of the folder. (3) The method proposed in this paper shortens the scope for users' re-finding and is efficient. It can dynamically sort according to the user's memory contents.

6 Conclusion

By analyzing the behavior and memory characteristics of mobile phones re-finding, this paper proposed to tag the mobile phone documents with four features and construct a knowledge graph. Then we proposed the quantitative calculation method of tags and sorting which was verified by experiments. The results showed that the method proposed in this paper took less time for re-finding and was an efficient method. But this is only a preliminary study. In future research, we will continue to improve the method with optimizing the value of the weights and improve the sorting method.

References

1. Stryjak, J., Sivakumaran, M.: The Mobile Economy 2019. GSMA Intelligence, Barcelona (2019)
2. 2016 Mobile Office Panorama Report (in Chinese). Mobile Information Researching Center, China (2016)

3. Tauscher, L., Greenberg, S.: Revisitation patterns in world wide web navigation. In: CHI97: ACM Conference on Human Factors & Computing Systems, pp. 3–10. Association for Computing Machinery, New York (1997)
4. Do, T.V., Ruddle, R.A.: The design of a visual history tool to help users re-find information within a website. In: Baeza-Yates, R., et al. (eds.) ECIR 2012. LNCS, vol. 7224, pp. 459–462. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-28997-2_41
5. Tangjian, D., Liang, Z., et al.: ReFinder: a context-based information re-finding system. *IEEE Trans. Knowl. Data Eng.* **25**, 2119–2132 (2013)
6. Xuan, L.: Design and Implementation of Personal Knowledge Management Platform Based on Knowledge Graph. Beijing University of Posts and Telecommunications, Beijing (2018)
7. Yan, L.: Study on the Fragmentation Knowledge Management Based on Knowledge Map in Mobile Learning. Sichuan Normal University, Sichuan (2018)
8. Liu, G., Feng, L.: A method to support difficult re-finding tasks. *Journal* **02**, 15–19 (2016)
9. Wikipedia Forgetting_curve. https://en.wikipedia.org/wiki/Forgetting_curve. Accessed 15 May 2020
10. Baadte, C., Bozana, M.: The picture superiority effect in associative memory: a developmental study. *Br. J. Dev. Psychol.* **37**(3), 382–395 (2019)
11. Qian, S., Zhang, T., Xu, C.: Boosted multi-modal supervised latent Dirichlet allocation for social event classification. In: International Conference on Pattern Recognition, pp. 1999–2004. IEEE (2014)
12. GuoYin, W., Hong, Y., DaChun, Y.: Decision table reduction based on conditional information entropy. *J. Comput. Chin. Ed.* **25**(07), 759–766 (2007)