2581-4575

International Journal For Recent Developments in Science & Technology

Crossref

A Peer Reviewed Research Journal

A Scalable Approach for Hiding Information Based on Web Text

D. Deepthi[1], A. V. Naga Mani[2]

[1]Student, V. S. Lakshmi Women's Degree & P.G. College, Kakinada

[2]Senior lecturer, V. S. Lakshmi Women's Degree & P.G. College, Kakinada

Abstract: Data mining is that the method of extracting valuable data from an over-sized information supply. Now a day World Wide Web has gained a lot of user's attention towards its information. In present days no images are having tagging or annotation concept in internet and hence users are facing a lot of problem to differentiate the images. Hence in this paper, we mainly try to design a model which can automatically extract the hidden data and extract the information which is present in the google.

Keywords: Big Data; Coverless Information Hiding; Frequent Words Hash; Rank Map; Steganography

I. INTRODUCTION

The concealed knowledge makes use of the senselessness of human meaning and multimedia redundancies to conceal in the optical carrier. It can primarily be separated into four categories: document hiding, picture hiding, visual hidting and audio hiding by the numerous digital carriers [1]. Text is the most commonly used media, so this paper concentrates on hiding text content.

Classical face annotation approaches are often treated as an extended face recognition problem, where different classification models are trained from a collection of well labelled facial images by employing the supervised or semi-supervised machine learning techniques. However, the "model-based face annotation" techniques are limited in several aspects. First, it is usually time-consuming and expensive to collect a large amount of human-labelled training facial images. Second, it is usually difficult to generalize the models when new training data or new persons are added, in which an intensive retraining process is usually required. Last but not least, the annotation/recognition performance often scales poorly when the number of persons/classes is very large.

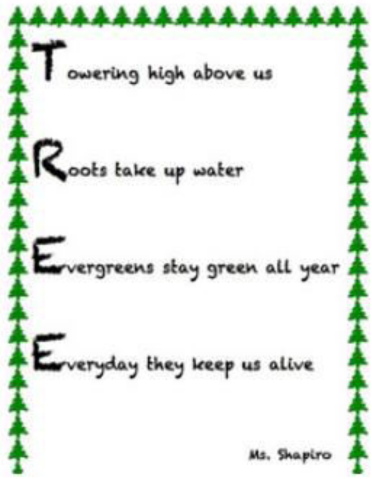
Recently, some emerging studies have attempted to explore a promising search-based annotation paradigm for facial image annotation by mining the World Wide Web (WWW), where a massive number of weakly labelled facial images are freely available. Instead of training explicit classification models by the regular model-based face annotation approaches, the search-based face annotation (SBFA) paradigm aims to tackle the automated face annotation task by exploiting content-based image retrieval (CBIR) techniques [7], [8] in mining massive weakly labeled facial images on the web. The SBFA framework is data-driven and model-free, which to some extent is inspired by the search-based image annotation techniques [1], [2], [3] for generic image annotations. The main objective of SBFA is to assign correct name labels to a given query facial image. In particular, given a novel facial image for annotation, we first retrieve a short list of top K most similar facial images from a weakly labeled facial image database, and then annotate the facial image by performing voting on the labels associated with the top K similar facial images.

Auto face annotation can be beneficial to many real world applications. For example, with auto face annotation techniques, online photo-sharing sites can automatically annotate users' uploaded photos to facilitate online photo search and management. Besides, face annotation can also be applied in news video domain to detect important persons appeared in the videos to facilitate news video retrieval and summarization tasks.

One challenge faced by such SBFA paradigm is how to effectively exploit the short list of candidate facial images and their weak labels for the face name annotation task. To tackle the above problem, we investigate and develop a search-based face annotation scheme. In particular, we propose a novel unsupervised label refinement (URL) scheme by exploring machine learning techniques to enhance the labels purely from the weakly labeled data without human manual efforts. We also propose a clustering based approximation (CBA) algorithm to improve the efficiency and scalability.

II. COVERLESS INFORMATION HIDING

Coverless information hiding is a new challenging research field. In fact, “coverless” is not to say that there is no carrier, but compared with the conventional information hiding, coverless information hiding requires no other carries [6]. The idea of coverless information hiding is often used in our daily life, and the acrostic poem is a classic example. An acrostic poem is shown in Figure 1 form which we can learn that the secret information is ”TREE”. Coverless information hiding is essentially the disclosure of secret information in the text. Its distinctive characteristic is ”no embedding”, that is, a carrier cannot embed secret information by modifying it [6].



Towering high above us

Roots take up water

Evergreens stay green all year

Everyday they keep us alive

Ms. Shapiro

Figure 1: An acrostic poem

III. PROPOSED METHOD

In our algorithm, web texts are retrieved from the web text library to hide information. The framework of the algorithm is shown in Fig.1. The sender segments the SI into some secret keywords, and the searched keywords are extracted by Text Rank algorithm for searching web text. A series of web texts related to the SI are obtained by using web spider technology and search engines. Then a web text library is constructed after processing these web texts. The web text set that containing secret keywords is retrieved from the web text library, and the optimal web texts are selected. The location information of each secret keywords in the selected web text is analyzed and recorded, which is encoded and compressed in a string form after generating a key. The string is appended as a parameter to the URL of the selected web text, which is packaged and delivered to the recipient. The recipient extracts the location information after receiving the URL. The web page is downloaded and processed to get the web text. The secret keywords are extracted from the web text according to the location information and the key. And the SI is obtained after these keywords are connected.

1. The secret keyword set is defined as $S=\{si|i=1,2,\cdots,N\}$, and searched keyword set is denoted as $L=\{li|i=1,2,\cdots,M\}$.

2. Crawled web page set is defined as $W=\{wi|i=1,2,\cdots\}$, the set of links parsed from W is denoted as $U=\{uj|j=1,2,\cdots\}$.

3. Pi is defined as a web text set containing si , $Ki=\{kij|j=1,2,\cdots,5\}$ is a similar word set of si .

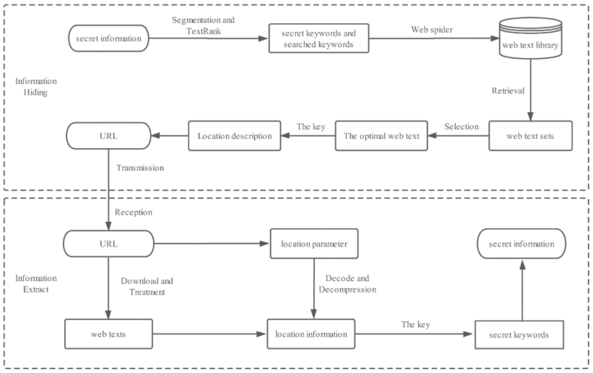
4. The retrieved result set is defined as $R=\{rk|k=1,2,\ldots,N\}$.

5. The hidden keyword set of a web text is defined as $O=\{oj|j=1,2,\cdots\}$, the position of secret keyword oj in web text is defined as loc(oj) .

6. The minimum number of bits representing the number of rows and columns are rb and wb, respectively.

7. The location string of a web text is defined as str , which is the result of the connection of location information for all secret keywords.

8. The location parameter is defined as seq, URLsec is the link sent to recipient.



The framework of the algorithm.

A. Information Hiding

In this section, we will get the URL of web text containing secret keywords and the corresponding location string of SI in the web text. The hiding process is shown in Algorithm 1. The hiding process can be divided into five steps as follows: (1) Segment SI to secret keywords, and extract the searched keywords by using TextRank; (2) Fetch the web texts and build a web text library; (3) Retrieve the web text set and select the optimal web text; (4) Encode the location information of secret keywords in selected web text to obtain the location parameter; (5) Connect the URL and location parameter, and send it to the recipient.

Algorithm 1 Hiding Algorithm

```
while secret information SI is not null do
Split SI into secret keyword set S from jieba analyzer
Select searched keyword set L by TextRank algorithm
Building web text library as Algorithm 2
The optimal web text is selected by Algorithm 3 and 4
Location parameter seq is generated by Algorithm 5
URLsec=URL+‘‘/pos"+seq , and send URLsec to recipient
end while
```

1) Construction of Web Text Library

Text coverless information hiding draws on text big data to hide information. In this paper, we select appropriate web texts from a large number of web pages to build a web text library, which can be considered as a simplified text big data. The details are shown in the Algorithm 2. The searched result will be different due to different search strategy in the search engine, thus multiple search engines are combined to search web pages based on meta search engine. In our algorithm, the domestic mainstream search engines, Baidu and Sogou as well as Google abroad are selected.

Algorithm 2 Building Web Text Library

```
Fetch web page set W from the result of search engines.
for i=1 to length(W) do
Parse wi , add URLs into link set U
end for
Remove duplicate links, update link set U .
for j=1 to length(U) do
Download and parse uj to get web text.
end for
```

Delete Numbers, English and punctuation, etc.

Split text contents to words and delete rare words.

Build full-text index, and the index entry is the keywords of web texts.

The number of searched web pages is usually very large, thus a threshold T is set to fetch the number of searched web pages. The process of spider is divided into two parts, one is to fetch all the links, the other is to crawl the text content. Obviously, there are some identical links in the searched results, so the duplicate links will be removed. The content of the web text contains Chinese and English characters, programming code, Numbers, etc. Due to the messy text format, the pretreatment must be performed. After pretreatment, all web texts are made up of Chinese characters and corresponding URL, and secret keywords are allowed to be retrieved in web text library.

2) Retrieving Web Texts

In our proposed algorithm, web text sets that contain secret keyword are retrieved in the order of word segmentation. The retrieval of secret keyword is failed when the set was empty, then the similarities of the top 5 words are calculated by using word2vec. These similar words are selected to replace secret keywords according to the similarity. If the retrieved result for all similar words is failed, the secret keyword is mismatched. As a result, a series of web text sets are obtained. Then we get the intersection of the sets in order. The optimal web text will be selected when the intersection is empty. The details are shown in the Algorithm 3.

Algorithm 3 Retrieving Web Texts

```
for i=1 to length(S) do
    Retrieve web text set Pi that contains the secret keyword si
    if Pi is null then
        Similar word set Ki is obtained by word2vec
        while j≤Ki do
            Replace kij with si , and retrieve Pi
            if Pi is not null then
                break;
            end if
        end while
        if Pi is null then
            Record si , ri is null and si is mismatched
        end if
    else if Pi is not null then
        Add Pi to the result set R
    end if
end for
Initialize I=r1 , V=null , then delete r1 from R
while R is not null do
    V=I∩r1
    if V is null then
        Record I , the optimal web text will be selected from I
        I=r1
    else
        I=V
    end if
    Delete r1 from R
end while
```

3) Selection of the Optimal Web Text

In general, it is enough to randomly select the web text as stego-text in the retrieved web text library. However, this method cannot meet the security requirements. For example, the obtained web text is highly correlated with SI, and even the SI is a complete sentence in the web text.

Therefore, the optimal web text is selected by Algorithm 4.

Algorithm 4 Selecting the Optimal Web Text

```
Assume the set of retrieved web texts as I, and it can hide Y secret keywords
Define the distance set as D, and the variance set as V
**for** i=1 to length(I) **do**
    **for** j=1 to Y **do**
        dj=loc(o(j+1))−loc(oj)
        add dj to a distance set D
    **end for**
    Delete max(di) from D
    **if** *similar length* ≤ 5 **then**
        Record the var(D) , add it to variance set V
    **end if**
**end for**

Select the web text that variance is min(V)
```

It is mainly concerned with the distribution of keywords in the web text. The keywords are distributed as evenly as possible in the text, which means that the distances between adjacent keywords tend to be stable values. The distance is defined as the difference value between the positions of the two keywords in the text. The variance of all distances represents the level of uniformity of each web text. The web text of the smallest variance should be selected. In addition, the number of consecutive distances of 1 is defined as similar length, which is used to measure the security performance. Generally, the algorithm is secure when the number less than or equal to 5. If not, it is easy to be recognized by human eyes and the information hiding is meaningless.

4) Location Encoding of Secret Information

The two-dimensional coordinate system is constructed to represent the location of secret keywords in web texts, then location information is compressed and encrypted after encoding. Algorithm 5 shows the details. In this algorithm, wb can be used as key according to different recipients, thus the number of columns is determined. rb can be calculated by the optimal web text and the key, and the coordinate of each keyword is represented. In order to improve the efficiency of encoding string and remove the effects of invisible characters, base64 algorithm is used to encode the concatenated coordinate binary string. In this schema, each 24-bit binary string is processed and converted into 4 visible characters. Since base64 algorithm requires the length of the string to be a multiple of 8, several “0” strings are appended to the binary string. It allows the binary stream to be recovered losslessly after encoding. It is obvious that the number of 0 that we added to string ranges from 0 to 7. Therefore, we use 3 bits representing the number of 0 strings added as a prefix for the binary string. Finally, the location parameter is obtained, which is combined with the URL of the web page and sent to the recipient.

Algorithm 5 Encoding the Location Information

```
Defines the function that converts decimal t to b bits as binb(t) , default b is the minimum number of bits that can represent t
**Initialize** wb , the location string str is null
```

```
r=length(O)/2wb
rb=bin(r)
fori=1 to length(S) do
if si in O then
the abscissa si(x) and ordinate si(y) of si are calculated
qi=binrb(si(x))+binwb(si(y))
str=str+qi
end if
end for
num_add_col_bits=bin5(wb)
iflength(str) is not a multiple of 8 then
calculated the number of add “0”, denoted as num
num_add=bin3(num)
end if
res=num_add+num_add_col_bits+str+‘‘0"∗ num
seq=base64(res)
```

B. Information Extraction

In the process of information extraction, the recipient receives the URL and extracts the information by the number of column bits. And the location coordinates of each keyword are extracted by the key. The secret keywords are extracted by these location coordinates. All the secret keywords are connected, and then some Chinese grammar processing is performed to obtain the SI. The details are shown in Algorithm 6.

Algorithm 6 Extraction Algorithm

Divide URL sec into URL and location parameter

Download web page and get web text

Decode location parameter by base64 and the location coordinates are obtained

Construct the two-dimensional coordinate system, and secret keywords are extracted

Connect these keywords and get the extracted information

IV. CONCLUSION

This paper presented a coverless text information hiding method based on the frequent words hash. By using the words rank map and the frequent words hash, normal texts containing the secret information could be retrieved from the text database, and will be sent to the receiver without any modification. Because there is no embedding, the information hiding does not change the probability distribution of the covers. Therefore, the proposed method is theoretically safe, and could be able to escape from almost all state-of-the-art steganalysis methods.

REFERENCES

[1] E. O. Blass, T. Mayberry, G. Noubir, and K. Onarlioglu, “Toward robust hidden volumes using write only oblivious RAM,” in ACM SIGSAC Conference on Computer and Communications Security (CCS’14), pp. 203–214, Scottsdale, USA, 2014.

[2] S. Bo, Z. Hu, L. Wu, and H. Zhou, Steganography of Telecommunication Information, Beijing: National Defense University Press, 2005.

[3] J. T. Brassil, S. H. Low, and N. F. Maxemchuk, ”Copyright protection for the electronic distribution of text documents,”

Proceedings of the IEEE, vol. 87, no. 7, pp. 1181–1196, 1999.

[4] C. Cachin, "An information-theoretic model for steganography," in The Second Workshop on Information Hiding, pp. 306–318, Oregon, USA, 1998.

[5] X. Chen, S. Chen, and Y. Wu, "Coverless information hiding method based on the chinese character encoding," Journal of Internet Technology, vol. 18, no. 2, pp. 91–98, 2017.

[6] X. Chen, H. Sun, Y. Tobe, Z. Zhou, and X. Sun, "Coverless information hiding method based on the chinese mathematical express," in The First International Conference on Cloud Computing and Security (ICCCS'15), pp. 133–143, Nanjing, China, 2015.

[7] L. Huang, L. Tseng, and M. Hwang, "The study on data hiding in medical images," International Journal of Network Security, vol. 14, no. 6, pp. 301–309, 2012.

[8] S. Katzenbeisser, F. Petitcolas, Information Hiding Techniques for Steganography and Digital Watermarking, Artech House Publishers, 2000.

AUTHORS PROFILE:

A.V.NAGAMANI: Senior lecturer in Dept. of computer Science at V.S.lakshmi Women's degree & pg College, Kakinada since 2011. She has vast experience in handling projects on JAVA and Eclipse.