International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 9 (2018) pp. 7086-7091

Design of Algorithms for Detection of Intelligent Plagiarism

[1] Dhrubajyoti Baruah and [2]Dr. Anjana Kakoti Mahanta

[1] Assistant Professor, Department of Computer Application, Jorhat Engineering College, Jorhat, Assam -785007, India.
[2] Professor, Computer Science department, Gauhati University, Guwahati, Assam-781014, India

Abstract

Due to the availability of online resources, plagiarism in academic arena is increasing significantly and efficient tool to detect such offence is need of the hour. Several tools are developed to detect plagiarism, which are being used by academic institutions. Knowing this, literal plagiarism has changed its path to intelligent plagiarism where the author, instead of direct copy-pasting from source, does some degree of modification on the original source to prepare the content. Herby it is tried to provide the content a fresh aroma and to pass the scanning of plagiarism detection tools. This paper describes techniques that can detect intelligent plagiarism.

Keywords: Plagiarism, Web Content Mining, Similarity.

INTRODUCTION

Plagiarism is unacknowledged copying of documentary content from others. There is no two human, whatever may be extent of similarity in their thoughts, write exactly the same texts. Plagiarism is derived from the Latin word "plagiarius" which means kidnapper. Plagiarism is defined as the use or close imitation of the language and thoughts of another author and the representation of them as one's own original work [1]. Within academia, plagiarism by students or researchers is considered academic dishonesty or academic fraud, and offenders are subject to academic punishment including expulsion [1]. For example, Tezpur University of India, resolved to withdraw the M. Tech degree in Energy conferred by the University on one Pradeep Kumar Nath, on the charges of plagiarism in his M. Tech dissertation thesis.[2] Based on plagiarist's behaviour, plagiarism may be divided into two groups- (a) Literal Plagiarism (b) Intelligent Plagiarism. Literal plagiarism is a common and major practice wherein plagiarists do not spend much time in hiding the academic crime they committed. For example, they simply copy and paste the text from the Internet[3]. On the other hand, in Intelligent plagiarism, plagiarists try to deceive readers by changing the contents of others to appear as their own. Intelligent plagiarists try to hide and change the original work in various intelligent ways, including text manipulation, translation, and idea adoption[3]. This paper describes techniques to detect intelligent plagiarism.

CONSIDERED INTELLIGENT PLAGIARISM:

Two types of intelligent plagiarism – i) Marginal alteration in values and ii) Table row/column elimination are considered in our research.

MARGINAL ALTERATION IN VALUES:

Quantitative values of findings of research are sometimes changed slightly by other writers with an aim to deceive plagiarism detection tools. Let us consider the following figure1 showing original and plagiarised data reports-

Content 1

Ambient Air Quality data in different cities in India for the year 2012

State	Cities	SO_2	NO_2	PM_{10}
	Chitoor	4	9	40
	Guntur	5	11	75
Andhra Pradesh	Hydrabad	4	28	79
	Kakinada	5	11	63
	Kothagudem	4	9	74
	Kurnool	4	15	79
	Nalgonda	5	11	62
	Nellore	6	22	108
	Patencheru	6	11	82
	Ramagundam	4	9	37
	Tirupati	6	12	97
	Vijaywada	3	11	49
	Warangal	12	13	65
	Vishakhapatnam	5	11	63
Assam	Daranga	5	13	56
	Dibrugarh	6	13	56
	Guwahati	6	14	92
	Margherita	6	15	54
	Lakhimpur	2	2	45
	Nagaon	6	13	79

Original

Content 2

Ambient Air Quality data in different cities in India for the year 2012

State	Cities	SO_2	NO_2	PM_{10}
	Chitoor	4.5	9	40
	Guntur	5.2	11	75
Andhra Pradesh	Hydrabad	4.6	28	79
	Kakinada	5.1	11	63
	Kothagudem	4.1	9	74
	Kurnool	4.9	15	79
	Nalgonda	5.1	11	62
	Nellore	6.2	22	108
	Patencheru	6.4	11	82
	Ramagundam	4.2	9	37
	Tirupati	6.3	12	97
	Vijaywada	3.1	11	49
	Warangal	12.2	13	65
	Vishakhapatnam	5.5	11	63
Assam	Daranga	5.3	13	56
	Dibrugarh	6.6	13	56
	Guwahati	6.3	14	92
	Margherita	6.1	15	54
	Lakhimpur	2.5	2	45
	Nagaon	6.4	13	79

Copied and value changed

Figure 1

Third column of values of SO_2 are changed slightly from Content1 to create Content2. Plagiarism detection tools are developed on the basis of different similarity measures. As values of SO_2 changes, similarity between the contents drops down and tools cannot detect that Content2 was actually plagiarised from Content1.We analysed some of the tools like Plagium and Urkund and found that they are not sensitive towards slight or large changes in values. Plagium, when checked for plagiarism between Content 1 and Content 1 itself, shows 85.4% similarity in Figure 2.

Figure 2

Same result, 85.4% shown for comparison between Content1 and Content 2 in figure 3-

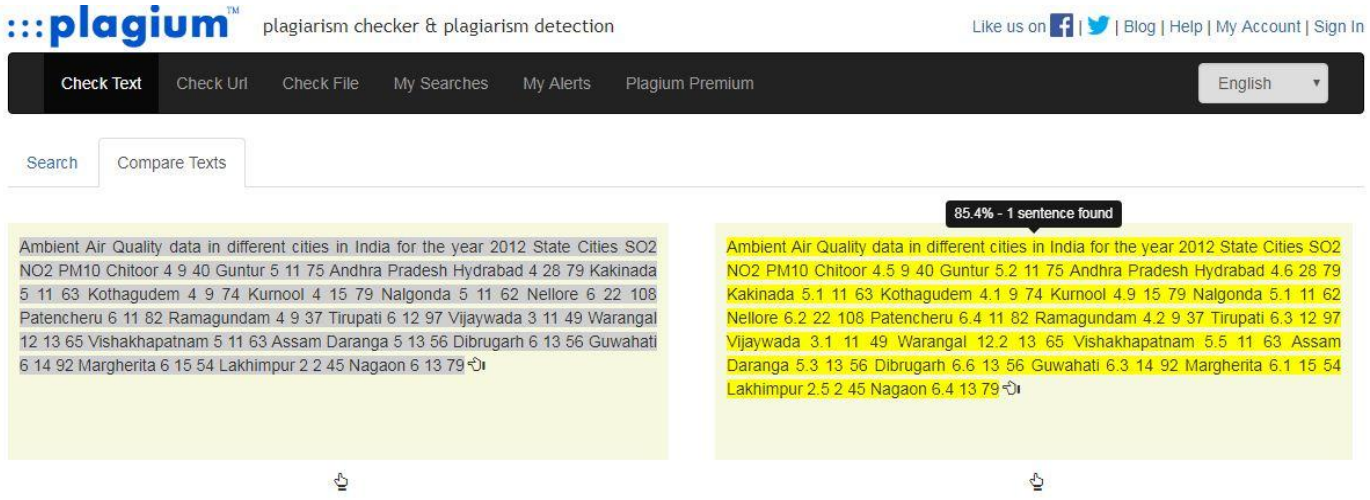

Figure 3

As a first impression, it is seen that plagium is either capable of holding slight changes in quantitative values or not at all sensitive towards value changes. We created a third content where SO_2 values are largely changed and this is tested against Content1-

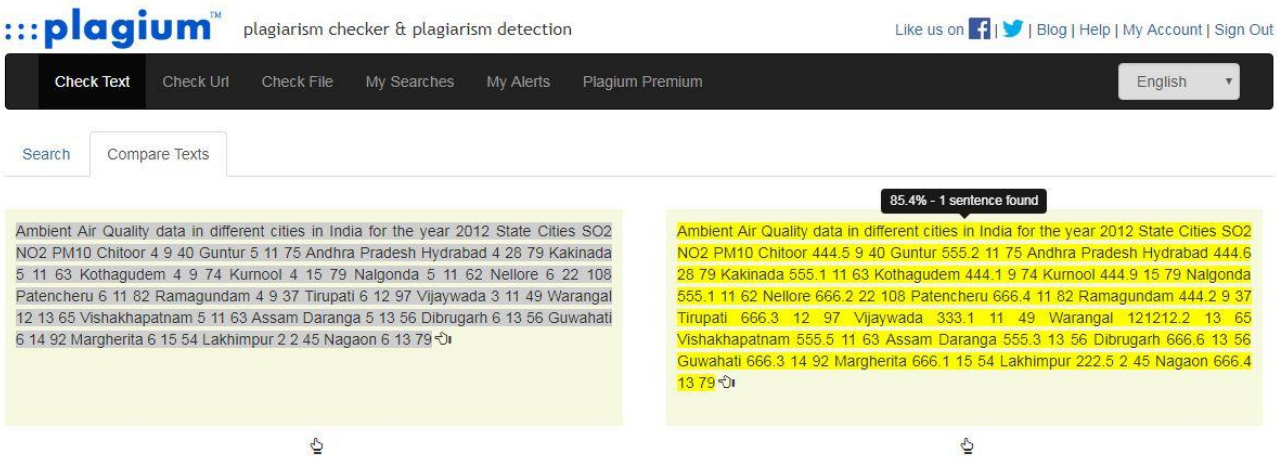

Figure 4

This also shows 85.4% similarity in figure 4 and thus indicates that plagium is not sensitive towards changes of values. Similar behaviour found for Urkund also.

OUR APPROACH FOR MARGINAL ALTERATION IN VALUES:

We have designed an algorithm to tackle minimal changes in values. Different contents may have different ranges of data. For example, in figure 1, SO_2 has data range between 12 and 2. In another content which have population data of Indian states, it may range between 207,281,477 (Uttar Pradesh) and 64,429 (Lakshadeep). [4] .Hence our approach is to normalize the data set which is done by standard formula.

Normalized data for X is calculated as –

$$Xnormlized = a + \frac{(X-A)(b-a)}{(B-A)}$$

This is called feature Scaling .Here,

A – smallest number of the data set

B- largest number of the data set

a- smallest no of the normalized data range

b- largest no of the normalized data range

x- data to be normalized.

If we want to get normalized values between 0 and 1

The formula become-

$$Xnormlized = \frac{(X-A)}{(B-A)}$$

After normalizing the data sets, difference between the corresponding data are calculated. If absolute value of the difference is less than a threshold, corresponding data are considered to be the same.

The algorithm is shown below-

Step1: Array1={Quantitative values of Content1}

Step2: Array2={Quantitative values of Content2 }

Step3: //sorting in ascending order. Smallest no. is sorted first in the inner for loop.

```
n= no. of elements of Array1.
for(i=0;i< n; i++)
  {
    for(j=i+1;j< n; j++)
    {
      if(Array1[i]> Array1 [j])
      {
        temp    = Array1 [i];
        Array1 [i]  = Array1 [j];
        Array1 [j]  =temp;
      }
    }
  }
```

Step4 : // normalized value calculated as X normalized $= a + \frac{(X-A)(b-a)}{(B-A)}$
//and replacement of Array 1 values with normalized values.

```
A= Array[0]; B= Array[n-1];
```

a= user defined minimum normalized value

b= user defined maximum normalized value.

```
for(i=0;i<n; i++)
```

{

Array1[i]=a+(((Array1[i]-A)(b-a))/(B-A))

}

Step 5: Repeat Step3 and step4 for Array2.

Step 6: // Item by item comparison of the two arrays. If absolute value of the difference between corresponding items (i.e quantitative content) is less than threshold then they are made same by replacement.

For(i=0;i<n;i++)

{

If (ABS(Array1[i]-Array2[i])<Threshold)

Array2[i]=Array1[i] ;

}

Step 7: Remove all quantitative values from Content1 and place Array1 values into it.

Step 8: Remove all quantitative values from Content2 and place Array2 values into it.

Step 9: Determine similarity between Content 1 and Content 2 by Proposed Similarity measure (Baruah and Mahanta) as

S=1-(|Content1-Content2|/|Content1|) [5]

Step 10: If calculated similarity is greater than a threshold value, Content2 is plagiarised from Content1.

(Algorithm1:To tackle marginal alteration in values)

This algorithm can detect minimal changes in values to determine plagiarism.

TABLE ROW/COLUMN ELIMINATION:

Some writers, instead of copying an entire table, eliminate some rows or columns and prepare a paragraph as its own. This decreases similarity percentage and plagiarism may not be detected. Let us consider a vehicle research result that lists how fast a vehicle reaches a speed of 100km/hr from 0.Considering different factors like Tires, elevation above sea level, weight of the driver, equipment used for testing, weather conditions and surface of testing track, researchers prepared the following list.[7]-

Table 1: Content3 (X)

Car	Fuel	Execution Time(0-100km) Second
Bugatti_Chiron	Diesel	2.3
LaFerrari	Petrol	2.4
Lamborghini_Huracan	Diesel	2.5
BMW_M5	Diesel	2.8

Now, another writer , by eliminating second column and fifth row prepares the following paragraph –

Bugatti_Chiron Car has Execution Time(0-100km) of 2.3 second. LaFerrari has 2.4 and Lamborghini_Huracan has 2.5 second.

This paragraph, Content4 (Y), is plagiarised. But, different similarity measures show low similarity percentage. For, example, Jaccard similarity shows-

$$J(x,y)=\frac{|x\cap y|}{|x\cup y|} \quad =10/18=0.55$$

55% similarity between Content3 and Content4. If we follow our algorithm, [5] –

S=1-(|X-Y|/|X|),here,
|X-Y|=|{BMW_M5,Fuel,Diesel,Petrol,2.8}|=5
And |X|=15

S=1-(5/15)=0.66 . This also shows low 66% similarity which may be less than threshold and thus plagiarism may not be detected.

OUR APPROACH FOR ROW/COLUMN ELIMINATION:

In our approach to tackle this, we are putting index values to all the tabular cells as RiCi. Here Ri is ith row and Ci is ith column.

Car (R1C1)	Fuel (R1C2)	Execution Time(0-100km) Second (R1C3)
Bugatti_Chiron (R2C1)	Diesel (R2C2)	2.3 (R2C3)
LaFerrari (R3C1)	Petrol (R3C2)	2.4 (R3C3)
Lamborghini_Huracan (R4C2)	Diesel (R4C2)	2.5 (R4C3)
BMW_M5 (R5C1)	Diesel (R5C2)	2.8 (R5C3)

If in content 4 there is not even a single data word with index Rj, we eliminate the row Rj from Content3. Same way we can

eliminate column Cj if there is not a single data word in content4 with index Cj. Applying this, we get the Content4 as-

Car (R1C1)	Execution Time(0-100km) Second (R1C3)
Bugatti_Chiron (R2C1)	2.3 (R2C3)
LaFerrari (R3C1)	2.4 (R3C3)
Lamborghini_Huracan (R4C2)	2.5 (R4C3)

Following algorithm is designed to implement the idea-

Step1: Remove all stop words from paragraph.

Step2: Constitute Set Y from all remaining words of the paragraph after step1.

Step3: Constitute Set X from the words of the table.

Step4: Provide index value to all cell data of the table as RiCi where Ri is ith row and Ci is ith column.

Step5:

n= total no of rows.

m=total no of columns.

```
for Ri where i=1 to n
{   count_variable=0;
    for Cj where j=1 to m
     {
      if cell_data(RiCj) Not= any of elements of X
       count_variable=count_variable+1;
     }
  if count_variable=m
  remove all data of Ri from X
}
For Cj where j=1 to m
{ count_variable=0;

                    For Ri where i=1 to n
{
 if cell_data(RiCj) Not= any of elements of X
       count_variable=count_variable+1;
     }
  if count_variable=n
  remove all data of Cj from X
}
```

Step6: Determine similarity between X and Y by Proposed Similarity measure (Baruah and Mahanta) as S=1-(|X-Y|/|X|) [5]

Step7: If calculated similarity is greater than a threshold value, Y is plagiarised from X.

(Algorithm2: to tackle row/column elimination)

Similarity between Content3 and Content4 determined by algorithm2 is 100%. Hence the algorithm is capable of detection of intelligent plagiarism committed in terms of row/column elimination from table.

CONCLUSION

Availability and ease of access to web content is the prime reason of increase in plagiarism. Although plagiarism detection tools are developed, most of them are not able to handle intelligent plagiarism. Different methodologies and techniques are being developed and this work is a small contribution towards this domain. Further work may be initiated to apply these algorithms to develop Graphical User Interface to detect plagiarism.

REFERENCES

[1] https://en.wikipedia.org/wiki/Plagiarism, last visited on March,2018

[2] http://www.tezu.ernet.in/notices/plagiarism.pdf. Notice dated 26/9/2012

[3] Understanding Plagiarism Linguistic Patterns,Textual Features, And Detection Methods By Salha M. Alzahrani, Naomie Salim, And Ajith Abraham, *Senior Member*, *IEEE*. IEEE Transactions on Systems, Man, and Cybernetics—part c: Applications and Reviews, vol. 42, no. 2, March 2012

[4] https://en.wikipedia.org/wiki/List_of_states_ and_union_territories_of_India_by_population last seen on March,2018

[5] Dhrubajyoti Baruah, Anjana Kakoti Mahanta, "A New Similarity Measure with Length Factor for Plagiarism Detection", International Journal of Computer Applications (0975 – 8887) Volume 72– No.14, May 2013

[6] https://en.wikipedia.org/wiki/List_of_fastest_ production_cars_by_acceleration last visited in march,2018.

[7] "SPLAT:A system for self plagiarism detection" by Christian collbarg, steven Koubhorov,Josua Louie and Thomas slattery, dept. of Computer Science, University of Arizona, Tuscan, AZ 85721

[8] Suphakit Niwattanakul, Jatsada Singthongchai, Ekkachai Naenudorn and Supachanun Wanapu, "Using of jaccard Coefficient for Keyword Similarity ", published in Proceedings of the International MultiConference of Engineers and Computer Scientists 2013 Vol I, IMECS 2013, March 13 - 15, 2013, Hong Kong

[9] B. Karthikeyan, V. Vaithiyanathan, C. V. Lavanya of Sastra University, India – "Similarity Detection in Source Code Using Data Mining Techniques" published in European Journal of Scientific Research ISSN 1450-216X Vol.62 No.4 (2011), pp. 500-505 © EuroJournals Publishing, Inc. 2011

[10] Salha M. Alzahrani, Naomie Salim, and Ajith Abraham, Senior Member, IEEE , Understanding Plagiarism Linguistic Patterns,Textual Features, and Detection Methods, published in Ieee Transactions On Systems, Man, And Cybernetics—Part C: Applications And Reviews, Vol. 42, No. 2, March 2012