Content-based Plagiarism Detection in Korean Document Using Ferret’s Trigram

Byung Ryul Ahn, Won-gyum Kim, Moon-Hyun Kim

Abstract

Document plagiarism means the unauthorized use of the original document of another author without recognition of the source. With the development of the Internet, the volume of digital information available and easily accessible has increased massively and detecting plagiarism manually is so expensive in terms of both time and effort. Although many copy detection techniques for digital document already have been released, their performance is still unsatisfactory. This paper proposes content-based copy detection for Hangul (Korean character) documents by improving the detection accuracy of existing Ferret’s trigram. The key of the proposed system to identify plagiarism is to use two elements: firstly the number of matching trigrams in the original document and secondly the weighting factor of the trigrams where they match sequentially. In this study we show that the proposed system is developed further by weighting results depending on the matching degree of trigram, thereby improving the accuracy of similarity detection in Hangul documents.

Keywords: Document Plagiarism, Copy detection, Copyright protection

1. Introduction

Diffusion and spread of the Internet has brought a major change in society as a whole. Access environment of the public media has changed to high-speed network channels such as Internet newspaper, IPTV and digital movies. This provides a wide range of opportunities for public users to use new content easily and also provides many chances for content providers to produce new content distribution businesses. In 21st century cultural content industry will determine the competitiveness of the nation, and in order to strengthen the competitiveness of the industry copyright-protected content should be done more than anything else in this way.

Document plagiarism means the unauthorized use of the original document of another author without recognition of the source, as if the plagiarist had created them. Examples of document plagiarism include the arbitrary use of information obtained from the media and the unauthorized reuse of the program source codes of another author without references. Plagiarism Detection is one of the most essential parts for copyright protection of digital document. However, with the development of the Internet, the volume of digital information available and easily accessible has increased massively, and detecting plagiarism manually is so expensive in terms of both time and effort. We, therefore, need techniques that can automatically detect document plagiarism. Although many copy detection techniques for digital document already have been released, their performance is still unsatisfactory in terms of accuracy.

In this paper we propose an algorithm to improve the detection accuracy of an existing program, which is Ferret’s trigram. The key idea of the algorithm to identify plagiarism is to use two elements: firstly the number of matching trigrams in the original document, and secondly weighting factor of the trigrams where they match sequentially. Chapter 2 briefly reviews previous studies related to plagiarism detection and Chapter 3 examines the Hangul document plagiarism detection system proposed in this paper. In Chapter 4 we discuss the performance of the proposed algorithm through example cases and Chapter 5 presents conclusions.

2. Related works

Many existing plagiarism detection systems work on the basis of content such as words and sentences of documents. In those systems, the comparison unit of the system compares original
and query documents and calculates the similarity between them. In COPS [1], original documents are broken up into sentences or sequences of sentences. Subsequent query documents are broken up into the same way and are compared against the original documents. As COPS performs comparisons in sentence units, its detection performance is greatly lowered by even minor variations. SCAM is based on word frequencies in documents [2]. SCAM performs word-unit examinations and may falsely detect plagiarism for documents on similar subjects. Semantic Sequence Kernel (SSK) detects plagiarism by finding the semantic sequences in a document and comparing them [3][4]. SSK produces fewer detection errors than these above two systems, but its algorithm, necessitating the discovery of semantic systems, is complex. PPChecker uses synonym information in plagiarism detection and presents plagiarism patterns [5]. It, however, exhibits weakness in the discrimination of sentence structure variations.

KOALA (Korea University Report Police System) was developed for copy detection of Hangul document at the Korea University [14]. After it extracts the Web documents by searching the Internet with multiple queries extracted from a document to be validated plagiarism on a system that relevant, it calculates the degree of similarity and determines whether plagiarism using the index words extracted from Web documents and query document.

The basic core of Ferret was developed by Lyon, Malcolm and Dickerson [6][7][9]. The main characteristic of Ferret is to provide very fast and fine-grained similarity detection in moderately large collections of documents by using triples of tokens to compute a measure of resemblance between each pair of documents in the collection. It compares every text with each other based on counting the number of distinct trigrams similar between the texts, and produces a list of file-pairs together with the similarity scores that ranked from the most similar pair to the least similar one. This count is used to calculate the resemblance measure, as the number of similar trigrams in a pair of documents, divided by the total number of different trigrams in the pair. In this paper we use the Ferret’s trigram. From the next section we describe how the Ferret’s trigram applies to detect plagiarism in Hangul documents.

3. Proposed system

In this chapter we explain the proposed plagiarism detection in Hangul documents based on the Ferret system that compares trigrams. As the detection targets are Hangul documents and they have many word-ending variations, we need to isolate morphemes beforehand. The isolated morphemes comprise the units of comparison in each trigram unit; the trigram units are then used for comparison between original and query Hangul documents. The following sub-sections describe the system architecture and comparative analysis processes of the proposed detection in detail.

3.1. System architecture

The proposed detection system consists of six modules described in Fig. 1. The functions of each module are as follows:

1) **Morphemic Parser**
   - It receives the queried document and the original document, and breaks them down into morphemes.

2) **Trigram Generator**
   - It organizes a set of trigrams from the set of morphemes of each document by grouping three morphemes into one trigram.

3) **Trigram Comparison Analyzer**
   - It compares trigrams, one by one, between the two documents. A set of all trigrams and a set of trigrams common to both documents are obtained. This system gives different weights to the matches based on the number of sequentially matched trigrams, and then determines the number of matches.

4) **Similarity Calculator**
   - Similarity is calculated from the values produced by the Trigram Comparison Analyzer. The similarity value 1 is obtained for a perfect match; and the more similar the documents are, the closer the similarity value is to 1.
5) **Plagiarism Decider**
   - The existence of plagiarism is decided based on the value produced by the Similarity Calculator being above an agreed reference value.

6) **Report Generator**
   - A report is generated, which summarizes the matching trigrams and provides information regarding the plagiarism outcome.

![Diagram](image)

**Figure 1.** Basic architecture of the proposed system

### 3.2. Morphemic Parser

The general definition of a morpheme is the smallest unit of meaning. Morphemes can be classified into grammar morphemes and word morphemes by their meanings or functions. Word morphemes are morphemes that have word meanings and indicate or describe objects, statuses or actions. Typically, nouns, verbs, adjectives, and adverbs belong to them. For example, in the Hangul sentence, "pyojeol tamjineun geunjeol doieoyahanda," "pyojeol," "tamji," "geunjeol," "doi," and "ha" are examples of word morphemes.

Grammar morphemes are those that have grammatical meanings. They are used together with word morphemes and show their relationships. In Hangul they are generally suffixes. In the above example, "neun," "eoya," and "da" are examples of grammar morphemes.

In this paper we did not develop a new morphemic parser, as the existing one was adequate. The Intelligent Morphemic Parser v1.0, developed by 21st Century Sejong Plan [8], is used for the proposed system. Using this we extracted morphemes from a Hangul document and removed the grammar morphemes to leave only word morphemes in order to exclude the highly variable character of Hangul grammar morphemes to improve plagiarism detection accuracy. In the example sentence given above only the five word morphemes, "pyojeol, tamji, geunjeol, doi," and "ha" would be used for plagiarism detection.

### 3.3. Trigram comparison morphemic parser

#### 3.3.1. Overall steps
1. Receive documents for comparison.
2. Use the words in each document as tokens, but with the endings removed.
3. Organize each document into a set of trigram units.
4. If there are \( M \) trigrams in the original document, number the trigrams from 1 to \( M \), and compare each of them with each of the trigrams in the queried document.
5. When the trigrams of the two documents match, update the intersection until a pair of trigrams in the sequence does not match, and add 1 to the count variable (perform the counter \((i, j)\) reflexive statement). If a sequence of five trigrams matches in a row, by the repetitive matching of \( i \) and \( j \), the value \( \sum \) becomes 15. Thus the higher the number of the sequential matches the greater the weight in proportion to \( k^2 \).
6. Calculate similarity, as follows:

\[
\text{If } x = \frac{n(A \cap B)}{n(A \cup B)} \text{ (rate of discovered trigrams), and } y = \frac{\sum}{\text{blockNum}} \text{ where } \text{blockNum} \text{ is the number of matching trigram groups, and } \text{contNum} \text{ is the number of sequentially matching trigram groups, (in the second equation, the numerator is the final value of the variable count, and the denominator corresponds to the perfect match. The higher the number of sequential matches and the more the matching trigrams there are, the greater the } y \text{ value.), the geometric mean } \sqrt[2]{\frac{x y}{x + y}} \text{ will be deemed the degree of similarity.}
\]

3.3.2. Matching algorithm

The basic idea of matching algorithm depicted in Fig. 2 is to compare each trigram in the original document with each trigram in the queried document. If the identical trigram is discovered, the next trigrams are compared until the trigrams do not match, and the count value is increased by \( \sum \) (\( \alpha \) is the number of sequentially matching trigrams). When a pair of trigrams does not match, comparison of separate pairs resumes until the next matching pair of trigrams from each document is found. Even if the number of matching trigrams is identical, a document with a higher continuity of matching trigrams is considered more similar to the original document than one with lower continuity, and the accuracy of plagiarism detection is thus improved.

![Figure 2. Matching steps based on the continuity of trigrams](image)

4. Experiment and Discussion

The performance of the proposed algorithm was tested as follows. Parts of the poems of \textit{Kim Swoel} were used as the first original document and the author’s own poem, containing some parts of \textit{Swoel’s} original poems, both changed and unchanged, together with a simple
Content-based Plagiarism Detection in Korean Document Using Ferret’s Trigram
Byung Ryul Ahn, Won-gyum Kim, Moon-Hyun Kim

explanation of the poem “Azalea” were used as the two queried documents. The second original document contained a simple explanation of data mining, and the two queried documents consisted of a plagiarized version of the original with some changes and a non-plagiarized document about the same subject. Thus, three of the queried documents contain plagiarism or quotations, and one was a non-plagiarized document on the same subject.

First, each document was reorganized into sets of morphemes. These were mechanically classified using the existing “Intelligent Morphemic Parser.” Then, three morphemes from the set of morphemes were grouped in a duplicated manner to create trigram sets. After that, the union and intersection of the two documents being tested were obtained and continuity weights were measured in accordance with the proposed algorithm. They were then put into the similarity calculator to determine similarity. The following shows the original Hangul and query documents used in this experiment.

---

<Original document 1 - text of the poems by Kim Soweol>

어체도 하루 벌 나그네 집에 가서 가까 가까가막 물여 새셨소. 오늘은 또 몇 십리 어리도 갈까.
산으로 올라가거든 들을 갈까 오는 것이 아닌가요. 말 마소. 내 집도 정주 꽃산 자고 바가는 곳이라오.

<Queried document 1-1 - Author’s own poem that quotes the poems by Kim Soweol>

[Queried document text here]

<Queried document 1-2 - Explanation of “Azalea” the poem by Kim Soweol>

Understanding and impression of “Azalea” the poem by Kim Soweool

This poem is the essence of Soweol’s poems. It artistically sublimates the traditional “affection and regret” through a woman who overcomes the sadness of separation through her will for resignation. This “affection and regret” is in line with the traditional emotions of Korean people, which have been expressed ceaselessly through such poems as “Gongmudohaga,” “Gasiri,” “Seogyeongbyeolgok,” and “Arirang.”

As we all know, the traditional Korean emotions of “regret and grief” and the traditional rhythms of Korean folk songs are the two main elements of, and inspirations for Soweol’s poems. Over 130 poems left by Soweol, who is regarded as the greatest Korean poet, were published in the collection “Azalea,” before his death. Another collection of his poems “Soweolsicho” was published after his death by Kim Eok and numerous others have been published since and become best sellers.

<Original document 2 - Data mining>

Summary: 많은 데이터 가운데 승출되는 유용한 상관관계를 발견하는 것. <대 데이터 해야하게우 데이터 마트가 사용자가 원하는 데이터들을 미리 만들어놓고 이를 가져올 수 있도록 하는 개념인 반면, 각 데이터의 상관관계를 만능 지능 기법을 통해 자동적으로 받아주는 것. 예를 들어, 비용 결정하는 사항에 대해 데이터가 있고 색깔에 대한 선호도와 관련된 데이터가 있다면 이 두의 관계를 발견 내는 기능을 수행한다. 즉 정확히 수치화하기 힘든 데이터 간의 연관을 찾아내는 역할을 한다. 데이터베이스에서 간략한 사건별 데이터 모델을 발견하여 미래에 정확한 정보를 추출해 내고 의사 결정에 이용하는 것이 일반적이다. 즉 데이터의 승출된 패턴과 관계를 찾아내며 관계를 찾아내어 정보를 발견해 내는 것이므로. 여기에서 정보 발견이란 데이터의 고급 통계 분석과 모델링 기법을 적용하여 유용한 패턴과 관계를 찾아내는 과정이다. 데이터베이스 마케팅의 핵심 기술이라고 할 수 있다.>

<Queried document 2-1 - Data mining (non-plagiarized text)>

[Queried document 2-1 text here]
The procedures for similarity calculations using the above documents are described below.

1) **Original Document 1 vs. Queried Document 1-1**
   Number of trigrams in the intersection of the two documents = 16
   Number of trigrams in the union of the two documents = 51 (number of trigrams in queried document) + 216 (number of trigrams in original document) – 16 (number of trigrams in the intersection between the two documents) – 14 (number of identical trigrams in original document) = 237, Because \( x = 16/237 = 0.0675 \) and \( y = 561/7260 = 0.0773 \), Similarity = 0.0490.

2) **Original Document 1 vs. Queried Document 1-2**
   Number of trigram intersections between the two documents = 33
   Number of trigrams in the union of the two documents = 120 (number of trigrams in queried document) + 216 (number of trigrams in original document) – 33 (number of trigrams in the intersection of the two documents) – 14 (number of identical trigrams in original document) - 3 (number of identical trigrams in queried document) + 3 (number of identical trigrams in the intersection of the two documents) = 289, Because \( x = 33 / 289 = 0.1142 \) and \( y = 561 / 7260 = 0.0773 \), Similarity = 0.0922.

3) **Original Document 2 vs. Queried Document 2-1**
   Number of trigrams in the intersection of the two documents = 1
   Number of trigrams in the union of the two documents = 120 (number of trigrams in queried document) + 188 (number of trigrams in original document) – 1 (number of trigrams in the intersection of the two documents) = 307, Because \( x = 1 / 307 = 0.0033 \) and \( y = 1 / 7260 = 0.0001 \), Similarity = 2.6431^-4 (approx. 0.00264).

4) **Original Document 2 vs. Queried Document 2-2**
   Number of trigrams in the intersection of the two documents = 15
   Number of trigrams in the union of the two documents = 84 (number of trigrams in queried document) + 216 (number of trigrams in original document) – 15 (number of trigrams in the intersection of the two documents) = 285, Because \( x = 15 / 285 = 0.0526 \) and \( y = 120 / 7260 = 0.0165 \), Similarity = 0.0252.

In the four tests, the similarity values of the plagiarized documents ranged from 0.0252 to 0.0922 and the similarity value of the non-plagiarized document was 0.00264. Thus, the tests demonstrated that the higher the degree of plagiarism, the higher the similarity value. The similarity was the highest between original document 1 and queried document 1-2, which had the longest sequences of matching trigrams. Between queried documents 1-1 and 1-2, 1-1 had a slightly higher trigram matching rate, but document 1-2 had longer sequences of matching.
trigrams because it copied the whole poem as it was; the similarity of document 1-2 to the original was thus greater. The plagiarism of document 1-2 from the original is more conspicuous, which shows the validity of this algorithm taking sequences of matched trigrams into account.

What, then, is the minimum threshold value of similarity that a document must have to be regarded as plagiarism? There is no absolute standard for this: it varies according to the type, size, subject, and other features of the documents. To get more accurate data, this study conducted more tests using 14 documents similar to those described above. Among 14 test documents there are 3 plagiarized documents. As a result, the following similarity values were obtained and showed in Table 1.

Table 1. The similarity values of plagiarized and non-plagiarized documents

<table>
<thead>
<tr>
<th>Original document No.</th>
<th>Plagiarism or not</th>
<th>Similarity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>O</td>
<td>0.0591</td>
</tr>
<tr>
<td>2</td>
<td>O</td>
<td>0.1007</td>
</tr>
<tr>
<td>3</td>
<td>O</td>
<td>0.1822</td>
</tr>
<tr>
<td>4</td>
<td>O</td>
<td>0.0260</td>
</tr>
<tr>
<td>5</td>
<td>O</td>
<td>0.0966</td>
</tr>
<tr>
<td>6</td>
<td>O</td>
<td>0.0915</td>
</tr>
<tr>
<td>7</td>
<td>O</td>
<td>0.1003</td>
</tr>
<tr>
<td>8</td>
<td>O</td>
<td>0.1363</td>
</tr>
<tr>
<td>9</td>
<td>O</td>
<td>0.0490</td>
</tr>
<tr>
<td>10</td>
<td>O</td>
<td>0.0922</td>
</tr>
<tr>
<td>11</td>
<td>O</td>
<td>0.0252</td>
</tr>
<tr>
<td>12</td>
<td>X</td>
<td>0.0000</td>
</tr>
<tr>
<td>13</td>
<td>X</td>
<td>0.0002</td>
</tr>
<tr>
<td>14</td>
<td>X</td>
<td>0.0026</td>
</tr>
</tbody>
</table>

In Table 1, document no. 12, 13 and 14 mean similarity values of non-plagiarized documents and the others show those of plagiarized or quoted documents. From the above experiments we can estimate that the reference similarity value of plagiarism is approximately 0.005. The above examples of plagiarism and quotation represent very high degrees of plagiarism. The similarity value will decrease for documents with lower degrees of plagiarism. The example of non-plagiarized documents in this study could have a high degree of similarity because they covered a similar aspect of the same subject. Considering these two cases, we could view the value 0.005 as the threshold of plagiarism detection. We can confirm the validity of this value through more experiments.

5. Conclusion

In this paper we presented content-based plagiarism detection for Hangul (Korean character) by improving the detection accuracy of existing Ferret’s trigram. The key characteristic of the conventional Ferret algorithm is the comparison of trigrams. In this study we examined the procedures for applying this idea to Hangul documents. To do this, we developed additional steps of parsing documents into morphemes. This study proposes improved detection performance by considering the length of sequences of matching trigrams, and demonstrated the effectiveness of this new technique comparing with the evaluation of similarity by the number of matching trigrams based on the Ferret algorithm. Even though the evaluation of similarity threshold for deciding plagiarism has a wide variety according to the characteristics of the documents under review, we could estimate an applicable threshold generally through experiments. The accuracy of this estimation could be improved through comparing a range of sample types. We need to take other considerations into account such as the exchange of similar words and more research is necessary to improve the accuracy of plagiarism detection system further.
References

Appendix I. Algorithm pseudo code based on C

```c
// Counts the number of sequentially matching trigrams based on the indices i and j.
int counter(int i, int j)
{
    if( i>M || j>N || origin[i] != query[j] )
        return 0;  // Returns zero if the end of one document is reached or two trigrams don’t match.
    else
    {
        if( origin[i] Intersection[] )          // If the trigram is not in the intersection
            Intersection[] = origin[i];  // Insert it in the intersection of trigrams.
        return counter(i+1, j+1) + 1;
        // If the compared trigrams match, the count is increased by 1, and this is reflexively performed to compare the trigrams of two documents.
    }
}

Main()
{
    // This variable increases by 1 each time trigrams match.
    count = 0;
    // The set of trigrams in the original document with word endings removed
    origin[] = { set of trigrams in the original document }
    // The set of trigrams in the queried document with word endings removed
    query[] = { set of trigrams in the queried document }
    Intersection[] = { }; // the sequential array of the trigrams in the intersection
    Union[] = { the union of trigrams in the two documents }; // the array of the union of trigrams
    similarity = 0; // This variable stores the similarity which is the final target.
    for(i=1; i<=M; i++)
        for(j=1; j<=N; j++)
            count += counter( i, j );
            // Counts the number of sequentially matching trigrams starting from the indices i and j, and stores it in count.
    // The rate of the intersection against the union of the trigrams in the two documents
    x = sizeof(Intersection[])/sizeof(Union[]);
    // The rate of weight by the continuity of matching trigrams
    y = count / ∑_{i=1}^M \sum_{j=1}^N // Similarity is determined by the geometric mean of x and y.
    similarity = x y // Plagiarism is decided by the threshold
    if(similarity > threshold)
        return COPIED;
    else
        return INDEPENDENT;
}```