ISSN 1870-4069

Efecto del pre-procesamiento en la detección automática de plagio para PAN 2014 y PAN 2015

Jovani Armeaga García, Yulia Ledeneva, René Arnulfo García-Hernández

Universidad Autónoma del Estado de México,
México

UAP Tianguistenco Instituto Literario, Toluca, Estado de México,
México

jovani_2807@hotmail.com, yledeneva@yahoo.com, renearnulfo@hotmail.com

Resumen. Dentro de la detección automática de plagio, el alineamiento de texto en [1] lo define como el descubrimiento de fragmentos similares de texto entre dos documentos. La cual puede utilizarse en: detección de plagio, identificación de autoría, detección de reúso de texto, recuperación de información, entre muchas otras. El pre-procesamiento consta de diversas técnicas que se aplica en la mayoría de las tareas del Procesamiento del Lenguaje Natural (PLN), en este caso, las heurísticas presentadas son tomadas de los trabajos [1] y [2] de las mejores participaciones en la competencia internacional de detección automática de plagio PAN 2014 y PAN 2015 en la sub-tarea alineamiento de texto monolingüe, con la finalidad de conocer el efecto que tiene la eliminación de *stopwords* y el uso o no de *stemming* en las heurísticas antes mencionadas, que son técnicas dentro del pre-procesamiento.

Palabras clave: Procesamiento de lenguaje natural, alineamiento de texto, detección automática de plagio, competiciones PAN 2014 y PAN 2015.

1. Introducción

El PLN es una sub-disciplina de la inteligencia artificial y rama de la ingeniería lingüística computacional, la cual busca construir sistemas y mecanismos que permitan la comunicación entre personas y máquinas por medio de lenguajes naturales. El lenguaje natural en [3] se entiende como el lenguaje hablado y escrito que tiene como propósito que exista una comunicación entre una o varias personas. Algunas de las aplicaciones del PLN son:

- Recuperación de información,
- Traducción automática,
- Extracción de información.

La recuperación de información según [4] es, teniendo una necesidad de información y un conjunto de documentos, se ordenan los documentos por relevancia para esa necesidad y se presenta un sub-conjunto de los más relevantes. Según [5] dice que

"cualquier sistema de recuperación de información puede ser descrito como un conjunto de ítems de información (DOCS), un conjunto de peticiones (REQS) y algún mecanismo (SIMILAR) que determine qué ítem satisface las necesidades de información expresadas por el usuario en la petición".

De la recuperación de información se desprenden diversas tareas como:

- Generación automática de resúmenes,
- Detección automática de plagio,
- Extracción automática de palabras clave.

En la detección de plagio, en [6] el plagio se define como, copiar en lo sustancial obras ajenas, dándolas como propias, es una de las definiciones más aceptadas, lo que se refleja en los trabajos [7, 8]. Una obra es toda creación original que puede reproducirse por cualquier medio o forma [7]. El plagio puede aparecer en diversas obras como se menciona en [9]:

- Literarias,
- Fotográficas,
- Programas de cómputo,
- Musicales,
- Arquitectónicas,
- Danzas,
- Esculturas,
- Programas de radio y TV,
- Cinematográficas y audiovisuales,
- Obras de arte.

En los últimos años el uso incremental de los medios digitales, ha provocado el incremento de plagio textual o en documentos, según [8] esto es tomar el texto de un autor y hacerlo pasar como propio. Esto abarca desde la copia sin modificar nada, hasta el parafraseado del documento que modifica las palabras, pero manteniendo la idea central del texto. Esto se debe a la enorme cantidad de información que se encuentra disponible en dichos medios.

Actualmente las instituciones académicas, es en donde más se presenta el plagio en tareas de los alumnos [7], siendo un acto muy poco castigado, surgiendo así los sistemas de detección automática de plagio. Los primeros sistemas en [10, 11] se menciona que fueron WcopyFind desarrollado por la universidad de Virginia, Ferret Plagiarism Detector por la universidad de Hertfordshire y SCAM (*Stanford Copy Analysis Mechanism*) por la universidad de Stanford, estos primeros sistemas no mostraron mucha eficiencia, para la detección de documentos plagiados, al final el humano decide que es plagio y que no.

Un ejemplo en donde se puede aplicar la detección automática de plagio textual para evitar duplicados, es en los sistemas de creación de documentos. En los trabajos [8, 12] mencionan dos formas en las que se puede detectar plagio. La primera es realizando un análisis intrínseco, el cual solo busca cambios de estilos de escritura en un documento; y el basado en documentos de referencia, en donde se comparan documentos sospechosos contra documentos fuente.

Dentro de la detección automática de plagio basado en documentos de referencia se encuentra el alineamiento de texto, es una tarea en donde normalmente, los documentos

contienen diferentes tipos de ofuscación, con la finalidad que el nuevo documento sea similar al original [13]. Generalmente los corpus que existen para la detección de plagio, implementan diferentes técnicas de ofuscación elaboradas por herramientas comerciales, en donde, hay variaciones semánticas de palabras, operaciones con el texto de forma aleatoria. Básicamente el plagio de los corpus es creado de manera artificial.

A partir del 2009 hasta la fecha según [14], PAN es la competición más grande de detección de plagio, identificación de autoría y mal uso de software social. Entre el 2012 y 2015 en la competición PAN, la tarea de detección de plagio se dividió en dos sub-tareas: recuperación de fuentes y alineamiento de texto. Para el alineamiento de texto [13, 14, 15], los sistemas deben identificar todos los pasajes de máxima longitud de texto reusado entre un par de documentos.

Como primera etapa de los enfoques mostrados des PAN 2012 a PAN 2015, se aplican distintas técnicas de pre-procesamiento, ésta es una etapa que se aplica en diversas tareas de PLN. En el caso de alineamiento de texto algunas técnicas que se han aplicado son:

- Eliminación de caracteres especiales: se refiere en [7, 16, 17, 18, 19] a remover los signos de puntuación y algunos caracteres que puedan generar ruido en el documento.
- Eliminación de números: en [20] la importancia de los dígitos no es prioridad y remueve los números que aparezcan en el texto.
- Eliminación de espacios en blanco: En [16, 21] se cambian los espacios en blanco de todo el texto por algún carácter, separando cada *token*, o simplemente se elimina el espacio en blanco.
- Conversión a mayúsculas: en [7] todo el texto se deja en un solo formato, con la finalidad de dejarlo normalizado.
- Conversión a minúsculas: en [1, 2, 18, 19, 20, 21, 22, 24, 25] todo el texto se deja en un solo formato, con la finalidad de dejarlo normalizado.
- Eliminación de *stopwords*: en [18, 19, 21, 22, 24] se menciona que son palabras que por sí solas no dicen nada del documento y pueden ser pronombres, artículos, preposiciones, etc.
- *Stemming*: en [1, 2, 19, 20, 21, 22, 26] obtiene la raíz de una palabra truncando una palabra en relación a otras. Por ejemplo sonrisa, sonríe, sonrían y sonreíste se obtiene la raíz *"sonri"*.
- División en *tokens*: en [1, 2, 18] el texto del documento es fragmentado, es decir, se divide en palabras o *tokens*.

En este artículo, se enfoca a la utilización de diferentes listas de *stopwords* en la etapa de pre-procesamiento, para saber cómo afecta la eliminación de esta información sobre las heurísticas de [1] y [2], las cuales están disponibles en código abierto[1].

[1] http://www.gelbukh.com/plagiarism-detection/PAN-2015

2. Estado del arte

En diversas aplicaciones del PLN se han hecho trabajos sobre pre-procesamiento uno de ellos es el de Ledeneva [27], en donde se analiza la importancia del pre-procesamiento, en la generación automática de resúmenes utilizando secuencias frecuentes maximales. Las técnicas de pre-procesamiento que utilizaron fueron análisis léxico como eliminación de signos de puntuación, normalización de números y algunas variantes de *stopwords* y *stemming*. Se detectó que al probar estas técnicas en el pre-procesamiento no afectan a la calidad de los resúmenes generados que comprueba que el método propuesto es bueno y no depende de la etapa de pro-procesamiento.

Al contrario en el trabajo de [28], se puede observar que al utilizar varias técnicas de pre-procesamiento, se mejoran los resultados considerablemente.

En la detección automática de plagio también se aplican técnicas de pre-procesamiento. En el trabajo de [7] se aborda la comparación de medidas de similitud en cadenas textuales, para identificar plagio en tareas escolares, en donde las técnicas de pre-procesamiento que utiliza son:

- Eliminación de números,
- Eliminación de espacios en blanco,
- Eliminación de signos de puntuación,
- Conversión a mayúsculas.

Aunque este trabajo se enfoca más a obtener una medida de similitud, es notable que inicialmente se aplique una etapa de pre-procesamiento, para reducir principalmente el ruido, que pueda ocasionar algunos caracteres.

Dentro de la competición internacional de plagio, identificación de autoría y mal uso de software social PAN, se desprende una sub-tarea que es alineamiento de texto en donde para la edición de PAN 2014 en [13] reportan que solo once participantes presentaron software para la evaluación y comparación, de los cuales solo diez reportaron la descripción de su enfoque.

Tabla 1. Técnicas de pre-procesamiento utilizadas por los participantes en la tarea de alineamiento de texto PAN 2014.

Pre-procesamiento	[1-2]	[23]	[29]	[16]	[17]	[18]
Eliminación de caracteres especiales	si	no	-	si	si	si
Eliminación de números	no	-	-	-	-	-
Eliminación de espacios en blanco	no	no	-	si	no	no
Conversión a mayúsculas	no	-	-	-	-	no
Conversión a minúsculas	si	no	-	no	no	si
Eliminación de *stopwords*	no	si	-	no	no	no
Stemming	si	no	-	no	no	no
División en *tokens*	si	no	-	no	no	si

Un análisis general de los métodos reportados, revela que para construir un algoritmo de alineamiento de texto se llevan a cabo las siguientes etapas: pre-procesamiento, pre-selección, extensión y filtrado. En esta edición la mayoría de los participantes se enfocó a predecir qué tipo de plagio se les presentaba. En la Tabla 1, se muestran las diferentes

técnicas de pre-procesamiento utilizadas, se marcó con "-", cuando no se menciona que técnica se utilizó.

En la edición de PAN 2013 se reportó la participación de nueve equipos, los cuales presentaron su software para la evaluación y comparación, pero solo seis de ellos presentaron la descripción de su enfoque. De acuerdo a [30] algunos de los equipos que participaron en alineamiento de texto en PAN 2013 usaron las técnicas de pre-procesamiento que se muestran en la Tabla 2, así como las etapas mencionadas anteriormente que son pre-selección, extensión y filtrado. El corpus de entrenamiento tanto en PAN 2013 y PAN 2014 es el mismo.

Tabla 2. Técnicas de pre-procesamiento utilizadas por algunos participantes en la tarea de alineamiento de texto PAN 2013.

Pre-procesamiento	[19]	[22]	[24]	[25]	[20]
Eliminación de caracteres especiales	si	no	no	no	no
Eliminación de números	no	no	no	no	si
Eliminación de espacios en blanco	-	-	-	-	-
Conversión a mayúsculas	no	no	no	no	no
Conversión a minúsculas	si	si	si	si	si
Eliminación de *stopwords*	si	si	no	no	no
Stemming	si	si	no	no	si
División en *tokens*	-	-	-	-	-

Como se puede observar en la Tabla 1 y la Tabla 2, las técnicas de pre-procesamiento utilizadas, en la mayoría son las mismas. Los documentos del corpus de entrenamiento, son en texto plano. Cada equipo utiliza ciertas técnicas conforme al enfoque planteado. En la Tabla 3, se muestran los participantes que reportaron haber realizado alguna técnica de pre-procesamiento en PAN 2012, en general no se dice mucho del pre-procesamiento que realizaron.

Tabla 3. Técnicas de pre-procesamiento utilizadas por algunos participantes en la tarea de alineamiento de texto PAN 2012.

Pre-procesamiento	[21]	[26]
Eliminación de caracteres especiales	si	no
Eliminación de números	-	-
Eliminación de espacios en blanco	si	no
Conversión a mayúsculas	no	-
Conversión a minúsculas	si	no
Eliminación de *stopwords*	si	no
Stemming	si	si
División en *tokens*	-	-

En la primer edición de alineamiento de texto en el 2012 en donde en PAN reporta que once presentaron software para su evaluación y comparación, y solo diez reportaron la descripción de su enfoque, como se muestra en la Tabla 3 no hay muchos detalles acerca de técnicas empleadas de pre-procesamiento. El análisis de los métodos mostró

[15] algo en común, que fue las etapas de como construyen los algoritmos de alineamiento de texto las cuales son: pres-elección, fusión de partes y extracción filtrada. En general estos enfoques están basados en reglas, los cuales en cierta forma limitan la detección de plagio.

En general las técnicas que se utilizan en la preparación de los datos de entrada son muy similares en PAN 2012, PAN 2013 y PAN 2014. Inicialmente en PAN 2012 el corpus tenía diferentes tópicos los cuales se muestran a continuación y se explica con más detalle en [31]:

– No hay plagio (*no-plagiarism*),
– Ninguno (*no-obfuscation*),
– Artificial bajo (*artificial-low*),
– Artificial alto (*artificial-high*),
– Traducción (*translation*),
– Paráfrasis simulada (*simulated-paraphrase*).

Para PAN 2013 y PAN 2014 el corpus se conformó por los siguientes tópicos, siguiendo la misma forma de creación por parte de [31]:

– No hay plagio (*no-plagiarism*),
– Ninguno (*no-obfuscation*),
– Aleatorio (*summary-ofuscation*),
– Traducción (*translation-obfuscation*),
– Resúmenes (*summary-obfuscation*).

Como menciona en [31], los diferentes tipos de plagio que se crearon en su mayoría son elaborados de manera artificial, creados por herramientas comerciales, herramientas en línea, por operaciones de texto, etc.

En la edición de PAN 2015 la tarea de alineamiento de texto cambio un poco en cuanto a la temática, los equipos ahora debían elegir entre generar una colección con casos reales de reutilización de texto o plagio, o teniendo en cuenta los pares de documentos, generar pasajes de texto reutilizado o plagiados. Aplicando un tipo de ofuscación.

Debido a que el trabajo de [1] no se pudo comparar con otro de la edición PAN 2015, y había tenido una mejora del aplicado en PAN 2014 [2], no se declaró como ganador, sin embargo, analizando el enfoque de [1], utiliza las mismas técnicas de pre-procesamiento que [2], siendo hasta el momento el mejor en cuanto alineamiento de texto.

Las medidas de evaluación empleadas en PAN para alineamiento de texto son: *granularity, recall, presicion* y *plagdet.* En [12, 31] se dice que, d_{plg} denota un documento que contiene plagio. Un caso de plagio en d_{plg} es una 4-tupla $s = \{s_{plg}, d_{plg}, s_{src}, d_{src}\}$, donde, s_{plg} es un pasaje plagiado en d_{plg}, y s_{src} es el pasaje original correspondiente en el documento de referencia d_{src}. De forma similar, un caso de plagio detectado se expresa como $r = \langle r_{plg}, d_{plg}, r_{src}, d'_{src} \rangle$; donde r asocia un pasaje supuestamente plagiado r_{plg} en d_{plg} con un pasaje r_{src} en d'_{src}. Decimos que r detecta s si y solo si $r_{plg} \cap s_{plg} \neq \emptyset$ y $d'_{src} = d_{src}$.

Denotamos un documento d como un conjunto de referencias a sus caracteres d = {(1, *d*),…, (|*d*|, d)}, donde (*i*, *d*) refiere el *i*-ésimo carácter en *d*. De esta forma un caso

de plagio *s* puede ser representado como $s = s_{plg} \cup s_{src}$, donde $s_{plg} \subseteq d_{plg}$ y $s_{src} \subseteq d_{src}$. Los caracteres referenciados en s_{plg} y s_{src} forman pasajes s_{plg} y s_{src} en la visión anterior. De forma similar una detección *r* puede ser representada como $r = r_{plg} \cup r_{src.}$ A partir menciona en [12] de esto podemos decir que *r* detecta *s* si y solo si $r_{plg} \cap s_{plg} \neq \emptyset$ y $r_{src} \cap s_{src} \neq \emptyset$. Por último, *S* y *R* denotan conjuntos de casos de plagios y detecciones respectivamente. Basado en estas representaciones *precision* y *recall* de *R* según *S* se define como:

$$prec(S, R) = \frac{1}{|R|} \sum_{r \in R} \frac{|\bigcup s \in S (s \sqcap r)}{|r|},$$

$$rec(S, R) = \frac{1}{|S|} \sum_{s \in S} \frac{|\bigcup r \in R (s \sqcap r)}{|s|},$$

donde

$$s \sqcap r = \begin{cases} s \cap r \text{ si } r \text{ detecta } s, \\ \emptyset \text{ otro caso}. \end{cases}$$

Además de *precision* y *recall* otro concepto importante que caracteriza la eficiencia de un algoritmo de detección de plagio, esto se refiere, si un caso de plagio es detectado como uno solo o en varias partes. Para esto se define *granularity* de *R* en *S*, en donde *S* y *R* denotan conjuntos de casos de plagio y de detecciones. Lo definen con la siguiente fórmula:

$$gran(S, R) = \frac{1}{|S_R|} \sum_{s \in S_R} |R_S|,$$

donde $S_R \subseteq S$ son los casos detectados por las detecciones de *R*, y $R_S \subseteq R$ son las detecciones de un caso *s* dado:

$$S_R = \{s \mid s \in S \wedge \exists r \in R : r \text{ detecta } s\},$$

$$R_S = \{r \mid r \in R \wedge r \text{ detecta } s\}.$$

El dominio de *gran(S, R)* es [1,|*R*|], el 1 indica la correspondencia deseada uno a uno y |*R*| indica el peor de los casos, donde un solo caso $s \in S$, es detectado una y otra vez.

Teniendo en cuenta *precision, recall* y *granularity* que permiten un orden parcial entre los algoritmos de detección. Para obtener un orden general, estas medidas se combinan de la siguiente forma:

$$plagdet(S, R) = \frac{F_a}{\log_2(1 + gran(S, R))}.$$

En donde F_a denota la Medida-F. Para las ediciones de PAN la medida armónica ponderada de *precision* y *recall* es $a = 1$, ya que no hay indicación de que una sea más importante que la otra.

3. Trabajo previo

De acuerdo a los equipos que participaron en alineamiento de texto en PAN 2014 y PAN 2015, en la Tabla 4, se reportan los mejores enfoques con el corpus de evaluación proporcionado por la competencia PAN.

Tabla 4. Resultados de [2], en alineamiento de texto PAN 2014.

Equipo	Plagdet	Recall	Precision	Granul.
Sanchez-Perez15 [1]	0.9010	0.8957	0.9125	1.0046
Sanchez-Perez14 [2]	0.8781	0.8790	0.8816	1.0034
Oberreuter [13]	0.8693	0.8577	0.8859	1.0036
Palkovskii [17]	0.8680	0.8263	0.9222	1.0058
Glinos [18]	0.8593	0.7933	0.9625	1.0169

Tomando como referencia los enfoques [1, 2, 16, 17, 18, 19, 20, 22, 23, 24, 25] en general se deducen las diferentes heurísticas en cuatro etapas: pre-procesamiento, pre-selección, extensión y filtrado.

Pre-procesamiento. Se refiere a preparar el texto antes de ser procesado en las siguientes etapas se puede hacer, por ejemplo: eliminar caracteres no alfanuméricos, eliminar palabras vacías, por mencionar algunas.

Pre-selección. Teniendo un documento sospechoso y un documento origen, el texto se divide en fragmentos, con el fin de encontrar coincidencias en ambos textos.

Extensión. En general, en esta etapa trata de formar pasajes de máxima longitud, con la unión de fragmentos, tanto en el documento origen como en el sospechoso.

Filtrado. Teniendo los pasajes alineados, se eliminan los que no cumplen ciertos criterios, esto se hace con el fin de maximizar el rendimiento de cada método.

Se cuenta con los trabajos de [1-2], los cuales están enfocados a la tarea de alineamiento de texto, en la Figura 1, se muestran las etapas del enfoque de [2].

En la Tabla 5, se muestran varios parámetros los cuales se enlistan a continuación:

- minSentLength: es la longitud mínima de un fragmento de texto, que son 3 palabras,
- th1: corresponde al parámetro de la similitud coseno,
- th2: corresponde al parámetro de la similitud de Dice,
- th3: corresponde al parámetro de similitud de estructura,
- minSize: longitud de pasajes pequeños,
- minPlagLength: longitud de un pasaje para ser considerado plagio,
- maxGap: define cuál es la separación máxima, en número de oraciones, que puede existir en dos oraciones seleccionadas para ser consideradas adyacentes.

Tabla 5. Lista de parámetros de la heurística [2].

Parámetro	Valor	Parámetro	Valor
minsentlen	3	maxgap	4
th_1	0.33	maxgap_least	2
th_2	0.33	minsize	1
Th_3	0.40	minplaglen	150

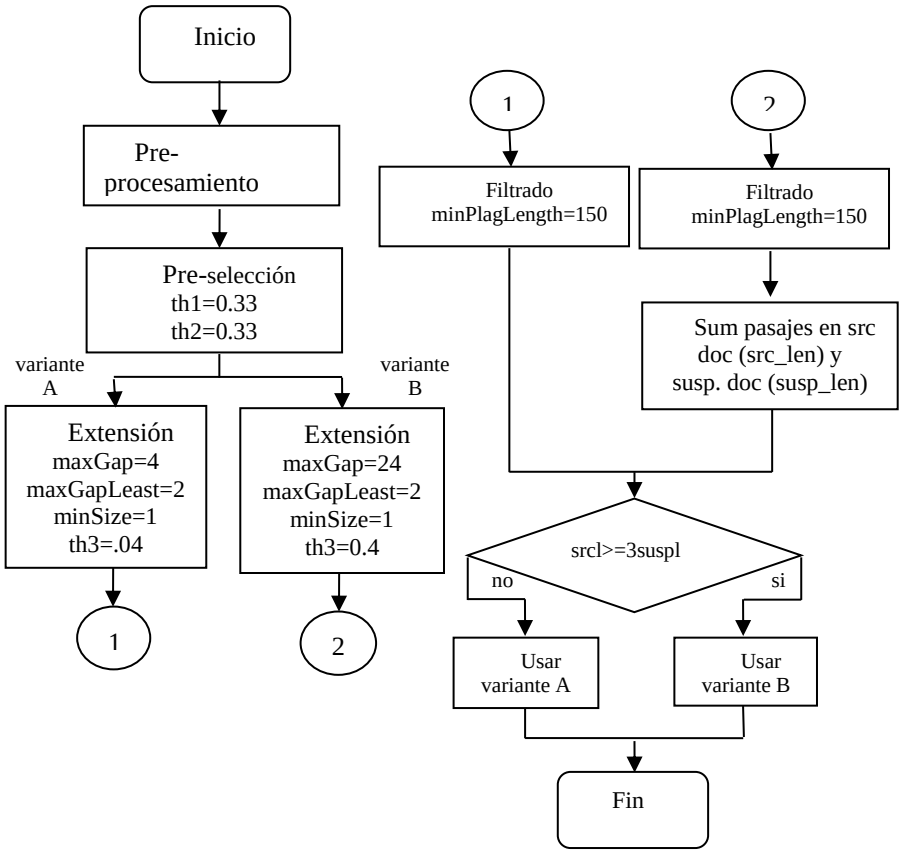

Fig. 1. Enfoque de [2], para la detección automática de plagio mediante alineamiento de texto PAN 2014.

El enfoque de [2] fue el que obtuvo el mejor desempeño en la tarea de alineamiento de texto, en la competencia de detección plagio PAN 2014. Para el 2015, el trabajo de [1] mejoró su enfoque, superando sus resultados obtenidos un año atrás, este enfoque se muestra en la Figura 2.

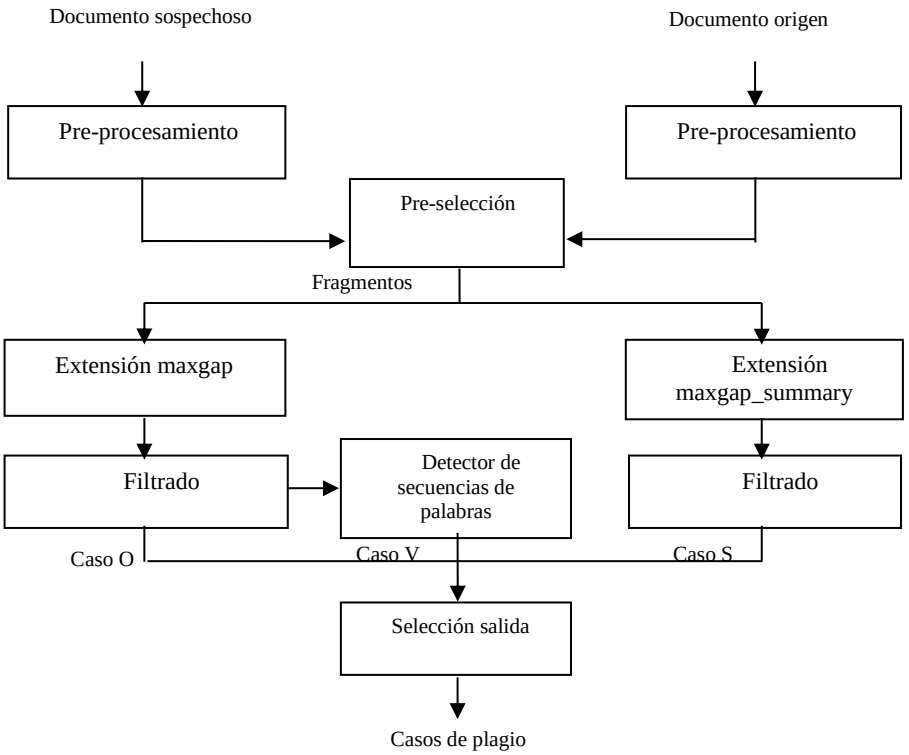

Fig. 2. Enfoque de [1], para la detección automática de plagio mediante alineamiento de texto PAN 2014.

Tabla 6. Lista de parámetros de la heurística [1].

Parámetro	Valor	Parámetro	Valor
minsentlen	3	maxgap_summary	24
th_cos	0.30	maxgap_least	0
th_dice	0.33	minsize	1
th_validation	0.34	minplaglen	150
maxgap	4	th_verbatim	256

Para [1], su enfoque va dirigido a la sub-área del corpus, *summary obfuscation*, siendo una de las más difíciles para detectar plagio. En la Tabla 4, se muestran los mejores resultados en el corpus de evaluación, dentro de la competencia PAN 2014 para alineamiento de texto, el enfoque de [1] es una mejora del enfoque [2], utilizando

el corpus de PAN 2014. Los parámetros del enfoque de [1] son los que se muestran en la Tabla 6.

La descripción de cada parámetro se muestra a continuación:

- th_cos: corresponde al parámetro de la similitud coseno,
- th_dice: corresponde al parámetro de la similitud Dice,
- th_validation: corresponde al umbral de validación,
- maxgap: separación máxima de oraciones que son consideradas adyacentes,
- maxgap_summary: separación máxima de oraciones que son consideradas adyacentes, enfocado a la parte de ofuscación en
- minsize: longitud de pasajes pequeños,
- minplaglen: longitud de un pasaje para ser considerado plagio,
- th_verbatim: longitud de secuencia de palabras en la etapa de filtrado.

4. Metodología propuesta

Método general. En cada experimento, seguimos los siguientes pasos:

- Pre-procesamiento: se mantuvieron las técnicas de los métodos de [1, 2] los cuales son los siguientes:
  - Conversión a minúsculas,
  - Eliminación de caracteres especiales,
  - *Stemming*,
  - División en tokens.

  En cuanto a la implementación de *stopwords* en los enfoques de [1, 2] no se eliminan *stopwords*. En nuestro enfoque es en donde probamos diferentes listas de *stopwords ShortList* y *BigList* (las listas contiene diferente cantidad de *stopwords*). Implementamos estas dos listas en los enfoques de [1, 2], así como las dos listas que ellos mencionan en [12].
  El uso de la de técnica *stemming* que implementa [1, 2] decidimos no utilizarla en algunos experimentos, para conocer su importancia dentro de los enfoques trabajados.
- Pre-selección: Se utilizó la fragmentación de texto, tanto en los documentos sospechosos como el origen.
- Extensión: Teniendo los fragmentos de texto se procede a agruparlos, formando pasajes de máxima longitud. Para [1] se agrega una etapa de validación de los grupos creados.
- Filtrado: teniendo los pasajes alineados, se eliminan los que no superan la longitud de 150 caracteres, también se eliminan pasajes solapados.

Corpus de entrenamiento. Utilizamos el corpus de entrenamiento proporcionado por la competencia PAN 2014, en la sub-tarea alineamiento de texto, el cual está

disponible[2]. El corpus está conformado por 5185 pares de documentos sospechosos de plagio en idioma inglés, están divididos en 5 formas de plagio: no hay plagio (*no plagiarism),* ninguno *(no obfuscation),* aleatorio *(random obfuscation),* traducción *(translation obfscation)* y resúmenes (*summary obfuscation).* En [31] menciona que algunas formas de plagio se realizaron con herramientas comerciales y en algunos casos se llegó a perder la coherencia de los textos. A continuación se dará una breve descripción de cada sub-área del corpus:

– No hay plagio (*no-plagiarism*). Esta parte del corpus no contiene ningún tipo de plagio.

– Ninguno (*no-obfuscation*). La ofuscación que se presenta solo es de *copy-paste.*

– Aleatorio (*summary-ofuscation*). Es una secuencia de operaciones de texto al azar, añadir, eliminar y reemplazar palabras o frases cortas en todo el texto.

– Traducción (*translation-obfuscation*). El texto se tradujo en por lo menos tres idiomas con diferentes herramientas comerciales, siendo inglés el idioma inicial y final.

– Resúmenes (*summary-obfuscation*). Incluye un resumen no atribuido en otro documento, las ideas principales del documento se mantienen. Puede ser visto como una forma de plagio de ideas.

Tabla 7. Resultados reportados por [2], en alineamiento de texto PAN 2014 sin eliminación de stopwords.

Resultados PAN 2014 sin *stopwords*				
Ofuscación	*Plagdet*	*Recall*	*Precision*	*Granul.*
Ninguna	0.8938	0.9782	0.8228	1.0000
Aleatoria	0.8886	0.8581	0.9213	1.0000
Traducción	0.8839	0.8902	0.8777	1.0000
Resúmenes	0.5772	0.4247	0.9941	1.0434
Total	**0.8773**	0.8799	0.8774	1.0021

Evaluación. El marco de evaluación fue propuesto por [31], en donde se propone una medida (*plagdet*) que está en función de precisión, recuerdo y granularidad, lo cual ya se explicó anteriormente.

La Tabla 7 y la Tabla 8, muestran los resultados obtenidos en los diferentes enfoques para PAN 2014 [2] y PAN 2015 [1] utilizando el corpus de entrenamiento, estos resultados se encuentran reportados en [1], el resultado final se encuentra en negritas.

[2] http://pan.webis.de/clef14/pan14-web/plagiarism-detection.html

Los resultados obtenidos utilizando la lista de las 50 stopwords más frecuentes en inglés, reportada por Stamatatos en [32], para el trabajo de [2] los resultados se muestra en la Tabla 9 y el para el trabajo de [1] los resultados obtenidos se muestran en la Tabla 10.

Tabla 8. Resultados reportados por [1], en alineamiento de texto PAN 2015 sin eliminación de stopwords.

Resultados PAN 2015 sin *stopwords*				
Ofuscación	*Plagdet*	*Recall*	*Precision*	*Granul.*
Ninguna	0.9812	0.9761	0.9933	1.0048
Aleatoria	0.8847	0.8699	0.8999	1.0000
Traducción	0.8792	0.9128	0.8481	1.0000
Resúmenes	0.6304	0.4862	0.9739	1.0404
Total	**0.9025**	0.8937	0.9164	1.0036

5. Resultados experimentales

En el trabajo de [2], se encuentran dos listas de *stopwords* que en sus experimentaciones no reporta, nos dimos a la tarea de realizar la experimentación con estas dos listas de *stopwords*, una está descrita como las 50 *stopwords* más frecuentes en inglés por Stamatatos en [32] y la otra está contenida en el paquete NLTK de Python, la cual se encuentra en el trabajo de [12], se utilizó el corpus de entrenamiento de PAN 2014 en todos los experimentos.

En esta sección presentaremos en primer lugar los resultados reportados utilizando el trabajo de [2] en PAN 2014. Los resultados obtenidos utilizando la lista de las 50 *stopwords* más frecuentes en inglés, reportada por Stamatatos en [32], se muestran en la Tabla 9. Los parámetros utilizados en los resultados de la Tabla 9 a la Tabla 13 para PAN 2014 se encuentran descritos en la Tabla 5.

Tabla 9. Resultados obtenidos de [2], en alineamiento de texto PAN 2014 eliminando las 50 *stopwords* reportadas por Stamatatos [32].

Resultados PAN 2014 50 *stopwords*				
Ofuscación	*Plagdet*	*Recall*	*Precision*	*Granul.*
Ninguna	0.9002	0.9722	0.8380	1.0000
Aleatoria	0.8653	0.8099	0.9288	1.0000
Traducción	0.8768	0.8687	0.8850	1.0000
Resúmenes	0.4890	0.3530	0.9939	1.0924
Total	**0.8662**	0.8518	0.8866	1.0043

Con la lista de *stopwords* del paquete de Python NLTK, para el trabajo de [2] los resultados se muestran reportados en la Tabla 10.

Tabla 10. Resultados obtenidos de [2], en alineamiento de texto PAN 2014 eliminando las *stopwords* del corpus NLTK.

Resultados PAN 2014 NLTK *stopwords*				
Ofuscación	*Plagdet*	*Recall*	*Precision*	*Granul.*
Ninguna	0.8968	0.9707	0.8334	1.0048
Aleatoria	0.8482	0.7846	0.9231	1.0000
Traducción	0.8677	0.8600	0.8755	1.0000
Resúmenes	0.4799	0.3517	0.9842	1.1136
Total	**0.8563**	0.8405	0.8797	1.0055

Las listas de *stopwords* que implementamos, en primer lugar en la Tabla 11, mostramos los resultados utilizando la lista de *stopwords ShortList.* La Tabla 12, muestra los resultados obtenidos con la lista de *stopwords BigList.*

Tabla 11. Resultados obtenidos de [2], utilizando el corpus de entrenamiento PAN 2014 en alineamiento de texto, con la lista de stopwords *ShortList.*

Resultados PAN 2014 con *stopwords ShortList*				
Ofuscación	*Plagdet*	*Recall*	*Precision*	*Granul.*
Ninguna	0.8906	0.9696	0.8235	1.0000
Aleatoria	0.8442	0.7826	0.9163	1.0000
Traducción	0.8677	0.8586	0.8770	1.0000
Resúmenes	0.4976	0.3628	0.9789	1.0903
Total	**0.8541**	0.8398	0.8745	1.0043

Tabla 12. Resultados obtenidos del [2] del utilizando el corpus de entrenamiento PAN 2014 en alineamiento de texto, con la lista de stopwords *BigList*.

Resultados PAN 2014 con *stopwords BigList*				
Ofuscación	*Plagdet*	*Recall*	*Precision*	*Granul.*
Ninguna	0.8818	0.9683	0.8096	1.0000
Aleatoria	0.8294	0.7574	0.9165	1.0000
Traducción	0.8621	0.8598	0.8644	1.0000
Resúmenes	0.5127	0.3829	0.9681	1.1000
Total	**0.8455**	0.8330	0.8647	1.0049

Se realizó un experimento sin implementar *stemming* ni eliminar *stopwords*, los resultados se muestran en la Tabla 13.

Tabla 13. Resultados obtenidos de [2], utilizando el corpus de entrenamiento PAN 2014 en alineamiento de texto, sin *stopwords* ni *stemming*.

Resultados PAN 2014 sin *stemming*				
Ofuscación	*Plagdet*	*Recall*	*Precision*	*Granul.*
Ninguna	0.9054	0.9785	0.8425	1.0048
Aleatoria	0.8891	0.8515	0.9302	1.0000
Traducción	0.8624	0.8286	0.8992	1.0000
Resúmenes	0.4969	0.3450	0.9932	1.0429
Total	**0.8721**	0.8540	0.8934	1.0019

A continuación se mostrarán los resultados utilizando el trabajo de [1] en PAN 2015, los parámetros utilizados de la Tabla 14 a la 18 se encuentran descritos en la Tabla 6. Los resultados obtenidos utilizando la lista de las 50 *stopwords* más frecuentes en inglés, reportada por Stamatatos en [32], se muestran en la Tabla 14. Los resultados finales se enmarcan con negritas en cada tabla.

Tabla 14. Resultados obtenidos de [1], en alineamiento de texto PAN 2015 eliminando las 50 *stopwords* reportadas por Stamatatos [32].

Resultados PAN 2015 50 *stopwords*				
Ofuscación	*Plagdet*	*Recall*	*Precision*	*Granul.*
Ninguna	0.9812	0.9761	0.9933	1.0048
Aleatoria	0.8847	0.8701	0.8998	1.0000
Traducción	0.8791	0.9128	0.8477	1.0000
Resúmenes	0.6304	0.4862	0.9739	1.0404
Total	**0.9025**	0.8937	0.9163	1.0036

Con la lista de *stopwords* del paquete de Python NLTK, para el trabajo de [2] los resultados se muestran reportados en la Tabla 15.

Tabla 15. Resultados obtenidos de [1], en alineamiento de texto PAN 2015 eliminando las *stopwords* del corpus NLTK.

Resultados PAN 2015 NLTK *stopwords*				
Ofuscación	*Plagdet*	*Recall*	*Precision*	*Granul.*
Ninguna	0.9812	0.9761	0.9933	1.0048
Aleatoria	0.8846	0.8701	0.8996	1.0000
Traducción	0.8794	0.9128	0.8484	1.0000
Resúmenes	0.6304	0.4862	0.9739	1.0404
Total	**0.9026**	0.8937	0.9165	1.0036

En la Tabla 16, mostramos los resultados utilizando la lista de *stopwords ShortList* y en la Tabla 17, reportamos los resultados obtenidos con la lista de *stopwords BigList.*

Tabla 16. Resultados obtenidos de [1], utilizando el corpus de entrenamiento PAN 2014 en alineamiento de texto, con la lista de stopwords *ShortList*.

Resultados PAN 2015 con *stopwords ShortList*				
Ofuscación	*Plagdet*	*Recall*	*Precision*	*Granul.*
Ninguna	0.9812	0.9761	0.9933	1.0048
Aleatoria	0.8851	0.8702	0.9006	1.0000
Traducción	0.8798	0.9125	0.8494	1.0000
Resúmenes	0.6291	0.4848	0.9737	1.0406
Total	**0.9028**	0.8936	0.9171	1.0036

Tabla 17. Resultados obtenidos de [1], utilizando el corpus de entrenamiento PAN 2014 en alineamiento de texto, con la lista de stopwords *BigList*.

Resultados PAN 2015 con *stopwords BigList*				
Ofuscación	*Plagdet*	*Recall*	*Precision*	*Granul.*
Ninguna	0.9812	0.9761	0.9933	1.0048
Aleatoria	0.8860	0.8705	0.9020	1.0000
Traducción	0.8799	0.9101	0.8517	1.0000
Resúmenes	0.6289	0.4846	0.9737	1.0406
Total	**0.9031**	0.8929	0.9183	1.0036

De igual forma como se hizo anteriormente, se realizó un experimento sin la implementación de *stemming* en el texto de entrada ni la eliminación de *stopwords*, los resultados obtenidos se encuentran reportados en la Tabla 18.

Tabla 18. Resultados obtenidos de [1], utilizando el corpus de entrenamiento PAN 2014 en alineamiento de texto, sin *stopwords* ni *stemming.*

Resultados PAN 2015 sin *stemming*				
Ofuscación	*Plagdet*	*Recall*	*Precision*	*Granul.*
Ninguna	0.9820	0.9758	0.9952	1.0048
Aleatoria	0.8879	0.8590	0.9188	1.0000
Traducción	0.8709	0.8641	0.8788	1.0008
Resúmenes	0.5726	0.4205	0.9775	1.0376
Total	**0.8990**	0.8710	0.9340	1.0037

6. Conclusiones

En este artículo, se experimentó la implementación de diferentes listas de *stopwords*, en la competición internacional de plagio PAN 2014 y PAN 2015, para la sub-tarea alineamiento de texto, tomando los enfoques de [1, 2], en la etapa de pre-procesamiento, para conocer el efecto de la ausencia de información en el corpus de entrenamiento, en este caso de *stopwords*, que es la técnica en donde se elimina más información de los documentos originales.

Las listas de *stopwords* utilizadas las llamamos *ShortList* y *BigList*, debido a la cantidad de *stopwords* que contiene cada lista, así como las propuestas en [12] que son dos listas: una es las 50 *stopwords* más comunes en inglés propuesta por Stamatatos [32] y la segunda es la que está incluida en la librería de Python NLTK. Tambien se reportaron resultados cuando no se implementó *stemming.*

En la Tabla 19 se muestran los resultados finales de nuestras experimentaciones de PAN 2014 y PAN 2015 con el corpus de entrenamiento de PAN 2014, comparando los resultados obtenidos en los trabajos de [1, 2], los mejores resultados se muestran resaltados en negritas.

Tabla 19. Resultados obtenidos en PAN 2014 y PAN 2015 en comparación con los reportados en los trabajos de [1] y [2].

Resultados PAN 2014		Resultados PAN 2015	
Experimentos	*Plagdet*	Experimentos	*Plagdet*
Sin eliminar *stopwords*[2]	**0.8773**	Sin eliminar *stopwords*[1]	0.9025
Eliminación de 50 *stopwords*	0.8662	Eliminación de 50 *stopwords*	0.9025
Eliminación *stopwords* NLTK	0.8563	Eliminación *stopwords* NLTK	0.9026
Stopwords ShortList	0.8541	*Stopwords ShortList*	0.9028
Stopwords BigList	0.8455	*Stopwords BigList*	**0.9031**
Sin *Stemming*	0.8721	Sin *Stemming*	0.8990

Como se puede observar, para PAN 2014 el mejor resultado es el reportado por [2] en el corpus de entrenamiento, para ese trabajo lo mejor es mantener la mayor cantidad de información, así como aplicar *stemming*, esto da resultados favorables. Para PAN 2015 el mejor resultado es con la eliminación de *stopwords* de la lista *BigList*. Con la eliminación de *stopwords*¸ se logra una mejor detección de pasajes similares, sin olvidar que la implementación de *stemming* también juega un papel importante en la preparación del texto, eliminando caracteres que no son importantes a lo largo del proceso de alineamiento de texto.

Como se describió en el estado del arte la mayoría de los equipos de detección de plagio mediante alineamiento de texto, utilizan técnicas muy similares de pre-procesamiento, ocho son las técnicas que se identificaron entre trabajos para alineamiento de texto entre PAN 2012 y PAN 2015 [33, 34, 35]. Es muy común el uso de técnicas de pre-procesamiento en tareas de detección de plagio, pero no hay un estudio sobre cómo estas técnicas afectan los resultados finales.

Como trabajo futuro se tiene enfocarse a las siguientes etapas del método y en conjunto con el estado del arte, saber que técnicas nos podrían ayudar a tener mejores resultados en comparación con los de [1]. La etapa de pre-selección y extensión son clave para la formación de pasajes plagiados en los pares de documentos.

La implementación de características de n-gramas sintácticos en el texto [36, 37] o distintos tipos de n-gramas [17, 38, 39] en la etapa de preselección, puede ayudar a tener una mejor generación de fragmentos para formar pasajes plagiados. La implementación de diversos agrupadores en la etapa de extensión en combinación con la etapa de pre-selección, es algo que a lo largo de las competiciones de PAN 2012 y PAN 2014 se ha presentado, y también es una opción implementar y analizar algunos agrupadores [40, 41],

Si bien ya se han registrado resultados favorables, la experimentación con nuevas heurísticas, nos abrirá el panorama, en cuanto la utilidad de diferentes técnicas tanto para resolver este problema como algunos otros que se relacionen.

Referencias

1. Sánchez-Pérez, M.A., Gelbukh, A.F., Sidorov, G.: Dynamically adjustable approach through obfuscation type recognition. In: Working Notes of CLEF 2015 - Conference and Labs of the Evaluation forum, Toulouse, France, September 8-11, 2015. CEUR Workshop Proceedings, vol. 1391, CEUR-WS.org, http://ceur-ws.org/Vol-1391/92-CR.pdf (2015)
2. Sánchez-Pérez, M.A., Gelbukh, A.F., Sidorov, G.: The Winning Approach to Text Alignment for Text Reuse Detection at PAN 2014. In: L. Cappellato, N. Ferro, M. Halvey, W. Kraaij (eds.). Notebook for PAN at CLEF 2014. CLEF 2014. CLEF2014 Working Notes. Sheffield, UK, September 15-18. CEUR Workshop Proceedings, ISSN 1613-0073, Vol. 1180, CEUR-WS.org, 2014, pp. 1004–1011 (2014)
3. Mateos, F. J., Ruiz, J. L.: Procesamiento del lenguaje natural. Dpto. Ciencias de la Computación e Inteligencia Artificial; Universidad de Sevilla. http://www.cs.us.es/cursos/ia2/temas/tema-06.pdf (2012)
4. Baeza-Yates, R., Ribeiro-Neto, B.: Modern information retrieval. Vol. 463, New York: ACM press (1999)
5. Salton, G., McGill, M. J.: Introduction to modern information retrieval. (1983)
6. Real Academia Española.: Diccionario de la lengua española. Vigésima tercera edición, 2014, http://www.rae.es/ Consultado Agosto (2016)
7. Armeaga, G.J.: Comparación de medidas de similitud en cadenas textuales, para la detección de plagio en tareas escolares. Unidad académica Profesional Tianguistenco, Universidad Autónoma del Estado de México, Tesis de licenciatura (2015)
8. Barrón Cedeño, L. A.: Detección automática de plagio en texto. Departamento de Sistemas Informáticos y Computación, Tesis desarrollada dentro del Máster en Inteligencia Artificial, Universidad Politécnica de Valencia (2008)
9. Instituto Nacional del Derecho de Autor.: Derechos de Autor; México; http://www.indautor.gob.mx/accesibilidad/accesibilidad_autor.html, Consultado Mayo (2015)
10. Sánchez Vega, J. F.: Detección automática de plagio basada en la distinción y fragmentación del texto reutilizado. Tesis sometida como requisito parcial para obtener el grado de: Maestro en Ciencias en el Área de Ciencias Computacionales Instituto Nacional de Astrofísica, Óptica y Electrónica (2011)
11. Balaguer, E. V.: Putting ourselves in SME's shoes: Automatic detection of plagiarism by the WCopyFind tool. In: Proc. SEPLN, pp. 34–35 (2009)
12. Sánchez-Pérez, M.A.: Detección automática de plagio a través de formación de pasajes. Centro de Investigación en Computación, Instituto Politécnico Nacional, Tesis de Maestría (2014)
13. Potthast, M., Hagen, M., Beyer, A., Busse, M., Tippmann, M., Rosso, P., Stein, B.: Overview of the 6th International Competition on Plagiarism Detection (2014)
14. Elizalde, V.: Estudio y desarrollo de nuevos algoritmos de detección de plagio. Doctoral dissertation, Tesis de Licenciatura en Ciencias de la Computación Departamento de Computación, Facultad de Ciencias Exactas y Naturales, Universidad de Buenos Aires (2011)
15. Potthast, M., Gollub, T., Hgaen, M., GraBegger, J., Kiesel, J., Michel, M., Oberlander, A., Tippman, M., Barrón-Cedeño, A., Gupta, P., Rosso, P., Stein, B.: Overview of the 4th International competition on Plagiarism Detection. In: Forner et al. [33].

16. Alvi, F., Stevenson, M., Clough, P. D.: Hashing and Merging Heuristics for Text Reuse Detection. In: Cappellato et al. [35].
17. Palkovskii, Y., Belov, A.: Developing High-Resolution Universal Multi-Type N-Gram Plagiarism Detector. In: Cappellato et al. [35].
18. Glinos, D. S.: A Hybrid Architecture for Plagiarism Detection. In: Cappellato et al. [35].
19. Leilei, K., Haoliang, Q., Cuixia, D., Mingxing, W., Zhongyuan, H.: Approaches for Source Retrieval and Text Alignment of Plagiarism Detection. In: Forner et al. [34]
20. Palkovskii, Y., Belov, A.: Using Hybrid Similarity Methods for Plagiarism Detection. In: Forner et al. [34].
21. Leilei, K., Haoliang, Q., Shuai, W., Cuixia, D., Suhong, W., Yong, H.: Approaches for candidate document retrieval and detailed comparison of plagiarism detection. In: Forner et al. [33].
22. Torrejón, D. A. R., Ramos, J. M. M.: Text Alignment Module in CoReMo 2.1 Plagiarism Detector. In: Forner et al. [34].
23. Abnar, S., Dehghani, M., Zamani, H., Shakery, A.: Expanded n-grams for semantic text alignment. In: Cappellato et al. [35].
24. Suchomel, Š., Kasprzak, J., Brandejs, M.: Diverse queries and feature type selection for plagiarism discovery. In: Forner et al. [34]
25. Shrestha, P., Solorio, T.: Using a Variety of n-Grams for the Detection of Different Kinds of Plagiarism. In: Forner et al. [34].
26. Palkovskii, Y., Belov, A.: Applying Specific Clusterization and Fingerprint Density Distribution with Genetic Algorithm Overall Tuning in External Plagiarism Detection. In: CLEF, (Online Working Notes/Labs/Workshop) (2012)
27. Ledeneva, Y.: Effect of preprocessing on extractive summarization with maximal frequent sequences. In: Mexican International Conference on Artificial Intelligence, pp. 123–132, Springer Berlin Heidelberg (2008)
28. Hassan, S., Mihalcea, R., Banea, C.: Random walk term weighting for improved text classification. International Journal of Semantic Computing, 1(04), pp. 421–439 (2007)
29. Leilei, K., Yong, H., Zhongyuan, H., Haihao, Y., Qibo, W., Tinglei, Z., Haoliang, Q.: Source Retrieval Based on Learning to Rank and Text Alignment Based on Plagiarism Type Recognition for Plagiarism Detection. In: Cappellato et al. [35].
30. Potthast, M., Hagen, M., Gollub, T., Tippmann, M., Kiesel, J., Rosso, P., Stein, B.: Overview of the 5th international competition on plagiarism detection. In: Forner et al. [34].
31. Potthast, M., Stein, B., Barrón-Cedeño, A., Rosso, P.: An evaluation framework for plagiarism detection. In: Proceedings of the 23rd international conference on computational linguistics: Posters, Association for Computational Linguistics, pp. 997–1005 (2010)
32. Stamatatos, E.: Plagiarism detection using stopword n-grams. Journal of the American Society for Information Science and Technology, 62(12), pp. 2512–2527 (2011)
33. Forner, P., Karlgren, J., Womser-Hacker, C.: CLEF 2012: Evaluation Labs and Workshop – Working Notes Papers, 17-20 September, Rome, Italy (2012).
34. Forner, P., Navigli, R., Tu s, D., Ferro, N.: Working Notes for CLEF 2013 Conference, Valencia, Spain, September 23-26, 2013, CEUR Workshop Proceedings, vol. 1179. CEUR-WS.org (2013)
35. Cappellato, L., Ferro, N., Halvey, M., Kraaij, W.: Notebook for PAN at CLEF 2014. CLEF 2014. CLEF2014 Working Notes. Sheffield, UK, September 15-18, 2014. CEUR Workshop Proceedings, ISSN 1613-0073, Vol. 1180, CEUR-WS.org (2014)
36. Sidorov, G.: Construcción no lineal de n-gramas en la lingüística computacional: n-gramas sintácticos, filtrados y generalizados. Sociedad Mexicana de Inteligencia Artificial (2013)
37. Posadas-Durán, J. P., Markov, I., Gómez-Adorno, H., Sidorov, G., Batyrshin, I., Gelbukh, A., Pichardo-Lagunas, O.: Syntactic N-grams as Features for the Author Profiling Task. In: Proceedings of CLEF (2015)

38. Efstathios, S.: Plagiarism Detection Using Stopword n-Grams. JASIST, 62(12), pp. 2512–2527 (2011)
39. Barrón-Cedeño, A., Rosso, P.: On automatic plagiarism detection based on n-grams comparison. In: European Conference on Information Retrieval, Springer Berlin Heidelberg, pp. 696–700 (2009)
40. Kryszkiewicz, M., Lasek, P.: TI-DBSCAN: Clustering with DBSCAN by Means of the Triangle Inequality. In: International Conference on Rough Sets and Current Trends in Computing, Springer Berlin Heidelberg, pp. 60–69 (2010)
41. Smiti, A., Eloudi, Z.: Soft dbscan: Improving dbscan clustering method using fuzzy set theory. In: 6th International Conference on Human System Interactions (HSI), IEEE, pp. 380–385 (2013)