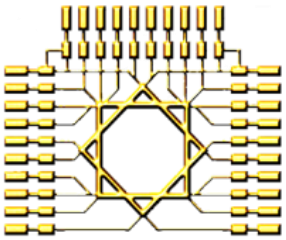
الجمهورية العربية السورية
المعهد العالي للعلوم التطبيقية والتكنولوجيا
قسم المعلوميات



مشروع سنة رابعة

اختصاص نظم معلوماتية

Plagiarism Detection

نظام كشف الغش في نصوص باللغة العربية

تقديم: رزان أسعد

بإشراف: د.غيداء ربداوي

م.رياض سنبل

9/9/2013

الملخص:

تم تطوير العديد من الأدوات الحساسة للغة ، من أجل بناء أنظمة لكشف الغش في المستندات المكتوبة باللغة الطبيعية، وخاصة للغة الإنجليزية.

توجد أدوات مستقلة عن اللغة، ولكنها تشكل عائقاً في بعض الحالات لأنها عادة لا تأخذ بعين الاعتبار ميزات لغة معينة.

يعدّ نظام كشف الغش في الوثائق العربية مهمة صعبة بسبب البنية اللغوية المعقدة للغة العربية.

نقدم في هذا المشروع أداة كشف الغش للمقارنة بين الوثائق العربية، وتحديد أوجه التشابه المحتملة.

تستند هذه الأداة على خوارزمية مقارنة جديدة تستخدم توابع تجريبية لمقارنة الوثائق المشكوك بمصداقيتها، ضمن مستويات هرمية مختلفة لتجنب المقارنة غير الضرورية .

تعتمد هذه الأداة على اختيار النصوص المشكوك بها، ثم معالجتها نصياً، نستخدم الخوارزمية السابقة لإعطاء بصمة لكل نص، وبعدها نقوم بعملية المقارنة بين النصوص، لإعطاء النتيجة النهائية.

Abstract:

Many language-sensitive tools for detecting plagiarism in natural language documents have been developed, particularly for English. Language-independent tools exist as well, but are considered restrictive as they usually do not take into account specific language features. Detecting plagiarism in Arabic documents is particularly a challenging task because of the complex linguistic structure of Arabic.

In this paper, we present a plagiarism detection tool for comparison of Arabic documents to identify potential similarities.

The tool is based on a new comparison algorithm that uses heuristics to compare suspect documents at different hierarchical levels to avoid unnecessary comparisons. First, we chose texts, and then we apply the text processing on texts and used the winnowing algorithm to fingerprint each text. After that, we show the result of comparison among texts.

الفهرس

مقدمة عامة:

جعل الاتصال السريع بالإنترنت وسهولة الحصول على المعلومات منه، عملية الغش عملية سهلة ومنتشرة بين الطلاب، وأصبح من الممكن الحصول على علامات جيدة من خلال وثائق ليس لديهم أدنى فكرة عنها.

تعددت طرق الغش: بين نسخٍ كامل للوثيقة وجملها ومفرداتها بدون ذكر اسم المصدر، أو غش في نقل الأفكار فقط أو غيرها.

حيث يوجد أنماط أخرى من الغش، مثل ترجمة محتوى من لغة للغة أخرى مع نسخ كامل للمعلومات المتواجدة ولكن بتغيير بسيط في الصور أو الفيديوهات واستخدام كود بدون إذن الناشر[1].

هناك عدة طرق لتقليص الغش في الوثائق منها طريقتان رئيسيتان: طريقة منع الغش وطريقة كشف الغش.

طريقة منع الغش تتضمن أسلوب عقاب، هذه الأساليب لها تأثير إيجابي على المدى الطويل، لكنها تتطلب وقتا طويلا لتنفيذها، نظرا لأنها تعتمد على التعاون الاجتماعي بين الجامعات والإدارات المختلفة للحد من الانتحال [1].

أما طريقة الكشف عن الغش فتشمل الطرق اليدوية وأدوات البرمجيات. وهي سهلة التنفيذ، لكن أثرها الإيجابي مؤقت. يمكن الجمع بين الطريقتين للحد من الاحتيال والغش. على الرغم من الأدوات البرمجية هي النهج الأكثر فعالية لتحديد الانتحال، ينبغي أن يتم الحكم النهائي يدويا [3].

يمكن اكتشاف الانتحال في النصوص المكتوبة باللغات الطبيعية أو بلغات البرمجة [2].

كشف الانتحال في التعليمات البرمجية سهل نسبيا حيث لا يوجد تداخل بين الكلمات في لغات البرمجة. على سبيل المثال: يمكن الكشف عن إعادة تسمية بعض المتغيرات في التعليمات البرمجية أو تعديل بعض البنى ضمن الكود[4]. أما في اللغات الطبيعية فهي أكثر صعوبة حيث كل كلمة قد يكون لها العديد من المرادفات والمعاني المختلفة[5].

من طرق الكشف عن الغش: طريقة لا تعتمد على اللغة، في حين طرق أخرى مخصصة للغة طبيعية واحدة.

تستند الطريقة التي لا تعتمد على لغة معينة على تقييم خصائص النص التي ليست ملازمة للغة معينة، مثل عدد الكلمات المفردة ومتوسط طولا لجملة [3]. أما الطرق الحساسة للغة معينة تستند على تقييم خصائص النص المحددة للغة واحدة. على سبيل المثال: عد تردد كلمة خاصة بلغة معينة [3].

هدف المشروع:

الهدف من المشروع هو بناء أداة تقوم بكشف الغش "APLAG" ، انطلاقا من مجموعة من النصوص المكتوبة باللغة العربية.

تأتي أهمية هذا النظام من الحاجة إلى كشف التشابه بين الوثائق العربية، بحيث يكون الدخل مجموعة كبيرة من الوثائق ويحتفظ النظام فقط بالوثائق التي تكون نسبة التشابه بينها كبيرة.

ثم يقوم بفحصها ليكشف المقاطع أو الجمل المتشابهة.

المراحل الرئيسية لهذه الأداة:

مرحلة المعالجة النصية، مرحلة المقارنة باستخدام خوارزمية تجريبية لمقارنة وثائق من مستويات منطقية مختلفة (على مستوى الوثيقة، الفقرة، أو الجملة).

الفصل الأول: متطلبات المشروع

نبين في هذا الفصل المهام المطلوب تنفيذها من قبل النظام

والقيود التي يجب اتباعها في تطويره.

1.1. المتطلبات الوظيفية:

1. يجب أن يتيح النظام خاصية اختيار مجموعة النصوص المراد كشف التشابه بينها.
2. يجب أن يستعرض النظام تقرير شامل بالنتائج.
3. يجب على النظام أن يتعامل مع الوثائق بالمستوى الصرفي والقواعدي.
4. يتعامل النظام مع وثائق باللغة العربية.

2.1.المتطلبات غير الوظيفية:

1. تنجيز النظام يجب أن يكون بلغة java.
2. يجب أن يقدم النظام واجهة سهلة الاستخدام.
3. يعيد النظام النتائج بزمن ودقة مقبولين.

الفصل الثاني: الدراسة المرجعية

نبين في هذا الفصل شرحاً مفصلاً للمبادئ العامة والأفكار النظرية
التي تم الارتكاز عليها في العمل ضمن المشروع.

1.2.التجارب السابقة في مجال كشف الغش:

سنقدم فيما يلي شرحاً مفصّلاً عن الطرق الرئيسية المستخدمة في مجال كشف الغش:

1.1.2. الطرق التقليدية في كشف الغش:

تتم مقارنة النصوص مع بعضها للكشف عن محتويات النسخ واللصق، أولتحديد أنماط الكتابة المختلفة ضمن الوثيقة.

الطريقة الثانية غير قابلة للتطبيق إذا كان للماتب أكثر من أسلوب للكتابة.

يمكن لمحركات البحث يمكن أن تدعم مثل هذه الأساليب للتحقق من الأجزاء المشبوهة ضمن الوثيقة والتي لا تتطابق الأسلوب الذي يستخدمه الكاتب عادة.

تشمل الطرق التقليدية أيضا تقنيات معتمدة على الضغط. ليكن لدينا وثيقتين D1، D2 وليكن d1، d2 هما الملف المضغوط لكل وثيقة باستخدام إحدى تقنيات الضغط.

لنجعل a= d1d2 تمثل سلسلة تجمع الملفين d1، d2. باعتبار أن B=D1D2 أيضا هي سلسلة تجمع الوثيقتين D1، D2 و b هو ملف الضغط ل B.

إذا كان D1 يختلف عن D2، يكون a و b لهما نفس الحجم. أما إذا كانا يحتويان على أجزاء إضافية زائدة فيكون حجم b أصغر من a. [3]

بشكل عام الطرق التقليدية سهلة التطبيق، ولكنها تتطلب الكثير من الوقت ورغم ذلك تكون أحيانا غير موثوقة وخاصة في حالة النصوص الكبيرة.

ولذلك نحن بحاجة إلى أدوات تساعد المستخدم في الكشف عن الغش بطريقة أسرع وأدق.

2.1.2. الطرق المعتمدة على المحتوى:

هذه الطرق تعتمد على المقارنات الصريحة بين محتوى الوثائق. تعتبر تقنية وضع بصمة للوثيقة إحدى أكثر الطرق شيوعا في هذا المجال. بحيث تعمل على قياس مقدار التشابه بين وثيقتين من خلال المقارنة بين بصمة كل منها.

يتم وضع البصمة لوثيقة من خلال تابع تجريبي يربط كل كلمة بقيمة من الأعداد الصحيحة يتم حسابها حسب الطريقة التي يختارها الخبير. وهكذا يمكن وضع بصمة لكل وثيقة.

تستند تقنيات وضع البصمة على مخطط k-grams (وهي سلسلة فرعية متتابعة طولها k)، والتي تعتبر أساس معظم تقنيات وضع البصمات.

يتم اختيار طريقة وضع بصمة وفقا لمخططات مختلفة مثل: ith hash ، $0 \bmod p$ hash، وطريقة ال Winnowing. [7]

- مخطط ith hash : يعتمد على تقسيم الوثيقة لأقسام وإعطاء قيمة عددية لكل قسم. هذه الطريقة سهلة التنفيذ، ولكنها غير دقيقة في حالة إدخال أو حذف أو إعادة ترتيب أجزاء من الوثيقة. مثلاً: عند إدخال حرف واحد للنص فإن البصمة لكل قسم سوف تزيد بمقدار واحد. وبالتالي لا يبقى أي بصمة مشتركة بين النسخة الاصلية والمعدلة .[7]
- مخطط p hash 0 mod: يربط كل موقع من النص بالعدد 0 mod حيثp هو عدد صحيح. وهذه الطريقة أيضا سهلة التنفيذ ولكنها ضعيفة في الكشف عن الغش.[7]
- طريقة Winnowing :هذه الخوارزمية تحدد البصمة من خلال مخطط k-grams الذي يقسم النص إلى أجزاء بحسب k. وتستخدم لكشف التشابه بين النصوص.

تعمل هذه الخوارزمية كما يلي: ليكن كل من t، k هما متحولان يعبر كل منهما عن عتبة معينة.

للكشف عن تشابه بين وثيقتين يجب أن تتحق الخاصيتين:

1. يؤخذ التشابه بعين الاعتبار إذا تم الكشف عن سلسلة فرعية مشتركة طولها أكبر أو يساوي عتبة الضمان t.
2. أي تشابه أقل من عتبة الضجيج k لا يؤخذ بعين الاعتبار.

تعمل هذه الخوارزمية كما يلي:

يجب تحديد حجم نافذة window size يكون له البعد t-k+1، وكل نافذة wi تحتوي على القيم العددية لتابع الhash : hi hi+w-1... يتم اختيار أصغر قيمة من قيم الhash لكل نافذة، إذا وجدت أكثر من قيمة تؤخذ القيمة الظاهرة أولاً. وهذه القيم مجتمعة تمثل البصمة الخاصة بكل وثيقة.

التحليل الدلالي الكامن (LSA):[8]

هو أسلوب يستخدم لوصف العلاقات بين مجموعة من الوثائق والمصطلحات.

يفترض في هذه التقنية أن تجتمع الكلمات المتقاربة من بعضها في المعنى ضمن مجموعات.

يمكن أن تصنف على شكل مصفوفات أسطرها تمثل الكلمات، والأعمدة تمثل النصوص.كل نص يحوي على مجموعة واحدة من الكلمات.

يستخدم التجزيء المفرد للقيم (SVD) لتقليص عدد الأعمدة مع الحفاظ على بنية التشابه بين الأسطر.

تتم مقارنة الكلمات بأخذ جب (cos) الزاوية بين الشعاعين المكونين من أي سطرين.

تمثل القيم القريبة من الواحد الكلمات المتشابهة، بينما القيم القريبة من الصفر تكون كلمات غير متشابهة.

آلية تحليل نسخ ستانفورد (SCAM):[9]

يستند هذا النظام على الكشف عن مخطط للكشف عن النسخ المسجلة، يتم تسجيل الوثائق في مستودع، ومن ثم تقارن مع الوثائق المسجلة مسبقا.

تتألف بنية مخدم النسخ من مستودع ومجزئ (chunker). يقوم المجزئ بتقسيم النص إلى جمل، كلمات ، أو جمل متقاطعة. يتم تقسيم كل وثيقة قبل تسجيلها في المستودع.

وكل وثيقة جديدة يجب أن تقسم بنفس وحدة التقسيم المستخدمة، وذلك قبل مقارنتها مع الوثائق المسجلة مسبقاً. يستخدم فهرس للتخزين لترتيب أقسام كل وثيقة مسجلة في المستودع.

يوجد مؤشر يربط بين كل دخول إلى قسم (chunk) من الوثيقة، بحيث يكون كل مؤشر له قسمان: اسم الوثيقة، ورقم القسم المتعلق به.

وتزداد احتمالية وجود تشابه بين الوثائق بتشابه الوحدات الصغيرة للأقسام، حيث وحدة القسم هي الكلمة.

تتم مقارنة الوثائق باستخدام نموذج التردد النسبي(RFM) الذي يتالف بشكل رئيسس من مجموعة من الكلمات التي لها نفس التردد في وثيقتين.

تستخدم طريقة استرجاع المعلومات المستخدمة في العثور على تشابه بين الاستعلام والوثائق، وتسمى هذه الطريقة بطريقة التصنيف، والتي تستخدمها محركات البحث ومعظم نظم الاسترجاع.[6]

يستخدم مقياس التشابه لحساب درجة التشابه بين الاستعلام والوثائق التي ترتب بشكل تنازلي حسب درجات التشابه، ومن ثم يتم إرجاع الوثائق المرتبة بشكل جيد.

يوجد أشكال متعددة من مقاييس التشابه لوضع درجة للتشابه. حيث اقترح هود وزوبل [6] صيغ مختلفة من مقاييش التشابه تعتمد على عدد تواجد الكلمات المتشابهة في الوثائق، طول الوثيقة، اختلاف تردد الكلمات بين الاستعلام والوثيقة، ومصطلح التوزين.

تبين النتائج المعلنة أن مصطلح التوزين هو من أفضل مقاييس التشابه، وخاصة بعد إزالة كلمات التوقف، وتقليص عدد كلمات الوثيقة.[10]

3.1.2. الطرق المعتمدة على أسلوب الكتابة:

تعتبر هذه الطريقة نهج إحصائي مستخدم لاكتشاف أسلوب الكتابة. يفترض أن لكل كاتب أسلوب فريد من نوعه. يمكن تحليل أسلوب الكتابة باستخدام عوامل داخل الوثيقة نفسها، أو من خلال المقارنة بين وثائق لنفس الكاتب.

يدعى نهج كشف الغش لنفس الوثيقة وبدون مراجع خارجية بنهج كشف الغش الجوهري.

يقوم بشكل عام على تقسيم النص إلى أقسام (فقرات أو جمل)، ثم تستخرج ملامح الأسلوب ويتم تحليلها.

- أهم السمات اللغوية للأسلوب:[11]
- إحصائيات النص التي تعمل على مستوى الحرف (عدد الفواصل، علامات الاستفهام، طول الكلمة..)
- السمات النحوية لقياس نمط الكتابة على مستوى الجملة (أطوال الجمل، استخدام كلمات الدلالة...)
- تصنيف أقسام الكلام لقياس استخدام الكلمات (عدد الصفات، الضمائر، الأفعال...)
- مجموعة كلمات صفوف مغلقة لعدّ الكلمات الخاصة (عدد كلمات التوقف، الكلمات الأجنبية، الكلمات "الصعبة"...)
- السمات الهيكلية التي تعكس تنظيم النص (أطوال الفقرات، أطوال الفصول...)

باستخدام هذه السمات، يمكن اشتقاق الصيغ لتحديد نمط كتابة المؤلف [11]:صيغة محددة للكاتب، وصيغة محددة للقارئ.

الصيغة المحددة للكاتب هي صيغة المؤلف نفسه. وتتضمن غنى المفردات وتعقيد وفهم المؤلف.

يقيس غنى المفردات عدد الكلمات المختلفة في الوثيقة، بينما يقيس التعقيد والفهم فهم الكاتب للوثيقة ويعطي نتيجة لهذه الوثيقة.

أما الصيغة المحددة للقارئ فهي تحدد مستوى قراء الوثيقة.

تمثل الأداة Glatt مثالاً عن نظام كشف الغش بطريقة تعتمد على نمط الكتابة. [12]

يمكن استخدام الطرق المعتمدة على نمط الكتابة في الكشف الداخلي والخارجي عن الغش، بينما تستخدم الطرق المعتمدة على المحتوى فقط في الكشف الخارجي عن الغش.

وإذا كان للكاتب أكثر من نمط كتابة واحد، فإن الطرق المعتمدة على الأسلوب يمكن أن تخطئ في الكشف عن الغش.

بشكل عام الطرق المعتمدة على المحتوى أفضل من تلك المعتمدة على الأسلوب من حيث الدقة [13]، حيث تقوم بتقديم استعراض عن نتائج الغش بالتفصيل.

تعتبر أقوى أدوات كشف الغش هي تلك التي تتحسس للغة، بحيث تراعي السمات اللغوية للغة معينة [13].

بينما بالرغم من أن الأدوات المستقلة عن اللغة تعمل لأكثر من لغة، إلا أنها تعطي نتائج سيئة غالباً.

حتى الآن، تعتبر APD هي الأداة الوحيدة المتواجدة القادرة على كشف الغش للغة العربية [14].

وهي تعتمد على وضع بصمة على كل وثيقة بأخذ مخطط 4-grams، ومقارنة المخططات الأقل تكراراً مع مجموعة داخلية من بصمات الوثائق.

يتم إجراء عملية الكشف عن التشابه بين الوثائق باستخدام تقنية استرجاع المعلومات على أساس مجموعات تقريبية.

2.2. ميزات اللغة العربية:

تنتمي اللغة العربية إلى مجموعة اللغات الأفرو–آسيوية، ولديها الكثير من الخصوصية مما يجعلها مختلفة عن غيرها من اللغات الهندو–أوروبية.

للغة العربية ثمانية وعشرون حرفاً أبجدياً (ا،ب،ت،.....،ي). حروف العلّة هي (ا، و،ي) والباقي حروف ساكنة.

يتغير شكل الحرف في اللغة العربية حسب موقعه في الكلمة، يمكن أن يكون ممدوداً بين حرفين.

تخطّ اللغة العربية من اليمين إلى اليسار، ولا تحتوي على أحرف كبيرة ولا صغيرة.

تُوضَع حركات التشكيل أعلى الحرف أو أدناه، وتفيد في تسهيل نطق الكلمة وتوضيح معناها. فغياب التشكيل في أغلب طرق التعبير العربية الإلكترونية والمطبوعة يشكل تحدياً كبيراً لفهم المعني الأصلي للكلمة.

تسمح اللغة العربية بإسقاط الفاعل أو المفعول به والتعويض عنهما بضمائر، كما في اللغة الإيطالية، والإسبانية والصينية.[15]

اللغة العربية هي لغة إعرابية للغاية. يمكن أن تكون الكلمة مركبة من جذر مضافاً إليه إحدى اللاحقات التي تعبر عن الزمن أو الجنس أو العدد، بالإضافة لوجود حروف الجر، والعطف، والضمائر.

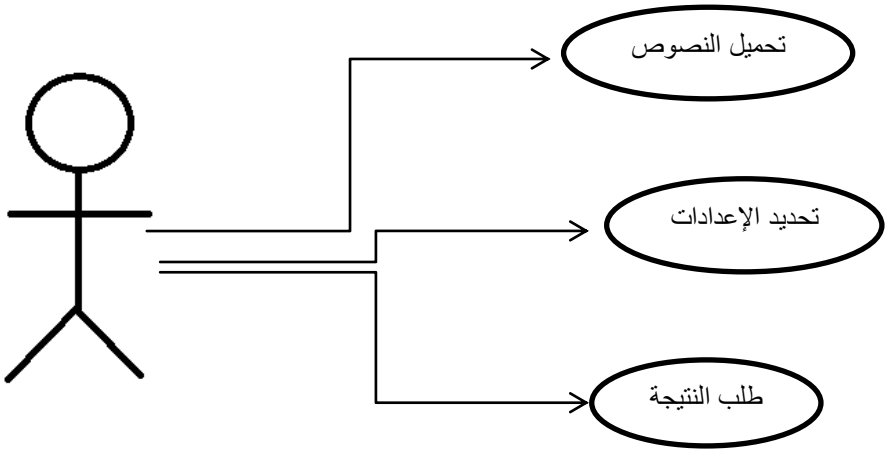
على سبيل المثال: كلمة المكاتب كلمة تدل على الجمع مفردها مكتب، وهذه الكلمة جذرها كتب.

الفصل الثالث: تحليل المتطلبات

يتضمن هذا الفصل حالات استخدام النظام حيث نستعرض فيه

مخططات حالات الاستخدام لأجزاء النظام مع وصف نصي لكل مرحلة

1.3. مخطط حالات الاستخدام:



2.3. الوصف النصي لحالات الاستخدام:

حالات الاستخدام	الفاعلون	الشرح
اختيار مجموعة النصوص	المستخدم: رئيسي	يقوم باختيار النصوص المراد الكشف عن التشابه بينها
كشف التشابه	المستخدم: رئيسي	يقوم البرنامج بفحص النصوص المختارة، ويطبق خوارزمية كشف الغش بينها.
استعراض تقرير بالنتائج	المستخدم: رئيسي	إظهار نتيجة التشابه بين النصوص، وعرضها على شكل تقرير مفصل.

الفصل الرابع: تصميم وتنجيز النظام

نستعرض في هذا الفصل تصميم بنيان النظام بشكل مفصل والخوارزميات المعتمدة ضمنه

إضافة إلى عرض لواجهات النظام

1.4. تصميم أداة كشف الغش باللغة العربية: APLAG

يجب أن يراعي نظام كشف الغش في اللغات الطبيعية الخصائص التالية:[7]

1- الحساسية لكل من علامات الترقيم والفراغات الإضافية والتشكيل... الخ.

2- الحساسية للتشابهات الصغيرة (يجب أن يكون التشابه كبيراً بما يكفي ليعبر عن الغش).

3- الحساسية للتبديلات في محتوى الوثيقة.

بنيت أداتنا لكشف الغش APlag باعتماد الأسلوب القائم على المحتوى والذي يحقق الخصائص الثلاث التالية:

- يتم التعامل مع الخاصية الأولى بمعالجة أي نص مدخل، بما في ذلك إزالة كلمات التوقف (الكلمات المكررة في اللغة والتي لا تدل على وجود تشابه)، التقطيع، إيجاد جذر الكلمة.
  ويستند APlag على وضع بصمة ممثلة بعدد محدد k.
- تتحقق الخاصية الثانية إذا لم تكن k طويلة كفاية لتجاهل التعابير الشائعة في اللغة العربية.
- تتجلى الخاصية الثالثة بنتائج الأداء على مجموعة البيانات "تغيير البنية".

يوضّح الشكل (1) البناء الرئيسي لل APlag:

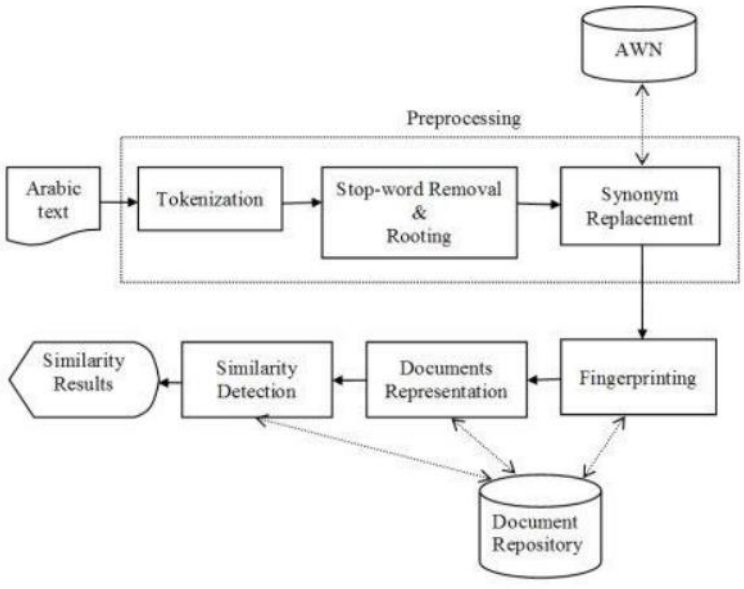


Fig. 1: Main architecture of APlag

الشكل (1)

تعتمد أداة APlag لكشف الغش على الطريقة القائمة على المحتوى، حيث يتم التحقق من الخواص الثلاث السابقة.

يتم التعامل مع الخاصية الأولى عن طريق المعالجة النصية للوثيقة، بما في ذلك التقطيع وإزالة كلمات التوقف، وإيجاد جذر الكلمة.

تتبى APlag في عملها مخطط k-grams ، حيث تكون الخاصية الثانية محققة إذا كان كبيرة كفاية ليتم تجاهل التعابير الشائعة في اللغة العربية.

بينما تظهر الخاصية الثالثة من خلال نتائج الأداء على مجموعة البيانات الهيكلية.

مسائل التصميم الأكثر أهمية هي المتعلقة بما يلي:

- التجهيز: التقطيع، إزالة كلمات التوقف، إيجاد جذر الكلمة.
- وضع البصمة :الاستفادة من مخطط k-grams، حيث k هي الوسيط الذي اختاره المستخدم.
- تمثيل الوثيقة : إنشاء بنية شجرة الوثيقة لكل وثيقة والتي تصف التمثيل الداخلي.
- اختيار بصمة تشابه :استخدام التشابه للعثور على أطول تشابه من سلسلتي التجزئة.

1.1.4. المعالجة النصية :

تحوي معظم طرق الكشف عن الغش المعتمد على المحتوى مرحلة التجهيز التي يتم فيها إزالة كلمات التوقف، ويتم إعادة كل كلمة لجذرها الأصلي.

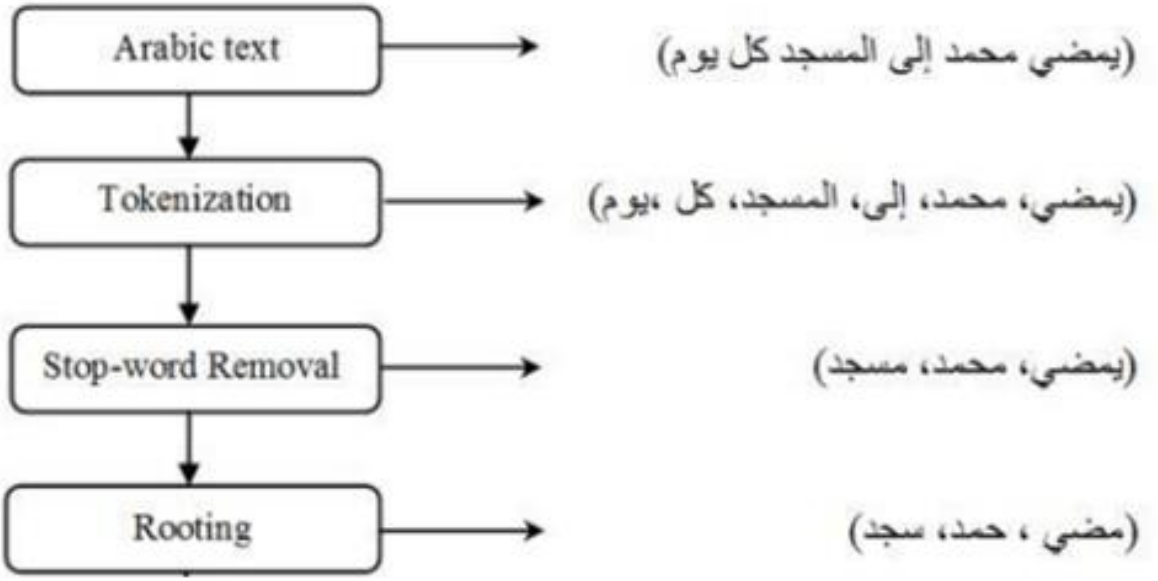
يتم تنفيذ الخطوات التالية لتحويل النص العربي إلى أفضل تمثيل منظّم ومنسق وملائم لكشف عملية الغش:

- التقطيع: تحويل النص المدخل إلى مقاطع (كلمات).
- إزالة كلمات التوقف: يوجد كلمات تستخدم بكثرة في أي نص، تعتبر هذه الكلمات من التشابهات غير المهمة بين الوثائق.

ولذلك تتم إزالتها من أجل تقليص التشابهات غير المفيدة، ولإعطاء نتائج أكثر دقّة.

إيجاد جذر الكلمة: أي إعادة كل كلمة لجذرها، ويستخدم نظام Khoja لإيجاد الجذر [16]، بحيث يعيد الكلمة إلى جذرها التي اشتقت منه الكلمة وذلك من خلال حذف اللاحقات ومن ثم مطابقة الكلمة الناتجة مع أنماط الأفعال والأسماء الموجودة في بيانات النظام.

فيوضح الشكل مراحل هذه المرحلة بالتفصيل:



الشكل (2)

2.1.4. مرحلة البصمات وتقييس التشابه:

لاستخراج تشابه وثيق سنقوم بداية بتقطيع النص إلى أجزاء أصغر[17]. يمكن أن تستخدم الجملة أو الكلمة كوحدة للتجزيء .في حالة التجزيء المعتمد على الجمل، سنعتمد على بعدn يحدد عدد الجمل في كل قسم من الوثيقة، فمثلا لو أخذنا وثيقة مكونة من خمس جمل s1 s2 s3 s4 s5 :واخترنا البعد n=3 ستقسم الوثيقة إلى ثلاثة أقسام هي: s1 s2 s3, s2 s3 s4, s3 s4 s5 ، وفي حالة التجزيء المعتمد على الكلمة، تقسم الوثيقة إلى أقسام معتمدة على وسيط التقسيم n حيث تعتبر كل n كلمة متتابعة عبارة عن قسم. على سبيل المثال: إذا كانت الوثيقة المعطاة تحتوي على الكلمات w1, w2, w3, w4, w5

وإذا كانت n=3 ستكون الأقسام هي: w1 w2 w3, w2 w3 w4, w3 w4 w5.

إن التقسيم المعتمد على الكلمة أكثر دقة في تحديد التشابه من التقسيم المعتمد على الجملة.

الأداة APlag تعتمد على التجزيء المعتمد على الكلمة، حيث في كل جملة من الوثيقة تقسم الكلمات أولاً ثم تعرّف كل كلمة برقم حسب تابع تجريبي يختاره الخبير.

من المهم اختيار تابع تجريبي hash function يقلص نسبة التصادمات . فمثلاً: تنجيز تابع تجريبي يربط كل مقطع chunk بمجموع الأعداد الصحيحة لقيم محارف المقطع. قد لا يكون التابع على قدر كبير من الدقة وذلك لأن المقطع قد يكون له نفس المحارف ولكن بترتيب مختلف.

في حالتنا : قمنا باختيار تابع تجريبي يقوم بحساب مجموع جداء القيمة العددية للمحرف في جدول الـ ASCII مضروبة بترتيب المحرف كما هو موضح في الثال:

جمع: (ج) 1* 9+2*34+3*22=143

حيث: 9 هي قيمة الحرف (ج) في جدول الـ ASCII مطروحا منه الحرف الأبجدي العربي الأول (أ) وذلك لتخفيف القيمة العددية، وهكذا....

يوجد الكثير من أبعاد التشابه الممكنة لمقارنة البصمات، مثل: مسافة Levenshtein، أطول سلسلة مشتركة LCS، وغيرها..[18]

وعلى سبيل المثال: تقيس مسافة Levenshtein أصغر عدد من عمليات: الإدخال، الحذف، والتعديل لتحويل كلمة إلى كلمة أخرى، أما طريقة أطول سلسلة مشتركة فهي تختص بالبحث عن المقطع الأطول المشترك بين سلسلتين مثلا: أطول مقطع مشترك بين طريق و تطريز هو طري.

إذا المسألة الرئيسية في كشف الغش هو عملية اختيار المقياس المناسب. لذلك، نلاحظ أن طريقة LCS ومسافة Levenstein هما الأنسب في حالتنا وخاصة أن الغش قد يشمل التعديل على نص (حذف أجزاء منه أو إضافة أجزاء..)

سوف نختار في نظامنا APlag طريقة السلسة الأطول المشتركة LCS، لأنها تعتمد على مفهوم التشابه أكثر من التقنيات الأخرى.

3.1.4. مقارنة النصوص:

يتم إنشاء تمثيل شجري لكل وثيقة من أجل وصف بنيتها المنطقية، بحيث يمثل جذر الشجرة النص نفسه، والمستوى الثاني يمثل الفقرات، أما أوراق الشجرة فتمثل الجمل.[10]

يهدف هذا التمثيل إلى تجنب المقارنات غير الضرورية بين وثائق متعددة، حيث يتم استكشاف الأشجار من الأعلى إلى الأسفل، وبعدها تجرى مقارنة أولية على مستوى النص، ثم على مستوى الفقرة، وأخيرا على مستوى الجملة.

نحدد خوارزمية المقارنات لكل مستوى في الشجرة بالخوارزميات الثلاث: خوارزمية على مستوى النص، خوارزمية على مستوى الفقرة، وخوارزمية على مستوى الجملة.

تتم مقارنة وثيقتين على مستوى النص وفقا للبصمة التي تم وضعها مسبقا وعتبة معينة. إذا كان عدد التقاطعات أكبر من العتبة المختارة، عندها كون التشابه محتمل بين الوثيقتين. في هذه الحالة، تجرى عملية المقارنة على مستوى الفقرة، فإن لم يتم الكشف عن التشابه يتم إيقاف العملية. أما إذا تم الكشف عن تشابه على مستوى الفقرة تستمر العملية على مستوى الجملة. فإما تتوقف العملية إذا لم يتم الكشف عن تشابه على مستوى الجملة أو إذا كان هناك تشابه ممكن بين جملتين، يتم قياس نسبة التشابه بواسطة خوارزمية السلسلة المشتركة الأطول LCS. إذا كان طول هذه السلسلة هو أكبر من طول لجملة الأقصر مضروبا بعتبة معينة، يتم أخذ الجمل المتشابهة بعين الاعتبار وإلا تستمر العملية على الجملة التالية.

4.1.4. الخوارزميات المتبعة في النظام:

Algorithm 1: Document level heuristic
Input: DocA, DocB // Two input documents
Output: similarity
Begin
DocMinSize = min (|DocA|, |DocB|)
DocIntersectionSize = |DocA∩DocB|
If (DocIntersectionSize>= DocMinSize*DocThreshold)
Then
//Possible similarity
//Check similarity at paragraph level
similarity = true
Else
similarity = false
End

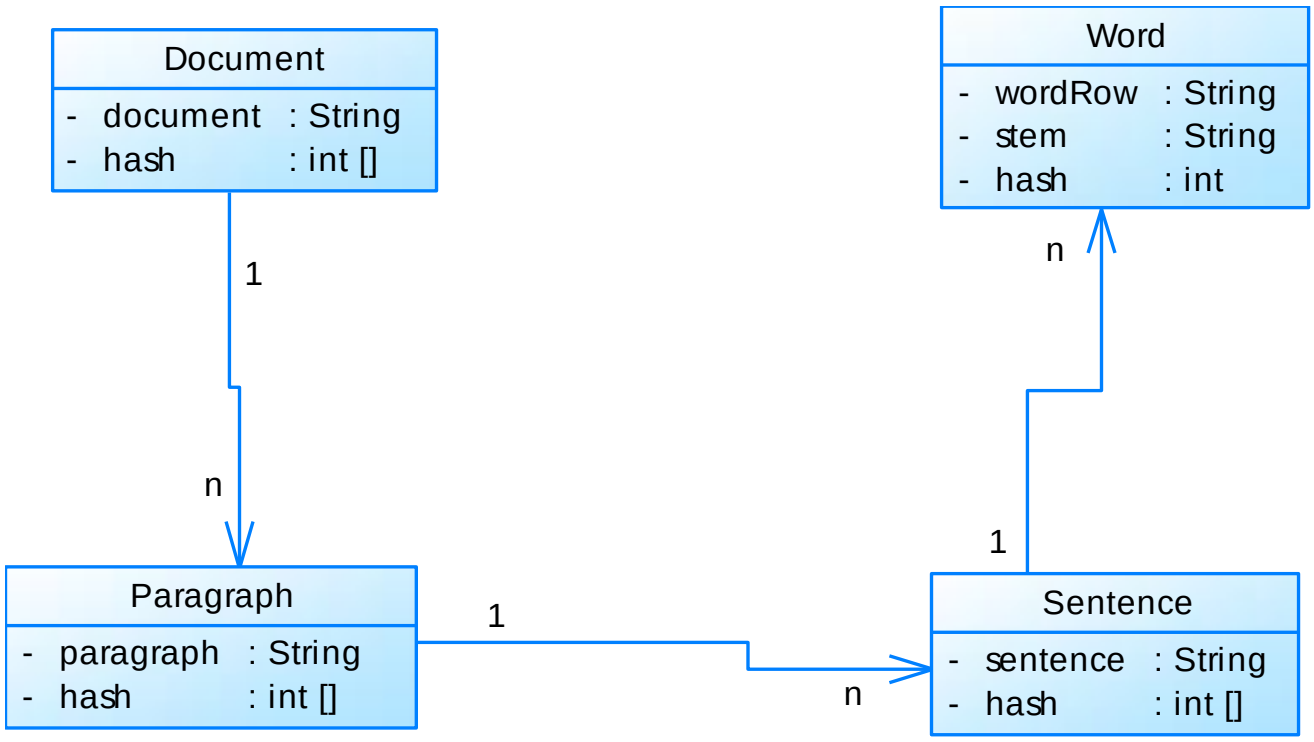
Algorithm 2: Paragraph level heuristic
Input: ParA, ParB // Two input paragraphs
Output: similarity
Begin
ParMinSize = min (|ParA|, |ParB|)
ParIntersectionSize = |ParA∩ParB|
If (ParIntersectionSize>= ParMinSize*ParThreshold)
Then
//Possible similarity
//Check similarity at sentence level
similarity = true
Else
similarity = false
End

Algorithm 3: Sentence level heuristic
Input : SenA, SenB
Output: similarity, similar substrings in SenA and SenB
Begin
SenMinSize = min(|SenA|, |SenB|)
SenIntersectionSize = |SenA∩SenB|

```
        If (SenIntersectionSize>= SenMinSize*SenThreshold)
                Then
                LongestCommonSeq = LCS (SenA, SenB)
                If (|LongestCommonSeq| >= SenMinSize*SimilarityThreshold)
                Then
                //Similarity detected
                //Determine similar
                //substrings
                similarity = true
                Else
                similarity = false
        Else
        similarity = false
End
```

2.4. تنجيز النظام:

1.2.4. مخطط الأغراض :



قمنا بتنفيذ قاعدة بيانات كما يوضح الشكل حيث:

تمثل الكلمة كياناً يحتوي على واصفات تعبر عن الكلمة الأصلية، والجذر الذي نبحث عنه، ويحوي البصمة الخاصة بالكلمة.

تمثل الجملة بكيان يحتوي على مجموعة من الكلمات، ومصفوفة البصمات لأقسام الجملة.

تمثل الفقرة بكيان يحتوي على مجموعة من الجمل، والبصمات.

وأخيرا تمثل الوثيقة الكيان الأعلى الذي يتألف من مجموعة من الفقرات، ومصفوفة بصمات.

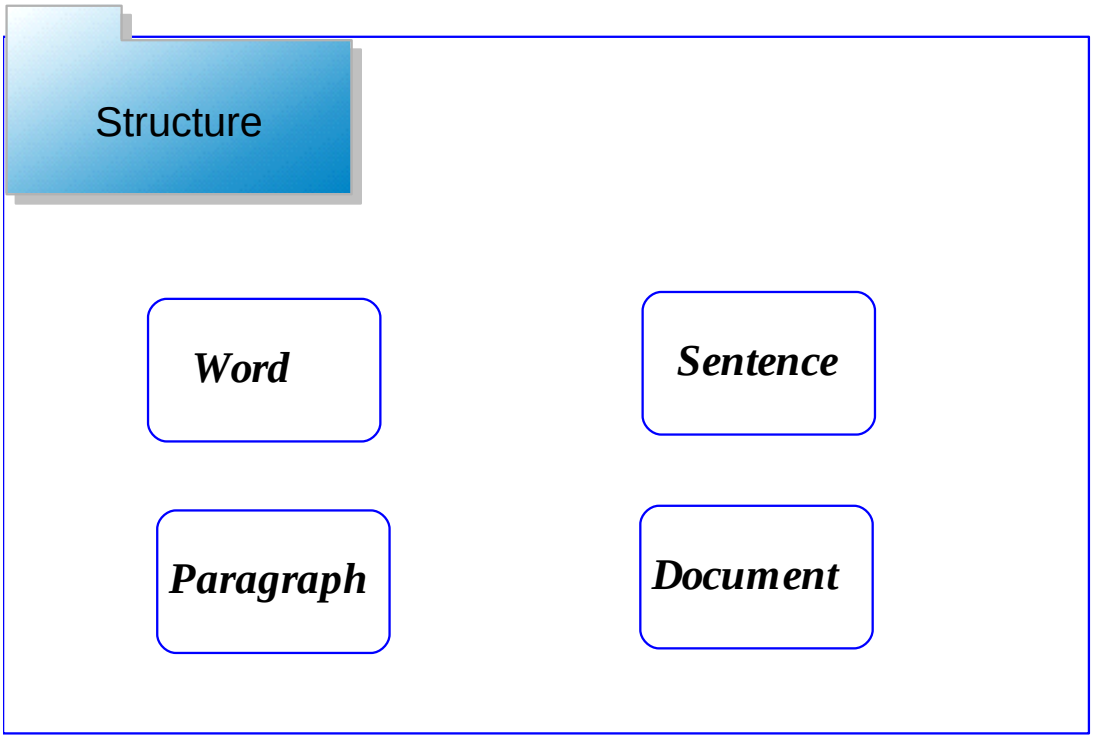
2.2.4. مخطط الصفوف:

يوضح الشكل التالي البنية المقترحة للنظام كاملاً ومخطط عمله:

الحزمة الأولى:

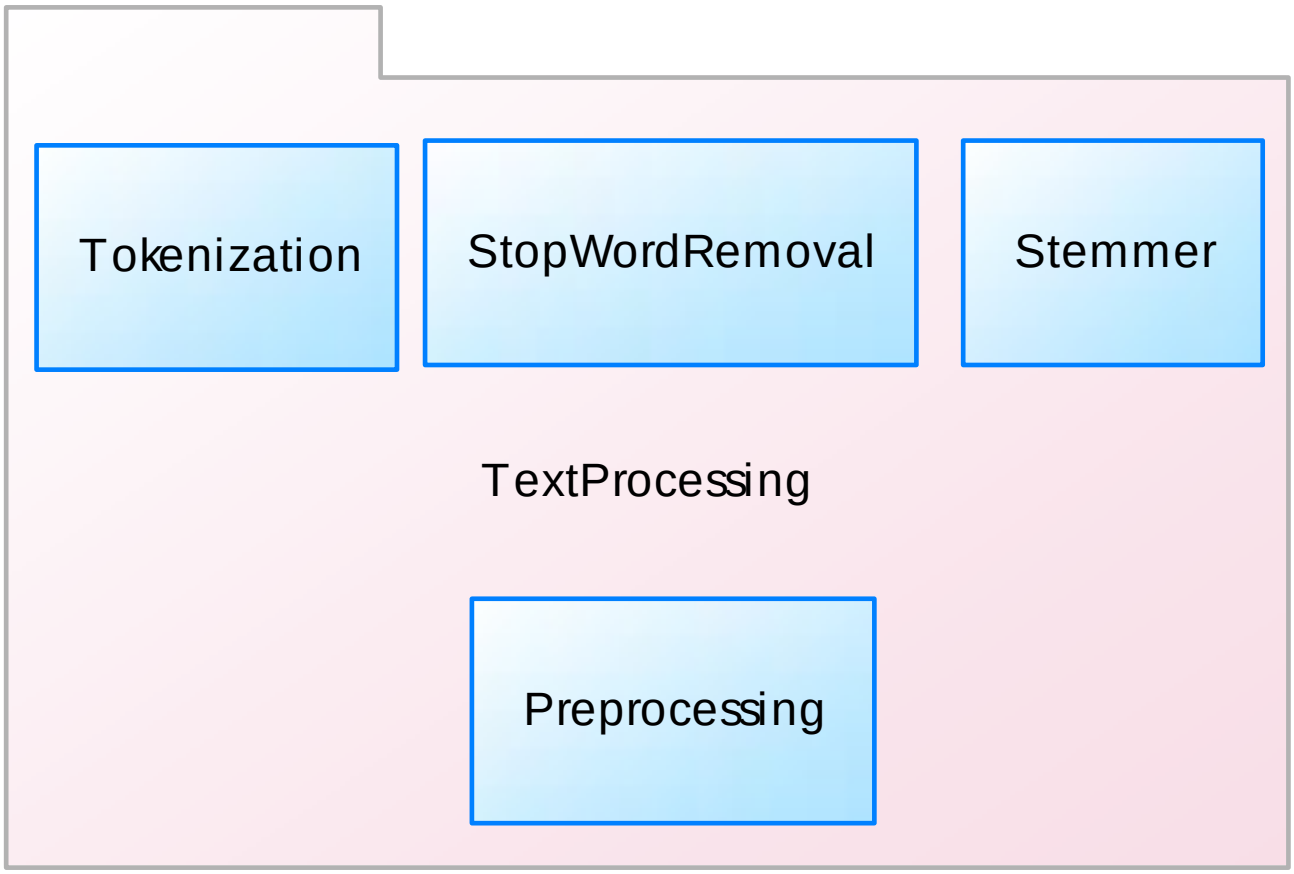
هي حزمة بنية المعطيات التي قمنا بتنفيذها: Structure

وتحتوي على الصفوف: Word, Sentence, Paragraph, Document.



الحزمة الثانية:

هي حزمة تنجيز النظام حيث تحتوي على مراحل المشروع كافة:

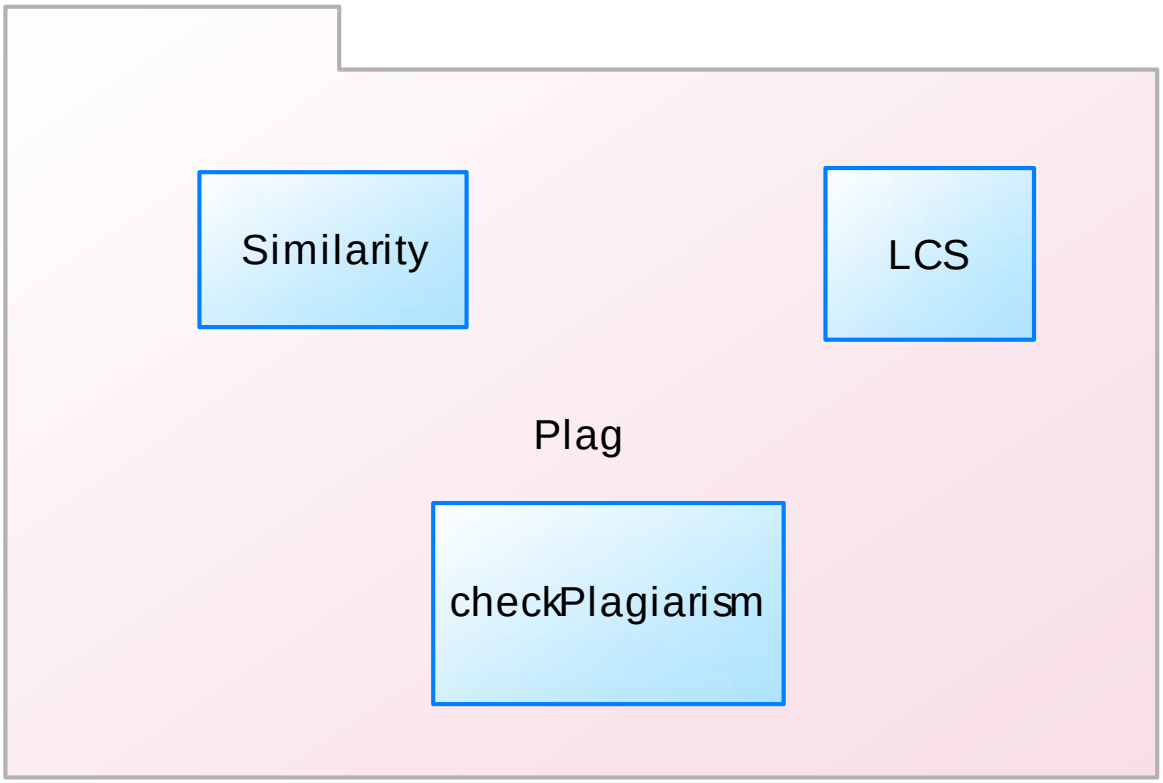


- **الصف Tokenization**: يقوم هذا الصف بتجزئة النص وإزالة علامات الترقيم وحركات التشكيل، وكل الرموز التي لا تفيد في كشف التشابه بين الوثائق.
- **الصف StopWordRemoval**: يقوم هذا الصف بإزالة كل الكلمات التي لا يفيد وجودها بكشف أي تشابه بين الوثائق، مثل: إلى ، في، عليكم، منكم...
- **الصف StemHandler**: هو الصف الذي يقوم بتنجيز التوابع الموجودة في الصف Stem بالشكل الذي يناسب نظامنا.
- **الصف preprocessing**: هو الصف الذي يقوم بتنفيذ الصفوف السابقة وربطها مع البنية المناسبة، بحيث يدخل النص كما هو، يقوم الصف بتجزئته وإزالة كلمات التوقف وإيجاد الجذر.

ويعتبر هذا الصف هو الرابط بين مركبات النظام والحزمة التي تحتويه، إذ يمكن لأي تعديل على الحزمة أن يكون في هذا الصف فقط، وأي مبرمج يريد التعرف على هذه الحزمة يكفي أن يتطلع على هذا الصف.

الحزمة الثالثة:

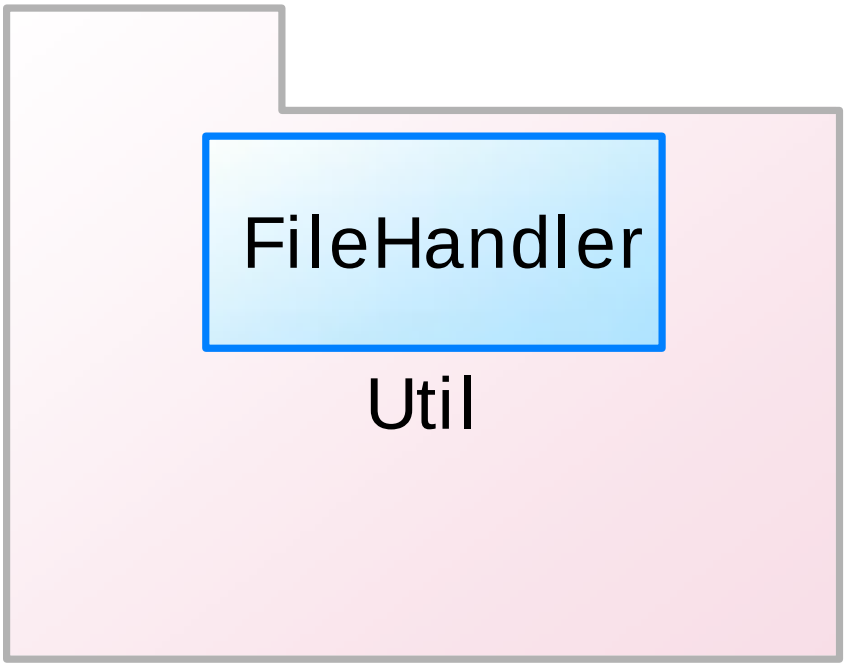
هل الحزمة التي تربط جميع مكونات النظام، وفيها:



- **الصف Similarity**: هو الصف الذي يقوم بكشف التشابه حسب البصمات التي تم وضعها مسبقا لكل كلمة، جملة، فقرة ووثيقة.
- **الصف checkPlagiarism**: هو الصف الذي يقوم بإدارة جميع العمليات بدءا من تحميل النصوص على شكل ملفات مروراً بالمعالجة النصية، وضع البصمة، والمقارنة.. وهكذا حتى الوصول إلى نتجة الكشف عن الغش.
- **أما الصف LCS**: فهو صف منجّز يحوي شرح مفصّل عن خوارزمية السلسلة المشتركة الأطول، حيث يستخدمها لنظام عند الكشف عن التشابه على مستوى الجمل.

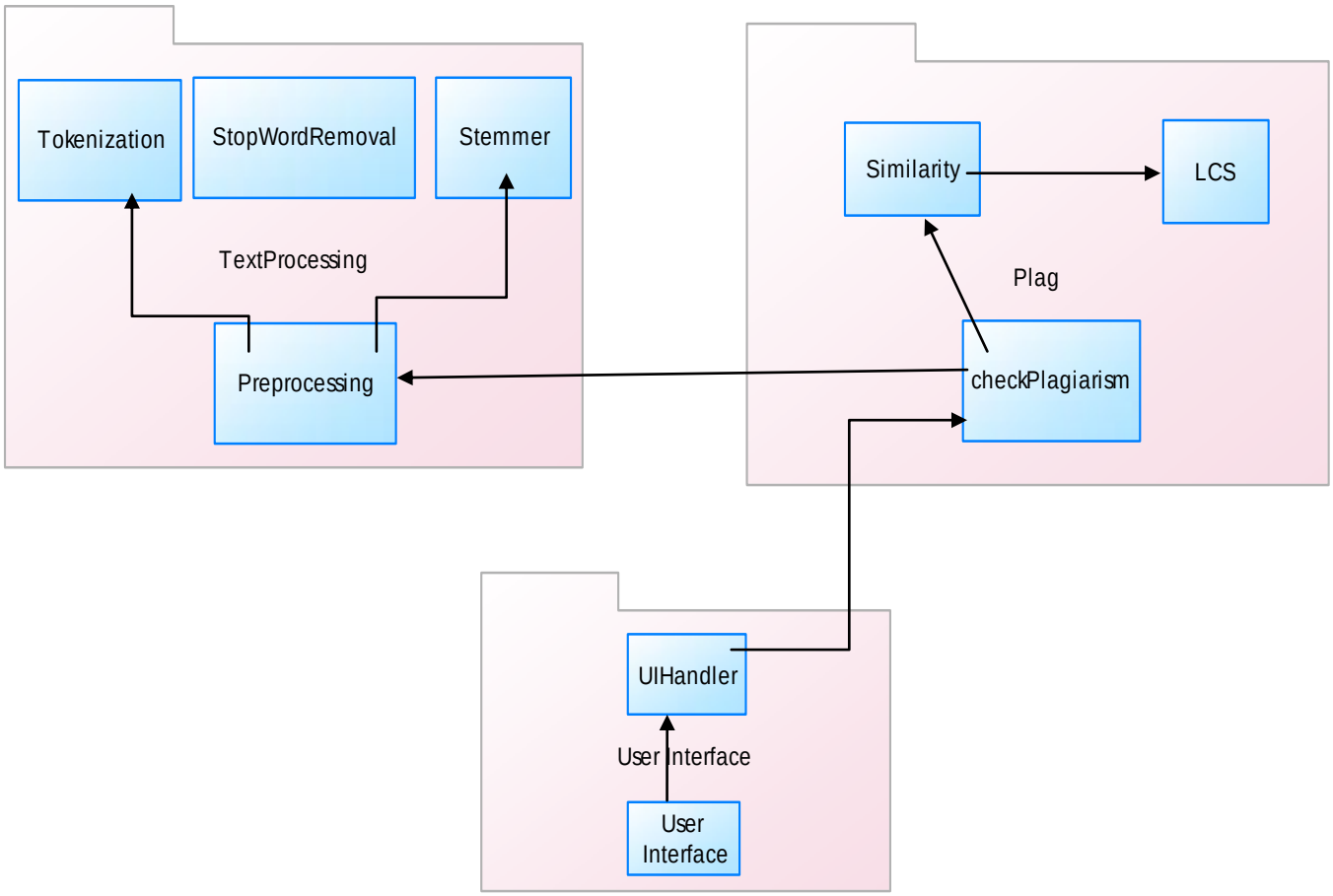
الحزمة الرابعة:

هي حزمة لا تتعلق بمتطلبات المشروع بشكل مباشر، وإنما هي صفوف نحتاجها بشكل عام، مثل قراءة ملفات واستخدام خوارزميات عامة كما سنرى بالشكل:

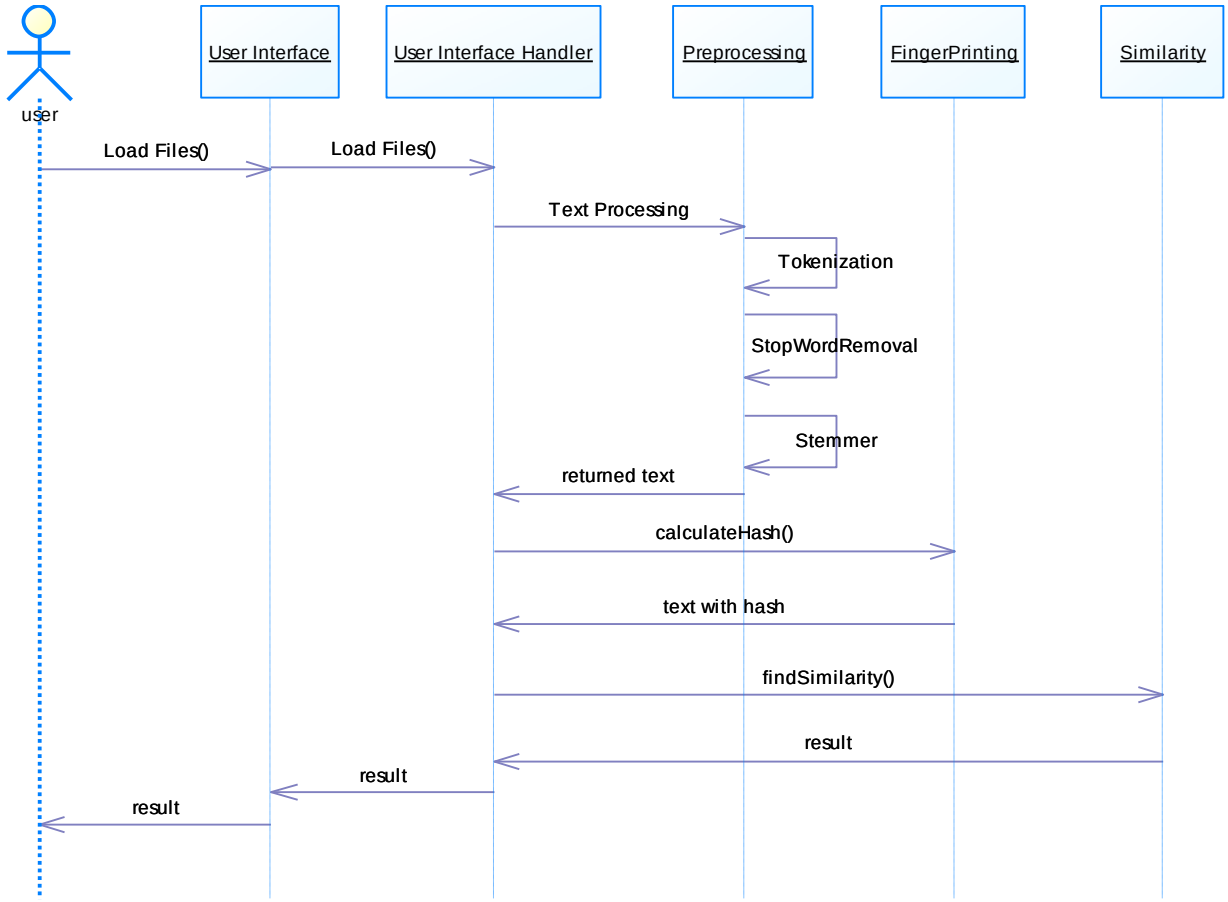


يحوي الصف ***FileHandler*** على توابع تسهل قراءة الملفات، وتحويلها إلى مختلف الصيغ التي نحتاجها في المشروع.

فيكون المخطط النهائي للنظام هو كالتالي:

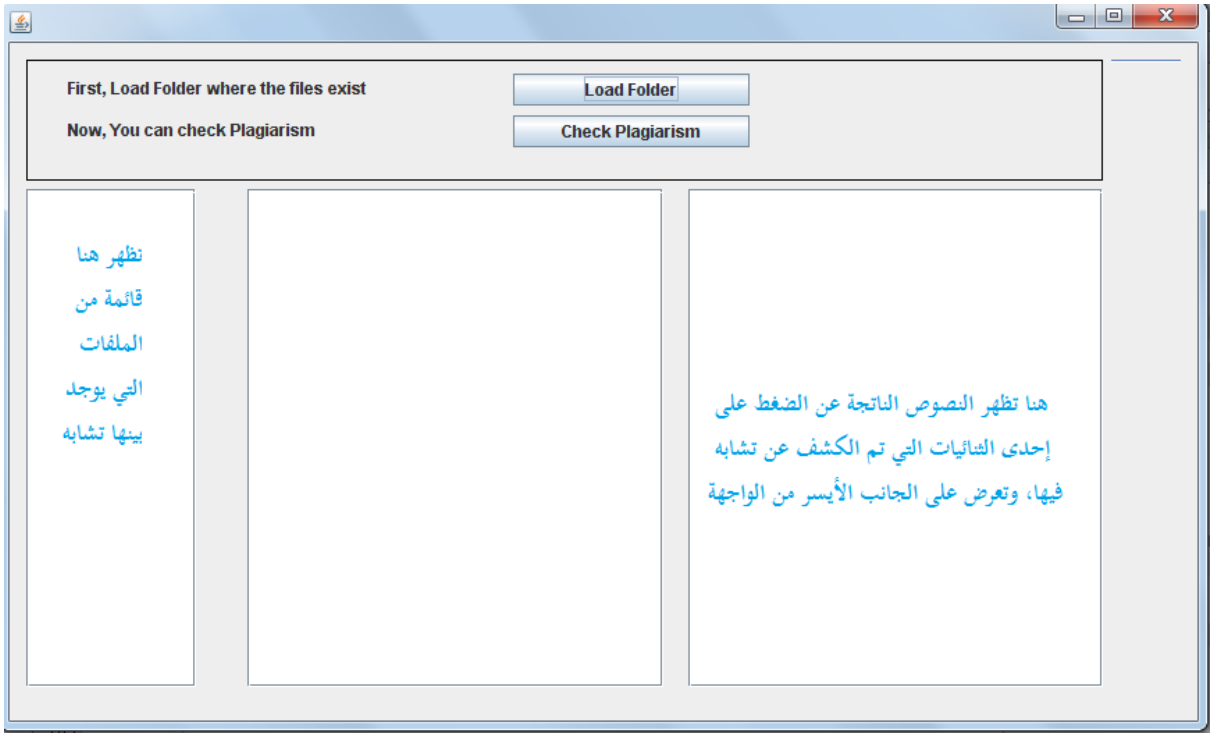


نوضح فيما يلي مخطط التتالي للنظام:



3.4. واجهات النظام:

استخدمنا لتوصيف النظام واجهة وحيدة.



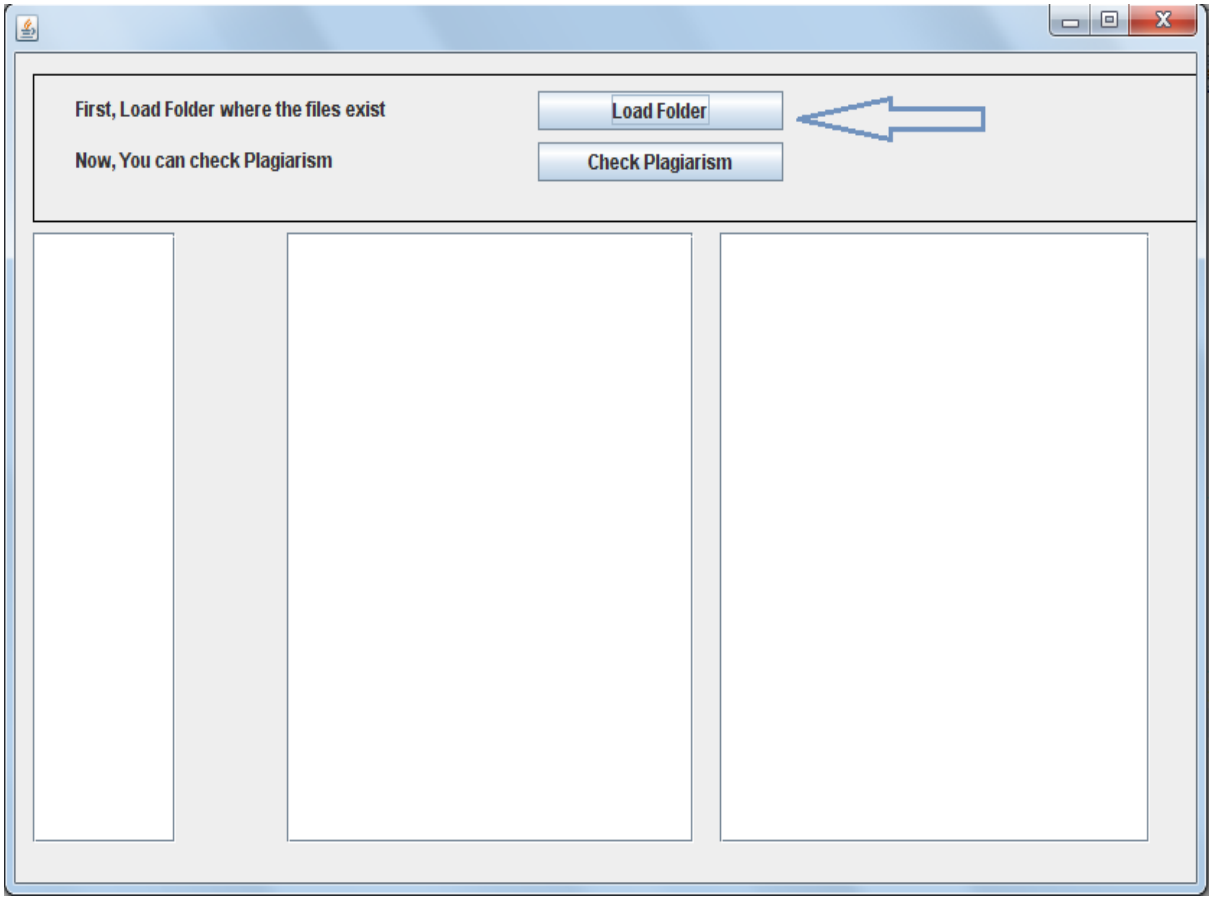
الفصل الخامس: التنفيذ والاختبارات

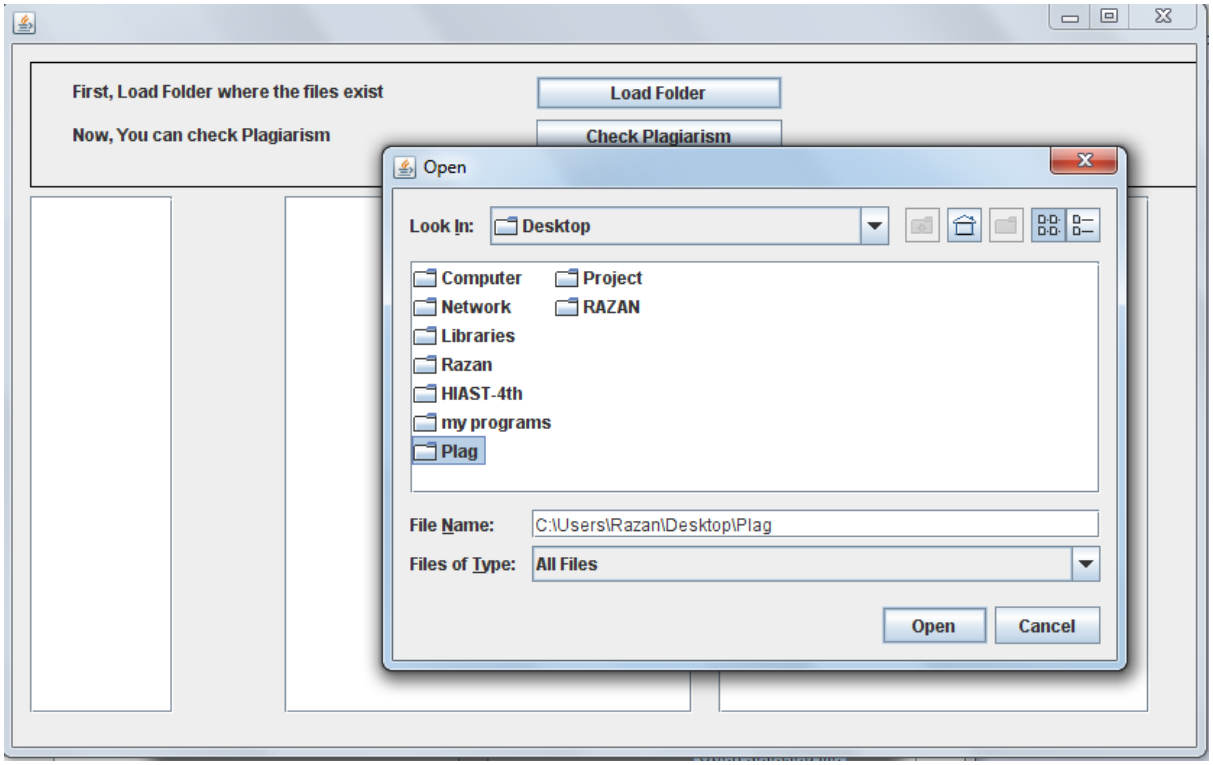
نستعرض في هذا الفصل بيئة العمل المعتمدة وخطة الاختبارات والآفاق المسقبلية للمشروع

1.5. بيئة العمل المعتمدة

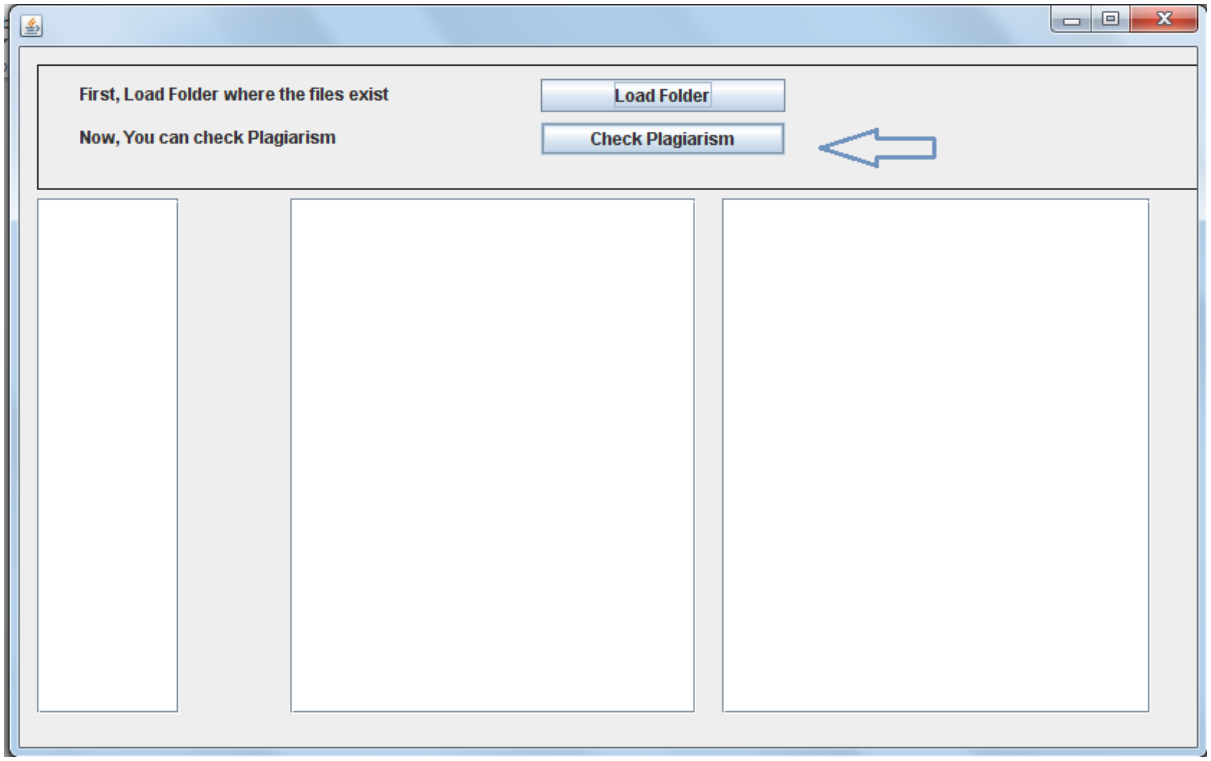
2.5. خطة الاختبارات:

يمكن من خلال الواجهة اختيار مجلد يحتوي على الملفات المراد فحصها:

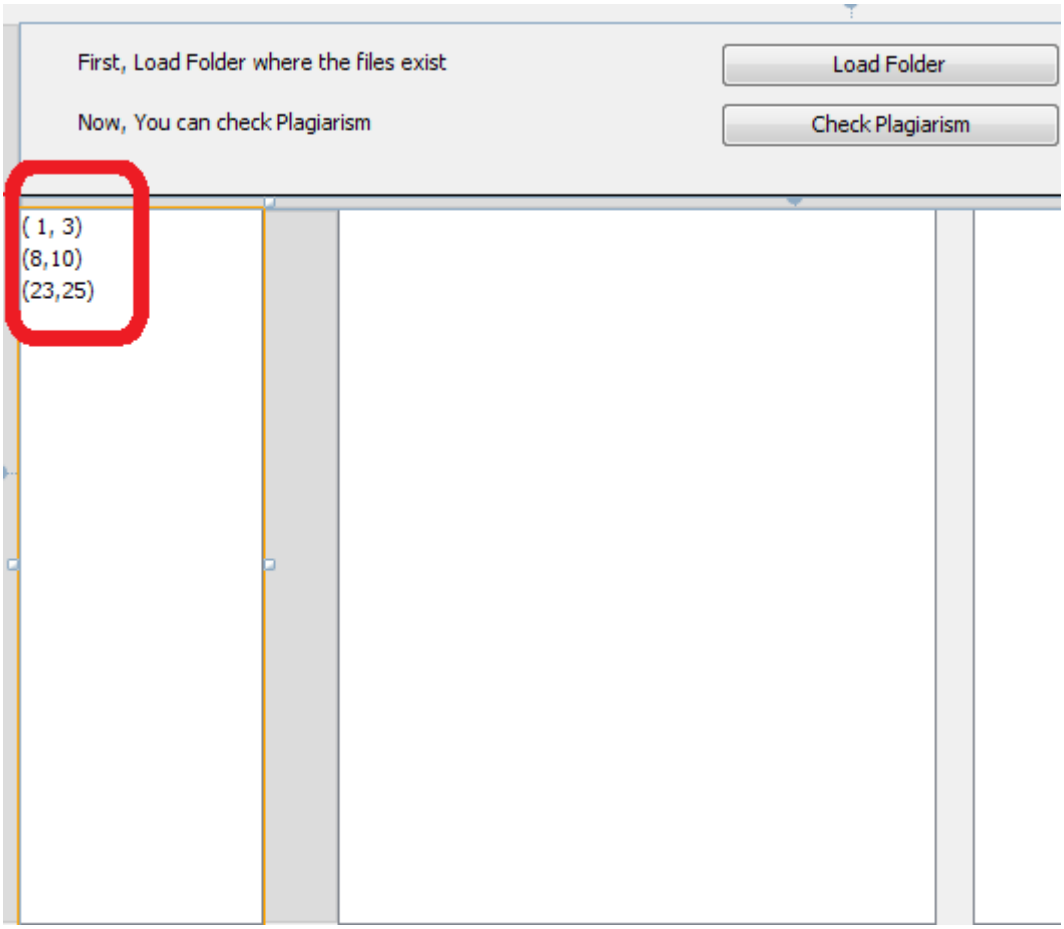




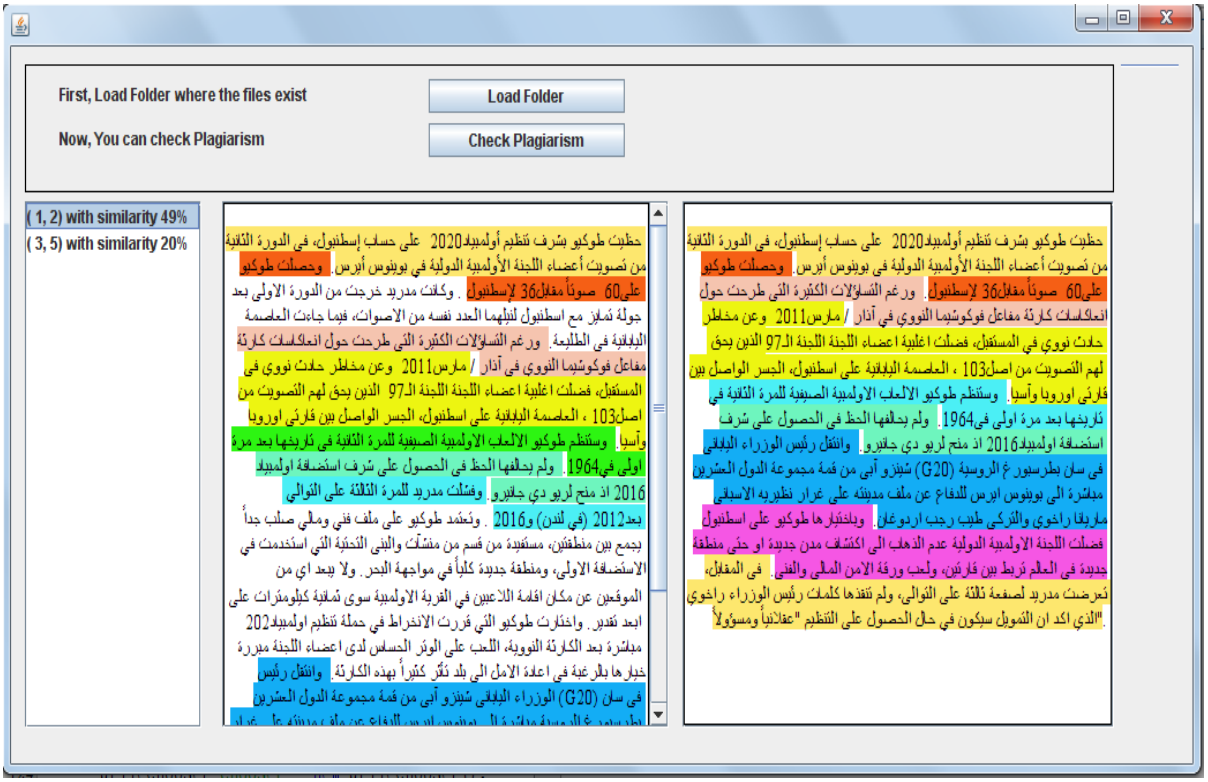
ثم يقوم المستخدم بطلب كشف الغش:



فيظهر على الواجهة ثنائيات من الوثائق التي يوجد تشابه فيما بينها، إذا ضغطنا على إحدى الثنائيات، تظهر المقاطع المشتركة بين الوثيقتين.



في حال ضغطنا على إحدى الثنائيات يظهر النصين المقابلين، ويتحدد المقاطع التي تم الكشف عن الغش فيها بلونٍ مختلف.



3.5. الآفاق المستقبلية:

من الجيد إجراء اختبارات إضافية على قواعد البيانات لمرادفات كلمة، وإجراء تغييرات في الأبعاد المستخدمة مثل: بعض العتبات والقيم التجريبية.

كما يمكن إضافة مرحلة متقدمة للكشف عن الغش وهي مرحلة استبدال الكلمة بإحدى مرادفاتها،إذيتم تحويل الكلمات إلى مرادفاتها الأكثر شيوعاً، والتي تساعد على كشف أشكال متقدمة مخفيّة من الغش.

يمكنالاستعانة بمرادفات كلمة من البيئة(AWN) Arabic WordNet بحيث يعتبر المرادف الأول لكلمة معينة في قائمة المرادفات هو المرادف الأكثر شيوعاً.

الخاتمة:

قدمنا بنية مقترحة من نظام كشف الغش للوثائق العربية APlag، التي يمكن من خلاله الكشف عن بعض الأشكال الخفية من الغش، مثل: تغيير بناء الجملة واستبدال كلمة بمرادفاتها.

وصفنا المكونات الرئيسية وخوارزميات الكشف عن التشابه بمقارنة البصمات بين الوثائق العربية على مستويات منطقية مختلفة (النص، الفقرة، الجملة).

وأخيرا لإثبات فعالية النظام قدمنا وناقشنا سلسلة من التجارب على مجموعة كبيرة من الوثائق العربية. وأشارت النتائج إلى قدرة APlag على الكشف عن النسخ طبق الأصل، والتغيرات في بناء الجملة، واستبدال الكلمة بإحدى مرادفاتها.

المراجع:

[1] Lukashenko R., Graudina V., Grundespenkis J. Computer-based plagiarism detection methods and tools: an overview [C]. In: Proceedings of the International Conference on Computer Systems and Technologies, Bulgaria, 2007, 14-15.
[2] Maurer H., Kappe F., Zaka B. Plagiarism – A survey [J]. Journal of Universal Computer Science, 2006, 12(8): 1050-1084.
[3] Gruner G., Naven S. Tool support for plagiarism detection in text documents [C]. In: Proceedings of the ACM symposium on Applied Computing, Santa Fe, New Mexico, 2005, 13-17.
[4] Menai M.B., Al-Hassoun N.S. Similarity detection in Java programming assignments [C]. In: Proceedings of the 5th International Conference on Computer Science & Education, Hefei, China, 2010, 356-361.
[5] Mozgovoy M., Kakkonen T., Sutinen E. Using natural language parsers in plagiarism detection [C]. In: Proceedings of the SLaTE Workshop on Speech and Language Technology in Education, Farmington, Pennsylvania, USA, 2007.
[6] Hoad C., Zobel J. Methods for identifying versioned and plagiarized documents [J]. Journal of the American Society for Information Science and Technology, 2003, 54(3): 203-215.
[7] Schleimer S., Wilkerson D., Aiken A. Winnowing: local algorithms for document fingerprinting [C]. In: Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data, San Diego, California, USA, June 2003, 9-12.
[8] Dumais S.T. Latent Semantic Analysis [J]. Annual Review of Information Science and Technology, 2005: 38-188, doi:10.1002/aris. 1440380105.
[9] Shivakumar N., Garcia-Molina H. SCAM: a copy detection mechanism for digital documents [C]. In: Proceedings of the 2nd International Conference on Theory and Practice of Digital Libraries, Austin, Texas, USA, June 1995.
[10] Si A., Leong H., Lau R. CHECK: a document plagiarism detection system [C]. In: Proceedings of ACM Symposium for Applied Computing, Feb. 1997, 70-77.
[11] Eissen S., Stein B., Kulig M. Plagiarism detection without reference collection [C]. In: Proceedings of the 30th Annual Conference of the German Classification Society, Berlin: Freieuniversity, 8–10 Mar. 2006, 359-366.
[12] http://www.plagiarism.com/self.detect.htm, visited: 15 Jan. 2012.
[13] Lancaster T., Culwin F. Classifications of plagiarism detection engines [J]. ITALICS, 2005, 4(2).
[14] Alzahrani S.M., Salim N. Statement-based fuzzy-set IR versus fingerprints matching for plagiarism detection in Arabic documents [C]. In: Proceedings of the 5th Postgraduate Annual Research Seminar (PARS 09), Johor Bahru, Malaysia, 2009.
[15] Farghaly A., Shaalan K. Arabic natural language processing: challenges and solutions [J]. ACM Transactions on Asian Language Information Processing, 2009, 8 (14): 1-22.
[16] Khoja S. Stemming Arabic Text [R]. 1999.
http://zeus.cs.pacificu.edu/shereen/research.htm
[17] Pataki M. Plagiarism detection and document chunking methods [C]. In: Proceedings of the 12th International WWW Conference, Budapest, Hungaria, May 20-24, 2003.
[18] Levenshtein V.I. Binary codes with correction for deletions and insertions of the symbol 1 [J]. Probl.Peredachi Inf., 1965, 1(1), 12–25.
[19] **Mohamed El Bachir Menai,** Department of Computer Science, College of Computer and Information Sciences, King Saud University, P.O. Box 51178, Riyadh 11543, Saudi Arabia ,menai@ksu.edu.sa.