People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research

Constantine 2 University
Faculty of New Technologies of Information and Communication
Department of Fundamental Computer Science and its Applications

Plagiarism Detection: A focus on the Intrinsic Approach and the Evaluation in the Arabic Language

by

Imene Bensalem

Thesis submitted in fulfilment of the requirements for the degree of

Doctor of Sciences

Defended on 29 Feb. 2020 in front of the scientific committee members:

President	Prof. Ramdane Maamri	Full professor at Constantine 2 University
Local Advisor	Prof. Salim Chikhi	Full professor at Constantine 2 University
External Advisor	Prof. Paolo Rosso	Full professor at Universitat Politècnica de València
Examiners	Dr. Alberto Barrón-Cedeño	Senior assistant professor at Università di Bologna
	Prof. Yacine Lafifi	Full professor at University of Guelma
	Dr. Sihem Mostefai	Associate professor at Constantine 2 University

The author of this thesis is a member of:

Laboratory of Modelling and Implementation of Complex Systems, SCAL Group, Constantine 2 University, Algeria

(Head: Prof. Salim Chikhi)

And benefited from 2 short internships at:

Natural Language Engineering Laboratory, Universitat Politècnica de València, Spain

(Head: Prof. Paolo Rosso)

Acknowledgements

Firstly, I would like to express my sincere gratitude to my local Advisor Prof. Salim Chikhi and my external advisor Prof. Paolo Rosso. I owe my research achievements to your diligent supervision.

Prof. Paolo, I have been lucky enough to have a chance to hold my doctorate research under your supervision. Although most of the time your guidance comes from behind the screen, you were a great inspiration (professional and personal) on the path to my current expertise. Thank you for receiving me twice in your lab as an intern, and for your continuous guidance and encouragement throughout all the doctorate years. I am particularly indebted to you for giving me the opportunity to learn the necessary skills for research: proper evaluation, scientific event organisation, and papers reviewing, among others. I am deeply grateful to your unreserved feedback, perceptive advice, patience, flexibility, support and understanding during the awkward moments. Grazie mille for everything!

Prof. Chikhi, I am deeply grateful to you for the freedom you have given me in choosing my research path and in working in my way and at my pace. I owe a sincere Thank you for your insightful advice, patience, positivity, encouraging words, and support that helped a lot in reducing the stress of the doctorate journey. I would like to express my appreciation of your wide perception that was always of help to better structure and present my written work for different categories of readers, regardless of their familiarity with the subject. Chukran djazilen!

Besides my advisors, I would like to sincerely thank the scientific committee members of the defence: Dr. Alberto Barrón-Cedeño (Università di Bologna), Prof. Yacine Lafifi (8 Mai 1945 University of Guelma), Dr. Sihem Mostefai (Constantine 2 University), and Prof. Ramdane Maamri (Constantine 2 University), for the time they reserved to read and evaluate my thesis. I am particularly grateful to Dr. Alberto Barrón-Cedeño for his constructive and meticulous review, which helped me improve the last version of the thesis and inspired me with ideas for future work.

I would like to thank Imene Boukhalfa (Doctorate candidate in Constantine 2 University), and Dr. Lahsen Abouenour (École Mohamadia d'Ingénieurs, Mohamed V Rabat University, Morocco) for their collaboration to create ExAra corpus.

Special thanks are reserved for my sister Abir (Marketing MBA), my friend Hanene Zitouni (Maître assistante in Constantine 2 University), her sisters Marwa and Loubna (Students at Mentouri University) and my friend Khouloud Meskaldji (Maître assistante at École Normale Supérieure of Constantine) for their assistance in the creation of some manual plagiarism cases for the ExAra corpus.

My sincere thanks also go to Mr. Larbi Benkhabchech who gave me access to material facilities at École Supérieure de Comptabilité et de Finances of Constantine. Without his support, it would not be possible to conduct the last experiments of my research.

I would like to mention that the research for this thesis was partially funded by the École Supérieure de Comptabilité et de Finances of Constantine and also the bilateral research project AECID-PCI AP/043848/11 (Application of Natural Language Processing to the Need of the University) between Constantine 2 University and the Universitat Politècnica de València in Spain.

Last but not least, I would like to express my immense gratitude and my love to my family (my parents and siblings Walid and Abir) for the emotional support, love and encouragement throughout this challenging project. Special thanks go to my mother: mama, words are powerless to thank you; without your love and support, this thesis might have probably never seen light.

Imene

Abstract

With the advent of the Internet and the widespread use of digital documents, access to information from the four corners of the globe has become easier and easier. This was accompanied by the copy-paste phenomenon that curtailed the appropriation of others' work (i.e., plagiarism) to a few clicks.

Since the '70s of the last century, researchers have begun developing software to automatically detect textual plagiarism. Still, as the techniques of these programs evolve, the plagiarists develop their tactics to escape them. Therefore, the plagiarism detection tools that have the potential to resist are the ones that are able to fight against this misconduct in different ways. Moreover, in the wake of globalisation, these tools should be able also to handle documents in multiple languages. Thus, given the perpetuation of this problem, the acquisition of the latest plagiarism detection technologies has become like an arms race for a never-ending battle.

This thesis deals with two major topics: *plagiarism detection in Arabic documents*, and plagiarism detection based on the writing style changes in the suspicious document, which is called *intrinsic plagiarism detection*. This approach is an alternative to the text-matching approach, notably, in the absence of the plagiarism source. Our key contributions in these two areas lie **first**, in the development of Arabic corpora to allow for the evaluation of plagiarism detection software on this language and, **second**, in the development of a language-independent intrinsic plagiarism detection method that exploits the character n-grams in a machine learning approach while avoiding the curse of dimensionality. Representing texts with character n-grams is one of the most successful text modelling approaches to some stylistic analysis applications. However, studies on the best character n-grams in the context of intrinsic plagiarism detection are almost non-existent. Hence, our **third** key contribution is an attempt to narrow this gap by investigating which character n-grams, in terms of their frequency and length, are the best to detect plagiarism intrinsically. We carried out our experiments on standardised English corpora and also on the developed Arabic corpora using the method we developed and one of the most prominent intrinsic plagiarism detection methods. The findings of our analysis can be exploited by the future intrinsic plagiarism detection methods that use character n-grams.

In addition to the above-mentioned technical contributions, we provide the reader with comprehensive and critical surveys of the literature of Arabic plagiarism detection and intrinsic plagiarism detection, which were lacking in both topics.

Keywords Intrinsic plagiarism detection . Arabic plagiarism detection . Character n-grams . Stylistic analysis . Evaluation corpora

مُلخص

مع ظهور الانترنت وانتشار استخدام المستندات الرقمية، أصبح الوصول إلى المعلومات من كافة بقاع الأرض أسهل فأسهل. ترافق ذلك مع ظاهرة النسخ واللصق التي اختزلت سرقة أعمال الآخرين (الانتحال) إلى بضع نقرات.

منذ سبعينات القرن الماضي، بدأ الباحثون في تطوير برامج الكشف الآلي لانتحال النصوص. ولكن كلما تطورت هذه البرامج طور معها المنتحلون أساليبهم للإفلات منها. مع ذلك كلما كانت الأدوات المستخدمة قادرة على التعامل مع وثائق بعدة لغات وبطرق كشف متعددة كلما زادت فرصة اكتشاف الانتحال. وبالتالي فإن اقتناء أحدث التكنولوجيات لكشف الانتحال أصبح أشبه بسباق تسلح لمعركة يبدو أنها دائمة بدوام البشرية.

هذه المذكرة تهتم بمبحثين رئيسيين ألا وهما كشف الانتحال في النصوص العربية، وكشف الانتحال بناءً على عدم استقرار أسلوب التعبير في النص المشبوه وهو ما يسمى بالكشف الجوهري عن الانتحال، والذي قد يكون بديلا عن أسلوب مقارنة النصوص في حال عدم توفر مصدر الانتحال. مساهمتنا الأساسية في هذين المجالين تكمن **أولا**، في تطوير ذخائر نصية لتقييم برامج كشف الانتحال في اللغة العربية و**ثانيا**، في تطوير أسلوب كشف جوهري يعتمد على نمذجة النص بطريقة مختزلة بواسطة ن-غرام الحروف في إطار التعلم الآلي. كما يمكن تطبيق هذا الأسلوب على أي لغة. ن-غرام الحروف هي من انجح طرق نمذجة النصوص لغرض تحليل الأسلوب. إلا ان دراستها بشكل واف في إطار الكشف الجوهري عن الانتحال تكاد تنعدم. وبالتالي، فإن مساهمتنا الرئيسية **الثالثة** هي محاولة لتقليص هذه الفجوة من خلال التحقيق في أي حروف ن-غرام –من حيث تواترها وطولها– هي الأفضل لكشف الانتحال بشكل جوهري. اعتمدنا في تجاربنا على الأسلوب الذي طورناه للكشف الجوهري للانتحال وأحد أبرز الأساليب الموجودة في هذا المجال. طبّقنا هذين الأسلوبين على ذخائر مرجعية تحتوي على نصوص باللغة الإنجليزية وكذلك على الذخائر التي صممناها بأنفسنا والتي تحتوي على نصوص باللغة العربية. يمكن استغلال نتائج دراستنا في الطرق المستقبلية للكشف عن الانتحال الجوهري التي تستخدم حروف ن-غرام.

إضافة إلى هذه الإسهامات التقنية، ساهمنا في ترتيب ونقد أدبيات الكشف عن الانتحال في النصوص العربية والكشف الجوهري عن الانتحال، الأمر الذي كان مفتقدا في كلا المجالين.

الكلمات المفتاحية الكشف الجوهري عن الانتحال . الكشف عن الانتحال في النصوص العربية . حروف ن-غرام . تحليل الأسلوب . ذخائر التقييم

Résumé

Avec l'avènement d'Internet et l'utilisation généralisée des documents numériques, l'accès à l'information des quatre coins du monde est devenu de plus en plus facile. Cela s'est accompagné avec le phénomène de copier-coller qui a limité l'appropriation du travail des autres (plagiat) à quelques clics de souris.

Depuis les années 70 du siècle dernier, les chercheurs ont commencé à développer les logiciels permettant de détecter automatiquement le plagiat textuel. Cependant, à mesure que les techniques de ces programmes évoluent, les plagiaires développent des tactiques pour leur échapper. Par conséquent, les outils de détection du plagiat qui ont le potentiel de résister sont ceux qui sont capables de lutter contre cette tricherie de différentes manières. De plus, à la suite de la mondialisation, ces outils devraient également pouvoir gérer des documents dans plusieurs langues. Ainsi, compte tenu de la perpétuation de ce problème, l'acquisition des dernières technologies de détection du plagiat est devenue une course aux armements pour une bataille sans fin.

Cette thèse traite deux sujets principaux : la *détection de plagiat dans les documents arabes* et la détection de plagiat basée sur les changements de style de rédaction dans le document suspect, appelée *détection de plagiat intrinsèque*. Cette approche est une alternative à l'approche par appariement de texte, notamment en l'absence de la source du plagiat. Nos principales contributions dans ces deux domaines concernent, **premièrement**, le développement de corpus arabes permettant l'évaluation des logiciels de détection de plagiat sur cette langue, et, **deuxièmement**, la mise au point d'une méthode de détection de plagiat intrinsèque qui est indépendante de la langue. Cette méthode exploite les n-grammes de caractères dans une approche d'apprentissage automatique tout en évitant la dimensionnalité. Représenter des textes avec des n-grammes de caractères est l'une des approches de modélisation de texte les plus réussies pour certaines applications d'analyse stylistique. Cependant, les études sur les meilleurs n-grammes de caractères dans le contexte de la détection intrinsèque du plagiat sont presque inexistantes. Par conséquent, notre **troisième** contribution clé est une tentative de réduire cet écart en recherchant les meilleurs n-grammes de caractères, en termes de fréquence et de longueur, pour détecter le plagiat de manière intrinsèque. Nous avons effectué nos expériences sur des corpus anglais normalisés ainsi que sur les corpus arabes que nous avons développés. Notre travail expérimental est basé aussi bien sur la méthode que nous avons développée que sur l'une des méthodes de détection de plagiat intrinsèque les plus importantes. Les résultats de notre analyse pourraient être exploités par les futures méthodes de détection de plagiat intrinsèque qui utiliseront les n-grammes de caractères.

Outre les contributions techniques susmentionnées, nous fournissons au lecteur des études exhaustives et critiques de la littérature relative à la détection de plagiat dans le texte Arabe, et à la détection de plagiat intrinsèque, qui faisaient défaut dans les deux domaines.

Mots clés Détection de plagiat intrinsèque . Détection de plagiat dans le texte Arabe . N-grammes de caractères . Analyse stylistique . Corpus d'évaluation

Contents

List of Figures

List of Tables

Chapter I. Introduction

> *One of the prominent aims of education is to bring up honest people. Academic integrity is one of the fundamental values of being honest.*
>
> (Baysen et al. 2017)

1 Motivation

Academic integrity as defined by the international centre for academic integrity[1] is a commitment to a set of moral values, "without them, everything that we do in our capacities as teachers, learners, and researchers loses value and becomes suspect" (Fishman 2013). Although the concretisation of this concept might be specific to each culture (Bretag 2016), there is no doubt that it is critical to the educational process and the proper accumulation of knowledge. Not only that, indeed, the importance of academic integrity goes beyond academia since an honest academic system is also fundamental and inevitable to build an honest society.

Breaching the academic integrity (also known as academic misconduct or academic dishonesty), which can be viewed also as a kind of corruption (Drinan 2016), is a worldwide concern in education and research. It can take several forms including *plagiarism*, which is the act of "copying another person's ideas, words or work and pretend that they are your own"[2]. Unlike other forms of academic dishonesty as the falsification and fabrication, plagiarism does not enter errors to the literature[3] (Vaux 2016), nonetheless, it is considered a serious problem because it is counter-productive and ineffective for the advance of knowledge. Moreover, it

[1] https://academicintegrity.org/

[2] Oxford Advanced Learner's Dictionary, 8th edition.

[3] Falsification and fabrication of data may lead to false scientific conclusions. But plagiarism is the act of copying existing information. Therefore, its main consequence is not to introduce erroneous conclusions to science, but to create redundancy and confusions on the origin of information.

deceives the reader, allocates undeserved credit to the plagiarist, and from a forensic viewpoint, it is an infringement of copyright[4].

Text plagiarism can occur not only in academia but also in other domains such as journalism and literary works. It can be intentional or unintentional (Maurer et al. 2006). The latter may occur for two reasons: (i) when the writer lacks knowledge on the writing rules and particularly on how to cite others' work, or (ii) when the writer suffers from cryptomnesia (Meuschke and Gipp 2013), which is a memory bias that gives a person a wrong impression that she/he is the origin of an idea.

In academia, plagiarism is widespread among students and even senior researchers and professors (Martin 2016). While statistics on its prevalence among professors are lacking, a large scale study by McCabe (2005) on 83 different campuses in the United States and Canada demonstrates that 57% of undergraduates and 68% of grad students have paraphrased or copied few sentences from Internet sources without footnoting them.

The Arab region is no exception or perhaps the situation is even worse. According to a study in a Lebanese university (McCabe et al. 2008), 80% of students admitted to a violation of academic honesty. There are also some small-scale studies (such as, (Elgendy 2014) and (Guimeur 2013)) and numerous journalistic reports that show alarming statistics about the prevalence of plagiarism among the Arab students[5]. As a citizen of an Arab country and I have been working in academia for several years, I can provide the reader with some facts. Indeed, students in my country are not taught research practices including writing skills such as citing references and paraphrasing techniques. This is reflected in the student's conception of research. As I have observed, students prepare research works by patchwriting, i.e. to copy-paste excerpts from many sources with some few editing to form eventually an essay on a certain subject and often without citing the references. That is, they plagiarise without being aware that it is a dishonest act. Another fact is that students in intermediate and high schools are used to buy, from cyberspaces, ready-made research works on the common subjects of the curriculum without being penalised by their teachers.

I agree with the study presented in (Madkhali 2017) that one of the reasons that lead the students to commit plagiarism in the Arab world is the *rote learning* method[6] followed in all levels of the educational system starting from the primary school to university. This method

[4] The infringement of copyright occurs especially when the plagiarised text is considerable. Otherwise, under the law of some (US- influenced) countries, this may be considered fair use.

[5] See for example the series of journalistic investigations published on The New Arab magazine (The Arabic version: www.alaraby.co.uk) on the problem of plagiarism in several Arab countries such as Algeria, Tunisia, Egypt, Qatar, and Sudan.

[6] The rote learning is "The process of learning something by repeating it until you remember it rather than by understanding the meaning of it." (Oxford Advanced Learner's Dictionary, 8th edition)

promotes and rewards memorisation and does not value creativity and critical thinking[7]. For example, in Algeria, pupils at exams are expected to repeat verbatim what the instructor taught them in order to obtain a high mark. In contrast, in other parts of the world such as the United States, students should provide answers using their own words and according to their standpoints on the topic (Madkhali 2017).

The lack of awareness on the definition of plagiarism in the Arab world nowadays might give the impression that this concept is new in the region. However, this phenomenon was rather a serious concern of the ancient Arab scholars. Interestingly, studying text reuse was an active field in Arabic literary criticism in the Middle Ages, especially in the context of poetry and literary works (Al Manasrah 2009; Tabana 1956; von Grunebaum 1944). For example, Al-Farazdaq (c. 641-c. 730) and Al-Mutanabbi (c. 915-c. 965), who are well-known Arab poets, were accused by plagiarising some of their poems.

As is evident, plagiarism is a perpetual and a universal problem, and it seems that fighting against it is an endless battle. In addition, in the wake of the Internet, and the proliferation of electronic documents, access to knowledge became easier, which further facilitated plagiarism.

Technologies used as countermeasures against plagiarism have been a subject of research since the ’70s of the past century. Still, most of these technologies are based on text-matching techniques, which show their limitations when the sources of plagiarism are not available for a reason or another[8]. On the other hand, the majority of the developed tools concern documents written in English, thus depriving a large class of users, who write in other languages, of taking advantage of this technology.

The contributions of this thesis concern the so-called *intrinsic plagiarism detection*, which is an approach that spots plagiarism on the basis of the inconsistencies of the writing style in the suspicious document. Unlike, the text comparison approach (which is called *external plagiarism detection*), *intrinsic plagiarism detection* analyses the suspicious document solely and does not require the availability of the potential sources of plagiarism. In addition, we are concerned about fostering research on the *detection of plagiarism in Arabic documents*, as much of the existing works are dispersed and difficult to compare.

2 Research Objectives and Contributions

The objectives of this thesis concern generally the domain of plagiarism detection but specifically they revolve around two sub-areas of this main domain, which are:

[7] Statistics from the Global Competitiveness Report (2018) of World Economic Forum show that Algeria and Egypt (which are amongst the largest Arab countries) are ranked, respectively, 105 and 123 among 140 countries, in the promotion of critical thinking in education.

[8] For example, if the source is not digitised or not indexed by the plagiarism detection system.

(1) *Plagiarism detection in Arabic documents,* and

(2) *Intrinsic plagiarism detection.*

A major issue in *Arabic plagiarism detection* is the disparate literature rendering difficult to draw from it any conclusion on the current state of research in this sub-area. Therefore, our objective concerning this context is to gain insights into the existing works and to discern the aspects that need attention from researchers. Eventually, our contributions in this regard are:

(1.a) Building benchmark corpora for the evaluation of the external and the intrinsic plagiarism detection approaches on Arabic documents to ensure the comparability of the methods, which is an attempt to narrow the gap of the disparate literature. These corpora have been released through the organisation of a shared task, which is an effort to increase the interest in fighting against plagiarism on Arabic documents using dedicated software.

(1.b) Providing the reader with a critical review of the existing Arabic plagiarism detection works including an appraisal of the quality of the publications.

Intrinsic plagiarism detection, which is the second sub-area of our research, is a very challenging approach, and it is still unsolved. Despite this fact, painstaking attempts to understand and study the performance of the stylistic features specifically to detect plagiarism are very few, as most of the works relied on what is already known about these features in other related domain such as authorship attribution. Moreover, although this research problem is relatively not new (the first methods dated back to the beginning of the millennium), there is to date no literature survey on it, which may slow down research progress on this topic. With all that said, our objective is to increase our understanding of this problem and the building blocks of its solutions, independently of the related problems. In this connection, the present thesis has three contributions:

(2.a) A language-independent intrinsic plagiarism detection method is proposed. It is based on new features that allow for an intuitive description of potential plagiarism in terms of character n-grams. These features have the advantage of being a reduced representation of texts using character n-grams.

(2.b) The best frequency and length of character n-grams in the context of the proposed method and another seminal state-of-the-art method is investigated. Unlike other studies on character n-grams[9], this study provides insights into the use of these features for intrinsic plagiarism detection specifically. The findings of this study can be exploited by future intrinsic plagiarism detection methods.

(2.c) The first systematic survey of intrinsic plagiarism detection literature is provided to the research community.

[9] There exist numerous studies on the best ways of using characher n-grams in the context of other research areas such as authorship attribution, but they are lacking in the context of intrinsic plagiarism detection.

Figure I.1 summarises the above contributions.

3 Thesis Outline and Chapters Summary

This thesis contains six chapters. Apart from the present chapter, which exposes the motivations and objectives of this dissertation, the remaining chapters are described below.

Chapter II provides a survey on the current methods of detecting plagiarism in Arabic documents. The survey shows that almost all the methods are based on uncovering plagiarism by comparing the suspicious document to the potential sources of plagiarism (the external approach). This motivates us to conduct the first experiments on Arabic documents that attempt to detect plagiarism by spotting the writing style changes (the intrinsic approach). In the light of these experiments, that utilise a small ad-hoc corpus, we felt the necessity to build a larger evaluation corpus that allows for a better assessment of the task performance on Arabic documents.

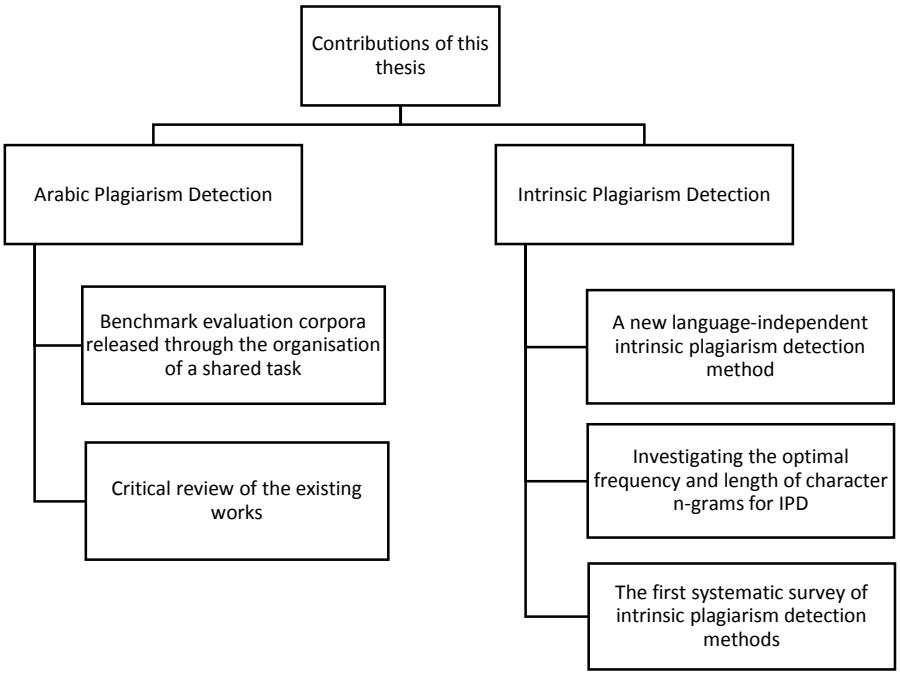


Figure I-1. The contributions of the thesis

Besides the technical aspect of Arabic plagiarism detection, this chapter discusses another important aspect, which is the quality of the publications in this research area.

Chapter III describes our contribution to try to narrow one of the gaps exposed in the previous chapter, which is the lack of standardised evaluation corpora for Arabic plagiarism detection. Through this chapter, the existing approaches of building plagiarism detection

evaluation corpora are presented, and then, we describe the corpora that we have built to evaluate both the external and the intrinsic plagiarism detection approaches in Arabic documents. These corpora have been made publicly available through the organisation of a shared task where researchers and practitioners have been invited to evaluate their methods. Then, the rest of Chapter III describes the methods of the participants and concludes with the lessons we learned from this experience.

Chapter IV situates intrinsic plagiarism detection with regard to some related tasks and surveys the existing plagiarism detection methods by organising them according to their building blocks, which are the pre-processing, segmentation, feature extraction, plagiarised fragments identification and the post-processing. This chapter is a contribution to advance the understanding of intrinsic plagiarism detection especially with the lack of survey papers on this task. The chapter concludes by pointing out the difficulty of this task and its need to further research.

Chapter V addresses the problem of intrinsic plagiarism detection on the basis of new character n-gram-based features (n-gram frequency class proportions). We show that the proposed features are able to discriminate between plagiarised and original text fragments with a performance that rivals the state of the art but with a small dimension of the text representation. Besides, this chapter studies the performance of character n-grams of different lengths and frequencies in the context of the proposed method and also Stamatatos' (2009a) method, which is a well-known state-of-the-art approach. We demonstrate experimentally that the low- and the high-frequency n-grams are not equally relevant for intrinsic plagiarism detection, and their performance depends on the way they are exploited.

Chapter VI concludes the thesis and exposes the current limitations of intrinsic plagiarism detection. We discuss the possibility to overcome those limitations if some constraints and assumptions imposed by the current view of the approach are alleviated. In light of that discussion, we provide our standpoint on the future of this task.

4 Published Work

Through my doctorate journey, I co-authored eight papers (6 as the first author). Many parts of this thesis are reused from some of them. Table I-1 (see the next page) displays the publications and the thesis chapters supported by each paper.

Table I-1. List of the published papers and the related chapters

Papers on my core contributions	
Bensalem, I., Rosso, P., Chikhi, S.: On the use of character n-grams as the only intrinsic evidence of plagiarism. Language Resources and Evaluation, 53(3). pp. 363–396. Springer (2019). doi:10.1007/s10579-019-09444-w **[Impact Factor (2018) : 1.029][Class A]**[10]	Chapter IV Chapter V
Bensalem, I., Boukhalfa, I., Rosso, P., Abouenour, L., Darwish, K., Chikhi, S.: Overview of the AraPlagDet PAN@FIRE2015 Shared Task on Arabic Plagiarism Detection. In: Majumder, P., Mitra, M., Agrawal, M., and Mitra, P. (eds.) Post Proceedings of the Workshops at the 7th Forum for Information Retrieval Evaluation (FIRE 2015), Gandhinagar, India. pp. 111–122. CEUR-WS.org (2015).	Chapter III
Bensalem, I., Rosso, P., Chikhi, S.: Intrinsic Plagiarism Detection using N-gram Classes. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, October 25-29. pp. 1459–1464. Association for Computational Linguistics (2014). **[Core A conference]**	Chapter V
Bensalem, I., Rosso, P., Chikhi, S.: A New Corpus for the Evaluation of Arabic Intrinsic Plagiarism Detection. In: Forner, P., Müller, H., Paredes, R., Rosso, P., and Stein, B. (eds.) CLEF 2013, LNCS, vol. 8138. pp. 53–58. Springer, (2013).	Chapter III
Bensalem, I., Rosso, P., Chikhi, S.: Building Arabic corpora from Wikisource. 2013 ACS International Conference on Computer Systems and Applications (AICCSA), Fes/Ifran, Morocco. pp. 1–2. IEEE (2013).	Chapter III
Bensalem, I., Rosso, P., Chikhi, S.: Intrinsic Plagiarism Detection in Arabic Text : Preliminary Experiments. In: Berlanga, R. and Rosso, P. (eds.) 2nd Spanish Conference on Information Retrieval (CERI 2012). pp. 325–329. , Valencia, Spain (2012).	Chapter II
Papers related to my work but outside the core contributions	
Franco-Salvador, M., Bensalem, I., Flores, E., Gupta, P., Rosso, P.: PAN 2015 Shared Task on Plagiarism Detection : Evaluation of Corpora for Text Alignment. Working Notes of CLEF 2015 - Conference and Labs of the Evaluation forum, Toulouse, France, September 8-11, 2015. CEUR-WS.org (2015).	
Meskaldji, K., Chikhi, S., Bensalem, I.: A New Multi Varied Arabic Corpus. 2018 3rd International Conference on Pattern Analysis and Intelligent Systems (PAIS). pp. 1–5. IEEE (2018).	

[10] Language Resources and Evaluation is a **class A** journal according to the 2019 criteria of journal classification of the ministry of higher education and scientific research of Algeria.

Chapter II. Arabic Plagiarism Detection: Critical Review

> *"I do not round verses and steal them*
> *I do not need to do it. The worse people are those who steal*
>
> Ṭarafah ibn al-'Abd (543-569)[1]

1 Introduction

Automatic plagiarism detection (APD) is a classical problem, which has been tackled with a range of researches in the last decades. Nowadays, plagiarism becomes easier due to the widespread use of electronic documents and the easy access to knowledge through the Internet. Consequently, its detection becomes more challenging because of the huge number of documents that can be used as a source of plagiarism.

This chapter starts by outlining the two main approaches to detecting plagiarism automatically (Section 2) to pave the road to our core subject, which is detecting plagiarism in the Arabic documents. Our contribution thereof is two-fold:

(i) providing the reader with a review of the existing works that encompasses three aspects: the quality of publications, the techniques, and the evaluation datasets (Section 3), and

(ii) describing the first experiments on detecting plagiarism in Arabic documents based

[1] Ṭarafah ibn al-'Abd (https://en.wikipedia.org/wiki/Tarafa) is a pre-Islamic poet. The verses quoted above are often cited by scholars to show that plagiarism was a concern of the old Arab poets. The translation of the verses to English is brought from (Madkhali 2017). The original verses in Arabic are:

ولا أغيرُ على الأشعار أسرقُها عنها غنيتُ، وشرُّ الناس مَنْ سرقا

on the changes of the writing style as this approach is underexplored in the Arabic language (Section 4).

2 Plagiarism Detection Approaches

When reading a text, one of the clues to plagiarism is to have a strong sense of "déjà vu" or to remember outright the source of the read material. Another traditional hint of plagiarism is what we like to call the "writing style salad", which means the existence of different writing styles in the same document. For example, it happens to find in the same document a sloppily written paragraph with some impressive sentences, or a paragraph that shows a high-level language with another that is full of language errors. This kind of observations triggers the suspicion that the text is, unexpectedly, multi-authored. The main approaches to the automatic plagiarism detection are primarily inspired by these traditional indications. The *external* approach is based on discovering the similarities with other texts and the *intrinsic* approach is based on detecting the writing style's inconsistencies within the document[2].

As per (Clough 2003), the assumption behind the *external plagiarism detection* is that it is unlikely that two texts, written independently, share exactly the same expressions even if the texts are about the same topic. This assumption relies upon the intuition that people express themselves differently.

The intuition behind *intrinsic plagiarism detection* (IPD) pertains to *stylometry* that assumes each author has a distinctive and quantifiable writing style (Holmes 1998). Clough (2003) was among the first to address IPD[3]. He argued that this form of detecting plagiarism deserves more attention since according to a survey on 321 academics (Bull et al. 2001), it turns out that inconsistencies in the writing style are the most common trigger of suspicion on the originality of the students' work.

Note that the changes of the writing style within a document are not always associated with plagiarism; they can be due to expected reasons such as quotations and multi-authorship. Similarly, reusing texts can be legitimate such as proper citations and common phrases reuse. Automatic plagiarism detectors are expected to distinguish between plagiarism and those fair cases of style change or text reuse. However, in reality, the software may fail to make this

[2] The oldest reference we were able to find distinguishing between these two approaches and associating to them different terms is (Kimler 2003, p. 7). It seems that, the term *external plagiarism detection* was used for the first time in this master thesis to represent the text matching approach. However, the approach based on stylometry is called in this reference ***internal*** *plagiarism detection.* The term ***intrinsic*** *plagiarism detection* appears for the first time in (Meyer zu Eißen and Stein 2006). The distinction between these two approaches and the terminology associated to them became later more known after the first plagiarism detection shared task (Potthast et al. 2009). All that said, it should be noted that Clough (2000) was among the first researchers who suggested the consideration of the writing style inconsistencies as a mean of detecting plagiarism automatically.

[3] Clough (2003) performed a small experiment of detecting plagiarism on the basis of the style changes within a single document. On the other hand, as mentioned previously, the term *intrinsic plagiarism detection* was coined by Meyer zu Eißen and Stein (2006).

distinction. Therefore, humans have to examine the output of plagiarism detection software to make a final judgment on them. This examination is necessary for the output of the external approach and, a fortiori, the intrinsic approach. Since the latter does not provide the source of plagiarism, the human expert may need to make further investigation (e.g., checking whether the document is multi-authored or trying to find the source of the suspicious passage) to confirm or reject the cases detected by this approach.

The next two sections define the two approaches formally and outline briefly their techniques and related domains.

2.1 External Plagiarism Detection

2.1.1 Definition and Techniques

Given a document *d* and a potential source of plagiarism *D*, detecting plagiarism by the *external approach* consists in identifying pairs of passages (*s*, *s'*) from *d* and *d'* (*d'* ∈ *D*) respectively, such that *s* and *s'* are highly similar. This similarity could have many levels: *s* is an exact copy of *s'*, *s* was obtained by obfuscating *s'* (e.g., restructuring, rewording, etc.) or *s* is semantically similar to *s'* but uses different words (a.k.a. plagiarism of ideas) or even a different language (a.k.a. cross-language plagiarism) (Barrón-Cedeño 2012; Potthast et al. 2010b).

This problem has been tackled by many researchers in the last decade using a plethora of techniques related to information retrieval (IR), near-duplicate detection (Broder 2000), and textual similarity detection (Gomaa and Fahmy 2013).

Clough (2003) stated that plagiarism detection is deeper than information retrieval since it seeks the similarity of content rather than topic. The discriminators in IR are, for example, proper nouns and names. However, plagiarism detection is concerned with the similarity of expressions.

While detecting verbatim plagiarism is no longer a challenge (Potthast et al. 2011), detectors are still struggling to detect obfuscated plagiarism. The last trend that has been observed in the last PAN plagiarism detection competitions[4] is to develop methods that detect a specific kind of plagiarism obfuscation (Potthast et al. 2014b). For instance, Sanchez-Perez et al.'s method (2014) is oriented to detect plagiarism cases that summarise the source passage. Besides, the detection of semantic textual similarity (whose plagiarism of ideas detection is one of its applications) is still an active research area (Cer et al. 2017).

[4] PAN (https://pan.webis.de) is a lab that organises a series of shared tasks on text forensics; plagiarism detection is among the addressed problems. More details on it are provided in the next chapter (pp. 31-32).

2.1.2 Generic Process

A complete process of external plagiarism detection involves mainly two phases: *source retrieval* and *text alignment* (Potthast et al. 2013a). For the given suspicious document *d*, the *source retrieval* phase consists of selecting, from the available set of source documents *D*, a subset *D'* of documents that are the most likely sources of plagiarism. Note that *D* is extremely large in comparison with *D'*. The former can be, e.g., the Web or a large local document collection, and the latter should be only a handful of documents. With regard to *text alignment*, it is the process of extensively comparing *d* with each document in *D'* in order to uncover similar passages.

Figure II-1 depicts the building blocks of these two phases. PAN competition series on plagiarism detection has contributed significantly to the definition of these phases and to coining standard terminology[5]. Therefore, we refer the reader to PAN overview papers

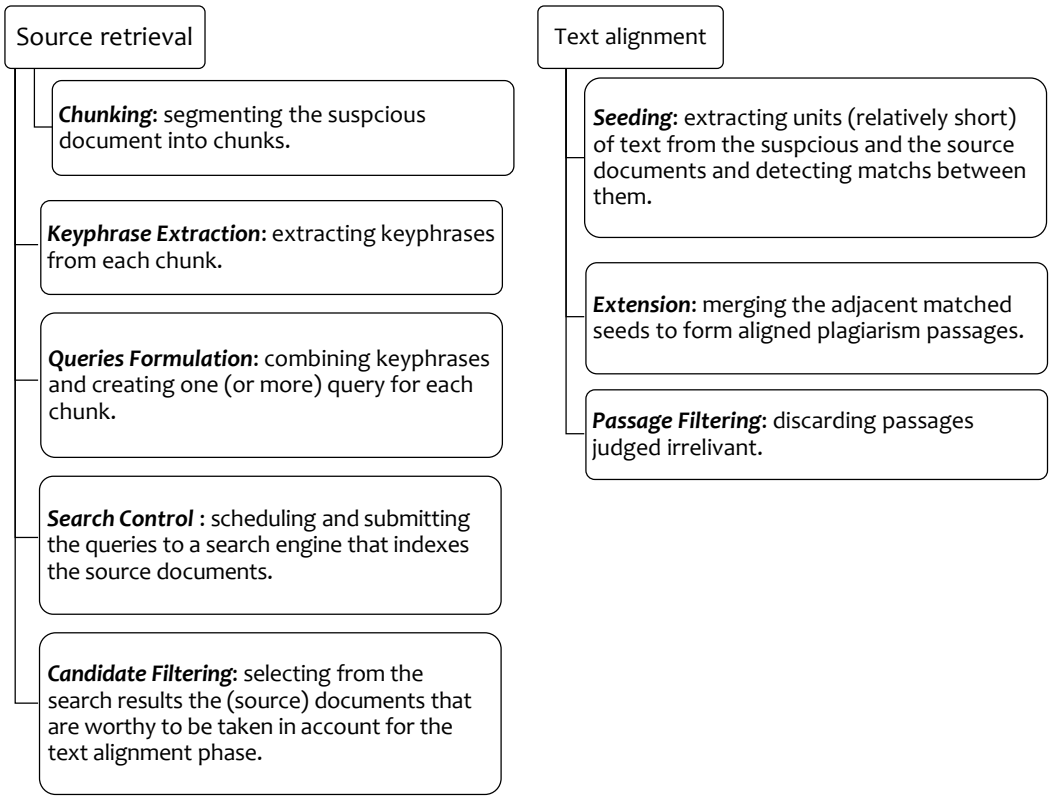


Figure II-1. External plagiarism detection methods building blocks

[5] The source retrieval phase is also called heuristic retrieval and candidate retrieval. The text alignment phase has been called also detailed analysis and detailed comparison.

(Potthast et al. 2009, 2010a, 2011, 2012, 2013a, 2014b), where the above phases with their building blocks are explained in detail. In this chapter (and also in the next chapter), we adopt PAN's terminology to describe the methods developed for Arabic.

2.2 Intrinsic Plagiarism Detection

2.2.1 Definition and Techniques

Given a document *d*, detecting plagiarism by the *intrinsic approach* consists in identifying in *d* the set of passages *S*, such that each $s \in S$ is different from the rest of the document with respect to the writing style. Then, this approach relies on the textual features that are able to distinguish the writing styles of different authors in one document.

Intrinsic plagiarism detection is strongly related to authorship attribution (Stamatatos 2009b), paragraph authorship clustering (Brooke and Hirst 2012) and detection of inconsistencies in multi-author documents (Akiva and Koppel 2012). The used techniques are related to feature extraction and classification. For instance, Stamatatos (2009a) used character n-grams as features and a distance function for classification. Stein and Meyer zu Eißen (2007) used a vector space model of lexical and syntactic features and supervised classification.

The intrinsic approach is an essential alternative in situations where the plagiarism source is not available (e.g., an old book) or a part of the text has been directly written by another author without copying from any source (e.g., the case of students who asks someone else to write for them parts of their essay or thesis)[6].

The automatic analysis of the writing style, which is the heart of intrinsic plagiarism detection, is an important component of many other natural language processing applications. For some of them, the input document is considered as a whole when analysing its style, which is the case of the authorship identification (Stamatatos 2009b) and the authorship verification (Koppel and Seidman 2013). For other applications, the document is perceived as a set of fragments, for each of them the writing style needs to be analysed individually. Examples of such applications include paragraph authorship clustering (Brooke and Hirst 2012), authorial segmentation of multi-author documents (Akiva and Koppel 2013), detection of stylistic inconsistencies in collaborative writing (Glover and Hirst 1996; Graham et al. 2005) and plagiarism direction identification (Grozea and Popescu 2010).

For intrinsic plagiarism detection, it is crucial to analyse the writing style at the fragment level and sometimes also at the document level, which is one of the difficulties of this task. In fact, this task is still unsolved and evidence of that is the low performance of the existing methods in comparison with the external plagiarism detection methods.

[6] *Ghostwriting* is the precise term referring to the misconduct of asking a third party to write our work. There are researchers (such as Martin (2016) and Foltýnek et al. (2019)) who consider it as a form of plagiarism.

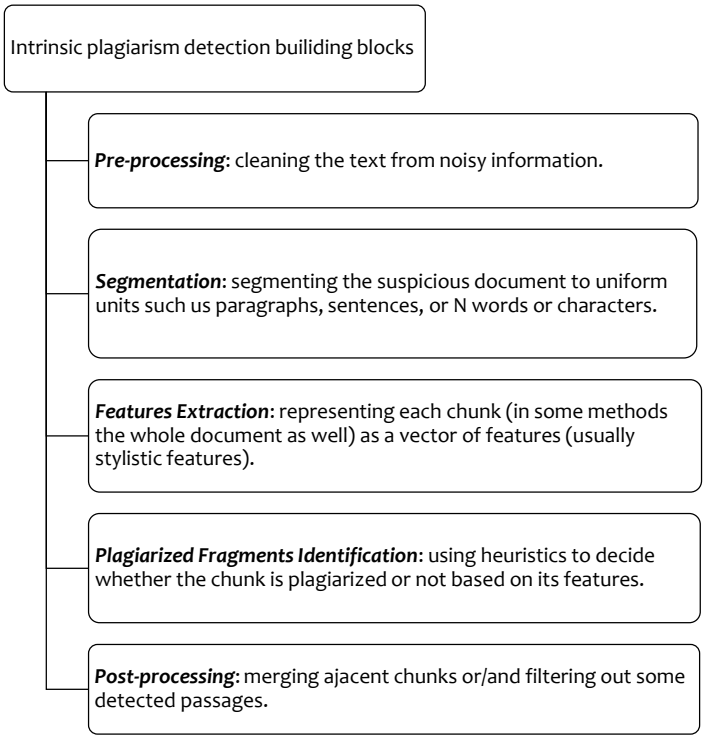


Figure II-2. Intrinsic plagiarism detection methods building block

2.2.2 Generic Process

Most of the existing intrinsic plagiarism detection methods entail five main building blocks, which are depicted in Figure II-2. These are inspired by the authorship verification approach (Stein et al. 2011) and have not been changed since the past decade. More details on this approach and its building blocks are provided in Chapter IV.

3 Survey of Arabic Plagiarism Detection Works

This section provides the reader with an overview of the state of research on Arabic plagiarism detection. Our main questions are:

- What are the approaches to plagiarism detection applied to Arabic documents?
- Which techniques have been used to adapt the approaches to Arabic?
- What are the addressed kinds of plagiarism?

To answer these questions we followed the steps below:

1. Publication gathering and filtering
2. Quality appraisal and unreliable publication exclusion
3. Survey of the existing methods and corpora

The following sections detail the aforementioned steps.

3.1 Publication Gathering and Filtering

We searched publications on Arabic plagiarism detection through the search engine Google Scholar[7] by using the query Arabic + "plagiarism detection". The total number of publications that we eventually gathered after discarding irrelevant papers (i.e., papers that contain the keywords but not related to the subject) are 59 publications, published from 2008 to 2019[8].

Although research on plagiarism detection in general dated back to the '70s (see, e.g., (Ottenstein 1976)), it seems there is no work on this topic that addressed the Arabic language before 2008. Therefore, it is highly probable that the documents we collected represent almost all the available publications on Arabic plagiarism detection until the time of writing this chapter. Figure II-3 depicts the distribution of the number of published works on Arabic plagiarism detection over the years. It is clear from the figure that the number of publications on this topic has increased in the last 5 years[9].

As shown in Figure II-4, almost all the collected publications are papers that describe methods, tools or experiments on plagiarism detection. These have been published in

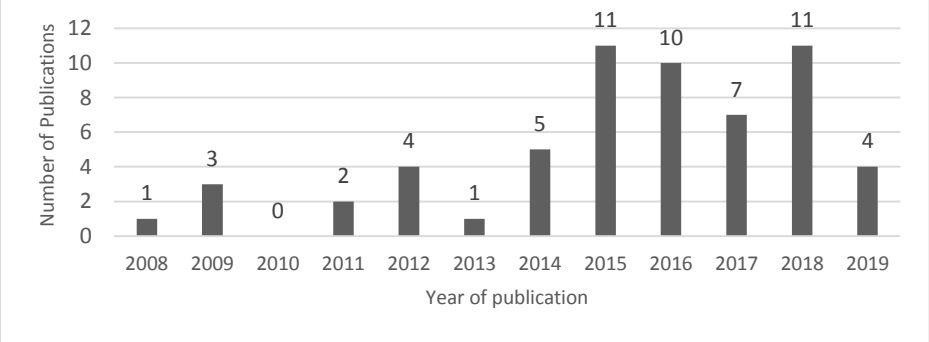


Figure II-3. Arabic plagiarism detection papers published from 2008 to June 2019

[7] https://scholar.google.com/

[8] Note that the number of papers published in 2019 are counted until June (the time of writing this chapter). Another point to mention is that our papers on Arabic plagiarism detection have not been included in this investigation since they are contributions to bridge some of the gaps that we detected in the existing work, and we will describe them in details as part of this thesis. By adding our papers, the number of publications dealing with plagiarism detection on Arabic documents becomes 65.

[9] Some of the works, published after 2015, make use of our proposed corpora (see the next chapter for details).

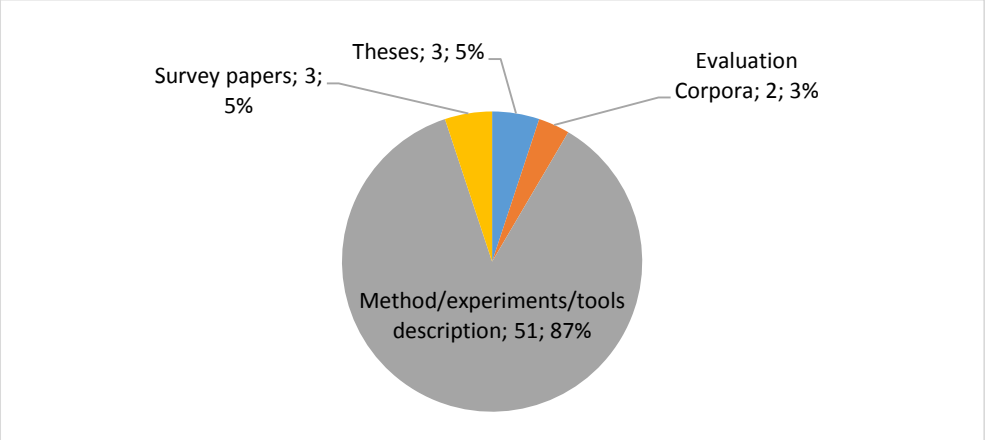


Figure II-4. Types of Arabic plagiarism detection publications

conferences, workshops, or academic journals.

Since, in this chapter, we are interested in gaining insights into the existing methods, we discarded the papers that solely describe the existing work. We discarded also the theses after noticing that their authors have papers in our collection. Consequently, 53 is the number of the remaining papers that we will consider in the next step, which is the quality appraisal.

3.2 Quality Appraisal

3.2.1 Methodology

Poor-quality studies may slow down and hinder the progress of research. They lead, in addition, to providing practitioners looking to benefit from academia with unreliable information (Ivarsson and Gorschek 2011). Thus, before going through surveying the existing approaches to detecting plagiarism on Arabic documents, we start by appraising the quality of the collected papers. This step has two benefits. On the one hand, it will help us to identify the unreliable papers and consequently discarding them as they may lead to misconceptions about the actual state of research in the field. On the other hand, evaluating the quality of papers is itself an aspect that provides a further understanding of the current state of research in the field.

Simplistically, the quality of research can be associated with the number of citations, or the venue of publication. However, these pieces of bibliographic information do not necessarily reflect the relevance and the quality of the paper as there are many other factors that influence how many times the paper is cited and why the venue of publication is chosen, especially if the publication stemmed from regions where academic research is not an established domain: the case of many Arab countries.

Instead of a simplistic approach of including/excluding papers from our literature review, we applied the approach proposed in (Menzies and Shepperd 2019), which consists in assessing the quality of papers in terms of twelve criteria. The authors called these criteria "bad smells" and defined them as the surface issues that might be detected in research publications and that can be indications of serious problems. The scope of Menzies and Shepperd's investigation is software analytics. Still, the authors noted that while some of the proposed "bad smells" are specific to software analytics, others are general and then applicable to other scientific domains. Hence, we selected four "bad smells" (from twelve) that we judged appropriate for plagiarism detection research[11]. Table II-1 lists them in the first column. In the second column, we determine exactly how these "bad smells" emerged in the examined Arabic plagiarism detection papers.

Table II-1. Bad smells that we detected in Arabic plagiarism detection papers

Bad smells	Signs
Not interesting	• Conceptual description of techniques/methods/data without experiments or novel ideas.
Not using related work	• Unawareness of related work or the state of the art in the evaluation strategies of plagiarism detection methods.
Using deprecated or suspect data	• Using homemade evaluation datasets without sufficient description of how plagiarism has been created and/or annotated to build the ground truth. • Using evaluation datasets that are introduced in other papers but are not publically available without specifying how they have been obtained. • Using datasets that are not oriented to evaluate plagiarism detection without explaining how they have been adapted to the task.
Inadequate reporting	• Partial reporting of results e.g., only giving the precision without the recall. • Incomplete description of the evaluation measures, e.g., not providing their formulas or/and not specifying whether they are computed at the character, fragment or document level[10]. • Partial description of the proposed method. • Comparison with the results of other tools or methods that are not publically available without specifying how the computation of results has been done.

[10] See Chapter III (Section 5.3.1) for details on the standardised evaluation measures of plagiarism detection.

[11] Most of the "bad smells" that we did not consider concern the statistical significance, which is usually not utilised in plagiarism detection studies. Note that this does not mean that this technique is not applicable for plagiarism detection evaluation but rather its use is uncommon even in the best studies. This fact might be attributed to the lack of practical guidelines on hypothesis testing that may accompany the current plagiarism detection evaluation measures.

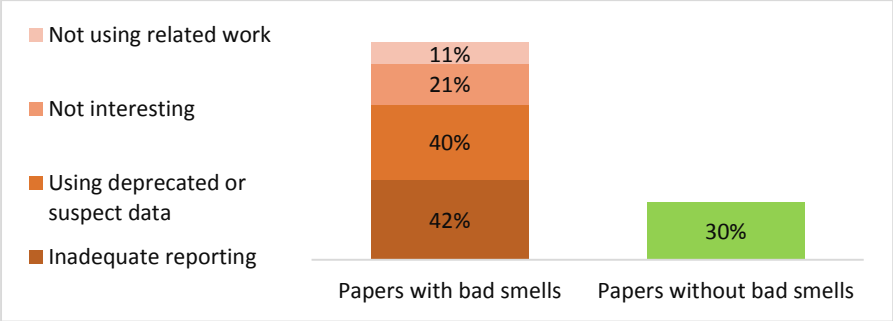


Figure II-5. Proportion of Arabic plagiarism papers with and without "bad smells"

Table II-2. Overview of the number of papers considered in our study

Total number of the collected publications	59
Number of papers after discarding theses and surveys	53
Number of papers retained after the quality appraisal	24

3.2.2 Results

Our analysis reveals that only 30% (16 papers among 53) of the collected papers are free of "bad smells". As depicted in Figure II-5, the *inadequate reporting* and the *use of deprecated or suspect data* are the most prevalent issues, as they are present in over the third of the examined papers. Note that some papers have more than one issue, which explains why the sum of the proportions in Figure II-5 is higher than 100%.

As pointed out previously one of our gaols of undertaking a quality appraisal is to exploit it as a decision-making tool on whether or not to include a paper to our survey of Arabic plagiarism detection methods. In this connection, we considered as reliable the papers that are free of "bad smells". Nevertheless, we also kept eight additional papers with the "bad smells" *inadequate reporting* and/or *deprecated or suspect data*. The "bad smells" in these papers are caused by some missing information on the proposed method or the evaluation corpus. However, the quantity of the missing information is small, and we felt that the quality of the work remains acceptable even with the presence of some issues. As a result, the final number of the papers we kept for the subsequent analysis is 24 out of 53 papers. Table II-2 recapitulates the number of the manipulated papers in each step.

Table II-3. Scope of the examined Arabic plagiarism detection papers

Scope	Papers
External plagiarism detection	(Alzahrani and Salim 2008), (Alzahrani and Salim 2009), (Jaoua et al. 2011), (Menai 2012; Menai and Bagais 2011), (Jadalla and Elnagar 2012a, 2012b, 2012c), (Soori et al. 2014), (Alzahrani 2015), (Hussein 2015, 2016), (Khan et al. 2015), (Magooda et al. 2015), (Lulu et al. 2016), (Ghanem et al. 2018), (Boukhalfa et al. 2018), (Nagoudi et al. 2018a), (Nagoudi et al. 2018c), (Khan et al. 2019)
Intrinsic plagiarism detection	(Mahgoub et al. 2015)
Tools description	(Alzahrani et al. 2009), (Kahloula and Berri 2016)
Evaluation corpora	(Siddiqui et al. 2014)

3.3 Methods and Evaluation Corpora

In this section, we review the 24 selected works (from the previous step) in terms of their *scope*, *techniques*, *locality of source documents*, *language dependence* and *evaluation strategies*.

3.3.1 Scope

Table II-3 shows that almost all the works on Arabic plagiarism detection concern the external approach. Apart from our works (Bensalem et al. 2012, 2013a, 2014a, 2019)[12], which are not listed in the table, the paper (Mahgoub et al. 2015) is the only work that concerns the application of the intrinsic approach to the Arabic language.

Besides the method-oriented papers, two papers describe plagiarism detection tools that were released for use as finished products with a graphical user interface. Alzahrani et al. (2009) presented their tool along with a paradigm of using it in e-learning systems. Unfortunately, Alzahrani tool is no longer available online. With regard to Kahloula and Berri (2016), they described the architecture of their tool that has been later made available online as a commercial application (http://almikshaf.com/)[13].

[12] Note that our works in (Bensalem et al. 2012, 2014a, 2019) are language-independent but the evaluation is made on our proposed Arabic corpora (Bensalem et al. 2013a, 2013b, 2015) in addition to the English corpora. We defer the description of our experiments of the papers (Bensalem et al. 2012) and (Bensalem et al. 2014a, 2019) to Section 4 of the present chapter and Chapter V, respectively

[13] Last consultation: 21/06/2019

Concerning the evaluation corpora, Siddiqui et al. (2014) describe the creation of a corpus of external plagiarism detection, which consists of 348 suspicious document-source document pairs. Despite the potential importance of such a corpus of semi-realistic plagiarism[14], it has not been released online and until now, only their authors exploited it (Khan et al. 2019).

Since virtually all papers on Arabic plagiarism detection are about the external approach, the next sections are devoted to this approach. We defer the description of the single IPD method cited in Table II-3 to the next chapter (p. 54), as it is among the participating methods in the shared task we have organised.

3.3.2 Techniques

We observe in Table II-4 that *fingerprinting* is the most used technique in Arabic external plagiarism detection methods. This technique can be employed in text alignment as well as in source retrieval. Therefore, we explain it in more detail in the next paragraphs.

Fingerprinting is a technique of copy detection that was first used to detect the exact copy of an input data but later used by Manber (1994) to detect partial copies. It is also used in many plagiarism detection programs including commercial ones (Maurer et al. 2006). A *fingerprint,* in the context of language processing, is a sequence of numbers computed from the input text. Hence, detecting the similarity between two texts using fingerprinting revolves around comparing relatively short sequences of *numbers* instead of comparing the texts themselves, which renders computation less expensive.

Detecting plagiarism in a suspicious document d_{spc} given a source document d_{src} using the fingerprinting techniques involves the following mains stages:

1. Segmenting d_{spc} and d_{src} into fragments. Let S_{spc} and S_{src} denote the sets of fragments of d_{spc} and d_{src}, respectively.
2. Computing a fingerprint from each fragment of both documents.
3. Comparing the fingerprints of d_{spc} to those of d_{src} using a similarity measure.
4. A fragment $s_{spc} \in S_{spc}$ is considered plagiarised from a fragment $s_{src} \in S_{src}$ if the similarity between their fingerprints exceeds a predefined threshold.

Fingerprinting a text (see step 2 above) comprises, in turn, the three steps below.

Chunking the text into substrings. The substrings represent the smallest unit from which the fingerprint is computed. The choice of the chunk length is crucial since it affects the accuracy of the similarity detection. A short chunk may lead to false alerts of similarity even

[14] We call semi-realistic plagiarism the plagiarism that has been created by asking people to write essays and plagiarise deliberately. See the next chapter (Section 3) for further information on the approaches of creating plagiarism detection evaluation corpora.

if the texts share only some few words such as idioms. On the other hand, long chunks may lead to skipping partial overlaps i.e., when only some words or letter are different between the two texts they will not be detected as similar.

Hashing. It is the operation of using a *hash function* that maps each chunk to a numerical code termed a *hash*.

Selection. For efficiency reasons, only a subset of the generated hashes is chosen to represent the whole text fragment. The sequence of the selected hashes constitutes the fingerprint of the input text. This step requires the use of a selection strategy which could be simple such as selecting the hashes that are multiples of a fixed k (i.e., hash mod k =0) or more sophisticated such as the Winnowing algorithm (Schleimer et al. 2003), which basically selects the smallest hash from a sliding window.

Note that the fingerprinted text could be expanded by the use of linguistic resources to allow for the detection of copies where words have been substituted by their synonyms or translated from other languages, which is e.g., the case of Menai's (2012) method.

Apart from fingerprinting, Arabic plagiarism detection methods used other techniques. For example, the method of Alzahrani and Salim (2008) adopts a fuzzy-sets information retrieval model used previously by (Yerra and Ng 2005) for copy detection on Web documents. Alzahrani and Salim's method computes the similarity between each sentence in the suspicious document and those of the source document based on the correlation factor between their words. According to the authors, the correlation factor between a pair of words, $\langle w_i, w_j \rangle$, is estimated based on thesauri or the occurrence of each word in a corpus of documents. Its highest value (= 1) is reached if the two words are synonyms or they always appear together in the corpus, i.e., each document that contains w_i, it contains also w_j. The proposed method succeeded in detecting the exact copy and the restructured copy of sentences but not the copies where words are substituted by their synonyms or antonyms. The authors attributed that to the limited list of the words used in the word-to-word correlation matrix. Then, in their subsequent work (Alzahrani and Salim 2009), the authors manually enhanced the matrix with further pairs of synonyms and antonyms, which serves in detecting reworded sentences.

Another example of methods is the one of (Hussein 2015). This method exploits latent semantic analysis to model the documents, which are subsequently compared using the cosine similarity measure. The results of detecting obfuscated plagiarism (restructured and reworded) are promising. However, these findings cannot be generalised since the evaluation corpus contains only one document with this kind of plagiarism.

The rest of methods in Table II-4 are essentially based on matching words or phrases which are selected and represented using different techniques, such as: matching compressed text (Soori et al. 2014), matching only important words (Jaoua et al. 2011), matching sentences (Ghanem et al. 2018; Magooda et al. 2015), matching n-grams (Alzahrani 2015), or matching

Table II-4. Papers on Arabic plagiarism detection using the external approach

Reference	Candidate retrieval	Detailed comparison	Main used techniques	Source documents locality
(Alzahrani and Salim 2008)		■	- Fuzzy-set information retrieval	Local
(Alzahrani and Salim 2009)		■	- Fingerprinting vs. Fuzzy-set information retrieval	Local
(Menai 2012; Menai and Bagais 2011)	■[15]	■	- Fingerprinting - Documents tree representation - Longest common substring	Local
(Jaoua et al. 2011)	■	■	- Lexical chains enriched with synonyms	Web-enabled
(Jadalla and Elnagar 2012a, 2012b, 2012c)		■	- Fingerprinting (Winnowing)	Local
(Soori et al. 2014)		■	- Lempel-Ziv compression method	Local
(Alzahrani 2015)	■	■	- Fingerprinting - K-overlapping n-grams	Local
(Magooda et al. 2015)	■	■	- Word matching - Sentence matching - Skip-gram matching	Local
(Khan et al. 2015)	■		- Variance in the readability - key-phrases extraction	Web-enabled
(Hussein 2015, 2016)		■	- Latent semantic analysis	Local
(Lulu et al. 2016)	■		- Fingerprinting	Web-enabled
(Boukhalfa et al. 2018)		■	- Study on the effect of stemming on plagiarism detection	Local
(Nagoudi et al. 2018c)		■	- Fingerprinting - Word embedding	Local
(Nagoudi et al. 2018a)		■	- Word embedding - Support vector machine, decision trees, and random forests	Local
(Ghanem et al. 2018)		■	- Sentence matching - VSM with Tf-Idf weighting - Feature-based semantic similarity	Local
(Khan et al. 2019)	■	■	- key-phrases extraction - Jaccard similarity (between documents) - Cosine similarity (between sentences)	Web-enabled

[15] This method does not use exactly the principle of creating queries to a search engine to retrieve the candidate documents but it compares the suspicious and the source documents at three levels starting from the document level then the paragraph level and finally the sentence level. If no similarity is detected at the document level, the followings levels will not be considered. Thus, we consider the document-level comparison as the candidate retrieval module.

different levels of the document starting from the sentence level to the whole document (Menai 2012).

To detect semantic similarity, some methods expanded the compared texts with additional related words extracted from Arabic WordNet (Ghanem et al. 2018) or learned from a corpus (Nagoudi et al. 2018c).

3.3.3 Locality of the Source Documents

External plagiarism detection methods compare the suspicious document either to a local database of documents or to documents downloaded from the web. In the latter case, the method should be equipped with a (source) documents retrieval module that interfaces with an online search engine. As shown in Table II-4, some methods are web-enabled. Almost all of them used Google API for this purpose.

3.3.4 Language Dependence

By analysing the Arabic plagiarism detection methods, we observed that the dependence on Arabic emerges in two levels: processing level and resource level. The language-dependent processing used in many of the methods of Table II-4 is the stop-words removal[16] and stemming. Other methods required the use of Arabic resources such as Arabic WordNet (Ghanem et al. 2018; Jaoua et al. 2011; Menai 2012).

3.3.5 Evaluation Strategies

The aforementioned methods were evaluated using different corpora in terms of the type of plagiarism they contain (see Table II-5) and also in terms of their size that varies from 15 (Alzahrani and Salim 2009) to many thousands of suspicious documents (Jadalla and Elnagar 2012b). Moreover, these corpora have been built using different strategies, for example, Jadalla and Elnagar 2012 (2012b) tested their method on a collection of real suspicious documents, which consists of students' projects and Wikipedia articles. To build the ground-truth, the real plagiarism in these documents has been annotated automatically using an n-grams-based method that performs an exhaustive pairwise comparison of the documents. Other corpora are created by collecting a relatively small set of documents and generating from each document an exact copy and several near-duplicate copies by restructuring and rewording sentences and inserting new excerpts (Alzahrani and Salim 2009; Hussein 2016; Menai 2012). Khan et al. (2015, 2019) constructed a corpus by asking a group of students to write essays and borrow texts from the web while keeping the source. Although this approach

[16] Stop-word removal can be language-independent if it is based on computing the frequency of the words in the given document (stop-words in all the languages are very frequent). However, most of the methods use a predefined list of stop words, which is, of course, specific to each language.

is the closest to the real scenario of plagiarism generation, the obtained suspicious documents cannot be thought of as absolutely real since they were not produced naturally. Therefore, we call them semi-realistic suspicious documents.

It should be noted that until the end of 2015, there was no benchmark to allow for a fair comparison of Arabic plagiarism detection methods. To bridge this gap, we organised the shared task AraPlagDet[18] through which we made available evaluation corpora for training and testing Arabic plagiarism detection methods (the next chapter is devoted to AraPlagDet and the corpora we constructed for the task). Our corpora have been used to evaluate several methods during and after the competition. We listed them all in the last line of Table II-5.

Table II-5. Description of the corpora used to evaluate plagiarism detection methods on Arabic documents. The character '-' is used when no information is provided.

Methods \ Corpora	Suspicious documents			Type of plagiarism				
	Real	Semi-realistic	Artificial	Verbatim	Synonym substitution	Restructuring	Translation	Paraphrasing [17]
(Alzahrani and Salim 2008, 2009)			▪	▪	▪	▪		
(Jaoua et al. 2011)			▪	▪			▪	
(Jadalla and Elnagar 2012a, 2012b, 2012c)	▪					-		
(Menai 2012; Menai and Bagais 2011)			▪	▪	▪	▪		
(Soori et al. 2014)			▪	▪		▪		▪
(Hussein 2015, 2016)	▪		▪	▪	▪	▪		
(Khan et al. 2015), (Khan et al. 2019)		▪				-		
(Lulu et al. 2016)			▪			-		
(Alzahrani 2015), (Magooda et al. 2015), (Boukhalfa et al. 2018), (Nagoudi et al. 2018c), (Nagoudi et al. 2018a), (Ghanem et al. 2018)			▪	▪	▪	▪		▪

[17] Add or delete words but keep the meaning

[18] The organisation committee members are: The author of the present thesis, Paolo Rosso (PRHLT research centre, Universitat Politècnica de València), Kareem Darwish (Qatar Computing Research Institute), Salim Chikhi (MISC Lab, Constantine 2 University), and Imene Boukhalfa (MISC Lab,Constantine 2 University).

Another aspect of the evaluation is the performance measures. Indeed, the majority of methods have been evaluated using precision and/or recall computed using the number of detected and plagiarised sentences or excerpts. Lulu et al. (2016), who proposed a method of source retrieval, measured the performance of their method using only a document-level recall that is the proportion of the correctly retrieved source documents to the total number of the actual source documents.

The performance of methods that have been evaluated on our corpora (see again the last line of Table II-5) have been measured by character-level precision and recall. Potthast et al. (2010c) have introduced these measures specifically for plagiarism detection evaluation and publicised them at the PAN plagiarism detection shared tasks. Further details on these measures and PAN are provided in the next chapter.

3.4 Discussion

As stated before, some years ago, the comparison of Arabic plagiarism detection methods was elusive until we released our evaluation corpus via the organisation of AraPlagDet shared task. One limitation of this corpus is the absence of cross-language cases. It is worth to note that the cross-language plagiarism detection has generally lost its importance with the popularity of machine translation (Potthast et al. 2019), unless paraphrasing occurred on top of the translation (Franco-Salvador et al. 2016). On the other hand, the cross-language semantic textual similarity (STS) (which can be one of the building blocks of cross-language plagiarism detection (Ferrero et al. 2017)) is still a popular research subject[19], and the good news is the huge number of participants (more precisely, 45 participants) in the Arabic-English STS SemEval-2017 shared task (Cer et al. 2017), which shows the prospective importance of this track. Therefore, it remains interesting to evaluate cross-language plagiarism detection on a benchmark dataset where Arabic is the target language and English is the source since serious attempts to address this task are missing in the Arabic plagiarism detection literature[20].

In addition to the technical perspective, we shed light in our review on the quality of Arabic plagiarism detection papers. We showed that despite the growing number of Arabic plagiarism detection publications in the last years, sadly, less than half are of good or acceptable quality. The most common flaw in the papers we considered of poor quality is the lack of the evaluation, evaluation with strange or erroneous formulas of the performance measures, and the use of doubtful evaluation datasets. These results suggest that there may be a need to review the current research practices in the Arab region.

[19] See for example, the recent SemEval shared tasks on this topic (Agirre et al. 2016; Cer et al. 2017).

[20] In fact, there are some papers on Arabic-English cross-language plagiarism detection, but they are among the papers we discarded from our review because, unfortunately, all of them suffer from either the lack of evaluation or the use of deprecated or suspect data.

Concerning the plagiarism detection approaches applied to the Arabic language, we showed that the intrinsic approach is almost not used. We attempted to mitigate this deficiency by building a corpus of Arabic documents to evaluate the intrinsic approach and harnessing it to evaluate our method. We defer the description of this corpus and the proposed method to the Chapters III and V, respectively. What we will handle in the rest of the present chapter, is the description of the first experiments that we carried out to understand intrinsic plagiarism detection and its applicability to the Arabic language.

4 Preliminary Experiments on Intrinsic Plagiarism Detection in Arabic Documents

We showed in the previous section that hardly any intrinsic plagiarism detection works exist on Arabic documents. Therefore, this section describes a set of preliminary experiments that check the effectiveness of some well-known language-independent stylistic features in discriminating between plagiarized and not plagiarized sentences. We used STYLYSIS tool (Barrón-Cedeño et al. 2012)[21] to measure these features on a small-sized corpus. The next sub-section describes the process of building and pre-processing the evaluation corpus, and the remaining sections describe the experiments and discuss their results.

4.1 Corpus Building and Pre-processing

Since we were the first to deal with Arabic intrinsic plagiarism detection, there were no evaluation corpora. Therefore, we built manually a small corpus of 10 documents with different sizes and topics. Some statistics about this corpus are provided in Table II-6.

We tried to simulate real plagiarism in terms of inserting in each document sentences in relation to its topic. However, we did not obfuscate them (with paraphrasing for example).

Table II-6. Corpus statistics

Number of documents	10
Number of all sentences	336
Number of plagiarized sentences	63
Number of words	13309
Number of tokens	6765

[21] STYLYSIS is an online tool developed in Natural Language Engineering Lab. in the Universitat Politècnica de València (Spain). It allows computing some stylistic features on English and Spanish texts. The tool is no longer available online but a brief description of it could be found in (Barrón-Cedeño 2012, p. 117) and (Rosso 2015, p. 233).

This is because the purpose of our experiments is to detect the sentences with a different writing style from that of the document (i.e., outliers), and paraphrasing the inserted sentences does not guarantee the preservation of the presumed writing style inconsistency between them and the host document.

After the building step, the corpus was pre-processed with the intention to prepare it for the stylistic analysis using STYLYSIS tool. Two tasks were performed in this regard. First, the corpus was transliterated with the Buckwalter scheme[22] (Buckwalter 2000), because STYLYSIS does not support the Arabic language. Second, each document was split into sentences considering full stop as the separation character. STYLYSIS considers lines (all words before newline character) as the text segments on which features are computed regardless of their lengths and the punctuation they contain. For example, if a line contains a paragraph, features such as sentence length and average word length represent the paragraph length and the average length of its words. Hence, in order to compute the features at the sentence level, we pre-processed the documents so that each line contains exactly one sentence.

4.2 Experiments 1: Insight into some Stylistic Features

4.2.1 Description

STYLYSIS computes six lexical features, namely *Gunning Fog Index*, *sentence length*, *average word length*, *Honore's R function*, *Yule's K function,* and *Flesch-Kincaid readability test*. Its results are displayed as graphs with upper and lower boundaries. Sentences over and under these boundaries are considered outliers of the general writing style of the document in terms of the considered feature (discriminator).

We recorded the state (outlier or not) of each sentence from the graphs of four discriminators: *word average length*, *sentence average length*, and *R* and *K functions*. We did not experiment with the two remaining features because their calculation is based on the number of syllables, which could not be computed faithfully from a transliterated Arabic text, especially because our texts are without diacritics, which are represented by vowels after the transliteration[23].

Consequently, each sentence is recorded as a vector of four Boolean values, which represent the sentence states with regard to the four considered discriminators (i.e., features). A sentence is considered plagiarized by a discriminator if it is an outlier.

[22] The Buckwalter transliteration replaces each Arabic letter by a Latin letter or a symbol. For example the transliteration of the Arabic word: انتحال (Plagiarism) is: AntHAl.

[23] For example, the word كتب (wrote) is transliterated as ktb. However, if it is with diacritics it will be كَتَبَ and its transliteration will be kataba.

Table II-7. Performance evaluation

Discriminator	Precision	Recall	F-measure
Average word length	0.160	0.190	0.174
Average sentence length	0.194	0.190	0.192
R function	0.205	**0.413**	0.274
K function	**0.244**	0.349	**0.287**

4.2.2 Results and Discussion

We used three measures to evaluate the performance of discriminators: Precision (equation 1), Recall (equation 2), and F-measure, which is their harmonic mean. Results are shown in Table II-7.

$$Precision = \frac{|\text{ sentences detected as plagiarized } \cap \text{ actual plagiarized sentences }|}{|\text{ sentences detected as plagiarised }|} \quad (1)$$

$$Recall = \frac{|\text{sentences detected as plagiarized } \cap \text{ actual plagiarized sentences }|}{|\text{ actual plagiarized sentences}|} \quad (2)$$

As can be seen from these results, average word length (AWL) has the lowest performance; therefore, it seems it is an unreliable stylistic discriminator of the Arabic text. This is consistent with Abbasi and Chen's (2005) research on authorship analysis[24]. These authors attributed the unreliability of AWL to the small range over which the lengths of Arabic words are distributed.

In contrast to its performance with English text (Meyer zu Eißen et al. 2007), average sentence length (ASL) seems also to be a poor stylistic discriminator of Arabic text as shows our results based on our small corpus.

R and *K* functions, which are used to measure the vocabulary richness, are known as not robust stylistic features with short English texts (Meyer zu Eißen et al. 2007). Surprisingly, our results suggest that they are relatively the most prominent detectors, especially in terms of recall, despite the small length of our corpus sentences.

It should be noted that the *average sentence length* and *R* and *K* functions have already been used among other features in a machine learning method to detect anomalous Arabic texts in a mid-sized corpus (Abouzakhar et al. 2008), but unfortunately, their effectiveness was not tested separately.

[24] Stylistic analysis is the core task of the authorship analysis, which makes the latter a very close domain to intrinsic plagiarism detection.

Table II-8. Combination's results: baseline vs. the most precise voting schemes

Discriminator	Precision	Recall	F-measure
Baseline: Or(ASL, AWL, R, K)	0.215	**0.730**	0.332
And(Or(ASL, R, K), Not(AWL))	**0.245**	0.540	**0.337**

4.3 Experiment 2: Combining Discriminators

In this section, we propose to combine the considered discriminators using voting schemes.

4.3.1 Description

In order to have a basis of comparison, we introduced a lenient baseline method, which considers a sentence as plagiarized if it is detected as an outlier, at least, by one of the four considered discriminators. The recall of this baseline method is the best we can obtain from combining these four lexical discriminators. Moreover, we experimented using other voting schemes in the aim to raise the baseline precision and keep the recall relatively high.

Besides the baseline results, Table II-8 shows the best combination in terms of precision. Voting schemes used to combine discriminators are presented as logical expressions.

4.3.2 Results and Discussion

The expression "Not(AWL)" in the used voting scheme means sentences detected by average word length are discarded even though they are positive with the remaining discriminators. Since AWL is the most imprecise detector (as shown in the previous experiment), many false positive sentences were eliminated by using it as a filter, which yielded an increase of 3% in precision.

An examination of the outliers of the AWL values led us to the two following remarks. First, sentences with a great average word length are mainly: (i) sentences with diacritics (generally Quran verses), or (ii) sentences that contain foreign words. Second, most outliers under the average word length are long sentences composed of several phrases linked by the conjunction "And". This latter is in Arabic a one-letter word "و" which leads to a decrease in the sentence average length. Therefore, we can conclude that the average word length is not a feature of discrimination between different writing styles of Arabic text, and then it is not effective as a plagiarism detector. Nonetheless, it may be useful to filter false positive sentences with the characteristics mentioned above.

5 Conclusion

In this chapter, we presented a review of Arabic plagiarism detection papers. We showed that despite the growing number of papers about this topic in the last years, around 70% of them suffer from quality issues, mainly because of the inadequate reporting and the use of suspicious evaluation data. We leveraged the results of the quality appraisal to filter out from our review the papers of poor quality. The remaining papers are almost all about the external approach. Thus, the last section of this chapter was devoted to preliminary experiments on intrinsic plagiarism detection in Arabic text. We conducted these experiments using a small corpus; hence the need to build large corpora to allow for more advanced studies. In this regard, the next chapter is about the first competition of Arabic plagiarism detection whereby we made available evaluation corpora for both the external and the intrinsic approaches.

Chapter III. Evaluation of Plagiarism Detection on Arabic Documents

> *"The blessing of science comes from attributing it to those who said it.*
>
> Jalal al-Din al-Suyuti (1454-1505)[1]

1 Introduction

A shared task is a computer science competition that invites researchers and practitioners to evaluate their methods to solve a particular research problem[2]. Shared tasks have become important events that foster the development of approaches to unsolved research problems through providing the participants with benchmark datasets against which the performance of the methods is measured (see (Potthast et al. 2014a, 2015) for more information on the organisation of shared tasks).

PAN (standing for **P**lagiarism analysis, **A**uthorship identification, and **N**ear-duplicate detection) as defined in their website[3] is "a series of scientific events and shared tasks on digital text forensics and stylometry". PAN events are organised yearly since 2007 as part of conferences and forums, mainly CLEF (Conference and Labs of the Evaluation Forum)[4] and FIRE (Forum of Information Retrieval Evaluation)[5] in several times.

---

[1] Jalal al-Din al-Suyuti is an Egyptian of Persian origin historian, biographer, jurist, teacher and scholar of Islamic theology (https://en.wikipedia.org/wiki/Al-Suyuti). The original quote in Arabic is "من بركة العلم عزوه إلى قائله".

[2] In the present thesis, the terms *shared task* and *competition* are used interchangeably.

[3] https://pan.webis.de

[4] http://www.clef-initiative.eu (see (Ferro and Peters 2019) for further information on CLEF with all its evaluation initiatives)

[5] http://fire.irsi.res.in

Since the beginning of the organisation of PAN, the organisers have been interested in addressing topics related to plagiarism detection through information retrieval (IR) techniques to search for sources of the reused texts and through stylometry techniques to uncover the doubtful style changes within the document. In fact, PAN shared tasks were instrumental in establishing a standardised evaluation framework for plagiarism detection (Potthast et al. 2010c), which consists of a set of quality measures and a series of evaluation corpora involving automatically or semi-automatically created suspicious documents. This framework served actively in pushing forward research on plagiarism detection[6].

Although research on plagiarism detection has gone a long way in establishing evaluation frameworks, we still know very little about the effectiveness of existing software when used with Arabic texts[7]. We attribute this lack of knowledge to the absence of a dedicated standardised corpus of Arabic documents.

One may ask why not to evaluate methods only on English documents and presume that their performance will be similar in other languages. We argue that building a corpus is necessary to evaluate the detection of plagiarism in Arabic documents for the following reasons:

- The ability to test methods that take into account Arabic peculiarities (e.g. diacritics, inflexion, …, etc.) or are based on language-specific processing like parsing or stemming.
- The ability to tune the parameters of language-independent methods for Arabic. For example, in Chapter V (pp. 116, 120-121) we show that, within two intrinsic plagiarism detection methods, the optimal parameters for Arabic and English are different.
- In the context of intrinsic plagiarism detection, writing style discriminators in English are not necessarily discriminators in Arabic, e.g., we have shown in preliminary experiments in Chapter II (Section 4.2.2) that the average sentence length is not a worthy style discriminator for Arabic. In other experiments (Meyer zu Eißen et al. 2007), however, this feature was among the best ones to differentiate English passages of different styles.

To solve the evaluation issue of plagiarism detection in Arabic documents, we organised AraPlagDet, which is the first shared task that addresses this problem. AraPlagDet[8] was held under the 7th edition of FIRE[9]. Similar to past PAN competitions on plagiarism detection in English documents (Potthast et al. 2009, 2010a, 2011), AraPlagDet has two sub-tasks to evaluate the main plagiarism detection approaches, namely external plagiarism detection and intrinsic plagiarism detection. As part of AraPlagDet organisation, we developed evaluation

[6] See (Rosso et al. 2019) for an overview of all PAN shared tasks including the plagiarism detection task.
[7] Corpora in several languages have been submitted to PAN 2015 text alignment shared task (Potthast et al. 2015). We participated in the evaluation of the quality of some of these corpora (Franco-Salvador et al. 2015), but none of them is in Arabic.
[8] The official web page of the shared task is: http://misc-umc.org/AraPlagDet . But it is also available on PAN website: https://pan.webis.de/fire15/pan15-web/araplagdet.html
[9] http://fire.irsi.res.in/fire/2015/home

corpora for training and testing the methods of the participants. Eventually, eight systems have been tested on these corpora.

This chapter describes our contribution to fostering the development and the standard evaluation of plagiarism detection software for the Arabic language. The rest of the chapter is organised as follows. Section 2 exposes our motivations to organise AraPlagDet. Section 3 is a brief description of the approaches used to build plagiarism detection corpora. Sections 4 introduces the AraPlagDet shared task. Sections 5 and 6 provide detailed discussions on the evaluation corpora that we created and the methods submitted to the external and intrinsic plagiarism detection sub-tasks, respectively. Section 7 draws some conclusions.

2 Motivation

Despite the lack of large-scale studies on the prevalence of plagiarism in the Arab world, the huge number of news on this phenomenon in media[10] attests its pervasiveness. Moreover, some studies show the lack of awareness, among Arab students, on the definition of plagiarism and the seriousness of this offence (Elgendy 2014; Hosny and Fatima 2014). These studies suggest the use of plagiarism detection software as one of the solutions to tackle the problem. In the last years, some papers on Arabic plagiarism detection have been published (Alzahrani and Salim 2008; Bensalem et al. 2012; Hussein 2015; Khan et al. 2015; Menai 2012; Soori et al. 2014). However, it is difficult to draw a clear conclusion on the performance of these methods since they were evaluated, using different strategies and corpora. For instance, Jadalla and Elnagar (2012) compared their system with a baseline method using a number of documents that were supposed to be suspicious. Alzahrany and Salim (2009), as well as Menai (2012), evaluated their methods using respectively 15 and 300 suspicious documents constructed by rewording and restructuring some parts of the documents. Jaoua et al. (2011) created 76 suspicious documents by the manual insertion of text fragments obtained by queries to a search engine, using keywords in relation with the subject of the document that will host the plagiarism.

As stated in the introduction, one of our motivations to organise the AraPlagDet shared task was to make available an evaluation corpus that allows for proper performance comparison of the Arabic plagiarism detectors. In addition to that, we had also other motivations, which are:

- To contribute to raising awareness in the Arab world on the seriousness of plagiarism and the importance of its detection.
- To promote the development of plagiarism detection techniques that deal with the Arabic

[10] Some news stories on plagiarism in Algeria: http://gulfnews.com/news/uae/culture/plagiarism-costs-ba-li-zayed-book-award-1.702316 and in Egypt: http://www.universityworldnews.com/article.php?story=20080717165104870

language.

- To encourage the adaptation of existing software packages for Arabic, as it is one of the most widespread languages in the world[11].

Since a standardised evaluation corpus is essential for the organisation of a shared task, we will first provide, in the next section, a summary of the approaches used to build plagiarism detection corpora before going further into the description of AraPlagDet competition.

3 Approaches to Creating Plagiarism Detection Evaluation Corpora

The assessment of the quality of plagiarism detection software requires a collection of documents (a.k.a. a corpus) that contains plagiarised passages i.e., a collection of suspicious documents. The plagiarism detector is then run over those documents to appraise its ability to spot the plagiarised passages without confusing them with the parts of text that are actually written by the supposed author. Of course, the plagiarised passages should be annotated in the suspicious documents to allow for a comparison with the outcome of the software.

However, there is a concern in this regard: collecting a sufficient number of documents incorporating real plagiarism might be unpractical for several reasons. Potthast et al. (2010c) report, among other reasons, the difficulty of procuring such documents since they are commonly concealed. The authors state, in addition, that making publically available such corpus (as an evaluation benchmark) is ethically and legally questionable.

To avoid these issues, Potthast et al. (2010c) introduced a relatively effortless way to build suspicious documents, which consists in creating synthetic plagiarism cases and inserting them in random positions in host documents. For that, one has to compile two sets of documents: (i) the source documents, from which passages of text are borrowed; and (ii) the host documents, in which the aforementioned passages are inserted after (optionally) obfuscating them. The obfuscation is applied to simulate the behaviour of a plagiarist who tries to camouflage its act by paraphrasing the stolen text (see Figure III-1 for an illustration of these steps). In the literature, if the obfuscations are created automatically (e.g., through shuffling words) the plagiarism is called *artificial* or *automatic*. Whereas, if the plagiarism is obfuscated by manual paraphrasing it is called *simulated*.

Note that in order to make the corpus publicly available, it is substantial that the source and the host documents (used to build the corpus) are copyright-free to avoid any legal problems with the owners of these texts. Meeting this condition is not always straightforward since many texts available online are protected by copyright. On the other hand, as stated in one of the

[11] See for example, some statistics on the countries where Arabic is an official language: https://en.wikipedia.org/wiki/List_of_countries_where_Arabic_is_an_official_language.

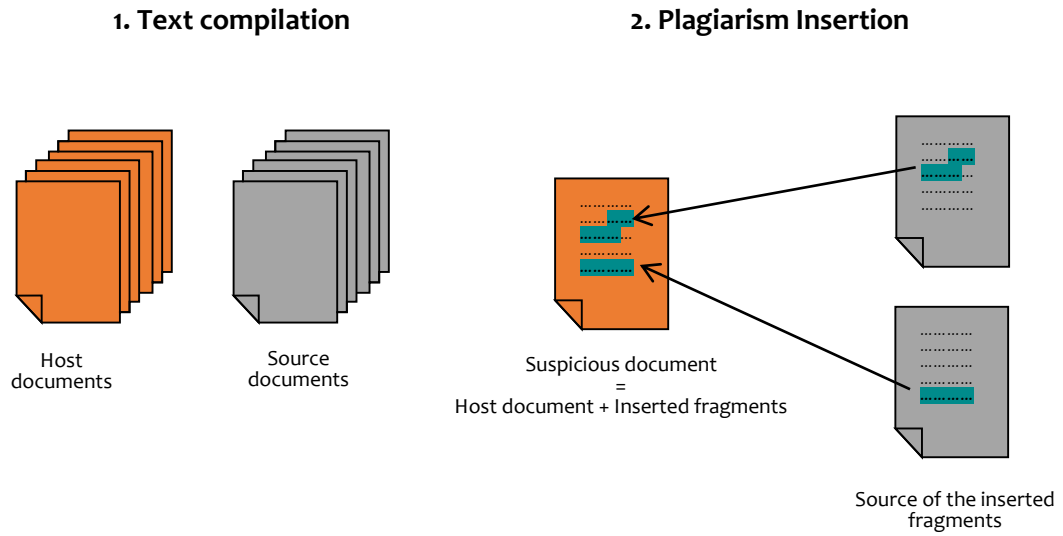


Figure III-1. The insertion based approach of building plagiarism detection evaluation corpora

important books of building corpora (McEnery et al. 2006), the copyright-free documents are generally old texts which are not useful for projects requiring contemporary texts.

Fortunately, since plagiarism detection evaluation corpora are not intended to study the language (as the case of linguistic corpora for example), it is not essential that the texts included in these corpora be contemporary. Indeed, a number of standardised plagiarism detection evaluation corpora were built from the classical texts provided by the Gutenberg website[12].

On the other hand, the copyright issue can be avoided as well by creating completely new texts instead of collecting and processing existing ones. This is another approach of creating suspicious documents, and it is based on *crowdsourcing*, which consists in hiring a group of people to write essays on designated topics. Those writers are asked to reuse texts deliberately from a predefined source of documents (Barrón-Cedeño et al. 2013a; Clough and Stevenson 2011; Potthast et al. 2013b). While this approach produces semi-realistic plagiarism, it is cost-prohibitive due to its requirements in terms of material and human resources and time (Potthast et al. 2013b), and hence it is difficult to scale in terms of the number of suspicious documents and their lengths. Table III-1 summarises the differences between the aforementioned approaches to creating suspicious documents.

To create AraPlagDet corpora, we adopted the automatic approach since it has several advantages in comparison with the other approaches as shown in Table III-1. Furthermore,

[12] http://www.gutenberg.org (last consultation: 30/3/2019).

Table III-1. Comparison between approaches to creating suspicious documents. The symbol ✓ indicates an advantage, and ✗ indicates a disadvantage.

	Collecting documents with real plagiarism	Automatic insertion of plagiarism in host documents	Crowdsourcing
Scalability in document lengths	✗	✓	✗
Scalability in the number of documents	✗	✓	✗
Realism of the plagiarism scenario	✓	✗	✓
Copyright issue	✗	✗	✓
Ethical issue	✗	✓	✓
Cost (time, material and/or human resources)	✗	✓	✗

this approach has been used to build several corpora of PAN plagiarism detection shared tasks. Particulars of how our corpora were created are provided later in the sections dedicated to each of the AraPlagDet sub-tasks.

4 AraPlagDet Shared Task Description

As mentioned from the outset, AraPlagDet shared task involves two sub-tasks, namely: external plagiarism detection and intrinsic plagiarism detection. Each participant was allowed to submit up to three runs[13] in one or both sub-tasks. From 2009 to 2011, PAN plagiarism detection competitions have been organised with these two sub-tasks[14]. The evaluation corpora in these shared tasks were mostly English. Thus, AraPlagDet is the first plagiarism detection shared task addressing the Arabic language.

External and intrinsic plagiarism detection are significantly different approaches for plagiarism detection (see the previous chapter, Section 2). In the external plagiarism detection sub-task, participants were provided with two collections of documents, namely suspicious and source documents, and the task was to identify the overlaps (exact or not) between them. In the intrinsic plagiarism detection sub-task, participants were provided with suspicious

[13] We use the term *run* to refer to the output of the methods submitted by the participants as XML files.

[14] In PAN 2012, the intrinsic plagiarism task has been organised as a subtask of the author identification shared task. From 2012 to 2016, the external approach has been addressed in two subtasks, which are source retrieval and text alignment.

Table III-2. AraPlagDet shared task schedule

	Training period start	Test period start	Competition end (runs submission deadline)
External Plagiarism Detection subtask	July 16*	September 21	October 18
Intrinsic Plagiarism Detection subtask	June 20	September 21	October 18

*July 16 is the release date of a sample of the training corpus. The complete training corpus has been released on August 10.

documents and the task was to identify in each document the inconsistencies with respect to the writing style if any.

A total of 18 teams and individuals from different countries (six of them are not Arab) registered in the shared task, which shows the interest of practitioners and researchers in this topic. However, only three participants submitted their runs[15].

As the case of most shared tasks, AraPlagDet has two phases: the training period that started by the release of the training datasets and the testing period that started by the release of the test datasets. Table III-2 shows the dates of these periods. Note that we released the training datasets with their ground truth. However, the ground truth of the test datasets was made available to the participants only after the end of the competition. This means that during the test phase, the participants were not allowed to compute themselves the performance of their methods on the test data. We rather asked them to submit the output of their methods as XML files that determine the positions of the detected plagiarism and it was our work, as organisers of the shared task, to compute the performance by comparing the methods' outputs to the none-released ground truth (Section 5.3.1 shows how the performance is computed).

5 External Plagiarism Detection Sub-task

In this section, we describe the evaluation corpus and the submitted methods for the external plagiarism detection sub-task.

5.1 Corpus

5.1.1 Source of Text

To build our corpus for external plagiarism detection sub-task (ExAra corpus), we used

[15] This might be due to the fact that the competition started in the summer season, but the call for participation was launched a month before that. Maybe it would have been better if we had released the training corpora before summer.

documents from the Corpus of Contemporary Arabic (CCA)[16] (Al-Sulaiti and Atwell 2006) and Arabic Wikipedia[17]. The CCA involves hundreds of documents in a variety of topics and genres. Most of them have been collected from magazines. Our motivation to use the CCA as the main source of texts for our corpus is three-fold:

- The documents of the corpus have a variety of topics and genres. Such a variety is desirable because it makes plagiarism detection corpus more realistic.
- Each document is tagged with its topic, which is an advantageous feature to the process of creating artificial suspicious documents. In this process, which attempts to simulate real plagiarism, the topic of the inserted plagiarism cases should match the topic of the suspicious (host) document.
- The corpus is freely available and their developers strove to get the copyright permissions from the owners of the collected texts to use them for research purposes[18].

Note that for each topic, we used some documents as a source of the inserted fragments and some others as host documents. Besides CCA, we included in our corpus – specifically in the source documents set – hundreds of documents from Arabic Wikipedia. We collected them manually by selecting documents that match the topics of the suspicious documents. These documents have been incorporated in the corpus to baffle the detection, and only a few cases have been created from them. Surprisingly, we realised[19] that many of the collected Wikipedia articles (notably biographies) contain exact or near-exact copies of large passages from the CCA documents[20]. This fact resulted in plagiarism cases that are not annotated in the corpus. To address this issue, we applied a simple 5-grams method to identify these cases of 'real' plagiarism between the suspicious documents and the collected Wikipedia documents, and we discarded from the corpus the Wikipedia documents involving the detected passages[21].

5.1.2 Obfuscations

We created two kinds of plagiarism cases: artificial (created automatically) and simulated (created manually). For the automatically created cases, we used the strategy of *phrase shuffling* and *word shuffling*. To avoid producing cases that have the same pattern of shuffling,

[16] http://www.comp.leeds.ac.uk/eric/latifa/research.htm

[17] http://ar.wikipedia.org

[18] From our side, we contacted Eric Atwell (the co-developer of CCA) who gives us the permission to use CCA documents in our corpus.

[19] We started to be aware of this issue thanks to AraPlagDet participants who pointed out the existence of some plagiarism cases that have not been annotated in ExAra sample corpus, which has been released before the official training corpus.

[20] By a quick investigation into some of the plagiarised passages, it seems that the concerned texts have been plagiarised by the Wikipedia contributors from online magazines' articles (which some of them are part of the CCA corpus), and not the inverse. This throw doubt on the originality of the Arabic Wikipedia content, hence the need for a thorough investigation into this subject.

[21] Annotating the plagiarism in these documents would be a better solution but we chose to discard them because of time limitation with respect to the deadline of publishing the corpus to the participants to allow them to start the evaluation of their methods.

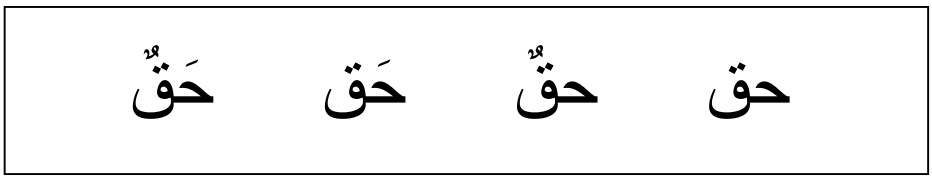


Figure III-2. Different representations of the same word with and without letters' diacritics.

we applied to the cases of the test corpus a different algorithm than the one used for the training corpus.

Regarding manually created plagiarism, we employed two obfuscation strategies: *synonym substitution* and *paraphrasing*. Both of them are described below.

Manual Synonym Substitution

To create plagiarism cases with this obfuscation, we did the following:

- We manually replaced some words with their synonyms. The sources of synonyms we used are Almaany dictionary[22], the Microsoft Word synonym checker, Arabic WordNet Browser[23], and the synonyms provided by Google translate[24]. It should be noted that an Arabic singular noun might have multiple plural forms that are synonymous. For example, the word 'جزيرة' (*jazira*– island) has the plurals 'جزائر' (*jazair*) and 'جزر' (*juzur*).
- We added diacritics (short vowels) to some words. Note that diacritics in Arabic are optional and their inclusion or exclusion are orthographically acceptable. Consequently, we can have for a word *w* whose length is *n* letters, at least 2^n different representations. For example, the different representations of the word 'حق' (*haq*– truth) with and without diacritics are depicted in Figure III-2.

We decided to manually substitute words with their synonyms (regardless of it is time-consuming) after many attempts to perform this task automatically. Despite our efforts to obtain exact synonyms by using part of speech tagging and lemmatisation, our attempts produced unintelligible passages, passages with totally different meanings from the original ones (poor precision) or passages with hardly any substituted word (poor recall). These unsuccessful attempts could be respectively attributed to:

(i) *The high ambiguity of the Arabic language*: researchers estimated the average number of ambiguities for a token in the Arabic language is 8 times higher than it in most of the other languages (Farghaly and Shaalan 2009). Therefore, it is not surprising to find it difficult to select automatically the appropriate synonym in a given context.

22 http://www.almaany.com
23 http://globalwordnet.org/arabic-wordnet/awn-browser
24 https://translate.google.com

Table III-3. Statistics of the ExAra corpus

		Training	Test
Generic information	Total number of documents	1174	1171
	Source documents	48.30%	48.68%
	Suspicious documents	51.70%	51.32%
	Total number of cases	1725	1727
Plagiarism per document	Without plagiarism	27.84%	28.12%
	With plagiarism	72.16%	71.88%
	Hardly (1%-20%)	33.77%	36.94%
	Medium (20%-50%)	36.74%	32.95%
	Much (50%-80%)	1.65%	2.00%
Document length	Very short (< 1 p)	22.57%	17.51%
	Short (1-10 pages)	73.34%	76.26%
	Medium (10 -100 pages)	4.09%	6.23%
Case length	Very short (< 300 chars)	21.28%	21.25%
	Short (300-1k chars)	42.43%	42.50%
	Medium (1k-3k chars)	28.46%	28.26%
	Long (3k-30k chars)	7.83%	7.99%
Plagiarism type and obfuscation	Artificial	88.93%	88.94%
	Without obfuscation	40.35%	40.30%
	Phrase shuffling	11.25%	10.42%
	Word shuffling	37.33%	38.22%
	Simulated	11.07 %	11.06%
	Manual synonym substitution	9.80%	9.79%
	Manual paraphrasing	1.28%	1.27%

(ii) *The limited coverage of lexical resources*: in our experiments, we used Arabic WordNet as a source of synonyms. Unfortunately, this resource, which is one of the most important and freely available linguistic resources for Arabic, contains only 9.7 % of the estimated Arabic vocabulary (Abouenour et al. 2013). Hence, the very low recall of the automatic synonym substitution is quite justified.

Manual Paraphrasing

Cases produced with this obfuscation strategy are the most realistic ones in our corpus. This is because the passages to be obfuscated have been *manually selected* from the source and then *manually paraphrased*. The results are plagiarism cases that are very close in terms of their

topic to the suspicious documents that host them. In this type of obfuscation, all kinds of modifications were applied (restructuring[25], synonym substitution, removing repetitions, etc.), provided that the meaning of the original passage is preserved.

Due to the dullness and slowness of the manual process[26], we produced 338 cases with synonym substitution obfuscation and only 44 cases with paraphrasing obfuscation. See Table III-3 for more detailed statistics.

5.2 Methods Description

Three participants submitted their runs. Since multiple submissions were allowed, two participants submitted three runs. Therefore, we collected a total of seven runs. Two participants among the three submitted working notes describing their methods. Following, we summarise the work of these two participants.

5.2.1 Participants Methods

We describe in this subsection the methods of Magooda et al. (2015) and Alzahrani (2015). Magooda et al. used two different approaches for source retrieval and three for text alignment and combined them in different ways in the three submitted methods: Magooda_1, Magooda_2, and Magooda_3. Alzahrani submitted one method. Table III-4 and Table III-5 provide details on these approaches. Following, we discuss the submitted methods regarding two aspects: scalability and language dependence regardless of their performance that will be discussed later.

Scalability

First, it should be noted that our evaluation corpus could be considered medium-sized especially in comparison with the PAN English corpora (Potthast et al. 2009, 2010a, 2011, 2012, 2013a, 2014b). Furthermore, we did not determine in the instructions of the competition the retrieval techniques to utilise. Nonetheless, to avoid being merely a lab method, any plagiarism detection approach needs to deal with large sets of documents by using appropriate retrieval techniques. Magooda et al. in their three methods used the Lucene search engine and two indexing approaches as shown in Table III-4. Therefore, their methods could be used with a large collection of source documents, and they could be adapted to be deployed online with a commercial search engine, which is an obvious solution to adopt if the source of plagiarism is the web as pointed out by Potthast et al. (2012).

[25] Restructuring a text consists in repositioning the words or the phrases of a sentence while preserving the meaning, e.g., changing the voice of sentence from passive to active or vice versa.

[26] We are aware about the possibility to use crowdsourcing to allow the creation of a large number of plagiarism cases manually (Potthast et al. 2010c). However, because of the lack of financial resources, we crafted the mannual cases ourselves (apart from some few volunteers) in collaboration with two colleagues: Imene Boukhalfa, from Constantine 2 University and Lahsen Abouenour from Mohammadia School of Engineers and Mohamed V Rabat University, Morocco.

Table III-4. Source retrieval approaches with their building blocks used in the participants' methods. Each column describes an approach in terms of its building blocks. The first line provides a concise description of the approach, and the second line indicates the methods that employed each approach. For example, Magooda_2 method used two approaches: sentence-based and keyword-based indexing.

Sentence-based indexing approach	keyword-based indexing approach	Fingerprinting approach
Magooda_1, Magooda_2, Magooda_3	Magooda_2, Magooda_3	Alzahrani
Chunking		
Splitting the document into sentences	Splitting the document into paragraphs	–
Keyphrase extraction		
–	Named entities with high *idf*. Terms with high *idf*.	–
Queries formulation		
All sentences	Two kinds of queries are extracted from each paragraph: (i) Combination of named entities and terms that have the highest *idf*. (ii) 10-grams that contain the maximum terms and named entities with the highest *idf*. Stemming is applied to queries.	–
Search Control		
–	–	–
Candidate Filtering		
1- Ranking the source documents according to the number of queries used to retrieve them. 2- Keeping the first ranked document for each query.	Keeping the top 10 retrieved documents for each query.	1- Generating word 3-grams for both suspicious and source documents, and computing Jaccard similarity between them. 2- Keeping the source document if Jaccard ≥ 0.1.

With respect to Alzahrani's method, it is suitable to an offline scenario, i.e., when the source of plagiarism is local and not too large, as in the case of detecting plagiarism between students' assignments. This is for two reasons: (i) its retrieval model is not structured to be used with search engines (for example, there is no query formulation, see Table III-4); and (ii) it is based on fingerprinting all the source documents and entails an exhaustive comparison between the n-grams of the suspicious document and those of each source document, which is not workable if the source of plagiarism is extremely large, like the web. Still, even with the intention to be used offline, it would be better to use retrieval techniques that allow for the processing of a large number of documents in a reasonable time such as inverted indexes. Malcolm and Lane (2009) discuss the importance of scalability even for offline plagiarism detectors.

Table III-5. Text alignment approaches with their building blocks used in the participants' methods

Sentence-based approach	Common word approach	Skip-grams approach	N-grams similarity approach
Magooda_1, Magooda_2, Magooda_3	Magooda_1, Magooda_2	Magooda_2, Magooda_3	Alzahrani
Seeding			
Matching sentences	Matching words	Matching the *n*-skip-3-grams extracted from windows of 5 words after stemming.	Matching K-overlapping 8-grams if the similarity between them > threshold. The computed similarity is based on the n-gram correlation factor.
Extension			
Keeping the pair of sentences if the distance between it and a neighbouring pair is less than a threshold.	From a window of *n* words, creating a passage that contains the closest word matches.	Grouping the adjacent matched skip-grams if the distance between them < threshold.	Merging the consecutive matched 8-grams if the distance between them is ≤ 300 characters.
Passage Filtering			
Keeping the pair if the passages are equivalent, else discard it if: - passages length < threshold - the number of the words matches < threshold			–

Language Dependence

Regarding this aspect, Magooda et al. reported the use of two language-dependent processing in the source retrieval phase: stemming queries before submitting them to the search engine and extracting named entities. In the text alignment phase, words are stemmed in the skip-gram approach. Moreover, their methods pre-process the text by removing diacritics and normalising letters[27]. Alzahrani method is nearly language-independent. The only reported language-specific process was stop-words removal. It was applied as a pre-processing step on the suspicious and the source documents.

Since the external plagiarism detection is a retrieval task, we think that challenges of Arabic IR hold for Arabic plagiarism detection. Arabic IR is challenging because of the high inflexion of Arabic and the complexity of its morphology. Arabic stems are derived from a set of a few thousand roots by fitting the roots into stem templates. Stems can accept attached prefixes and suffixes that include prepositions, determiners, and pronouns. Those are sometimes obstacles to match similar texts (Larkey et al. 2007). Moreover, unlike many other languages, Arabic writing includes diacritics that are pronounced, but often not written. As opposed to Latin languages, the use of diacritics in Arabic is not restricted to some letters: they

[27] Diacritics removal, and letters normalisation are not reported in Magooda et al.'s (2015) working notes. We found out about that because of a discussion with the first author.

could be rather placed on every letter. Indeed, in Arabic IR, diacritics are typically removed (Darwish and Magdy 2013; Habash 2010). Another issue that affects Arabic IR and consequently Arabic plagiarism detection is the fact that Arabic has some letters that are frequently used interchangeably such as: (ي, ى), (ا,أ, إ , آ) and (ه, ة) hence the need for a letter normalisation pre-processing. If the orthographic normalisation (diacritics removal and letter normalisation) is not employed, a plagiarism detection system may fail to match similar passages even if they have exactly the same words. See Figure III-3 for an illustration.

5.2.2 Baseline

We employed a simple baseline, which entails detecting common chunks of word 5-grams between the suspicious documents and the source documents and then merging the adjacent detected chunks if the distance between them is smaller than 800 characters[28]. Short passages (< 100 characters) are then filtered out. Since it is primarily based on matching n-grams, this baseline should detect mainly plagiarism cases that are not obfuscated.

5.3 Evaluation

5.3.1 Measures

We evaluated the methods using the character-based macro-averaged precision and recall in addition to the granularity, and ranked them using the plagdet that combines these measures in one measure. These measures have been introduced by Potthast et al. (2010c) specifically to evaluate plagiarism detection methods.

In these tailored measures, which became a standard for evaluating plagiarism detection methods, the plagiarised and detected fragments are perceived as sets of characters. As shown in equations 1 and 2, precision and recall are computed for each fragment and then averaged over the total number of fragments. For example, if the length of an actual plagiarism fragment s_{act} is 100 characters and the method marked as plagiarised half of this fragment, then the recall computed at the fragment level will be 0.5. Similarly, if the length of a detected fragment s_{det}

عاشت "إنديرا غاندي" أول رئيسة وزراء للهند الحياة السياسية بكل تقلباتها

عَاشَت "انديرا غاندى" اول رئِيسَة وُزرَاء للهنْد الحياه السّياسيّة بِكلّ تقلبَاتهَا

Figure III-3. Two passages with the same words but the second passage contains some letters with diacritics (highlighted in green) and a substitution of some interchangeable letters (highlighted in yellow). A simple plagiarism detector may fail to match them.

[28] This threshold has been determined experimentally.

is 100 characters, and only half of it is actually plagiarised, then the precision computed on this fragment will be 0.5.

Note that in the equations below, *Act* is the set of the plagiarism cases annotated in the corpus (the *Act*ual cases) and *Det* is the set of the plagiarism cases detected by the method (the *Det*ected cases). The symbols |*Act*| and |*Det*| are the number of actual and detected cases, respectively. The symbols $|s_{act}|$ and $|s_{det}|$ are, respectively, the lengths of s_{act} and s_{det} in characters.

$$precision(Act, Det) = \frac{1}{|Det|} \sum_{s_{\text{det}} \in Det} \frac{\left|\bigcup_{s_{\text{act}} \in Act} (s_{\text{act}} \cap s_{\text{det}})\right|}{|s_{\text{det}}|} \quad (1)$$

$$recall(Act, Det) = \frac{1}{|Act|} \sum_{s_{\text{act}} \in Act} \frac{\left|\bigcup_{s_{\text{det}} \in Det} (s_{\text{act}} \cap s_{\text{det}})\right|}{|s_{\text{act}}|} \quad (2)$$

$$granularity(Act, Det) = \frac{1}{|Act_{det}|} \sum_{s_{\text{Act}} \in Act_{det}} |Det_{s_{\text{act}}}| \quad (3)$$

$$plagdet(Act, Det) = \frac{\text{F} - \text{measure}}{\log_2\left(1 + granularity(Act, Det)\right)} \quad (4)$$

For a single actual plagiarism case, s_{act}, a plagiarism detection method may output multiple detections (separate or overlapping). Thus, granularity is used to average the number of the detected cases for each actual case as depicted in formula 3. $Act_{det} \subseteq Act$ is the set of the actual cases that have been detected, and $Det_{s_{\text{act}}} \subseteq Det$ is the set of the detected cases that intersect with a given actual case s_{act}. The optimal value of the granularity is 1, and it means that for each actual case s_{act}, no more than a single case has been detected (i.e. not many overlapping or adjacent cases).

To rank methods, a combination of the three measures is applied in the plagdet as expressed in the formula 4 where F-measure is the harmonic mean of precision and recall.

See (Potthast et al. 2010c) for more information on plagiarism detection evaluation measures.

5.3.2 Overall Results

Table III-6 provides the performance results of the baseline as well as the participants' methods on the test corpus. As shown in the table, four methods outperform the baseline in terms of the plagdet. In terms of precision, the majority of methods are good, but none of them performed better than the baseline. Regarding the recall, the best three methods have acceptable scores, but the rest of the methods' scores are more or less close to the baseline. All the methods have a granularity of more than 1.05, which is not a very good score in

Table III-6. Performance of the external plagiarism detection methods on the test corpus

Method	Precision	Recall	Granularity	Plagdet
Magooda_2	0.852	**0.831**	1.069	**0.802**
Magooda_3	0.854	0.759	1.058	0.772
Magooda_1	0.805	0.786	**1.052**	0.767
Palkovskii_1	0.977	0.542	1.162	0.627
Baseline	**0.990**	0.535	1.209	0.608
Alzahrani	0.831	0.530	1.186	0.574
Palkovskii_3	0.658	0.589	1.161	0.560
Palkovskii_2	0.564	0.589	1.163	0.518

comparison with what has been achieved by the state-of-the-art methods (see for example the results of PAN 2014 shared task (Potthast et al. 2014b)).

5.3.3 Detailed Results

The goal of this section is to provide an in-depth look at the behaviour of the methods. Table III-7 presents the performance of the participants' methods on the test corpus according to some parameters namely cases length, type of plagiarism and obfuscation.

Table III-7 reveals that, interestingly, the three methods of Magooda et al. are the only ones that detect cases with word shuffling obfuscation. This explains the low overall recall of Palkovskii's and Alzahrani's methods. It seems that the algorithm employed to shuffle words generates cases that are difficult to detect by the fingerprinting approach used in Alzahrani source retrieval phase. Magooda_1 and Magooda_2 methods perform better than Magooda_3 with respect to the word shuffling cases. This is thanks to the common words approach, which is able to match similar passages no matter the order of words. Regarding the impact of the case length, all the methods perform better with medium-sized cases.

All the methods achieved a very high recall in detecting cases without obfuscation. On the other hand, the manual paraphrasing cases are the most challenging to detect after the word shuffling cases.

5.3.4 Analysis of the False Positive Cases

Typically, it is easy to obtain a reasonable precision. This could be observed in the majority of the results in Table III-6. This behaviour was observed also in PAN shared task on plagiarism detection (Potthast et al. 2014b). Since Palkovskii_2 method is the least precise

among all the submitted methods, we were keen to understand the underlying reason behind its poor precision score. An examination of its outputs revealed that around 60% of the utterly false-positive cases (cases whose precision is 0) stem from documents with religious content. We went one step further and looked into the text of these cases. It turned out that the phrase "صلى الله عليه وسلم" was the underlying seed of many false-positive cases. This phrase, which translates as "May Allah honour him and grant him peace", is a commonly used expression in Arabic (written and even spoken) after each mention of the prophet Muhammad. Another kind of false-positive cases that stem from religion-related texts are quotations from Quran and Hadith (sayings of the prophet Muhammad). Some false-positive cases in the Palkovskii_2's run and even in the other methods' runs are of this kind. For instance, our examination of the detected cases in Magooda_2's run reveals that Quranic verses represent 6% of the utterly false-positive cases. It turned out that those detected Quranic verses exist also in some of the source documents, which led to flagging them as plagiarism.

An important characteristic for any plagiarism detection system is not to flag common phrases and quotations as plagiarism cases unless they appear as part of a larger plagiarism case[29]. In Arabic texts, and notably in texts about religious topics, quotations from Quran and Hadith are very common. Moreover, there are some religious phrases that could be repeated many times in documents. The expression "صلى الله عليه وسلم" ("may Allah honour him and grant him peace") is an example of such common phrases. In the ExAra test corpus, it appears 185 times in the suspicious documents and 171 times in the source documents. This increases the risk of obtaining many short false-positive cases. Still, this issue could be addressed simply by filtering out the very short detected cases. In the baseline method, for example, we applied such a filter and we obtained very high precision.

Another problem is that the common religious phrases may appear many times even in the same document. For example, the expression "صلى الله عليه وسلم" ("may Allah honour him and grant him peace") occurs 29 times in the 'suspicious-document0014' and 52 times in 'source-document00223'. This increases not only the risk of obtaining short false-positive cases (of some few words) but also longer cases when the adjacent seeds are merged in the extension step. We observed many cases of this kind in the output of Palkovskii_2 method. See Figure III-4 for an illustration.

As we already said, citing religious texts is common in Arabic writing. Moreover, many of the Arab countries are incorporating religion in their public schools' curricula (Faour 2012). Therefore, we believe in the need to have plagiarism detectors that are able to cope with religious citations and expressions.

[29] To be fair with regard the participating methods, it is important to mention that the participants in AraPlagDet external plagiarism detection sub-task were not asked to filter out the religious quotations (but this was among the rules of the intrinsic plagiarism detection sub-task). This explains why the methods mark the quotations as plagiarism.

Table III-7. Detailed performance of the participants' methods. In each measure, the underlined values are the highest per parameter.

	Precision								Recall								Granularity								Plagdet							
	Magooda_2	Magooda_3	Magooda_1	Palkovskii_1	Baseline	Alzahrani	Palkovskii_3	Palkovskii_2	Magooda_2	Magooda_3	Magooda_1	Palkovskii_1	Baseline	Alzahrani	Palkovskii_3	Palkovskii_2	Magooda_2	Magooda_3	Magooda_1	Palkovskii_1	Baseline	Alzahrani	Palkovskii_3	Palkovskii_2	Magooda_2	Magooda_3	Magooda_1	Palkovskii_1	Baseline	Alzahrani	Palkovskii_3	Palkovskii_2
Case length																																
very short	.753	.763	.693	.935	.978	.616	.493	.404	.747	.600	.679	.483	.431	.470	.548	.548	1.017	1.000	1.019	1.000	1.011	1.005	1.000	1.000	.741	.672	.677	.637	.594	.531	.519	.465
short	.862	.853	.807	.997	.998	.925	.647	.551	.850	.783	.818	.513	.505	.494	.554	.554	1.011	1.003	1.009	1.008	1.083	1.020	1.002	1.002	.850	.814	.807	.674	.634	.635	.596	.551
medium	.912	.910	.866	.999	.995	.961	.926	.866	.893	.867	.839	.645	.660	.637	.682	.682	1.025	1.024	1.029	1.127	1.171	1.290	1.039	1.039	.886	.873	.835	.720	.710	.641	.764	.742
long	.953	.947	.940	.999	.998	.800	.988	.988	.739	.676	.717	.491	.526	.511	.562	.562	1.641	1.583	1.412	2.506	2.462	2.077	2.987	3.026	.594	.576	.640	.364	.384	.384	.359	.357
Plagiarism Type																																
artificial	.891	.891	.834	.981	.994	.863	.683	.589	.835	.754	.795	.555	.536	.538	.558	.558	1.077	1.066	1.034	1.140	1.238	1.192	1.190	1.194	.818	.781	.795	.646	.599	.586	.543	.505
simulated	.819	.822	.850	.993	.979	.850	.825	.814	.800	.797	.716	.442	.523	.469	.845	.845	1.000	1.000	1.213	1.325	1.000	1.148	1.017	1.017	.809	.809	.678	.503	.681	.548	.825	.819
Obfuscation																																
none	.903	.912	.810	.978	.991	.894	.668	.583	.982	.974	.904	.999	.992	.988	.984	.984	1.003	1.003	1.003	1.000	1.022	1.220	1.000	1.000	.939	.940	.852	.989	.976	.816	.796	.732
word shuffling	.890	.871	.890	.000	.000	.000	.000	.000	.657	.492	.657	.000	.000	.000	.000	.000	1.081	1.044	1.081	-	-	-	-	-	.715	.610	.715	-	-	-	-	-
Phrase shuffling	.863	.865	.752	.999	.999	.860	.890	.889	.921	.869	.879	.870	.743	.772	.954	.954	1.360	1.382	1.000	1.689	2.124	1.084	1.933	1.949	.719	.692	.811	.652	.519	.768	.593	.590
Manual synonym substitution	.828	.833	.859	.993	.978	.854	.818	.806	.798	.796	.703	.493	.573	.516	.894	.894	1.000	1.000	1.243	1.333	1.000	1.155	1.012	1.012	.813	.814	.663	.539	.722	.581	.847	.841
Manual paraphrasing	.746	.746	.774	1.00	1.00	.778	.903	.903	.809	.809	.811	.051	.137	.107	.468	.468	1.000	1.000	1.000	1.000	1.000	1.000	1.067	1.067	.776	.776	.792	.097	.241	.187	.588	.588

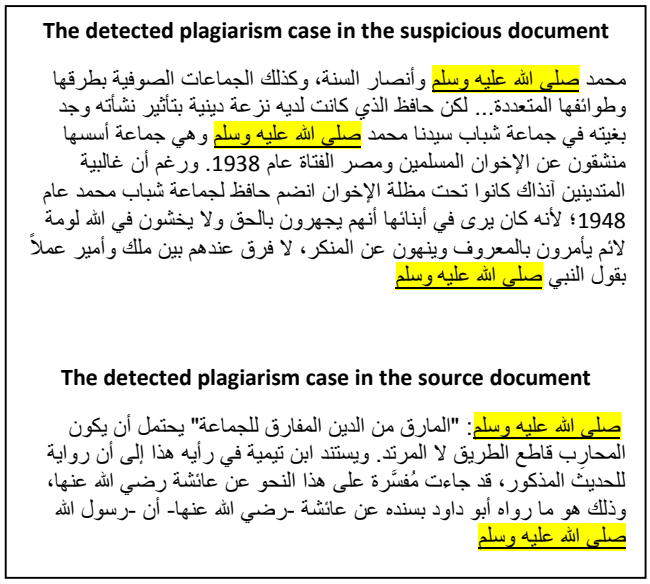


Figure III-4. Illustration of a false positive plagiarism case (detected by Palkovskii_2 method). It is obvious that this case has been detected because the common phrase "صلى الله عليه وسلم" ("may Allah honour him and grant him peace") has been used as a seed. The extension step produces a pair of passages that are completely different.

6 Intrinsic Plagiarism Detection Sub-task

Only one participant submitted a run to this sub-task. Following, we describe the corpus, the participant's method and its evaluation.

6.1 Corpus

Since the IPD is not concerned by finding the sources of plagiarism, the documents set from which the plagiarised excerpts are appropriated are omitted from the evaluation corpus of this approach. Thus, in this corpus, only the positions of plagiarism cases and their length in the suspicious document are annotated.

For AraPlagDet intrinsic plagiarism detection sub-task, we built a corpus that we called InAra. Table III-8 provides statistics of both training and test sub-corpora. Further information on the creation of this corpus is the subject of the following subsections.

6.1.1 Text Compilation

Criteria of Texts

Before collecting the texts, we set several criteria that should be verified in the host documents (documents where the plagiarised fragments are inserted).

Table III-8. Statistics of InAra corpus

		Training	Test
Documents		1024	1024
Cases		2833	2714
Plagiarism per document	Without plagiarism	20%	20%
	With plagiarism	80%	80 %
	Hardly (1%–20%)	37 %	35%
	Medium (20%–50%)	41%	41 %
	Much (50%–80%)	2%	5%
Document length	Short (< 10 pages)	75%	75%
	Medium (10 – 100 pages)	19%	19%
	Long (> 100 pages)	6%	6%
Case length	Very short (< 300 chars)	14%	15%
	Short (300–1k chars)	33%	34%
	Medium (1k–3k chars)	22%	23%
	Long (>3k chars)	31%	28%
Plagiarism type and obfuscation	Artificial without obfuscation	100%	100%

Criterion 1 Each host document must be written by one author only. If the document is multi-authorship, it may contain many writing styles, which could complicate the intrinsic plagiarism detection even further, i.e., if a writing style inconsistency is identified, it will be unclear whether this inconsistency is due to plagiarism or multi-authorship.

Criterion 2 The *author* and the *topic* of each text should be known. Those two pieces of information are important in the plagiarism insertion phase of the corpus creation (see Condition 1 in Section 6.1.2)

Criterion 3 Host documents should be diverse in terms of length but not too short. Indeed, based on a related research (Stein et al. 2011), we presume that the stylistic analysis becomes unreliable with the very short Arabic texts as it is with short English text (less than half a page approx.).

Criterion 4 Documents should not include many quotations. The purpose of this criterion is to avoid altering the evaluation by texts that are likely to be detected as plagiarism cases, although they are actually legitimate cases of text reuse.

Criterion 5 Texts should be punctuated because they will undergo a style analysis where the punctuation is an important feature[30]. This criterion seems obvious, but we decided to set it because, at a late stage of the text compilation, we noticed a lack of punctuation in some texts. This could be attributed to the poor editing or to the nature of Arabic text, which is known by its inconsistent use of punctuation (Alotaiby and Alkharashi 2007).

Source of Text

Since we planned to make the corpus publicly available, it was crucial to gather texts from a copyright-free source. For this reason and also because of the criteria mentioned above, sources of text have become very limited[31]. We finally decided to build our corpus from Arabic Wikisource, which is a library of heritage books and public domain texts. Apart from offering copyright-free texts, Wikisource was our choice as a source of text for the following reasons:

- Most of Wikisource texts are books. These are generally written by one author, which fulfils Criterion 1. Note that Wikipedia texts, for example, cannot be used since its texts stem from many contributors, which may lead to a mixture of styles in the same document.
- Most of Wikisource documents are tagged with topics and author names, which fulfils Criterion 2.
- Since the documents of Wikisource are books, it is possible to create suspicious documents in different lengths (which fulfils Criterion 3) by considering a whole book as one long document or by creating from it several documents of different lengths corresponding to its sections. This is not possible for example with news stories, which are generally limited in length.

We also added some texts from other sources, after making sure that they are without copyright. Table III-9 presents the sources of our document collection.

6.1.2 Insertion of Plagiarism

This step consists in extracting random fragments from the source documents and inserting them in random positions in the host documents. The operationalization of this step considers two conditions:

[30] For example, without the presence of the full stop character it becomes hard to compute the average sentence length.

[31] Note that chronologically speaking, we built InAra before ExAra. At the time of building InAra, we did not use the Corpus of Contemporary Arabic (that we used later to build ExAra) although it contains texts whose publishers have given permission to exploit them. This is because this corpus contains many texts from news wires that may have been edited by multiple journalists (Criterion 1 unsatisfied) or whose authors are unknown (Criterion 2 unsatisfied).

Table III-9. Sources of texts used to build InAra corpus

Source of text	Percentage of documents from this source in our corpus	URL
Arabic Wikisource	98%	https://ar.wikisource.org
Create your own country blog	2%	http://diycountry.blogspot.com
KSUCCA corpus (Alrabiah et al. 2013)		https://mahaalrabiah.wordpress.com/2012/07/20/king-saud-university-corpus-of-classical-arabic-ksucca
Islamic book web site		http://www.islamicbook.ws

Condition 1 The host document and the sources of the inserted plagiarism must have the same topic but different authors. This condition allows constructing a suspicious document with plagiarism cases that are different from the rest of the document in terms of its writing style (due to the difference of the author) but are not detectable with a topic analysis (due to the similarity of the topic).

Condition 2 The inserted cases should not be obfuscated to preserve their original writing style, which should be different from that of the host document as a consequence of Condition 1. In fact, it is unknown whether obfuscating plagiarism keeps or not the writing style difference between it (i.e., the plagiarism case) and the host document.

To generate suspicious documents with a variety in terms of the plagiarism proportion and the length of cases, we split the host documents into 6 sets according to the document lengths. Each set is divided arbitrarily into 4 equal subsets. Finally, plagiarism is inserted in each subset with a predefined percentage and various case lengths.

6.1.3 Difficulties

The creation of the InAra corpus was a challenging task mainly because of the lack of textual resources that fulfil all the required criteria of such a corpus. We checked several websites and blogs and concluded that the only resource that clearly declares that its texts are without copyrights and at the same time, it satisfies some of the required criteria is Wikisource. Notwithstanding, we have encountered several issues when we started to exploit it, which relatively slowed down the corpus building process. Following, we summarise the obstacles that we faced:

– Most of the texts in Arabic Wikisource are about religion. Many documents in this topic

are not relevant to build a corpus for intrinsic plagiarism detection because they comprise a lot of citations from the holy books, which may alter the study of the author writing style (a violation of Criterion 4 above). This drove us to filter out a big portion of the collected texts that contain a considerable number of quotations. We operationalized the identification of the quotations by utilising a simple heuristic that detects the excerpts between some quotation marks (e.g., “”, {}, etc.)[32]. Thus, eventually, it turned out that some concerned texts were not filtered out because they contain citations that are not bounded by quotation marks.

- Wikisource texts are very noisy. For the sake of the quality of its servers’ activities, Wikisource prevents downloading pages of books using crawlers. However, all its content is downloadable as a large XML file (a.k.a., a dump). Unfortunately, this file is very noisy meaning that it does not store books content only but also other pages such as users’ information, templates, Wikisource main page, help pages, etc. Moreover, books content itself is not clean. It contains the wiki markups, which are special tags and characters used to format the pages and to annotate them with metadata.
- The presentation and the organisation of texts lack coherence. Indeed, we noticed from the content of Wikisource pages that many contributors work with their own fashions and do not follow the text editing instructions. The following two points illustrate this incoherence.
  - Some pages are not linked to the books to which they belong, which make difficult the automatic association of these pages to the name of their author.
  - Some texts lack punctuation, which is important stylistic information as indicated in Criterion 5.

To mitigate the difficulties of extracting clean text from the Wikisource’s XML dump, we developed a tool for text cleaning and filtering (Bensalem et al. 2013b) that helped us to filter out irrelevant information and texts (such as, the wiki markups and the religious texts, respectively). Nonetheless, since also not religious books may contain quotations, and the content of Wikisource is not always well edited, we recognise that Criteria 4 and 5 have only been partially met.

6.2 Methods Description

6.2.1 Participant’s Method

In this section, we describe the method of Mahgoub et al. (2015), which is the only participant in the intrinsic plagiarism detection sub-task. Mahgoub et al. reported in their working notes that their method is similar to the one proposed by Zechner et al. (2009). It is based on computing the cosine distance between the Vector Space Model (VSM) of the suspicious document and the VSM of each chunk. Table III-10 describes the method according to the

[32] We determined the list of the quotation marks to use in our heuristic after the examination of some texts.

Table III-10. Description of Mahgoub et al.'s intrinsic method

Pre-processing
-
Document segmentation
Sliding window of 500 alphanumeric characters and a step of 250 characters.
Feature extraction
1. Frequency of Stop words 2. Frequency of Arabic punctuation marks 3. Frequency of Part Of Speech (POS) 4. Frequency of word classes
Plagiarised fragments identification
Cosine-distance-based heuristics that compares the document model with the fragments' models.
Post-processing
Merging adjacent chunks.

generic framework.

Language Dependence

It seems that the feature extraction is the most affected part by the language of the processed document. Three features extracted in Mahgoub et al.'s method are dependent on the language: it is obvious that any language has its approaches for POS tagging and its list of stop words. Moreover, being a right-to-left language, Arabic has some punctuation marks adapted to that, such as the comma (،) and the question mark (؟).

6.2.2 Baseline

We used a method based on character n-gram classes as features and naïve Bayes as a classification model. It is almost the same method described in (Bensalem et al. 2014a) (see Chapter V). This method is language-independent, and its performance scores are comparable to the ones of the best intrinsic plagiarism detection methods such as Oberreuter and Velásquez's (2013) and Stamatatos' (2009a) methods. The evaluation measures are the same used for external plagiarism detection (refer back to section 5.3.1).

6.3 Evaluation

6.3.1 Overall Results

As shown in Table III-11, Mahgoub et al.'s method performance is lower than the baseline.

Table III-11. Performance of the intrinsic plagiarism detection methods

Method	Precision	Recall	Granularity	Plagdet
Baseline	0.269	0.779	1.093	0.375
Mahgoub	0.188	0.198	1.000	0.193

This is in line with the performance of the original method (Zechner et al. 2009) that obtained a plagdet score of 0.177 on the PAN09 corpus (Potthast et al. 2009).

6.3.2 Detailed Results

Unlike the external approach, we presume that the performance of the intrinsic approach could be influenced by the document length and the percentage of plagiarism it incorporates. Table III-12 presents the performance of Mahgoub et al. and the baseline methods on the test corpus according to the aforementioned parameters in addition to the case length. The segmentation strategy of the baseline does not produce short chunks; therefore, the precision is not computed on the detected short cases. However, the actual short cases are detected with high recall. For both methods, the best performance is obtained on the medium cases, the short documents, and the documents with much plagiarism. Nonetheless, since we have only two methods, we cannot generalize any observed pattern.

7 Conclusion

In this chapter, we described the AraPlagDet shared task that we organised to evaluate plagiarism detection methods on Arabic texts. Participants were allowed to submit up to three runs in both the external and intrinsic plagiarism detection sub-tasks and a total of eight runs were finally submitted.

In the external plagiarism detection sub-task, most of the participants' methods were able to detect with high performance the plagiarism cases without obfuscation, whereas the obfuscated ones are quite challenging. This is consistent with the methods tested on PAN English corpora (Barrón-Cedeño et al. 2013b).

As for the intrinsic plagiarism detection, AraPlagDet shared task confirms that this problem is very challenging regardless of the language of the documents, and hence it needs more attention from researchers.

Besides, the AraPlagDet shared task taught us that considering the peculiarity of the Arabic language texts that is to quote commonly from Quran is essential to avoid the false positive detections. In this context, we recommend to include in the Arabic plagiarism detection tools

Table III-12. Detailed performance of the intrinsic plagiarism detection methods

		Precision		Recall		Granularity		Plagdet	
		Mahgoub	Baseline	Mahgoub	Baseline	Mahgoub	Baseline	Mahgoub	Baseline
Case length	very short	0.119	-	0.192	**0.759**	**1.000**	**1.000**	**0.147**	-
	short	**0.129**	0.121	0.179	**0.858**	**1.000**	**1.000**	0.150	**0.212**
	medium	**0.231**	0.223	0.215	**0.876**	**1.000**	1.007	0.223	**0.353**
	long	0.200	**0.283**	0.215	**0.672**	**1.000**	1.161	0.207	**0.358**
	very long	0.159	**0.361**	0.175	**0.301**	**1.000**	2.590	0.166	**0.178**
Document length	very short	0.000	**0.033**	0.000	**1.000**	-	**1.000**	-	**0.064**
	short	0.197	**0.221**	0.191	**0.944**	**1.000**	1.006	0.194	**0.356**
	medium	0.163	**0.228**	0.197	**0.764**	**1.000**	1.151	0.179	**0.318**
	long	0.159	**0.387**	0.221	**0.255**	**1.000**	1.591	0.185	**0.224**
Plagiarism per document	hardly	0.082	**0.158**	0.178	**0.783**	**1.000**	1.050	0.112	**0.254**
	medium	0.329	**0.445**	0.206	**0.761**	**1.000**	1.118	0.253	**0.518**
	much	0.495	**0.571**	0.219	**0.913**	**1.000**	1.079	0.303	**0.665**

a module that detects and filters out the Quranic citations. Such a module can rely on the external approach, whereby the whole text of the document is compared to the Quran corpus to identify the citations. Alternatively, it can rely on the intrinsic approach, whereby the Quranic citations are detected based on their peculiar style in comparison with the ordinary text[33]. See the work of (Meskaldji et al. 2018), which is a preliminary attempt toward the latter direction.

In a more general context, which is text reuse detection, recent efforts revealed a big quantity of reused texts in the old Arabic books (Belinkov et al. 2019). The reused texts include Quranic verses, sayings of the prophet Muhammad (Hadith) and common anecdotes. It would be interesting to develop evaluation frameworks to assess the performance of models in detecting these kinds of reused texts.

The major gain of the AraPlagDet shared task is the evaluation corpora that we built

[33] Hadith (i.e., the sayings of the prophet Muhammad) is also a common source of quotations in the Arabic text, and it would be interesting also for a plagiarism detection software to identify it to avoid false positives. However, in contrast to Quran, which has only one book, there are many books of Hadith, and sometimes one saying may have several versions spread over various sources. This fact may reduce the efficiency of the detection of Hadith by the text matching techniques, which is a motivation to try the intrinsic approach to uncover this kind of citations.

following PAN standards (in the annotation) and made available online to the research community[34]. It is our hope that these corpora will serve as a benchmark to assess the performance of future plagiarism detection tools on Arabic documents. Indeed, the ExAra corpus has been already used after the competition to evaluate other methods (Ghanem et al. 2018; Nagoudi et al. 2018c, 2018a) and a commercial tool (almikshaf.com)[35]. Additionally, it served to study the effect of stemming on plagiarism detection performance (Boukhalfa et al. 2018).

Future work in the context of Arabic plagiarism detection evaluation may include the construction of corpora comprising plagiarism cases translated from other languages, most notably, because some methods addressing English-Arabic semantic text similarity have been proposed recently (Alzahrani 2016; Nagoudi et al. 2018b), and it would be interesting to test them in the context of detecting cross-language plagiarism. What is also needed (as already discussed above) is an evaluation corpus involving annotated citations to allow the assessment of the detectors' behaviour with this kind of text reuse.

With regard to the methods, since research on the external plagiarism detection technologies is already mature, we think that instead of developing methods from scratch, researchers interested in Arabic processing should focus on adapting the state-of-the-art methods to this language and studying the effect of this adaptation on performance. This is because although we believe in the importance of language-dependent processing, it is still unknown to what extent this processing could improve the performance. More precisely, some questions need to be answered in this context, which are:

- To which extent can we rely on a plagiarism detector designed for English to detect plagiarism on Arabic documents?
- Is adapting an existing tool to Arabic worth the effort in terms of improving significantly the performance?
- Do language-dependent solutions perform better than the language-independent ones?

Providing answers to these questions would be important especially for practitioners.

Concerning the detection of the obfuscated plagiarism, we believe that techniques of *paraphrase identification* and *semantic textual similarity* (AL-Smadi et al. 2017; Cer et al. 2017) can improve uncovering such cases. Using these techniques, a recent work (Nagoudi et al. 2018c) reported an improvement of Arabic plagiarism detection performance in comparison with the results obtained by using the fingerprinting alone. However, in that study, the performance was measured on the whole ExAra corpus and not only on the documents

[34] The evaluation corpora can be downloaded from the webpages of AraPlagDet shared task: http://misc-umc.org/AraPlagDet or https://pan.webis.de/fire15/pan15-web/araplagdet.html.
[35] According to a personal communication with Dr. Boubaker Kahloula, the developer of the tool.

containing the obfuscated plagiarism. Thus, it is still unknown whether the reported improvement concerns the paraphrased plagiarism specifically. Therefore, further studies in this direction can be the subject of future work.

Chapter IV. Intrinsic Plagiarism Detection: a Survey

> *Your voice is like a record/album/CD. It is consistent and steady and memorable. (Listeners don't ever mix up Beyoncé with Madonna or Drake with Elton John.) Your voice will flow through everything you write and should be distinctive and recognizable.*
>
> Daphne Gray-Grant[1]

1 Introduction

As discussed in the previous chapters of this thesis, there exist different approaches to detecting plagiarism automatically. The most common one is the so-called *external plagiarism detection* (EPD) (Potthast et al. 2009), which is the process of comparing the suspicious document with the potential sources of plagiarism to seek similar passages. Another approach is *citation-based plagiarism detection* (CPD) (Gipp et al. 2011; Pertile et al. 2015), which is based on the assumption that the plagiarist uses patterns in citing references similar to the ones used in the source of plagiarism. This approach is applicable only to the documents that contain in-text citations and a bibliography section such as academic papers. Plagiarism could also be detected by *authorship verification* (AV). In this approach, the writing style of the whole suspicious document (van Halteren 2004) or its fragments (Burn-Thornton and Burman 2015) is compared to the one of a pre-compiled set of documents that are, unquestionably, written by the suspicious author. If the writing style of the suspicious document is not the same as the

[1] From her blog *What is writer's voice? And how can you find yours?* (https://www.publicationcoach.com/what-is-writers-voice)

set of documents, then it is likely that the actual author of the suspicious document is not the supposed one. This approach could be deployed in schools or e-learning environments where it might be feasible to create a database of students' essays and then use it to build a writing style model for each student. Afterwards, these models are used to check the authorship of students' subsequent submissions (van Halteren 2003). See (López-Monroy et al. 2020) for further information on the deployment of an authorship verification solution for e-learning.

As is clear from the descriptions above, the availability of external resources, which are either plagiarism sources (in the case of EPD and CPD) or a corpus of texts stemmed from the suspicious author (in the case of AV), is a crucial condition of the usability of the three approaches above. However, the fourth approach, namely *intrinsic plagiarism detection* (IPD), which is the focus of this chapter, requires none of the resources mentioned above. It is based rather on analysing the suspicious document solely with the aim to find internal evidence of plagiarism. Practically, this approach marks as plagiarised the passages whose writing style does not blend in with the whole document. Examples of such a case include a sloppily written paper involving some impressive sections, or a thesis where there exist, between its chapters, wide stylistic discrepancies.

Intrinsic plagiarism detection has been a subject of research since the early 2000s. Since then, a number of methods have been developed, particularly, during the last decade that has witnessed the organisation of shared tasks to address this research problem.

Figure IV-1 exhibits some important events related to intrinsic plagiarism detection.

Although research on IPD is two decades old, there is no publication surveying the body of knowledge accumulated on this subject during this period. Nonetheless, it is worth to mention that Stein et al. (2011) are the first to publish a long research paper on IPD explaining in detail the nature of the problem and its formal relationship with authorship attribution, authorship verification and external plagiarism detection. They also established a general structure of solutions to this problem and suggested a set of techniques that could be used in each step. In fact, their paper is a seminal reference; however, because of the lack of a considerable amount of IPD literature at the time of their publication, most of the cited works came from related domains such as authorship analysis. This chapter seeks to remedy the lack of survey papers on this subject by providing the reader with the first comprehensive review of the existing intrinsic plagiarism detection work.

Our survey focuses on the structural and the functional aspects of IPD methods. For this purpose, we reviewed more than 25 IPD methods published over the last two decades with special attention to understanding their different building blocks. As for the effectiveness aspect, we do not dwell much on it but only provide a brief overview aiming to point out the difficulty of the task. For more details on this aspect, we refer the reader to the overview papers of PAN shared tasks (Juola 2012; Potthast et al. 2010a, 2011; Stamatatos 2016) where many methods are compared in terms of their performance.

2001-2004	First Bachelor/ Master dissertations on detecting plagiarism based on the writing style changes	• (Hersee 2001) • (Kimler 2003) • (Bonsall 2004)
2003	Preliminary experiments on detecting plagiarism on the basis of the writing style changes	• (Clough 2003)
2006-2007	The first conference papers on IPD	• (Meyer zu Eißen and Stein 2006) • (Meyer zu Eißen et al. 2007) • (Stein and Meyer zu Eißen 2007)
2009-2011	PAN@CLEF plagiarism detection shared task with a sub-task for IPD	• (Potthast et al. 2009) • (Potthast et al. 2010a) • (Potthast et al. 2011)
2011	The first journal paper on IPD	• (Stein et al. 2011)
2012	PAN@CLEF Authorship Attribution shared task with a sub-task on Authorship clustering/IPD	• (Juola 2012)
2014	The first PhD thesis on IPD	• (Tschuggnall 2014)
2015	PAN@FIRE Arabic plagiarism detection shared task with a sub-task for IPD	• (Bensalem et al. 2015)
2016-2019	PAN shared tasks on new problems related to IPD	• **Author Diarization:** (Rosso et al. 2016; Stamatatos et al. 2016) • **Style breach detection:** (Tschuggnall et al. 2017) • **Style change detection:** (Kestemont et al. 2018; Zangerle et al. 2019)

Figure IV-1. Timeline of some milestones related to intrinsic plagiarism detection

It should be noted that we examined for this survey only the methods that concern the IPD task with its classic definition (unless mentioned otherwise). Other related tasks such as the multi-authorship document segmentation and style change detection (Kestemont et al. 2018; Tschuggnall et al. 2017), which could be seen as variants of the IPD problem with relaxed or additional constraints, are defined in Section 3. However, the examination of the building blocks of these tasks' methods is outside the scope of this survey.

The rest of this chapter is organised as follows. Section 2 sheds light on the use cases of intrinsic plagiarism detection. Section 3 outlines the research areas related to IPD. Section 4 surveys and analyses thoroughly the building blocks of the existing IPD methods. Next, a general overview of the performance of IPD methods is provided in Section 5. Finally, Section 6 concludes the chapter.

2 Use Cases

Literature on intrinsic plagiarism detection associates the use of this approach with the two following cases:

Use case 1 Using IPD as an alternative to EPD when the source of plagiarism cannot be found (Kasprzak and Brandejs 2010; Muhr et al. 2010). According to (Weber-Wulff 2015), failure to spot some plagiarism cases is a "frequently encountered problem" in the current plagiarism detection software. This happens, for example, when the plagiarist borrows the text from a non-digitalised reference, asks someone else to write for him parts of his work, or succeeds in defeating the external plagiarism detector by substituting, in the stolen text, some characters by their similar-looking ones from foreign alphabets[2] (e.g., replacing some Latin letters with Cyrillic ones or Arabic letter with Persian ones)[3]. Another circumstance could be added to this scenario, and perhaps it makes IPD the only way to try to uncover plagiarism, is the absence of other texts of the suspicious author; otherwise, detecting plagiarism could become an authorship verification problem.

Use case 2 Using IPD as a step in the source retrieval phase of the external plagiarism detection. The idea is to search for plagiarism sources by querying the search engine with texts extracted from the passages detected by an intrinsic method (Bru et al. 2011; Knight et al. 2004; Suchomel et al. 2012). The goal is to reduce the computational cost of the comparison between the suspicious document and the potential sources of plagiarism.

Since it is based on detecting stylistic changes, the current techniques of the intrinsic approach show their limitation and fail to unveil plagiarism when the suspicious document is

[2] External plagiarism detection tools could be defeated also by paraphrasing plagiarism (Sánchez-Vega et al. 2019).
[3] See (Heather 2010) and (Gillam et al. 2011) for further information on this kind of cheating.

written entirely in one style. This case happens, for instance, when a plagiarist buys an essay from a paper mill.

3 Similar Research Areas

In this section, we present some research areas that are closely related to intrinsic plagiarism detection (see Table IV-1 for a brief description). More precisely, we identify what makes these research areas similar to IPD.

3.1 Anomaly Detection

Intrinsic plagiarism detection in its essence could be seen as an anomaly-of-authorship detection at fragment level (Guthrie et al. 2007), where plagiarism is the anomaly, and the text written in the plagiarist's own style is the normal part. In fact, most of the current IPD methods deal with IPD as an anomaly detection problem. That is, they are based on the assumption that the normal data (original part) is the majority, and hence can be characterised, and the abnormal data (plagiarised part) is sparse and thus difficult to characterise. Therefore, methods based on this assumption build a writing style model of the whole suspicious document, and consider as plagiarism any fragment deviating from this general style (Mahgoub et al. 2015; Muhr et al. 2010; Oberreuter and Velásquez 2013; Stamatatos 2009a; Suárez et al. 2010; Zechner et al. 2009).

The major drawback of this perception emerges when the plagiarism constitutes the

Table IV-1 Intrinsic plagiarism detection and its related research areas

Intrinsic plagiarism detection	Given a suspicious textual document *d*, which passages are plagiarised without comparing *d* to potential sources of plagiarism?
Anomaly detection	Given a textual document *d*, which passages are outliers?
Multi-author document segmentation	Given a textual document *d* of unknown authorship and a number *N* of authors, which passages are written by each author?
Authorship verification	Given a textual document *d* and a set of textual documents *D* written by an author A; is A the author of *d*?
Plagiarism direction identification	Given two textual documents that share one or more passages, which of them is the source, and which one is the suspicious?
Linear text segmentation	Given a textual document, what are the positions that represent a topic change?
Speaker diarization	Given an audio or video recording that encompasses an unknown number of speakers, who spoke when?

majority of the suspicious document. In this case, the model built from the suspicious document becomes unreliable to represent the suspicious author's own style. In addition to that, if the source of plagiarism is only one (i.e., the plagiarised fragments are written in one style, and they constitute the majority of the document), a solution based on anomaly detection would mark the plagiarised part as original.

3.2 Multi-author Document Segmentation

This task consists of clustering/classifying the passages of a multi-author document according to authorship (Akiva and Koppel 2013; Aldebei et al. 2015, 2016; Giannella 2016; Glover and Hirst 1996; Graham et al. 2005; Koppel et al. 2011; Tschuggnall and Specht 2014). This should be done without employing any external text, which makes this problem very similar to IPD. It can be stated that three factors control the definition of this problem:

- Whether the number of authors, N, is known.
- Whether the mono-authorship (N = 1) is a possible case.
- Whether the document is already segmented so that each segment is written by one author.

The combination of the possible configurations yields distinct scenarios of this problem with different levels of complexity. The least complex scenario is: "*N is known, and the document is already segmented*[4]" (Akiva 2012; Brooke and Hirst 2012; Kern et al. 2012). Hence, the task is merely to group segments written by each author. The most complex scenario is: "*N is unknown, it could be equal to 1, and the document is not segmented*". In this case, the task should involve the prediction of N, the identification of the style shift positions, and the aggregation of fragments of similar style. Indeed, checking the existence of plagiarism in a document could be viewed as checking whether it is multi-author without possessing any information about the number of authors and the possible positions of writing style shift. Therefore, intrinsic plagiarism detection could be perceived as an authorship segmentation problem in its most complex scenario. In addition, IPD methods should decide which among the identified authorial parts have been plagiarised. On the other hand, if the number of detected authors is one, the document should be marked as plagiarism-free.

3.3 Authorship Verification

Given a document *d* of unknown authorship and a set of documents D written by an author A, the authorship verification task is to check whether A is the author of *d*. To this aim, *d* and D must be compared in terms of writing style. As suggested by Stein et al. (2011), intrinsic plagiarism detection problem is constituted of many instances of the authorship verification problem. To explain, in an IPD problem, (a) the question is to *verify the authorship* of a *set of*

[4] For example, it might be known that each paragraph is written by one author and there would be no need to look for style shift at sentence level.

passages obtained via segmentation of the suspicious document. (b) The whole suspicious document itself represents the text against which the writing style of passages is compared. Although they are closely related, IPD and AV are different because of two reasons. First, intrinsic plagiarism detection has to manipulate shorter texts (i.e., the document's fragments), which makes the quantification of writing style more difficult. Second, the suspicious document (that plays the same role as the set of documents of known authorship in AV) is mingled with plagiarism. Thus, it does not represent the alleged author's style faithfully as it is supposed to do.

3.4 Plagiarism Direction Identification

Given two documents that share one or more text fragments, this task is to determine which of them is the source and which one is suspicious. The proposed solutions to this problem (Grozea and Popescu 2010; Shrestha and Solorio 2015) are based on the idea that the plagiarised passage is more similar to the rest of the text in the source document than it is in the suspicious document. Thus, it is a matter of determining, for each document, whether the shared text fragment is an outlier, as done in intrinsic plagiarism detection.

3.5 Linear Text Segmentation

This task aims to segment the document into blocks according to topics so that the topical similarity is high between the sentences of the same block but low between the sentences of different blocks (Kern and Granitzer 2009). If the segmentation criterion is the writing style instead of the topic, the output will be the positions of the writing style shift. In this case, this task could be viewed as a segmentation module in intrinsic plagiarism detection. Recently, a shared task has been organised at the PAN Lab in 2017 to address this research direction (Tschuggnall et al. 2017).

3.6 Speaker Diarization

The research problems described above concern the textual documents. Recently, researchers noticed that intrinsic plagiarism detection is similar to speaker diarization[5] (Rosso et al. 2016; Stamatatos et al. 2016). This research problem that is closely related to speaker recognition[6] concerns the identification of the different speakers in an audio or video recording (Anguera et al. 2012), which is similar, in its principle, to the identification of the different authors in a textual document. Speaker diarization, in turn, is closely related, notably regarding techniques, to the problem of time series segmentation (Keogh et al. 2004).

[5] A shared task named Author Diarization has been organised at the PAN Lab in 2016 (http://pan.webis.de/clef16/pan16-web/author-identification.html). It involves three subtasks: traditional intrinsic plagiarism detection, diarization with a given number of authors, and diarization with an unknown number of authors.

[6] Speaker recognition is the identification of a person from her/her voice.

4 Building Blocks

Intrinsic plagiarism detection methods are composed of several heuristics that can be organised into 5 main building blocks (Stein et al. 2011), which we illustrate in Figure IV-2, and we delineate in the following subsections.

4.1 Pre-processing

The pre-processing heuristics are called so because they operate before the fragment-level analysis. These heuristics aim to *filter out the irrelevant information* that may disrupt the style analysis (through cleaning, normalisation and genre analysis) or *reduce the computation* by taking an early decision on the document (through checking whether the document is

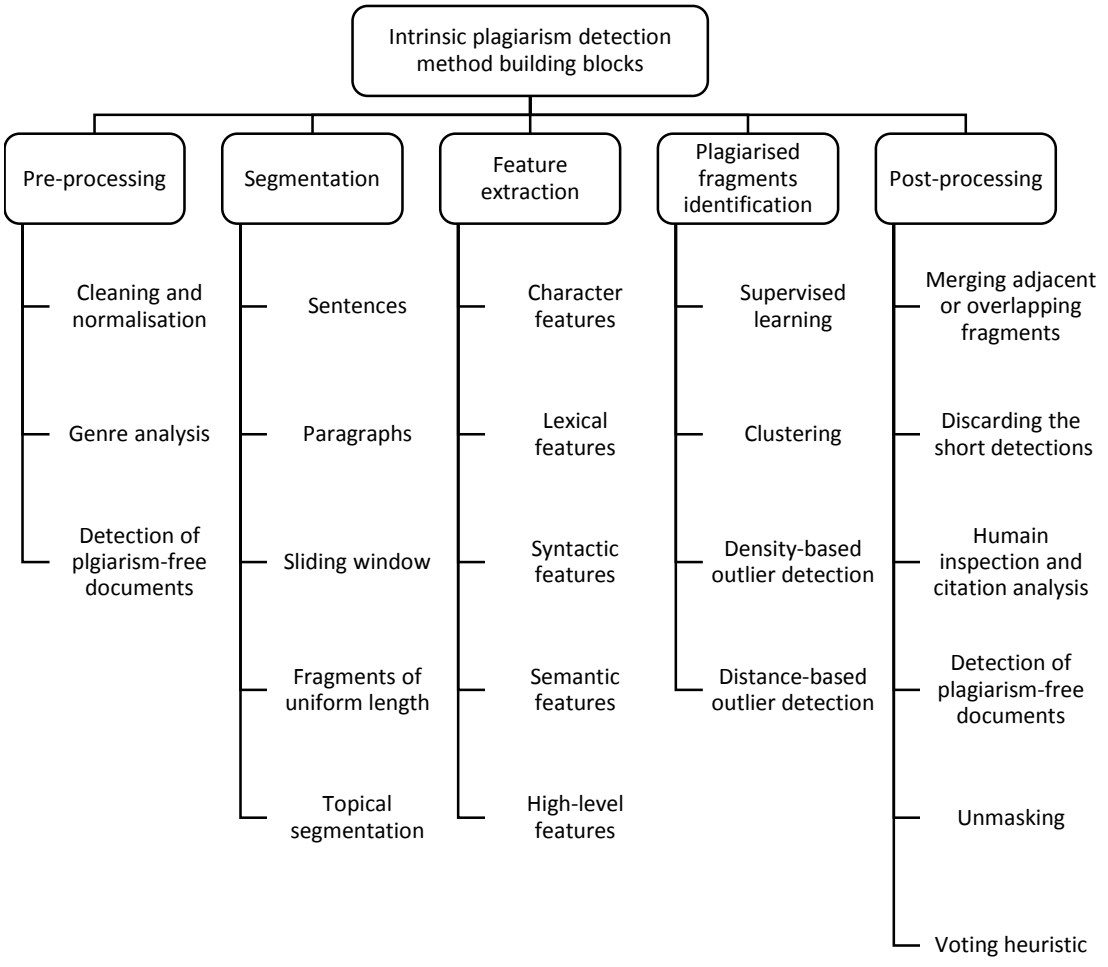


Figure IV-2. Taxonomy of the building blocks of intrinsic plagiarism detection methods

plagiarism-free). The subsections below describe these heuristics.

4.1.1 Cleaning and Normalisation

Those are the most used pre-processing heuristics in the existing IPD methods. Cleaning the text comprises removing non-alphabetic characters such as numerals, while normalisation is usually applied by lowercasing the text and, in some few methods, by stemming it. The goal is to eliminate the information of less importance in analysing the writing style. Hence, the selection of the textual information to discard or normalise should be done carefully because it may negatively influence the stylistic analysis. For example, punctuation and function words may be irrelevant information for external plagiarism detection but do not have to be eliminated in intrinsic plagiarism detection, as they are stylistic features.

4.1.2 Genre Analysis

Stein et al. (2011) propose heuristics in this context, namely document length analysis, genre analysis and analysis of issuing institution. The authors expect that the position of the plagiarised passages in the document and their length depend on that information. To illustrate what Stein et al. mean, let us take the example of analysing a thesis. Here, it would be useful to make an intra-document genre analysis to discard the parts of specific sub-genres where the analysis of the author's style makes little sense, such as the table of content and the reference list.

In fact, up to now, no method except the one of Stamatatos (2009a) considers the sub-genres that may contain the suspicious document[7]. The author proposes a simple heuristic that eliminates the fragments containing a certain proportion of non-alphabetic characters[8]. And based on empirical observations, he suggests that these fragments represent indexes and tables of content since the number of digits and spaces in these specific sections is more than in the ordinary text.

4.1.3 Detection of Plagiarism-free Document

This heuristic checks the existence of plagiarism in the document without identifying the positions thereof. Consequently, the plagiarism-free documents will not undergo further analysis that determines the boundaries of the plagiarised fragments. Note that it is possible to run this heuristic after the segmentation and the feature extraction, yet it should be before the plagiarism identification at the fragment level to be considered as pre-processing.

One implementation of this heuristic, which is used in (Stamatatos 2009a), considers a

[7] Stein et al. (2011) suggest the use of this kind of pre-processing heuristics but did not use them in the method they proposed.

[8] Practically, the actual length of a fragment is compared with its length after deleting the non-alphabetic characters.

Table IV-2. Pre-processing heuristics in intrinsic plagiarism detection methods

Heuristics	IPD Methods
Cleaning and normalisation	(Stamatatos 2009a) (Kern et al. 2012) (Kestemont et al. 2011) (Oberreuter et al. 2011b; Oberreuter and Velásquez 2013) (Tschuggnall and Specht 2013c, 2013a) (Kuta and Kitowski 2014) (Mahgoub et al. 2015) (Polydouri et al. 2017, 2018)
Genre analysis	(Stamatatos 2009a)
Detection of plagiarism-free document heuristics	(Stamatatos 2009a)

document as plagiarism-free if the variance of the style change function is not significant. Practically speaking, this implementation checks the significance of the style variance by comparing the standard deviation δ of the style change function to a predefined threshold τ_δ. If $\delta < \tau_\delta$ then the heuristic marks the document as plagiarism-free.

Note that investigating whether a document is plagiarised without determining the position of plagiarism can be thought of as a separate research problem, and in fact, this is a new direction to relax the IPD task. Recently, a shared task at the PAN Lab (Kestemont et al. 2018) has been dedicated to addressing this problem, meaning that the output of the methods is not the list of the plagiarised fragments but a binary result indicating whether a document is written in one style or a mixture of different styles. The performance results are promising, which confirms that this task is less challenging than the fragment-level IPD.

Table IV-2 lists the IPD methods that use the aforementioned pre-processing heuristics. Note that the papers that did not provide sufficient information on this step were not in the table.

4.2 Segmentation

Unlike other style analysis based applications such as authorship attribution (Stamatatos 2009a) and authorship verification (Koppel and Seidman 2013), where the writing style model is retrieved from the whole document, in the IPD methods, a document is perceived as a set of fragments; for each of them the writing style needs to be analysed individually. These fragments result from applying a decomposition strategy that distinguishes the natural structure of the text (sentences, paragraphs), segments the text into fragments of uniform lengths or segments the text according to other criteria such as the topic.

The segmentation is crucial in intrinsic plagiarism detection. Indeed, a granular segmentation may lead to an unreliable style analysis due to the difficulty of characterising the writing style of a short text. On the other hand, a coarse segmentation may prevent the

identification of the short plagiarised texts. In some works such as (Kuta and Kitowski 2014), the authors were interested in optimising the length of the fragments, and they show that the optimal length changes according to the length of the extracted character n-grams, which are the only used features in their experiments. Similarly, in (Zechner et al. 2009), it has been shown that the optimal fragment size varies according to the extracted feature. For instance, they reported that the optimal fragment length for the punctuation features differs from that for the part-of-speech tags features. Kestemont et al. (2011) show that the shorter the fragment, the higher the precision of the method.

As shown in Table IV-3, the most used segmentation heuristic is the sliding window, which is a special case of the uniform-length segmentation. The sliding window is characterised by two parameters: the size and the step. The size represents the length of each fragment, whereas the step corresponds to the interval between the beginning of two contiguous fragments. In other words, it is the length of text by which the window moves to extract the next fragment. If the window step is smaller than its size, this produces overlapping fragments.

4.3 Feature Extraction

The text is unstructured information, and to process it automatically, it is essential to represent it in a structured manner. Feature extraction in natural language processing (NLP) structures the text by converting it to a vector representation whose dimension is the number of features.

Table IV-3. Segmentation strategies used in intrinsic plagiarism detection methods

Sentences	(Zechner et al. 2009) (Seaward and Matwin 2009) (Suárez et al. 2010) (Tschuggnall and Specht 2012, 2013b) (Tschuggnall and Specht 2013a) (Kuznetsov et al. 2016) (Sittar et al. 2016)	
Paragraphs	(Meyer zu Eißen et al. 2007; Meyer zu Eißen and Stein 2006) (Seaward and Matwin 2009) (Suárez et al. 2010) (Kern et al. 2012) (Carnahan et al. 2014)	
Sliding window	of characters	(Stamatatos 2009a) (Kasprzak and Brandejs 2010) (Kestemont et al. 2011) (Rao et al. 2011) (Mahgoub et al. 2015)
	of words	(Oberreuter et al. 2011a; Oberreuter and Velásquez 2013)
	of sentences	(Zechner et al. 2009) (Tschuggnall and Specht 2013c) (Polydouri et al. 2017)
Uniform length	(Stein et al. 2011) (Akiva 2011) (Carnahan et al. 2014)	
Topical segmentation	(Muhr et al. 2010)	

Each feature value quantifies a characteristic of the text. In intrinsic plagiarism detection, the quantified characteristics concern usually the writing style.

Since the goal of IPD is to detect the writing style changes between the document's fragments, features are extracted at the fragment level. Nevertheless, some methods extract the features from the whole document as well, and considers the obtained feature vector as a model of the dominant writing style in the document. Note that if the same features are extracted from all the fragments, then the feature extraction yields a matrix where the number of rows is the number of fragments and the number of columns is the number of features (see Figure IV-3).

There is a large volume of published literature describing and classifying the stylistic features (see for example the references (Abbasi and Chen 2008; Holmes 1994; Juola 2006; Koppel et al. 2009; Stamatatos 2009b)). However, much of what we know about the stylistic features are based upon authorship attribution studies. Even Stein et al. (2011), in their well-known article on intrinsic plagiarism detection, surveyed the stylistic features used for authorship attribution and verification, as only few IPD methods existed at that time. Thus, in the next subsections, it is our contribution to survey specifically the features utilised in intrinsic plagiarism detection methods.

4.3.1 Character Features

To extract these features, the text is perceived as a sequence of characters. We classify the character features according to two criteria which are the *kind* and *meaningfulness* of the unit from which the value of the feature is computed. Concerning the first criteria, the character feature is typically the frequency or the number of occurrences of one of the following *three kinds of units*: (i) one character, (ii) a sequence of characters, or (iii) a class of characters. See,

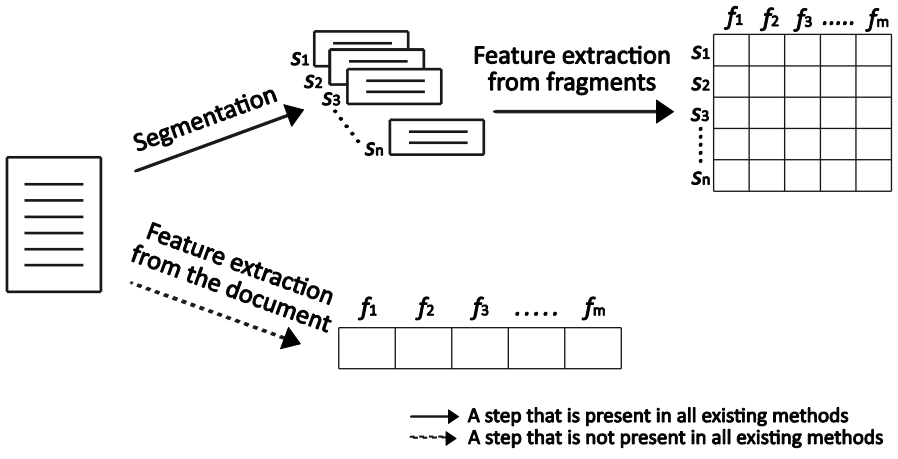


Figure IV-3. Feature extraction at fragment and document levels. The symbols $s_1, \ldots, s_n$ denote the fragments and $f_1, \ldots, f_m$ denote the features.

Table IV-4. The units from which the character features are extracted with examples extracted from a sentence

The sentence			
In character, in manner, in style, in all things, the supreme excellence is SIMPLICITY[9]			
Kinds of the units from which the feature is extracted		**Example of features**	
One character		Comma count:	4
Sequence of characters	Character n-grams	Count of the 5-gram , in :	3
	Character sound n-grams	Count of the 3-gram ***consonant-consonant-consonant*:**	2
	Affixes	Count of the suffix –**icity:**	1
Class of characters		Uppercase characters count:	11

in Table IV-4, an exemplification of these three kinds of features extracted from a sentence.

Concerning the second criteria, which is *the meaningfulness of the units* from which the features are computed, we classify character features depending on whether the unit is defined by considering or not the sense of the characters or any meaningful relationship between them. In this classification, we distinguish two categories: *character n-grams* and *linguistic units*.

Character n-grams are sequences of contiguous characters of a predefined length extracted from the text *without considering any linguistic relationship between them*. Despite their simplicity, these features have proven their effectiveness in many NLP applications, such as authorship attribution (Cavnar and Trenkle 1994; Stamatatos 2016), native language identification (Kulmizev et al. 2017) and opinion spam detection (Hernández Fusilier et al. 2015). Based on their reputability as stylistic markers notably for authorship attribution, they have been employed in intrinsic plagiarism detection. As a matter of fact, Stamatatos (2009a) was the first to develop a character-n-grams-based IPD method. Although it utilises only these language-independent features, Stamatatos' method was ranked first in the PAN 2009 intrinsic plagiarism detection shared task. This seminal method, by its simplicity, inspired other researchers, who reproduced it partially or fully in their works (Kasprzak and Brandejs 2010; Kestemont et al. 2011; Kuta and Kitowski 2014; Rao et al. 2011).

Instead of extracting the n-grams from the characters themselves, a couple of IPD methods (see (Carnahan et al. 2014; Stein et al. 2011, Table 4)) extract them from the sounds of characters. That is, n-grams are identified after representing each character by its sound: vowel (V) or consonant (C). For example, 'CVCV' is the representation of the word 'mama' by the sounds of its characters. Therefore, the 2-grams of character sounds that we can extract from

[9] Quotes of the poet Henry Wadsworth Longfellow.

this word are {CV, VC}. See another example in Table IV-4.

Character features of the category *linguistic units* involve properties of short strings that are *linguistically meaningful* but cannot be deemed standalone words. Features of this category that has been used in IPD methods include the frequency of suffixes (Kern et al. 2012; Muhr et al. 2010; Rao et al. 2011) and of some meaningful single characters, such as the space character (Sittar et al. 2016), each of the punctuation signs[10] (Kern et al. 2012; Kuznetsov et al. 2016; Mahgoub et al. 2015; Rao et al. 2011; Zechner et al. 2009), and digits (Kern et al. 2012).

4.3.2 Lexical Features

Tokenising the text into words is the primary operation in extracting the lexical features. Some features represent merely *statistics* computed on the words. Other features consist in assigning a score to each individual word. This score may represent, e.g., the frequency of the word in the processed text, and the obtained vector of words' scores is called the *vector space model* (VSM).

Other kinds of lexical features are extracted by computing the *ratio of a class of words* created according to a certain criterion such as their frequency ranges. The more sophisticated lexical features combine the aforementioned relatively basic features to produce features that indicate the *complexity/readability* of the text or the *vocabulary richness* of its author. The next four subsections provide details on each of the mentioned lexical features categories.

Vector Space Model

The chosen words reflect the preference of the author, meaning that representing the text as a vector of words is plausible for stylistic analysis. However, not all the words are equal in terms of their importance as writing style markers. Some words –more precisely, the so-called *content words*– reflect rather the topic of the text, and hence they could be discarded to avoid confusing the writing style analysis (Stamatatos 2018). In this context, various researches demonstrated that the VSM of *function words* is among the best representations to help identifying the author of a text. Function words are closed-class[11] words such as determiners, prepositions and pronouns. They are characterised by their high frequency in the text, and unlike the content words, they do not convey semantic information; their role is rather grammatical.

The IPD methods that use VSM of function words[12] frequency are (Meyer zu Eißen et al. 2007; Meyer zu Eißen and Stein 2006) (Stein and Meyer zu Eißen 2007) (Zechner et al. 2009)

[10] For several decades, punctuation has been admissible in USA courts as evidence for the identification of the writer of a disputed-authorship document (Chaski 1999).

[11] Unlike open-class words –such as verbs and adjectives– that constitute most of the vocabulary, closed-class words are limited in number, such as pronouns and articles. See (Manning and Schütze 1999, Chapter Linguistic essentials p. 82) for further details.

[12] Some authors call the function words the stop words as their name in Information Retrieval.

(Muhr et al. 2010) (Rao et al. 2011) (Kern et al. 2012) (Carnahan et al. 2014) and (Mahgoub et al. 2015).

Unlike the above methods that consider the function words only, Oberreuter et al.'s (2011b) method exploits all the words of the suspicious document to compute a style change function. Moreover, it is the only IPD method so far that represents the text by the frequency of words without including other features, and it ranked first at the PAN 2011 plagiarism detection shared task. However, since the content words are also part of this method's feature space, it has been criticised for benefiting from the possible difference between the topics of the plagiarised fragments and the suspicious document that host them, which is caused by the automatic creation of the evaluation corpus (Potthast et al. 2011). This means it is likely that this approach leads to detect the outlier fragments because they are distinct from the rest of the document in terms of their topic and not their writing style.

Basic Statistics

In addition to the frequency, other properties related to the words can be extracted such as their average length and their average number per sentences[13].

The *average word length* and the *average sentence length* are classical stylometric features and are used in several intrinsic plagiarism detection methods such as (Meyer zu Eißen et al. 2007; Meyer zu Eißen and Stein 2006) (Stein and Meyer zu Eißen 2007) (Stein et al. 2011) (Rao et al. 2011) (Carnahan et al. 2014) (Kern et al. 2012) (Kuznetsov et al. 2016) (Polydouri et al. 2017) (Sittar et al. 2016).

Readability and Complexity

These measures are computed using formulae that employ generally basic statistics on words or characters such as the word length, sentence length, and syllable count per word to generate a score that represents a metric of the easiness/complexity of the text or the educational level required to understand it.

Although some experiments demonstrated that readability measures are not reliable alone for authorship attribution (AA) (Chaski 1999), these measures have been included in some AA methods, such as (Luyckx and Daelemans 2005), where their performance alone was even better than some syntactic features in one of the experiments. On the other hand, these features are commonly used in author profiling. For example, in the PAN 2013 shared task on author profiling, they have been used in 7 among the 21 participating methods (Rangel et al. 2013). One of them (Gillam 2013), is based only on these features and it was ranked 8th in terms of performance on the English text, and first regarding the runtime.

The use of readability measures to detect plagiarism intrinsically assumes that the plagiarist

[13] In some methods, the sentence length is measured in characters instead of words.

borrows texts that might be different in their complexity from the text written by her/himself. This is a conceivable scenario since the plagiarist could be a person with low linguistic skills and appropriates texts that are better, in terms of readability, than his own text (Knight et al. 2004). The IPD methods that used these measures are: (Meyer zu Eißen and Stein 2006), (Stein and Meyer zu Eißen 2007), (Stein et al. 2011), (Carnahan et al. 2014) and (Polydouri et al. 2017).

Vocabulary Richness

The vocabulary richness of an author is known as a traditional characterisation of the writing style. To measure it, various formulae have been proposed. Most of them make use of one or more of the following three values:

- The length of the text in words (a.k.a., the number of tokens, N)
- The number of distinct words (a.k.a., the number of types, V).
- The number of words with a given frequency i, V_i.

The most straightforward measure is the type-token ratio V/N. There exist as well measures that consist in counting the number of words that occur once (hapax legomena, V_1), twice (dis legomena, V_2) or three times (tris legomena, V_3), etc. Other measures have more elaborated formulae than the aforementioned ones, such as K of Yule (1944) and R of Honoré (1979). Table IV-5 displays the formulae of some well-known measures.

Studies have shown that vocabulary richness measures should be used with caution to characterise the style of one author or distinguish it among others (Hoover 2003). Furthermore, earlier studies demonstrated that the majority of the vocabulary richness measures are unstable with short texts (Stamatatos et al. 2001), which discourages their use for intrinsic plagiarism

Table IV-5. Well-known vocabulary richness formulae

Yule's (1944) characteristic[14]	$K = 10^4 \left(\sum_{i=1}^{N} \frac{i^2 V_i}{N^2} \right) - \frac{1}{N}$
Honoré's (1979) R[15]	$R = \frac{(100 \log N)}{\left(1 - \left(\frac{V_1}{V}\right)\right)}$
Simpson's (1949) D[16]	$D = \sum_{i=1}^{V} V_i \frac{i}{N} \frac{i-1}{N-1}$

[14] The displayed formula is taken from (Miranda-García and Calle-MartÍn 2006)
[15] The displayed formula is taken from (Stamatatos et al. 2001)
[16] The displayed formula is taken from (Tweedie and Baayen 1998)

Table IV-6. Some linguistic aspects manipulated to produce different sentence structures

Sentence Type	Sentence Voice	Sentence beginning
Simple	Active	Noun
Complex	Passive	Pronoun
Compound		Adjective
Complex compound		Adverb
		Conjunction
		Preposition
		Gerund phrases
		Participial phrases
		Infinitive phrases

detection wherein the writing style is analysed at the fragment level. Nonetheless, these measures are included in numerous intrinsic plagiarism detection methods, which are: (Meyer zu Eißen et al. 2007), (Stein et al. 2011), (Kern et al. 2012), and (Carnahan et al. 2014). On the other hand, Meyer zu Eißen and his colleagues (2007; 2006) proposed a new vocabulary richness measure called Average Word Frequency Class, which is argued to be ideal for IPD due to its stability with different text lengths. Later, this feature has been used in other methods, such as (Stein and Meyer zu Eißen 2007), (Zechner et al. 2009), and (Carnahan et al. 2014). In addition, variants of this measure are used in (Polydouri et al. 2017).

4.3.3 Syntactic Features

Syntax is the set of rules that govern the order of words when forming phrases, clauses and sentences. Each language offers a plethora of ways to structure sentences by using the so-called rewrite rules[17]. By applying different rewrite rules, it is possible to express the same meaning using sentences of different structures. Table IV-6 displays some linguistic aspects that could be manipulated to produce different sentence structures.

While it is preferable to vary the structure of sentences to avoid monotony[18], the study of Feng et al. (2012) shows that the preferences of the author with respect to structuring sentences may dominate. Moreover, it has been proved that the syntactic analysis of the text could serve in the style-based text categorisation (Argamon et al. 1998) and most notably in the authorship

[17] A rewrite rule has the form: category → category*, where category is a tag that denotes a syntactic constituent i.e., a phrase of a determined type (nominal, verbal, prepositional, adverbial, adjective), a part of speech, or even a word. The rewrite rule means that the category in the left could be written as the categories in the right. For example, the sentence "The girl ate an apple" could be generated by applying the following rewrite rules:

S → NP VP NP → D N VP → V NP N → girl
N → apple V → ate D → the D → an

The tags S, NP, VP, N, V and D stand mnemonically for sentence, noun phrase, verb phrase, noun, verb, and determiner, respectively.

[18] The blog of the software Paper Rater (http://blog.paperrater.com/2015/06/sentence-beginnings.html) provides information on this subject.

attribution task (Baayen et al. 1996).

The extraction of syntactic features requires language-specific processing that annotates the words or phrases with tags that reflect the structure of sentences. Techniques used for this purpose include part of speech (POS) tagging, chunking and parsing. See Table IV-7 for an illustration of these three kinds of text processing. As shown in the table, the sentence structure could be represented either in a *flat* way i.e., as a sequence of contiguous syntactic constituents such as part of speech or phrase types tags, or in a *hierarchical* way (ordinarily, as a parse tree) where each syntactic constituent is part of another.

The *POS tags* are the most used syntactic features in the existing intrinsic plagiarism detection methods. Employing these features consists in representing the text fragments of the suspicious document by:

- an ordered sequence of their POS tags (Tschuggnall and Specht 2013a),
- POS tags with their frequencies (Kuznetsov et al. 2016),
- POS tag n-grams with their frequencies (Carnahan et al. 2014; Stein et al. 2011),
- high-level features crafted from POS tags such as the *compression rate* of a text fragment[19] (Polydouri et al. 2017, 2018; Seaward and Matwin 2009), and a syntactic complexity measure which is a weighted sum of POS tags frequencies (Carnahan et al. 2014).

Concerning the *parse trees*, to date, only one group of authors have attempted to exploit them (Tschuggnall and Specht 2012, 2013b, 2013c). The authors proposed a method based on representing each sentence of the suspicious document with a set of *pq-grams* extracted from the parse tree of sentences excluding their leaves i.e. excluding the lexicon. The concept of pq-grams has been first introduced by Augsten et al. (2010) in order to compute the distance between ordered labelled trees[20]. A pq-gram is defined as a sequence of nodes extracted from a tree following certain constraints controlled by the variables p and q, which determine respectively the number of nodes explored in the tree vertically and horizontally to form the sequence of the nodes. These sequences are indeed subtrees of p+q nodes where p of them constitute the stem and q of them are the leaves of the extracted subtree. For example, (VP, NP, NP, CC, NP) is a 2,3-gram extracted from the parse tree displayed in Table IV-7 (see the nodes surrounded with a rectangle).

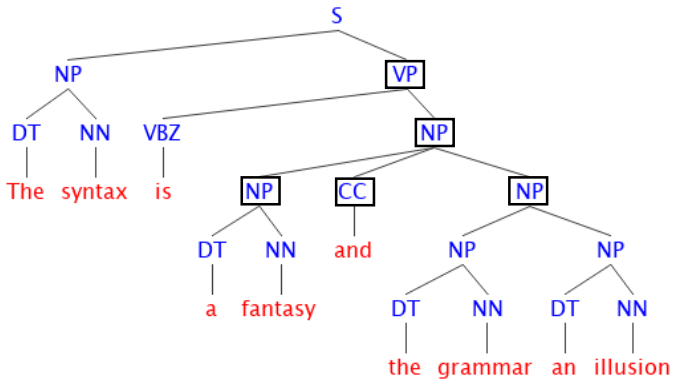
4.3.4 High-level Features

These features are obtained by feature engineering meaning that they are computed using other basic features. Some of the above-mentioned features could be already considered as high-

[19] In (Polydouri et al. 2017, 2018; Seaward and Matwin 2009), the compression rate is computed after representing the text by a binary sequence where the value 1 represents a particular POS tag and 0 otherwise. Afterwards, a compression algorithm (e.g., Lempel-Ziv) is used to reduce the length of the sequence.

[20] In graph theory, an ordered labelled tree is a rooted tree in which the nodes are labelled and their children are arranged in a specified ordering. This definition applies to parse trees where the label of nodes are tags representing syntactic constituents, and the order of siblings corresponds to the order of these constituents in the sentence.

Table IV-7. Examples of syntactic features and the tools used to extract them

The sentence	The syntax is a fantasy and the grammar an illusion[21]	**Tools (example)** [22]
Part of speech	The/**DT** syntax/**NN** is/**VBZ** a/**DT** fantasy/**NN** and/**CC** the/**DT** grammar/**NN** an/**DT** illusion/**NN**	POS Tagger (http://nlp.stanford.edu:8080/corenlp/process)
Chunks	[**NP** The syntax] [**VP** is] [**NP** a fantasy] and [**NP** the grammar] [**NP** an illusion]	Chunker (a.ka., Partial parser) (https://cogcomp.org/page/demo_view/ShallowParse)
Parse Tree		Parser (http://nlp.stanford.edu:8080/parser)

level features such as vocabulary richness and readability features (refer back to Section 4.3.2) and the compression rate of POS tags sequences (refer back to Section 4.3.3). In the next paragraph, we will describe other features that we were not able to classify among the aforementioned categories.

The method described in (Akiva 2011) represents each fragment of the suspicious document with a binary vector that marks the presence or the absence of the 100 rarest words of the document. Subsequently, a clustering algorithm creates two clusters of fragments according to the similarity between their vectors. Then, another set of features is extracted based on the *similarity of each fragment to the cluster* to which it belongs and the *number of fragments in each cluster* (i.e., the size of the cluster). Finally, these high-level features are the ones used to decide whether the fragment is plagiarised or not using a supervised classification method.

[21] The original quotation is in French : "Ils en conclurent que la syntaxe est une fantaisie et la grammaire une illusion". It is taken from: Gustave Flaubert, « Bouvard and Pecuchet », 1881.
[22] The tools mentioned as examples have been used to POS tag, chunk and parse the sentence of the example.

4.3.5 Semantic Features

Dealing with semantics is another level of NLP processing. Semantics has to do with *senses* and *references* of words. The sense of a word could be represented by its relations to other words in the lexicon (e.g., synonyms, antonyms…, etc.) while its reference is the real-world concept to which refers the word[23]. One may ask what the relationship is between the semantics and the style. Indeed, it has been proven that the choice to use a certain word among their synonyms may reflect the preference of the author (Clark and Hannon 2007). This means that two different authors may express the same meaning not only using different sentence structures but also different words.

Semantic features have been already exploited in authorship attribution (Gamon and Grey 2004). However, to our knowledge, none of the existing IPD methods utilises them, and that is why we do not mention them in Table IV-8, which displays all the categories of features used in IPD methods.

4.4 Plagiarised Fragments Identification

Once the features are extracted from each fragment of the suspicious document and also from the entire document in some methods, the plagiarism identification module leverages them to predict whether each fragment is plagiarised or not.

We argue that the existing approaches of plagiarism identification could be subsumed under four paradigms.

4.4.1 Supervised Learning

Intrinsic plagiarism detection can be thought of as a supervised binary classification task. In this case, a classifier should be trained using labelled data (i.e., the class is known) to be able to classify unseen data. In the context of IPD, methods of this paradigm need a dataset where the class (plagiarised or plagiarism-free) of each fragment of the suspicious document is known. The elements of this dataset are deemed as examples that feed a machine-learning algorithm. Once the algorithm learns its parameters from the provided examples, it can be used to predict the class of the fragments of a new suspicious document that has not been used in the training phase. Figure IV-4 illustrates the above explanation.

Unlike the majority of the supervised learning-based IPD methods which use classification algorithms, Kuznetsov et al.'s (2016) method employs a regression algorithm, namely Gradient Boosting Regression Trees[24]. The authors reported that they exploited regression to predict from the features a real value that represents the degree of mismatch between the

[23] For example, the word Java can be related to 2 real-world concepts : the programming language or the Indonesian island.

[24] Contrary to the common decision trees that associate an ordinal value (i.e., a class) to a given element, the regression trees map the element to a real value.

Table IV-8. Classification of the features used in intrinsic plagiarism detection methods

Character-based features		
Character n-grams frequency		(Stamatatos 2009a) (Kestemont et al. 2011) (Rao et al. 2011) (Kasprzak and Brandejs 2010) (Kern et al. 2012) (Kuta and Kitowski 2014) (Kuznetsov et al. 2016)
Character sound n-grams frequency		(Stein et al. 2011) (Carnahan et al. 2014)
Frequency of suffixes		(Rao et al. 2011) (Kern et al. 2012) (Sittar et al. 2016)
Frequency of a class of characters (digits, uppercases,…)		(Kern et al. 2012) (Sittar et al. 2016)
Frequency of individual characters (a particular punctuation mark, spaces, … etc)		(Zechner et al. 2009) (Rao et al. 2011) (Rao et al. 2011) (Kern et al. 2012) (Kuznetsov et al. 2016) (Mahgoub et al. 2015)
Lexical features		
VSM of words	All Words	(Oberreuter et al. 2011c; Oberreuter and Velásquez 2013)
	Closed-class words	(Meyer zu Eißen et al. 2007; Meyer zu Eißen and Stein 2006) (Stein and Meyer zu Eißen 2007) (Zechner et al. 2009) (Seaward and Matwin 2009) (Muhr et al. 2010) (Rao et al. 2011) (Stein et al. 2011) (Kern et al. 2012) (Carnahan et al. 2014) (Mahgoub et al. 2015) (Polydouri et al. 2017)
	Topic words	(Seaward and Matwin 2009)
	Discourse markers	(Rao et al. 2011)
	Common words	(Seaward and Matwin 2009) (Akiva 2012)
	Rarest words	(Akiva 2011)
Readability	Gunning Fog index	(Meyer zu Eißen et al. 2007) (Stein et al. 2011) (Carnahan et al. 2014)
	Flesch reading ease	(Stein et al. 2011) (Carnahan et al. 2014)
	Flesch-Kincaid Grade Level	(Meyer zu Eißen et al. 2007) (Stein et al. 2011) (Carnahan et al. 2014) (Polydouri et al. 2017)
	Dale-Chall formula	(Meyer zu Eißen et al. 2007)
	Average syllables per word	(Stein and Meyer zu Eißen 2007) (Stein et al. 2011) (Carnahan et al. 2014) (Polydouri et al. 2017)
	Amdahl's index	(Stein and Meyer zu Eißen 2007)
	Smog index	(Stein and Meyer zu Eißen 2007)
	Wiener Sachtextformel index	(Stein and Meyer zu Eißen 2007)

Ratios	Ratio of rarest words	(Kuznetsov et al. 2016)
	Ratio of Common words	(Kuznetsov et al. 2016)
Basic statistics	Average sentence length	(Meyer zu Eißen et al. 2007; Meyer zu Eißen and Stein 2006) (Stein et al. 2007) (Rao et al. 2011) (Kern et al. 2012) (Carnahan et al. 2014) (Kuznetsov et al. 2016) (Sittar et al. 2016) (Polydouri et al. 2017)
	Average word length	(Seaward and Matwin 2009) (Stein et al. 2011) (Kern et al. 2012) (Kuznetsov et al. 2016) (Sittar et al. 2016)
Vocabulary richness	Type-token ratio	(Kern et al. 2012)
	Hapax legomena	(Kern et al. 2012)
	Dis legomena	(Kern et al. 2012)
	Honoré's R	(Meyer zu Eißen et al. 2007) (Stein et al. 2011) (Kern et al. 2012) (Carnahan et al. 2014)
	Yule's K	(Meyer zu Eißen et al. 2007) (Stein et al. 2011) (Kern et al. 2012) (Carnahan et al. 2014)
	Simpsons' D	(Kern et al. 2012)
	Brunets' W	(Kern et al. 2012)
	Sichnels' S	(Kern et al. 2012)
	Word frequency class	(Meyer zu Eißen et al. 2007; Meyer zu Eißen and Stein 2006) (Stein and Meyer zu Eißen 2007) (Stein et al. 2011) (Zechner et al. 2009) (Carnahan et al. 2014) (Mahgoub et al. 2015) (Polydouri et al. 2017)
Syntactic features		
Ratio of interrogative sentences		(Sittar et al. 2016)
POS tags frequency		(Meyer zu Eißen et al. 2007; Meyer zu Eißen and Stein 2006) (Stein et al. 2007) (Zechner et al. 2009) (Seaward and Matwin 2009) (Mahgoub et al. 2015) (Kuznetsov et al. 2016) (Carnahan et al. 2014)
POS tags n-grams		(Carnahan et al. 2014) (Stein et al. 2011)
POS tags sequences		(Seaward and Matwin 2009) (Tschuggnall and Specht 2013a) (Polydouri et al. 2017)
Parse trees constituents		(Tschuggnall and Specht 2012, 2013b) (Tschuggnall and Specht 2013c)
High-level features		
Compression rate		(Seaward and Matwin 2009) (Polydouri et al. 2017)
Customised features		(Akiva 2011) (Kuznetsov et al. 2016) (Carnahan et al. 2014)

writing style of a fragment and that of the whole document. Then, all fragments with a regression value above a predefined threshold are marked as plagiarised.

It remains to say that the pitfall of IPD methods based on supervised learning is that they may suffer from the lack of training data. And even if it is available, there will be an imbalance in the number of plagiarised and the non-plagiarised examples since naturally the original texts are more abundant. This issue renders the IPD a classification problem with skewed classes, which is a known problem in machine learning that may lead to training biased classifiers. In (Polydouri et al. 2017, 2018), the authors attempted to mitigate this problem by using sampling techniques on the training corpus aiming to construct a balanced dataset. This problem can be also tackled by using classification algorithms designed to function with datasets of skewed classes, such as Complement Naïve Bayes (Rennie et al. 2003). In that context, we used this algorithm in one of our IPD experiments and it proved its effectiveness in comparison with the original Naïve Bayes (Bensalem et al. 2014b)[25].

The IPD methods that use supervised learning are listed in Table IV-9.

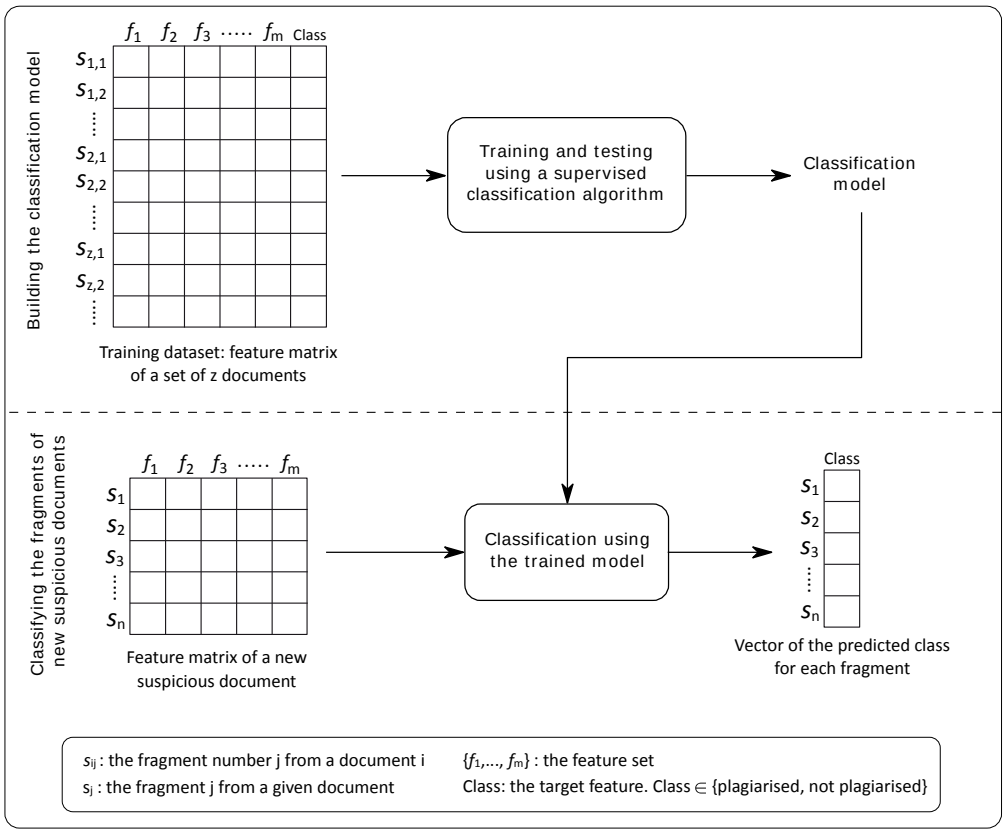


Figure IV-4. Steps of the supervised-learning-based intrinsic plagiarism detection

[25] The reference (Bensalem et al. 2014b) is an unpublished paper but has been accepted in CICLing 2014 conference (see https://www.cicling.org/2014/accepted.html –last consultation 15/03/2019). We described in this paper an early version of our IPD method in which we used a segmentation strategy into sentences and Complement Naïve Bayes as the classification algorithm. The evaluation has been made only on an earlier version of InAra Corpus (the Arabic corpus for IPD). Note that another version of our method is published later in (Bensalem et al. 2014a). This version uses the original Naïve Bayes and the sliding window for the segmentation, and we conducted the evaluation on InAra along with PAN IPD English corpora. A detailed description of our method is provided in the next chapter.

Table IV-9. The supervised learning-based methods used for intrinsic plagiarism detection

The supervised method	The IPD methods' references
Discriminant analysis	(Meyer zu Eißen et al. 2007; Meyer zu Eißen and Stein 2006; Stein and Meyer zu Eißen 2007)
Neural network	(Seaward and Matwin 2009)
Support vector machine	(Seaward and Matwin 2009) (Polydouri et al. 2017)
Decision trees	(Akiva 2011) (Polydouri et al. 2017)
Gradient boosting regression trees	(Kuznetsov et al. 2016)

4.4.2 Clustering

Clustering is an unsupervised machine learning approach that creates, from a given set of elements, subsets that group together the similar elements. The similarity between the elements is assessed based on their feature vectors. The number of clusters to create should be determined a priori for most of the algorithms. This paradigm is well suited for multi-author documents segmentation wherein each cluster involves the fragments of similar writing style (see, e.g., (Akiva 2012; Kern et al. 2012)), and hence, the number of the clusters represents the number of the authors involved in writing the document. In the existing intrinsic plagiarism methods, the number of the clusters created from the suspicious document fragments is typically two; one of them groups the plagiarism-free fragments and the other one contains the plagiarised fragments.

The IPD methods proposed in (Akiva 2011), (Carnahan et al. 2014) and (Sittar et al. 2016) are based on k-means, which is a well-known clustering algorithm.

The drawback of using clustering for IPD is that after the creation of the clusters (generally, two clusters) it remains unknown which of them is the plagiarism and which is the author's own text. Therefore, we need additional heuristic to make this decision. The heuristic of Carnahan et al. (2014), for example, assumes that the largest of the two clusters is the author's own text and the other one is the plagiarism. Another solution, used in Akiva (2011) (as mentioned in Section 4.3.4), is to use the clustering results to extract another set of features for each fragment such as its similarity to the cluster to which it has been assigned, and also the number of fragments in each cluster (i.e., the size of the cluster). These high-level features are then used to decide whether the fragment is plagiarised using a supervised classification method. The assumption of the authors is that plagiarised fragments are those that are close to the centroid of the small cluster but far away from the centroid of the whole document. In (Sittar et al. 2016) details on this step are not provided.

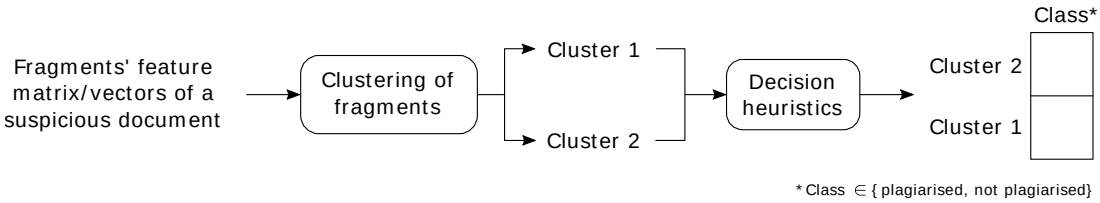


Figure IV-5. Steps of the clustering-based intrinsic plagiarism detection

Figure IV-5 illustrates the above-explained steps of the clustering-based paradigm.

4.4.3 Density-based Outlier Detection

This paradigm has been used by Stein et al. (2011). The method is based on assuming a Gaussian distribution of the feature values of the non-plagiarised fragments and a uniform distribution of the feature values of the plagiarised fragments (see Figure IV-6). Thus, two probabilities are estimated for a given fragment *s*:

(i) its probability of occurring among the plagiarised fragments, and

(ii) its probability of occurring among the non-plagiarised fragments.

To compute these probabilities using a single feature f_i, the Bayes rule is used and the predicted class is the one that has the maximum probability as shown in the following formula:

$$PredictedClass(s) = \underset{Class \in \{plag_0,\ plag_1\}}{\operatorname{argmax}} \frac{\mathrm{P}(f_i(s)|Class).\mathrm{P}(s)}{\mathrm{P}(f_i(s))},$$

where $\mathrm{P}(f_i(s)|\mathrm{Class})$ is the a-priori distribution, which is the likelihood that the fragment s has a certain feature value $f_i(s)$, given that the class of s is known. As mentioned earlier, the likelihood values for each class, i.e., $\mathrm{P}(f_i(s)|\mathrm{Class} = plag_0)$ and $\mathrm{P}(f_i(s)|\mathrm{Class}= plag_1)$, are estimated by assuming a Gaussian and a uniform distribution of the values of the feature f_i over the non-plagiarised class and the plagiarised class, respectively.

If more than one feature is involved in the class prediction, the formula below is used, where m is the number of features.

$$PredictedClass(s) = \underset{Class \in \{plag_0,\ plag_1\}}{\operatorname{argmax}} \mathrm{P}(Class).\prod_{i=1}^{m} \mathrm{P}(f_i(s)|Class).$$

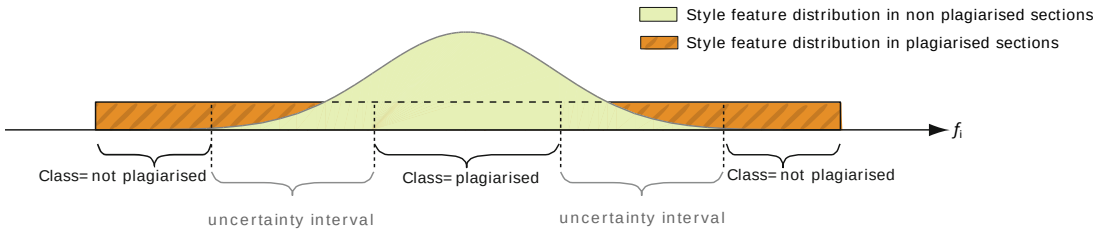


Figure IV-6. Illustration of the density-based outlier detection for intrinsic plagiarism detection. Plagiarised and non-plagiarised sections can be separated if their values of a feature *fi* are differently distributed (adapted after (Stein et al. 2011)).

As for the priors, i.e., P(Class = $plag_0$) and P(Class = $plag_1$) –which is the portion of each class among all the fragments[26]– the authors of this approach stated that they are estimated either by an impurity assessment (meta information on the document) or by the maximum likelihood estimator which assumes that the classes are uniformly distributed, i.e., half of the fragments is plagiarised and the other half is not. However, it has not been stated in the paper (Stein et al. 2011) which of these two options is adopted in the conducted experiments.

The application of the density-based methods on IPD has been proposed by Stein and Meyer zu Eißen (2007), and later, used by the same group of authors in their seminal paper (Stein et al. 2011). Yet, the experiments described in their paper were conducted only on a subset of PAN-PC-09 evaluation corpus, and therefore, it is still unknown how density-based methods compare to other paradigms. Besides, apart from Stein et al., only one group of young researchers from Carleton College[27] (Carnahan et al. 2014), adopted this paradigm, but unfortunately, the performance was not assessed with the standardised evaluation measures. Therefore, reproducing this method with the aim to compare it with others is good material for future work on IPD.

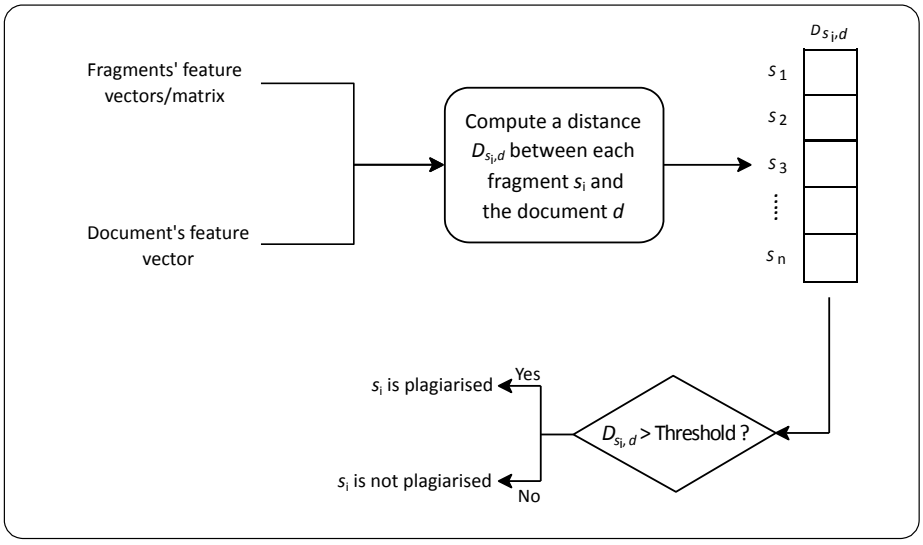
There exists another probabilistic method that has been applied to IPD, which is Hidden Markov Modelling (HMM) (Carnahan et al. 2014). But the authors provided neither a sufficient explanation nor the evaluation results. Hence, trying to solve IPD using HMM is a direction to explore in future work especially that this technique has been successfully used for a similar problem which is multi-author documents segmentation (Aldebei et al. 2016).

4.4.4 Distance-based Outlier Detection

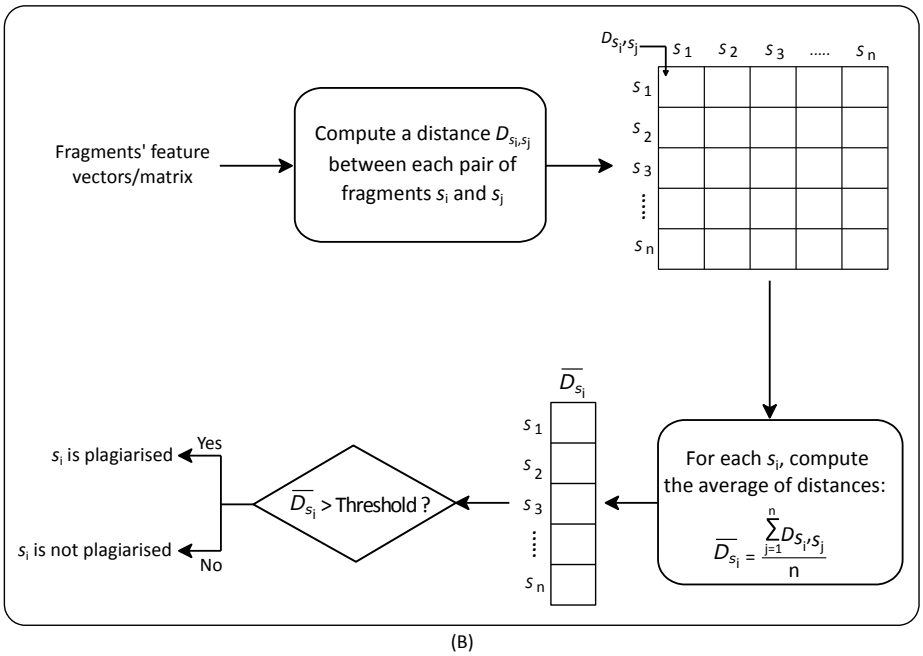
[26] For example, P(Class= $plag_1$) = 0.1 means that 10% of the document fragments are plagiarised, and consequently, P(Class= $plag_0$) is 0.9.

[27] http://www.cs.carleton.edu/cs_comps/1314/dlibenno/final-results/index.html (Last consultation: 15/03/2019)

The methods following this paradigms measure (i) the distance between the feature vector of each fragment and the one of the entire document, or (ii) the distance between the feature vectors of each pair of fragments. In this latter case, the distances that concern each fragment are averaged. Hence, for both cases, the document is represented with a vector of distances



(A)



(B)

Figure IV-7. Steps of the distance-based outlier detection for intrinsic plagiarism detection. In the figure (A), the distance is computed between the fragments and the document; and in the figure (B), the distance is computed between each pair of fragments.

wherein each dimension corresponds to one of its fragments. If the distance exceeds a predefined threshold, the corresponding fragment is considered as plagiarised (see Figure IV-7 for an illustration of the explanation above). In most of the examined methods, the threshold is defined as $\mu+\varepsilon\times\sigma$; where, μ is the mean of the distance vector, σ is its standard deviation, and ε is a small scalar value that is tuned experimentally in all the existing methods. Indeed, the distance can be thought of as a high-level feature. This is because, on the one hand, it is computed using other basic features and, on the other hand, the decision of whether the fragment is plagiarised is based on the distance and not on the extracted basic features directly, such as n-grams or words.

The intuition behind this paradigm is: to be considered as plagiarised, a fragment should have a style that is sufficiently different from the dominant style in the document. This viewpoint has a shortcoming. If there is no dominant style in the document (e.g., in the case of patchwriting from many sources) or if the dominant style is the one of the plagiarised parts (e.g., the majority of the text of the document is plagiarised from one source), then methods based on this paradigm becomes unreliable.

After the description of the paradigms used in the plagiarised fragments identification phase of the IPD methods, Table IV-10 lists the methods that use each paradigm.

4.5 Post-processing

There are three goals of the post-processing heuristics:

(1) to determine definitely the boundaries of the plagiarised fragments such that no overlap exists between them,

Table IV-10. Paradigms of the plagiarised fragments identification in intrinsic plagiarism detection methods

Plagiarism identification paradigm	References of IPD methods
Supervised learning	(Meyer zu Eißen et al. 2007; Meyer zu Eißen and Stein 2006; Stein and Meyer zu Eißen 2007) (Seaward and Matwin 2009) (Akiva 2011) (Kuznetsov et al. 2016) (Polydouri et al. 2017, 2018)
Clustering	(Akiva 2011) (Akiva 2012) (Kern et al. 2012) (Carnahan et al. 2014) (Sittar et al. 2016)
Density-based outlier detection	(Stein et al. 2011)
Distance-based outlier detection	(Stamatatos 2009a) (Zechner et al. 2009) (Muhr et al. 2010) (Suárez et al. 2010) (Kasprzak and Brandejs 2010) (Kestemont et al. 2011) (Rao et al. 2011) (Oberreuter et al. 2011b; Oberreuter and Velásquez 2013) (Kuta and Kitowski 2014) (Mahgoub et al. 2015) (Kuznetsov et al. 2016)

(2) to approve or disapprove the plagiarism suspicion at the fragment level,

(3) to approve or disapprove the plagiarism suspicion at the document level.

The next subsections describe the post-processing heuristics used in existing IPD methods.

4.5.1 Merging the Adjacent or Overlapping Detected Fragments

This heuristic operates by defining a threshold, $\tau_{Distance}$, on the distance, *dist*, between the boundaries of two consecutive detected fragments s_i and s_j. If $dist \leq \tau_{Distance}$, then s_i and s_j and the text between them (if any) will be considered as one detected fragment.

4.5.2 Discarding the Short Detections

This is typically done by defining a threshold, τ_{Length}, on the length, l, of a fragment, s_i, marked as plagiarised. If $l < \tau_{Length}$ then s_i is dropped from the set of the fragments marked as plagiarised.

4.5.3 Human Inspection and Citation Analysis

Those heuristics are suggested by Stein et al. (2011), and they may reveal that fragments flagged as plagiarised are actually quotations or passages that have been detected mistakenly because they contain jargon or punctuation marks that are not repeated in the rest of the document.

To date, none of the existing studies has appraised the result of IPD by human inspection. We conjecture that such a study may provide insight into the comparison between the abilities of human and machine in detecting plagiarism intrinsically. As for citation analysis, this could be manual or through an automatic citation detection method. We are not aware of any IPD method that uses such heuristic as a post-processing step. Moreover, the existing IPD evaluation corpora do not contain annotated citations, which may explain the absence of methods that address this point.

4.5.4 Detection of Plagiarism-free Documents Heuristic

It is a simple approach that rejects the plagiarism suspicion for the entire document if the proportion of the detected plagiarism to the whole document length is equal or under a predefined threshold. For example, this kind of approaches may consider as plagiarism-free a document in which the detected plagiarism is less than 5% of the document length. Recall that Stamatatos (2009a) used a similar *pre-processing* heuristic, whereas the decision of considering a document as plagiarism-free is based on the standard deviation of the style change function. But as post-processing, no method used this heuristic.

4.5.5 Unmasking

This is the most sophisticated approach used for post-processing in the existing IPD methods. It represents the suspicious document fragments with vectors of words labelled with the predicted class: plagiarised or not plagiarised. Then, a classifier is trained to separate the two sets of fragments S1 and S2, which are allegedly the plagiarised and the non-plagiarised texts as suggested by the plagiarism detection result. The next step is to eliminate the most discriminative words gradually in a series of "*training / feature elimination*" iterations. If the performance of the classifier heavily deteriorates after a certain number of iterations, S1 and S2 will be considered as written by the same author, meaning that the document is plagiarism-free. The rationale is that the most discriminative words are likely to be related to the topic or the genre of the text, and if their elimination did not harm the classification performance, this means the difference between S1 and S2 could not be limited to only the topic or the genre. Instead, the two sets of fragments are considered distinct in terms of the author writing style, which is represented by the small set of words that are still able to distinguish between S1 and S2 even after the elimination of the most discriminative words. Stein et al. (2011) used this heuristic in their method after its success for authorship verification (Koppel and Schler 2004), and they reported a great improvement in the plagiarism detection results. Nevertheless, as far as we know, no other IPD method has exploited this heuristic.

4.5.6 Voting Heuristic

If the used segmentation strategy produces overlapping fragments, as the case with the sliding window, then the plagiarism identification may yield more than one decision for one passage. For example, if the sliding window length is 3 sentences and its step is 1 sentence,

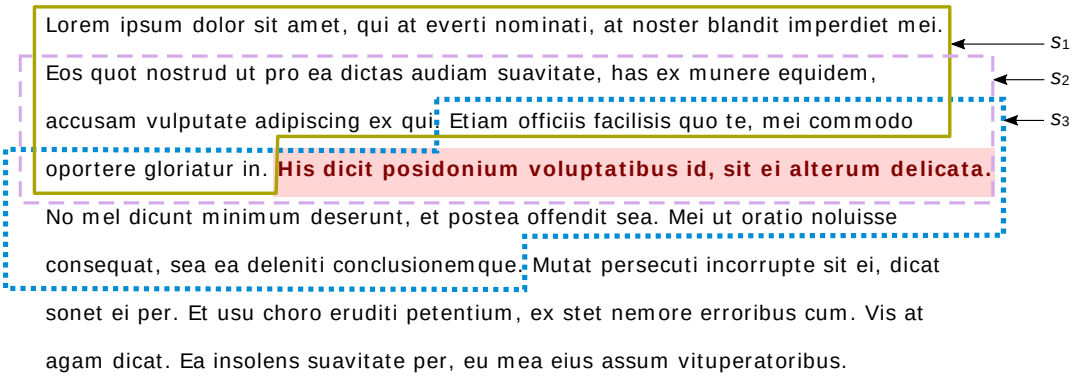


Figure IV-8. Illustration of the role of the voting heuristic with 3 overlapping fragments. The highlighted sentence belongs to the three fragments s_1, s_2 and s_3, and hence, it may receive three different plagiarism identification decisions. For example, if s_1 and s_2 are marked as plagiarised and s_3 as non-plagiarised, then the voting heuristics serves to determine a final decision for the sentence in the intersection.

then we may have a sentence that appears in 3 fragments and hence receives 3 plagiarism detection results (see Figure IV-8). The majority of the methods mark a passage as plagiarised if it belongs to a detected fragment no matter whether it is part of the intersection with a neighbouring undetected fragment. Unlike this simplistic decision, the voting heuristic considers all the decisions related to a passage to produce a final decision.

Table IV-11 lists the post-processing heuristics and the IPD methods that used them.

5 Performance of IPD Methods: a Brief Overview

To complete the picture on intrinsic plagiarism detection, it is necessary to talk about its effectiveness. In fact, despite the variety of heuristics and stylistic features used in the methods (as shown in Section 4), their performance scores are still poor. To the best of our knowledge, few methods, such as (Stamatatos 2009a) and (Oberreuter et al. 2011b; Oberreuter and Velásquez 2013), reached an F-measure greater than 0.3 using a standardised evaluation framework. Other methods, for instance Stein et al. (2011), Tschuggnall and Specht (2013c) and Polydouri et al. (2017, 2018) obtained relatively higher scores. Nonetheless, the two former methods have been evaluated on only subsets of the evaluation corpus, and the evaluation of the latter method is based on a modified version of the performance measures[28].

Table IV-11. Post-processing heuristics in intrinsic plagiarism detection methods

Heuristics	IPD Methods
Merging the adjacent or overlapping detected fragments	(Kuta and Kitowski 2014) (Zechner et al. 2009) (Mahgoub et al. 2015) (Muhr et al. 2010) (Suárez et al. 2010) (Kestemont et al. 2011) (Rao et al. 2011) (Tschuggnall and Specht 2012, 2013a, 2013c, 2013b)
Discarding the short detections	(Tschuggnall and Specht 2012) (Tschuggnall and Specht 2013a, 2013c, 2013b)
Voting heuristics	(Polydouri et al. 2017, 2018)
Unmasking	(Stein et al. 2011)
Detection of plagiarism-free documents heuristic	Not used up to now
Human inspection and citation analysis	Not used up to now

[28] The performance measure used by Polydouri et al. (2017, 2018) are computed based on the number of sentences and not the number of characters. For example, given a plagiarised fragment composed of 4 sentences, if the software detects 2 of them, the recall measured on this fragment, according to Polydouri et al. would be 0.5. However, the standardised recall score (Potthast et al. 2010c) could be more or less different since it is the ratio of the length, in characters, of the 2 detected sentences to the length of the full plagiarised fragment.

This makes the results of these methods not comparable to the others, and hence it is difficult to draw any conclusion on whether the techniques used by the latter methods are actually the best. On top of that, the performance of all IPD methods is significantly lower than the performance of external plagiarism detection methods whose accuracy exceeded 90% in detecting verbatim plagiarism.

A possible explanation for the difficulty of obtaining high accuracy might be the inherent constraints of the problem, which are the uncertainty on the number of the different writing styles present in the suspicious document and their boundaries. This makes the IPD not only a classification problem but also a segmentation problem.

6 Conclusion

This chapter is the first research so far surveying systematically the IPD methods published over the last two decades. In fact, the growth of research dealing with IPD is deemed slow as only a few tens of papers have been published since the beginning of the millennium. Moreover, most of the papers are PAN shared tasks' working notes, and there are only three articles published in journals: (Stein et al. 2011), (Oberreuter and Velásquez 2013) and (Polydouri et al. 2018). It is our hope that we provide the researchers with a base reference to speed up and foster in-depth research on this topic.

A solution to this research problem remains elusive (as shown in Section 5), and further efforts to resolve or better understand it are still needed. In Chapter V, we present our work aiming to increase our understanding of the use of character n-grams to detect and describe plagiarism. Besides, Chapter VI identifies the constraints of the current IPD solutions that must be overcome to advance research on this topic.

Chapter V. Character N-grams as the Only Intrinsic Evidence of Plagiarism

> *"May God forgive me, but the letters of the alphabet frighten me terribly. They are sly, shameless demons - and dangerous! You open the inkwell, release them; they run off - and how will you ever get control of them again!*
>
> Nikos Kazantzakis (1883, 1957)[1]

1 Introduction

One of the most straightforward and powerful text representation approaches used in style analysis-based tasks is character n-grams. Several studies have investigated the best ways of exploiting them in terms of their length, their frequency and even their position in the word (Houvardas and Stamatatos 2006; Jankowska et al. 2014; Kešelj et al. 2003; Sapkota et al. 2015; Stamatatos 2013; Zečević 2011). However, many of these investigations concern authorship attribution, and although there exist intrinsic plagiarism detection methods that use character n-grams, there is still a lack of in-depth works that optimise their use for this task specifically. Instead of task-oriented investigations, it has been granted in some works that the ways of using these features in authorship attribution remain the same for IPD.

The principal goal of this chapter is to investigate whether character n-grams of different

[1] Nikos Kazantzakis is a Greek writer. This citation is brought from: https://www.brainyquote.com/quotes/nikos_kazantzakis_746551

frequency and length are equivalent in terms of their relevance to intrinsic plagiarism detection. Our motivation to address this question is twofold:

- to optimise the effectiveness of the task by using only the set of n-grams that leads to the best results,
- to gain insight into the relationship between the frequency of character n-grams and plagiarism. In other words, to try to describe plagiarism in terms of character n-grams by considering their frequency ranges (frequent or infrequent).

We conduct our investigation using two character n-grams-based methods: our method (Bensalem et al. 2014a) that we will describe in this chapter, and the well-known IPD method of Stamatatos (2009a).

The rest of this chapter is structured as follows. Section 2 defines character n-grams and overview intrinsic plagiarism detection methods based on them. Sections 3 describes our method where selecting n-grams according to their frequencies is a core step. Sections 4 presents the datasets and the performance measures used in our experiments. In Section 5, the proposed method is compared to state-of-the-art methods. Sections 6 and 7 analyse the sensitivity of intrinsic plagiarism detection performance to the character n-grams frequency and length in the context of our method and Stamatatos' method, respectively. Finally, Section 8 summarises the main insights gained from this study.

2 Character N-grams

Representing a text using its character n-grams requires decomposing it into all the possible sequences of *n* successive characters. It is an approach to text description and manipulation that is in between words and individual characters (Rousseau 2002). Figure V-1 shows the n-grams extracted from a sentence. Note that the set of all the n-grams of a predefined length, *n*, extracted along with their frequencies from a given text, is generally referred to, in the literature, as the text's *n-gram profile.* Figure V-2 is a 3-gram profile computed from the same sentence of Figure V-1.

The first use of character n-grams dates back at least to Claude Shannon (McNamee and Mayfield 2004), an eminent scientist in information science. In his paper published in 1951 (Shannon 1951), Shannon used n-grams in the context of language modelling, which is the task of predicting the next letter given a sequence of its previous letters in a text. After this research, character n-grams have been implemented in the context of numerous other applications such as language identification (Cavnar and Trenkle 1994), native language identification (Kulmizev et al. 2017)[2], information retrieval (McNamee and Mayfield 2004),

[2] Language identification and native language identification are not the same task. The former consists in detecting the language of a given text and the latter is to ascertain the native language of the author from a text he wrote in a second language.

The n-grams of: “If you dream it, you can do it”*

1-gram	i	f	_	y	o	...	i	t

2-gram	If	f_	_y	yo	ou	_d	...	_i	it

3-gram	If_	f_y	_yo	you	ou_	_dr	...	o_i	_it

.....

10-gram	If_you_dre	f_you_drea	_you_dream	you_dream_	...	u_can_do_i	_can_do_it

* A quote of Walt Disney

Figure V-1. Illustration of the n-grams of a text where n=1..10. The character ‘_’ is used to represent the space

_yo	2
_it	2
ou_	2
you	2
if_	1
n_d	1
_ca	1
t,_	1
m_i	1
rea	1
u_d	1
do_	1

an_	1
it,	1
u_c	1
am_	1
dre	1
f_y	1
_do	1
can	1
,_y	1
eam	1
_dr	1
o_i	1

Figure V-2. An example of a 3-gram profile

writing style analysis (Stamatatos 2013) and spam detection (Kanaris et al. 2007) (See Section 2.1.2 for some other examples).

2.1 Advantages

Advantages of using character n-grams as a text representation could be different in each application, though, these low-level features are generally used for easiness and effectiveness

reasons as explained below.

2.1.1 Easiness

Unlike the language-specific processing such as stemming or parts of speech tagging –that are usually used with the bag of words representation– extracting character n-grams is straightforward and does not require non-trivial linguistic processing or expensive tools. Despite their effortless extraction, they are able to capture the morphological and syntactical features (Houvardas and Stamatatos 2006). That is, it is possible to easily obtain word stems and affixes without the need to language-specific processing. Consequently, a solution based on character n-grams can be readily applied to texts written in any language. For instance, Kešelj et al. (2003) applied their method of authorship attribution for English, Greek, and Chinese without the need to a complicated adaptation to each language.

Moreover, character n-grams are an excellent alternative to word-based representation for texts in some Asian languages where words are difficult to tokenise because of the lack of the space character between them. See for instance the work of Peng and Schuurmans (2003) who used these features to represent Chinese text for topic identification.

2.1.2 Effectiveness

In addition to being straightforward, it has long been known that character n-grams is an effective text representation approach in a wide variety of applications, such as topic identification (Cavnar and Trenkle 1994; Peng and Schuurmans 2003; Zhang et al. 2015), document clustering according to topic (Miao et al. 2005), sentiment polarity prediction[3] (Zhang et al. 2015) and language identification (Cavnar and Trenkle 1994).

One of the main reasons for their effectiveness is their tolerance to lexical errors. That is, it is possible to detect the similarity between words even if one of them is misspelt since the error will not occur in all the n-grams. For example, the words *straightforward* and *straitforward* are still sharing several n-grams despite the spelling error in the second word. This tolerance is especially important in applications where spelling errors may alter the results such as topic identification (Cavnar and Trenkle 1994; Khreisat 2009), topic change identification in user queries (Gencosman et al. 2014) information retrieval (Pearce and Nicholas 1996) and matching publications to references in bibliographic databases (Abdulhayoglu et al. 2016).

In addition to their tolerance to lexical errors, their ability to capture the morphological features makes the character n-grams effective for applications based on stylistic analysis. For example, by representing a text with 4-grams it is possible to obtain the frequency of the suffix

[3] Sentiment polarity prediction predicts whether a review (e.g. on a product) is negative or positive. See (Medhat et al. 2014) for further information on sentiment analysis research area.

"ing_" which could be used as an indicator of whether or not the author prefers to use gerunds. At the same time, n-grams may capture stems of words even if they appear in different inflected forms, which can help for example, to estimate the vocabulary richness accurately. Examples of stylistic analysis applications where character n-grams have been used successfully include authorship attribution (Kešelj et al. 2003; Stamatatos 2016), authorship verification (Jankowska et al. 2014), detection of opinion spam (Hernández Fusilier et al. 2015) and intrinsic plagiarism detection as we will detail in the next section.

2.2 Character N-grams in Intrinsic Plagiarism Detection Methods

Character n-grams are used in IPD methods alone (Kestemont et al. 2011; Stamatatos 2009a), or along with other features (Kern et al. 2012; Kuznetsov et al. 2016; Rao et al. 2011; Stein et al. 2011). The following summarises the intrinsic plagiarism detection methods that use character n-grams as the main features.

Stamatatos' (2009a) method represents the suspicious document and its fragments by 3-gram profiles[4]. The fragments are obtained through a sliding window, of around 1000 characters, that moves by 200 characters in each step. Then, a style change function is computed based on the dissimilarity between the n-gram profile of the entire document and the one of each fragment. By comparing the standard deviation of this function values with a threshold parameter, the method predicts whether the given document is plagiarism-free or not. If it is not plagiarism-free, a fragment is marked as plagiarised if its style change value is higher than a defined threshold that can be controlled by a parameter named by the author *sensitivity of plagiarism detection*.

Kestemont et al. (2011) hold the view that representing documents using all their n-grams is computationally expensive when dealing with long texts. Therefore, their method employs a predefined set of high-frequency 3-grams (extracted from a corpus) to represent the suspicious document fragments. This idea was inspired by authorship attribution research wherein high-frequency n-grams have been used successfully (Stamatatos 2009b). To detect outliers, this method uses the dissimilarity measure of Stamatatos (2009a) but computes it between each pair of the suspicious document fragments.

In Kuznetsov et al. (2016) method, each sentence is represented with a set of features, among others the frequency of the rarest n-grams, the frequency of the most frequent n-grams, and the mean of the relational frequency of n-grams. This latter is a new feature computed for each n-gram within a sentence. The more an n-gram is specific to a sentence (it appears in the sentence more than its occurrence in the rest of the document), the higher becomes its relational frequency. The authors reported that they determined the optimal lengths of n-grams (1, 3 and 4) after experimenting with different lengths. Next, gradient boosting regression trees

[4] The frequency of n-grams in this method is normalised.

are used to generate a model that combines features and predict a score for each sentence that represents its degree of mismatch with the style of the main author. Finally, all sentences with a score more than a certain threshold are marked as plagiarised.

Character n-grams have been used as well in other IPD methods but not as the main features (Kern et al. 2012; Potthast et al. 2010b; Rao et al. 2011). Table V-1 displays the lengths and frequencies of n-grams used in intrinsic plagiarism methods[5].

2.2.1 Discussion

In the examined methods, in which the n-grams have been selected according to their frequencies, the selection of the n-grams was not justified rationally based on an understanding of n-grams properties nor empirically based on n-grams performance.

For example, in (Kestemont et al. 2011), representing the text using only the most frequent n-grams extracted from a corpus was based on an efficiency reason which is to reduce the computation. However, no experiment has been done to check the impact of this reduction of the number of the used n-grams on performance or to prove that high-frequency n-grams are more effective than the rest of n-grams with lesser frequency. In (Kuznetsov et al. 2016), the frequencies of both rare and frequent n-grams in a sentence were among the features used to quantify the writing style incoherence between this sentence and the rest of the document. However, the rationale behind these choices has not been explained.

On the other hand, it is worth to mention the work of Kuta and Kitowski (2014) who replicated Stamatatos' (2009a) method with the aim of optimising its performance. The

Table V-1. The frequency and length of character n-grams in intrinsic plagiarism detection methods

	N-grams used to compute features	References
Frequency	All n-grams regardless of their frequencies	(Stamatatos 2009a) (Kuznetsov et al. 2016) (Kern et al. 2012)
	High-frequency n-grams	(Kestemont et al. 2011) (Rao et al. 2011) (Kuznetsov et al. 2016)
	Low-frequency n-grams	(Kuznetsov et al. 2016)
Length	1	(Kuznetsov et al. 2016) (Kern et al. 2012)
	2	(Kern et al. 2012)
	3	(Stamatatos 2009a) (Kestemont et al. 2011) (Kern et al. 2012) (Kuznetsov et al. 2016)
	4	(Kuznetsov et al. 2016)

[5] The table lists only the methods that provide information on the used character n-grams.

authors investigated the effectiveness of the most frequent n-grams (as they have been used in (Kestemont et al. 2011)) and unveiled their poor performance in IPD in comparison with the whole set of n-grams. However, the effectiveness of the low-frequency n-grams has not been investigated.

As stated in the introduction, this chapter aims to appraise the relation between IPD performance and the character n-grams' frequency and length for performance optimisation and task understanding reasons. We conduct our analysis in the context of two state-of-the-art intrinsic plagiarism detection methods (our method and Stamatatos' (2009a) method) where character n-grams have been exploited in distinct ways. Before starting the analysis, let us recall that Stamatatos' method is a well-known IPD method and we provided a brief description of it in Section 2.2. As for our method, it was first introduced in the short paper (Bensalem et al. 2014a), and we will provide a detailed description of it in the next section.

3 N-grams Frequency Classes Method

3.1 Intuition

We recall that in intrinsic plagiarism detection approach, a fragment is considered plagiarised if it deviates from the dominant writing style of the document. With respect to character n-grams, we posit that this deviation could emerge in two ways:

(1) The suspicious fragment could be a text in which we notice the *presence* of n-grams that are *infrequent* in the rest of the document, e.g., a punctuated passage while the rest of the document lacks punctuation.

(2) The suspicious fragment could be a text in which we notice a *lack* of n-grams that are relatively *frequent* in the rest of the document, e.g., a passage where there is a lack of using the function word 'of' – because a preference of using noun adjuncts instead – while 'of' is abundant in the rest of the document.

From the two aforementioned perspectives, we assume the following: given a document *d*, the *proportion* of its *infrequent* n-grams (the 1st perspective) and its *frequent* n-grams (the 2nd perspective) in a fragment of text *s* belonging to *d* could be a clue to whether *s* is plagiarised or not.

Describing n-grams just by being *frequent* or *infrequent* is vague, hence the need for a systematic way to determine the frequency boundaries of each category. Thus, the method we are proposing (1) classifies n-grams according to their frequencies in a given document, (2) computes for each fragment the proportion of n-grams that belong to a particular class, which quantifies the degree of the presence of the concerned subset of n-grams in that fragment, and (3) uses this *proportion* to reveal plagiarism, as we stated in the assumption above. The

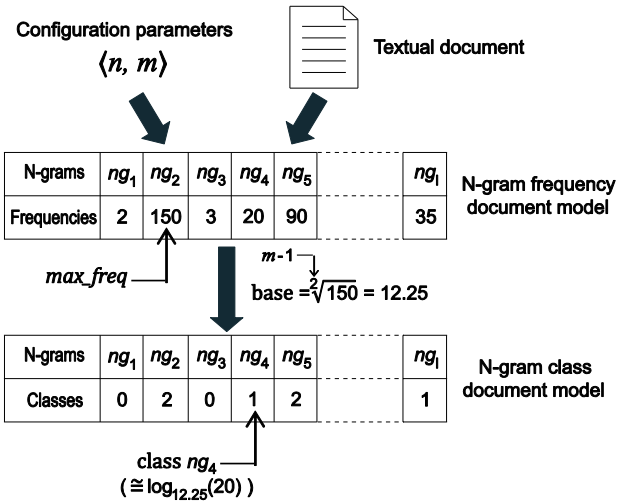


Figure V-3. Steps for computing the n-gram classes of a document. The parameter *n* is the length of n-grams and *m* is the number of classes. In this example *m* = 3 (class labels are from 0 to 2)

following subsections provide further details on these three stages.

3.2 N-gram Classification

N-gram frequency classes are created by grouping together the character n-grams of a particular length, *n*, that have similar frequencies in a given document. We represent the *frequency class of an n-gram* (or briefly *n-gram class*) by a natural number belonging to the interval [0..*m* − 1] such that *m* is the number of classes into which the character n-grams of a document are classified according to their frequencies in this document.

Concretely, to classify the n-grams of a given document, *d*, into *m* classes, first, the document profile is extracted, i.e., the document is represented by a 2 × *l* matrix (*l* is the total number of distinct n-grams extracted from *d*), where the first row contains the n-grams ng_i (i = 1..*l*) and the second one contains their number of occurrences, $freq_i$, in *d*. Let *max_freq* denote the maximum frequency, so:

$$max\_freq = \max\ freq_i,\ i = 1..l. \tag{1}$$

Then, the class of an n-gram, ng_i, is:

$$\text{class}\ ng_i = \text{round}\big(\log_{\text{base}}(freq_i)\big), \tag{2}$$

where round is a function that turns the real result of the logarithm into the nearest integer, and base is a variable computed as follows:

$$\text{base} = \sqrt[m-1]{max\_freq}\ . \tag{3}$$

By computing the base of the logarithm this way, the *high-frequency* n-grams in the document will be in the class $m-1$, and the *low-frequency* n-grams (e.g. the ones that appear only one time) will be in the class 0. If the number of classes is higher than two ($m > 2$), classes between 0 and $m-1$ will contain *medium-frequency* n-grams. Figure V-3 illustrates an example of computing the n-gram classes of a document.

3.2.1 Rationale

In the literature, selecting the n-grams by considering their frequencies is usually controlled either by:

(1) a threshold on the number of n-grams (Jankowska et al. 2014), e.g., selecting the 3000 most frequent n-grams, or
(2) a threshold on the frequency of n-grams (Stamatatos 2013), e.g., selecting n-grams whose occurrence is higher than 500.

These techniques are typically used to select n-grams based on their frequencies in a training corpus, whereas we are interested in selecting n-grams on the basis of their frequencies in each document separately. Therefore, the above techniques do not suit our purpose since it might be impractical to set a single threshold (on the n-grams frequency or number) to select n-grams from documents of different sizes. For example, while selecting the most frequent X n-grams makes sense for a long document, it leads to keeping all the n-grams of a document whose profile size is smaller than X n-grams.

To avoid the use of a threshold, our method classifies n-grams as a step towards their selection. Since the calculation of n-gram classes involves the variable maximum frequency, *max_freq* (see the equations (1-3)), we obtain classes whose boundaries adapt automatically to the document length. Besides, the parameter *number of classes*, *m*, allows controlling the frequency boundaries of classes (and consequently the number of n-grams in each class) without the need to set a threshold. As illustrated in Figure V-4, when $m = 2$, around half of the document's n-grams is assigned to the class 0, and the other half is assigned to the class 1. However, if $m = 10$, the number of n-grams in each class will be far less than half. To illustrate further, we also examined the n-grams frequencies in each class (of the same document used to create Figure V-4), and we observed that the class 0 comprises even the n-grams that occur 34 times when $m = 2$; whereas, the class 0 contains only the n-grams that occur once when $m = 10$.

3.3 Features Extraction

As mentioned earlier, the features we are introducing represent the proportions of the n-grams frequency classes in a given fragment. These features are extracted from the fragments of the suspicious document *d* according to the following steps:

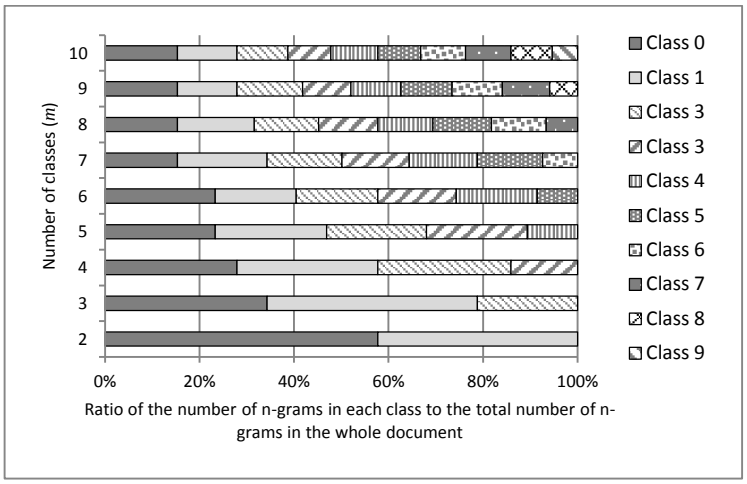


Figure V-4. The relation between the number of classes into which the n-grams are classified and the number of n-grams in the classes (these classes are computed on the 3-grams of suspicious-document03103 from the PAN-PC-11 corpus)

(1) ***Computing the document model.*** The n-gram class model of the given document *d* is built as explained in Section 3.1[6] (refer back to Figure V-3). Recall that this model is a vector representation of the document where the features are the whole set of the document n-grams, and their values are their classes.

(2) ***Segmentation.*** The document *d* is segmented into fragments by using the sliding window technique. Inspired by the segmentation strategy used in (Oberreuter and Velásquez 2013), we used different options for the window size depending on the document length, which are 100, 200 and 400 words applied to documents of fewer than 600 words, between 600 and 1800 words, and more than 1800 words, respectively. Let S denote the set of fragments s_p extracted by setting the window step equal to the quarter of the window size. The output of this step is a set of overlapping fragments.

(3) ***Computing the profile of each fragment.*** The n-grams are extracted from each fragment $s_p \in S$ (the overlapping fragments), and each n-gram is represented with its class obtained from the document model.

(4) ***Computing the proportion of classes.*** Finally, for each fragment $s_p \in S$, we compute the proportion of each class in the fragment. Each proportion is an NFCP feature.

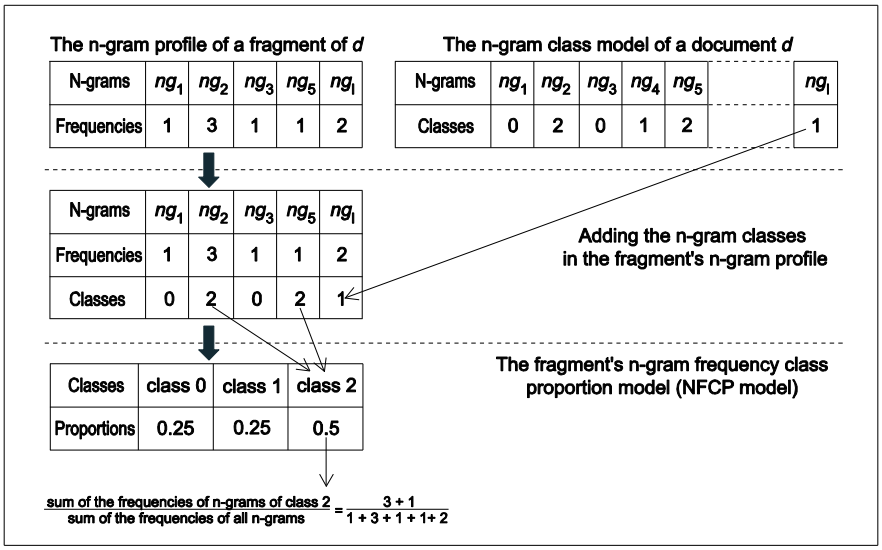
3.3.1 Variants of the Extraction Methods of NFCP Features

In this section, we describe different ways of extracting the NFCP features as well as the experiment we conducted to choose the variant that produces the best features. These variants are obtained by computing in different ways two values:

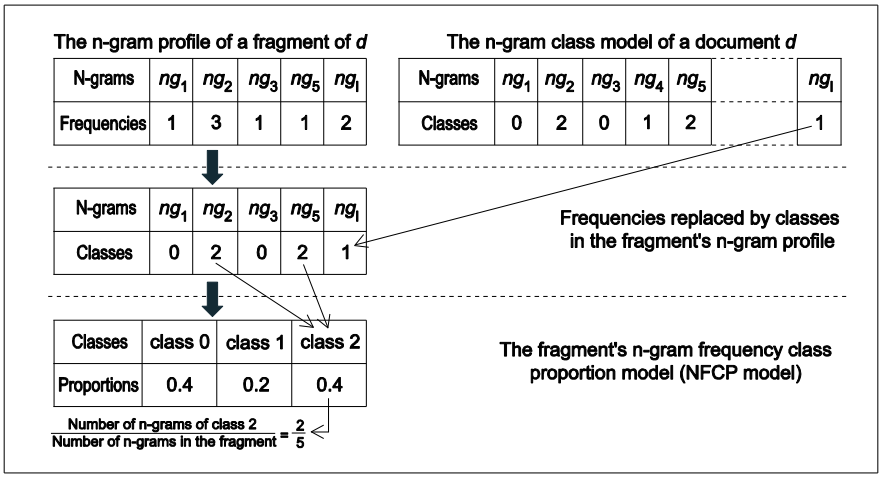
[6] Numerals have not been considered when extracting n-grams.

- the frequency of n-grams in the document, and
- the proportion of n-grams of a given class in a fragment.

To compute the *frequency of an n-gram* ng_i in document *d*, we thought of two ways: (S0) counting the number of all the occurrences of ng_i in *d* or (S1) segmenting *d*, and counting the number the occurrences of ng_i such that it is considered once per each fragment. In the latter case (S1), the minimum value that could take the frequency is 1 if ng_i appears only in one fragment, and its maximum value is the number of fragments in *d* if ng_i occurs in each fragment.



Computing the NFCP features by considering the repetition of n-grams in the fragment



Computing the NFCP features without considering the repetition of n-grams in the fragment

Figure V-5. Illustration of two ways of computing the proportion of n-gram classes in a fragment

We presume that computing the frequency as stated in (S1) better reflects the distribution of an n-gram over the document. That is, this frequency indicates that the n-gram occurs in distinct parts of the document, as opposed to the frequency computed with the ordinary way (as stated in (*i*)) that increases even if the n-gram's occurrence is concentrated in one fragment.

As stated earlier, the second value that we computed using two different ways is the *proportion of n-grams of a given class in a fragment* (the NFCP feature). In the first way (R0), each n-gram is considered once in the fragment no matter how many times it appears in the fragment. In the second way (R1), however, each n-gram is weighted with its frequency in the fragment. Figure V-5 illustrates these two ways. In this example, the fragment is represented with three NFCP features extracted from complementary classes. For the sake of simplicity, although unrealistic, we suppose that the fragment contains only five n-grams.

As a result of computing the n-gram frequencies and the class proportion using two distinct ways for each, we obtain four variants of the feature extraction algorithm. Thus, the question is: which variant is the best in terms of the effectiveness of the generated features? To answer this question, we conducted an experiment that compares the features generated from each variant. The next section describes this experiment, and Table V-2 provides the notation adopted to represent each of the four variants.

3.3.2 Selecting the Best Variant

Experimental Setup

Since at this stage we still do not know what the configuration of the extraction method –in terms of *n*-gram length and *n*-grams classification granularity[7]– that leads to the best features, we extracted 540 NFCP features by varying *n* from 1 to 10 and *m* from 2 to 10. We performed

Table V-2. Four variants for extracting the NFCP features and their notation

		How the frequency of an n-gram is computed in the document?	
		All the occurrences of the n-gram	*Segmenting the document and considering one occurrence per fragment*
Is the n-gram repetition in the fragment considered when computing the proportion of classes?	*No*	S0R0	S1R0
	Yes	S0R1	S1R1

[7] We use the terms n-gram classification granularity and the number of n-gram classes interchangeably.

this extraction using the four variants of the extraction method described above. The idea is to evaluate globally[8] the relevance of the features extracted by each variant in comparison with the other variants. The variant that generates the most relevant features would be considered the best.

Let $\mathcal{F}_v$ be the sets of the 540 NFCP features extracted from a corpus of suspicious documents using the variant v of the extraction method such that $v \in$ {S0S0, S0R1, S1R0, S1R1}. To find out the best variant we followed the steps below.

(1) Computing the information gain (InfoGain) of each feature $F_i \in \mathcal{F}_v$ ($i \in [1..540]$ and $v \in$ {S0R0, S0R1, S1R0, S1R1}) to assess its contribution in the task of classifying the fragment as plagiarised or not (see the next subsection for further information on information gain).
(2) Computing the average of the InfoGain ($AvIG_v$) of all the features $\mathcal{F}_v$ extracted by using a particular variant v.
(3) Considering as the best variant of the extraction method the one whose $AvIG_v$ is the greater in comparison with the other variants.

We conducted the experiment using a subset of 2389 documents from the part of PAN-PC-10 dedicated to the IPD evaluation[9]. In the next subsection, we describe the information gain and our motivation to use it to select the best variant of the feature extraction method.

Information Gain

Given a training set where each example is represented with a set of features and labelled with its class, the information gain is an information-theoretic measure (Manning and Schütze 1999) that gauges the contribution of a feature in reducing the uncertainty on guessing the class. In other words, it measures the amount of information provided by the feature to predict the class correctly. The higher the information gain the more pertinent is the feature for the classification.

Formally, it is defined as the difference between the entropy of the class and the conditional entropy of the class knowing the feature (See Equation (4)). Note that the entropy $H(X)$ of a variable X is a well-known information theory measure that could be loosely defined[10] as a measure that reflects the quantity of the distinct values that can take the variable X. The computation of the entropy in the equations (5), (6) and (7) includes the estimation of the class prior probability $p(c)$, the probability of a given feature value $p(f)$ and the conditional

[8] We are saying globally because we do not focus at this stage on each feature singularly, instead we are interested in getting an overview on the worth of the set of features generated by each variant of the feature extraction method in order to discern the best one.

[9] We also evaluated the features on a corpus of Arabic Text (InAra-train). Since we obtained similar results on PAN-PC-10 and InAra we present in the result section only the evaluation on PAN-PC-10. Still, all our conclusions hold true for Arabic text.

[10] Some references define it as the amount of the disorder in the variable. The book (Manning and Schütze 1999, sec. 2.2 Essential Information Theory) provides an excellent explanation of the Entropy with examples.

probability of the class knowing the feature value p($c \mid f$), such that c is one value among the possible values of the target variable Class and f is one value among the input variable F_i (i.e. the feature).

$$\text{InfoGain}(\text{Class}, F_i) = H(\text{Class}) - H(\text{Class} \mid F_i) \tag{4}$$

$$H(\text{Class}) = -\sum_{c \in \text{Class}} p(c) \log_2(p(c)) \tag{5}$$

$$H(\text{Class} \mid F_i) = \sum_{f \in F_i} p(f) H(\text{Class} \mid F_i = f) \tag{6}$$

$$H(\text{Class} \mid F_i = f) = -\sum_{c \in \text{Class}} p(c \mid f) \log_2(p(c \mid f)) \tag{7}$$

As can be seen from the formulas, the information gain evaluates each feature independently from the others. This is different from other kinds of feature evaluation techniques that take into consideration the dependency between features or evaluate the worth of a set of features together, which is not our goal at this stage.

In our experiment, the target variable (Class) has two possible values, which are 1 and 0 to show whether the fragment is plagiarised or not, respectively. On the other hand, the values f of an NFCP feature F_i are the proportion of a particular class of n-grams in the fragment, which is a continuous variable. Since the Entropy deals only with discrete variables, the NFCP features are first discretised before the InfoGain calculation. We used Weka implementation of InfoGain, which automatically utilises Fayyad and Irani's (1993) discretisation method when the variable is discrete.

Results

Figure V-6 shows that the variant of the feature extraction method that produces the best features is S1R0. Generally speaking, features generated when considering an n-gram once per segment (S1) are better than using the usual way of computing the frequency (S0) (compare S1R0 vs. S0R0 and S1R1 vs. S0R1). In addition, we can notice from the figure that computing the proportion of n-gram classes without taking into account the repetition of n-grams in the fragment is the most beneficial (compare S1R0 vs. S0R0 and S1R1 vs. S0R1).

Through the above experiment, we discerned the variant of the proposed feature extraction method that leads to the most pertinent features. This would be helpful for researchers interested to use the proposed features. Moreover, we showed an alternative of computing the n-gram frequency in a document, which sounds to be more significant (at least in the context of NFCP features) when classifying the n-grams according to their frequency. Note that in all

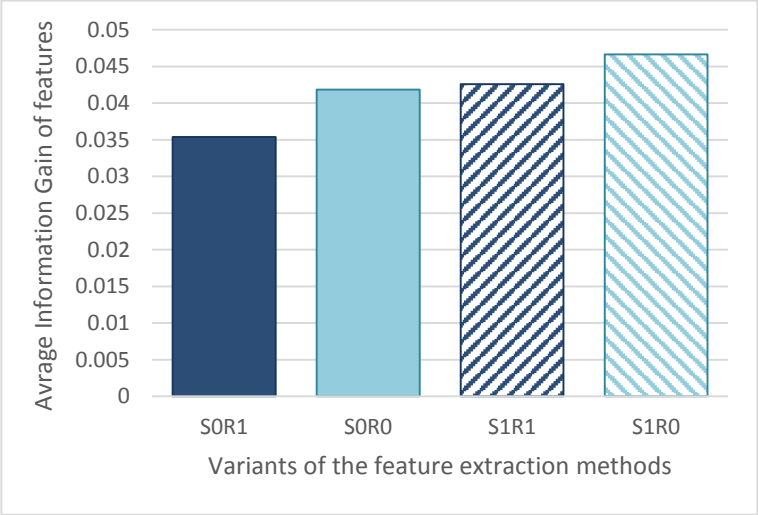


Figure V-6. Average of InfoGrain of the features generated by different variants of the extraction method

the subsequent experiments, we will adopt the best variant (S1R0) without mentioning that every time.

3.4 Plagiarism Identification

Once the suspicious document fragments are represented by features, a fundamental phase in the process of the intrinsic plagiarism detection is to decide whether a fragment is plagiarised or original. This phase has been implemented in the literature methods using different techniques, notably clustering (Kern et al. 2012), supervised classification (Meyer zu Eißen et al. 2007), comparing the values of a high-level feature with a threshold (Oberreuter and Velásquez 2013; Stamatatos 2009a) and density-based classification (Stein et al. 2011). (See Chapter IV, Section 4.4 for further details)

Our IPD method is based on supervised classification using Naïve Bayes[11]. Therefore, we built a training dataset where each fragment, $s_p \in S$, is represented by the target feature (Class) and a selected set of NFCP features. As mentioned earlier, the target feature value is either 1 (i.e., *plagiarised*) if the intersection between s_p and the plagiarism cases annotated in the corpus exceeds 50% of s_p length in characters, or 0 (i.e., *original*) otherwise. Subsequently, we used the training dataset to construct the classifier, which is then employed to identify the plagiarised fragments in any given document.

[11] The used implementation of Naïve Bayes is the one of the software Weka (Hall et al. 2009). We trained and tested other classification algorithms implemented on Weka software, and the best results were obtained with Naïve Bayes.

Table V-3. Statistics on the evaluation corpora

	PAN-PC-09	PAN-PC-10	PAN-PC-11	InAra-Training	InAra-Test
Language	English	English	English	Arabic	Arabic
# documents	3092	4766	4753	1024	1024
# plagiarism cases	10471	12851	11443	2833	2714

4 Datasets and Performance Measures

We used for our experiments three evaluation corpora in English and one corpus in Arabic with its two parts training and test. The English corpora (Potthast et al. 2010c) have been developed for the international competition on plagiarism detection (PAN)[12] of the years 2009, 2010 and 2011 to evaluate the IPD methods (Potthast et al. 2009, 2010a, 2011). We used specifically the test part of each corpus[13]. The Arabic corpus (InAra) (Bensalem et al. 2013a, 2013b) has been built by ourselves, following PAN annotation standards, and has been used in AraPlagDet 2015[14], the first plagiarism detection competition on Arabic documents (Bensalem et al. 2015).

These corpora are collections of annotated suspicious documents which have been created automatically by inserting, within a set of mono-authored documents (host documents), passages of different lengths borrowed from other texts. The inserted passage and the host document should have similar topics but written by different authors. Moreover, these suspicious documents comprise only verbatim cases of plagiarism. This is because disguising plagiarism may alter its writing style, which may further complicate its identification by the intrinsic approach. Table V-3 shows the statistics of the used corpora. Refer back to Chapter III for further information on plagiarism detection evaluation corpora.

As regards the performance measure, we use the F-measure for all the experiments in this chapter, which is the harmonic average of the precision and recall[15]. Precisely, we use a version of precision and recall adapted by Potthast et al. (2010c) for plagiarism detection evaluation. In these tailored measures, which became a standard for evaluating plagiarism detection methods, the plagiarised and detected fragments are expressed in terms of their lengths in

[12] http://pan.webis.de

[13] The corpora could be downloaded from: https://webis.de/data/data.html#pan-corpora

[14] http://misc-umc.org/AraPlagDet

[15] There is another performance measure of plagiarism detection, which is the granularity. This measure does not gauge the efficacy of the method to spot plagiarism but instead its ability to merge the overlapping and the adjacent detections into one segment. We did not use this measure in our experiments because it is rather sensitive to the post-processing methods used to merge the identified plagiarism cases, which is outside our experiments' scope.

characters. More precisely, we used the macro-averaged version where the precision and recall are computed at the fragment level and then averaged. Their formulas are presented in the equations 8 and 9[16], where *Act* is the set of the plagiarism cases annotated in the corpus (the *Act*ual cases) and *Det* is the set of the plagiarism cases detected by the method (the *Det*ected cases). Let s_{act} denote an actual case, and let s_{det} denote a detected case. The symbols $|s_{act}|$ and $|s_{det}|$ are, respectively, the lengths of s_{act} and s_{det} in characters. The symbols $|Act|$ and $|Det|$ are the number of actual and detected cases respectively.

$$precision(Act, Det) = \frac{1}{|Det|} \sum_{s_{\mathrm{det}} \in Det} \frac{\left| \bigcup_{s_{\mathrm{act}} \in Act} (s_{\mathrm{act}} \cap s_{\mathrm{det}}) \right|}{|s_{\mathrm{det}}|} \quad (8)$$

$$recall(Act, Det) = \frac{1}{|Act|} \sum_{s_{\mathrm{act}} \in Act} \frac{\left| \bigcup_{s_{\mathrm{det}} \in Det} (s_{\mathrm{act}} \cap s_{\mathrm{det}}) \right|}{|s_{\mathrm{act}}|} \quad (9)$$

5 Evaluation of the NFCP Features-based Method

The proposed feature extraction method allows extracting, through one configuration, ⟨*n, m*⟩, as many NFCP features as the chosen number of classes, *m,* from the n-grams of a determined length, *n,* of a given document. Let us call these features the complementary NFCP features since they are extracted from complementary n-gram classes.

The first idea that came to our mind to evaluate our assumption that the proportions of the n-gram classes are relevant to identify plagiarism is to represent the fragments by *m* complementary NFCP features. Therefore, we created 90 training sets by parameterising the extraction method with all the possible pairs ⟨*n*, *m*⟩ ∈ [1..10] × [2..10]. The parameters adopted for the test on English and Arabic texts are, respectively, the ones that yielded the highest F-measure through validation on PAN-PC-10 and InAra-Training, namely ⟨4, 3⟩ for English texts and ⟨1, 8⟩ for Arabic texts.

We tested the method on PAN-PC-09, PAN-PC-11 and InAra-Test. On the two English corpora, we compared it with Stamatatos' (2009a)[17] and Oberreuter and Velásquez's (2013) methods, which are the top-ranked methods in PAN09 and PAN11 competitions, respectively, and also with the method of Kestemont et al. (2011), being a character n-grams based method that has been evaluated on both corpora. On the Arabic corpus, the comparison is made with the method of Stamatatos (2009a)[18] and the method of Mahgoub et al. (2015), which, to the

[16] Those are the same formulas provided in Chapter III (p. 45). We put them here to make the chapter self-contained.

[17] The results of Stamatatos' method on the PAN-PC-11 corpus are available in (Potthast et al. 2011).

[18] The evaluation of Stamatatos' method on InAra-Test is performed by ourselves using the original implementation of the method (and the same parameters used for English) with an adaption of the code to support the Arabic language.

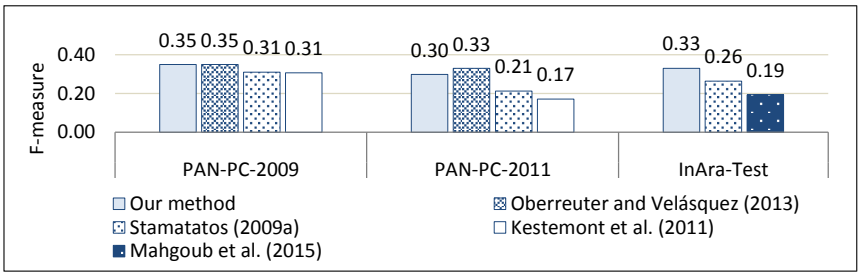


Figure V-7. F-measure of our method in comparison with the best methods in the PAN intrinsic plagiarism detection competitions

best of our knowledge, is the only method tested previously on the InAra-Test corpus[19]. The methods of Stamatatos and Kestemont et al. are both based on a style dissimilarity function computed on character 3-grams as outlined in Section 2.2, whereas Oberreuter and Velásquez's method compares word frequencies between the whole document and its segments. As for Mahgoub et al.'s method, it is based on computing the cosine distance between the document and its fragments represented by some syntactical and lexical features, such as parts of speech and stop words frequencies.

As shown in Figure V-7, the performance of our method is comparable to that of state-of-the-art approaches, which indicates that the NFCP are promising features to identify the style change in a fragment. This confirms our assumption concerning the usefulness of these features to mark the potentially plagiarised fragments.

6 Sensitivity Analysis of NFCP Features Performance to N-grams Frequency and Length

In this section, we examine the performance of the NFCP features extracted from different classes of n-grams. This examination is important for three reasons:

- The first reason is to optimise the performance of the proposed feature extraction method. Therefore, one can use it readily without going through a tuning phase of the parameters $\langle n, m \rangle$.
- The second reason is to gain insight into the relationship between the frequency of character n-grams and plagiarism. In Section 3.1, we presented two descriptions of plagiarism based on character n-grams: (1) it is the passage wherein we notice the presence of infrequent n-grams or (2) it is the passage wherein we notice the lack of frequent n-grams. However, it is still unknown which of them is the most pertinent description. In other words, what is the most relevant characteristic of a plagiarised fragment in terms of n-gram classes? Is it

[19] In the AraPlagDet competition, participants were more interested in the external plagiarism detection approach.

its relatively high proportion of the low-frequency n-grams or its relatively small proportion of the high-frequency n-grams? Or maybe the proportion of medium frequency n-grams is the most discriminative. Alternatively, all n-grams, whatever their frequencies, may be equally important. The experiments in the present section allow answering these questions.

— The third reason is to help choose the best performing *subset* of NFCP features (see Section 6.3).

6.1 Experimental Setup

As stated earlier, our approach of computing n-gram classes deals with two parameters ⟨*n, m*⟩, which represent the length of n-grams and the number of classes, respectively. Since our goal is to study the effect of n-grams' frequency and length on the performance of NFCP features, we extracted features by using all the possible values of the pair ⟨*n, m*⟩ ∈ [1..10] × [2..10]. That is, each document is represented with ten distinct n-gram profiles corresponding to the different n-gram lengths (from 1 to 10). Then, the n-grams of each profile are categorised into 2 classes, 3 classes …, and 10 classes. Therefore, the total number of classes (and consequently NFCP features) obtained from n-grams of a chosen length is 54 ($\sum_{m=2}^{10} m$). Since our experiments concern ten different n-gram lengths, the total number of the resulted classes is 540 (54 × 10). We name the classes labelled 0 the *low-frequency classes*, and we name the classes labelled *m-1* the *high-frequency classes*. The remainder of the classes are named *medium-frequency classes*. See Figure V-8 for an illustration.

In total, features have been extracted from 12611 English documents and 2048 Arabic documents including 34765 and 5547 plagiarism cases, respectively. Once the 540 features have been extracted, we evaluated the performance of each of them separately from the others. Practically, for each language, a total number of 540 classifiers (in each iteration), corresponding

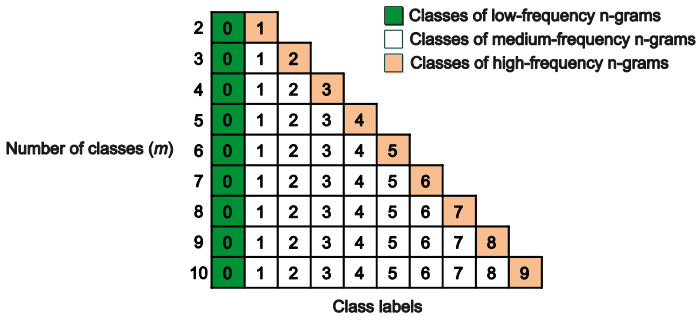


Figure V-8. The 54 classes obtained from the n-grams of a document by classifying them into different number of classes, *m*. For example, when *m* = 2 (the top of the figure), this means that the n-grams of the document are classified into 2 classes labelled 0 and 1. The former represents n-grams of low frequency, and the latter represents n-grams of high frequency

Table V-4. Evaluation setting of NFCP features

Training	Test	
PAN-PC-09	PAN-PC-10 PAN-PC-11	Iteration 1 Iteration 2
PAN-PC-10	PAN-PC-09 PAN-PC-11	Iteration 3 Iteration 4
PAN-PC-11	PAN-PC-09 PAN-PC-10	Iteration 5 Iteration 6
InAra-Training	InAra-Test	Iteration 1
InAra-Test	InAra-Training	Iteration 2

to the 540 NFCP features, have been trained and tested using the five datasets described in Section 5. Explicitly, cross-validation has been performed between each couple of corpora, i.e., each corpus is used separately, on the one hand, for training a classification model and on the other hand, for testing the models trained on the other corpora of the same language. Consequently, we obtained for each NFCP feature six classification results on English corpora and two classification results on the Arabic corpus as illustrated in Table V-4. Then, the F-measure scores are averaged for each language to be used in our analysis.

6.2 Results and Discussion

6.2.1 Sensitivity to N-gram Frequency Classes

Figure V-9 depicts the distribution of the F-measure of low-, medium-, and high-frequency NFCP features. As shown in the figure, half of the least-frequent features have an F-measure of more than 0.28 and 0.17 on English and Arabic corpora, respectively. However, more than half of the medium and high-frequency features perform poorly as illustrated through their lower medians in comparison with the median of the least-frequent features. The high-frequency features, notably, are the most likely to perform poor as 75% of them have an F-measure less than 0.17 in English texts and less than 0.09 in Arabic texts.

We can conclude from the above observations that the n-grams of a given fragment that do not appear frequently in the document are likely to assist in deciding whether it is plagiarised or not more than its n-grams that appear more frequently. In other words, the more an n-gram is frequent in the document, the less likely it is to be effective in detecting plagiarism intrinsically using NFCP features method.

Finally, it is also interesting to observe that performance scores of the features in each super-class (i.e., low, medium or high) are spread out on relatively large intervals. We can see in the figure that in the same super-class (in both Arabic and English corpora), there exist

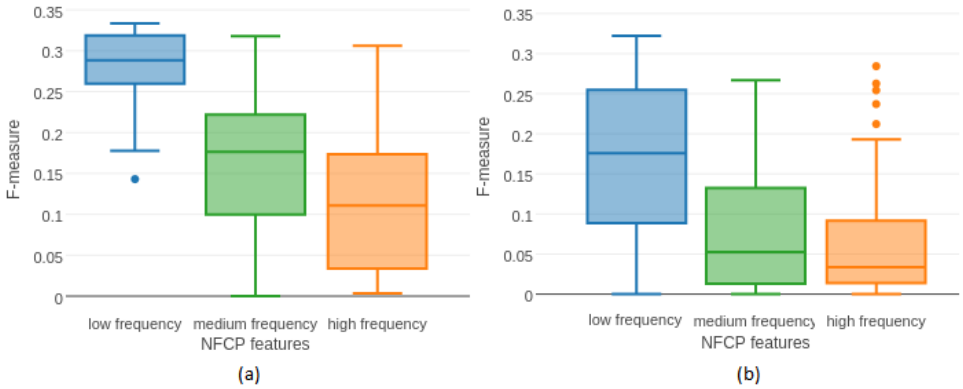


Figure V-9. The distribution of performance of the NFCP features computed on English text (a) and Arabic text (b)

features that reached an F-measure higher than 0.25 and other features with an F-measure lower than 0.15. This indicates that the NFCP features performance is influenced not only by the frequency of the selected n-grams (high, medium or low) but also by other parameters. Those parameters could be the *number of classes* into which n-grams are classified according to the frequency, which affects the number of the n-gram in each class, and obviously the *length of n-grams*. In the next subsections, we discuss the sensitivity of performance to these two parameters.

6.2.2 Sensitivity to the Number of Classes

The question addressed in this section is: when classifying n-grams into *m* classes in an experiment and into *m'* classes in another experiment, will the performance of the NFCP features extracted from the same super-class (e.g. the class of low-frequent n-grams) in both experiments be the same?

The graphs in Figure V-10 represent line charts of the performance of the NFCP features as a function of the number of classes. The features of each super-class are plotted in separate graphs. Each line relates the performance of the features extracted from the same n-gram length.

Recall that each point in the graphs is the average F-measure of one NFCP feature, which is computed using the scores obtained from the different test iterations. However, for the graphs of the medium-frequency features, each point represents the average F-measure of two or more features when the number of classes is greater than 3. For example, classifying n-

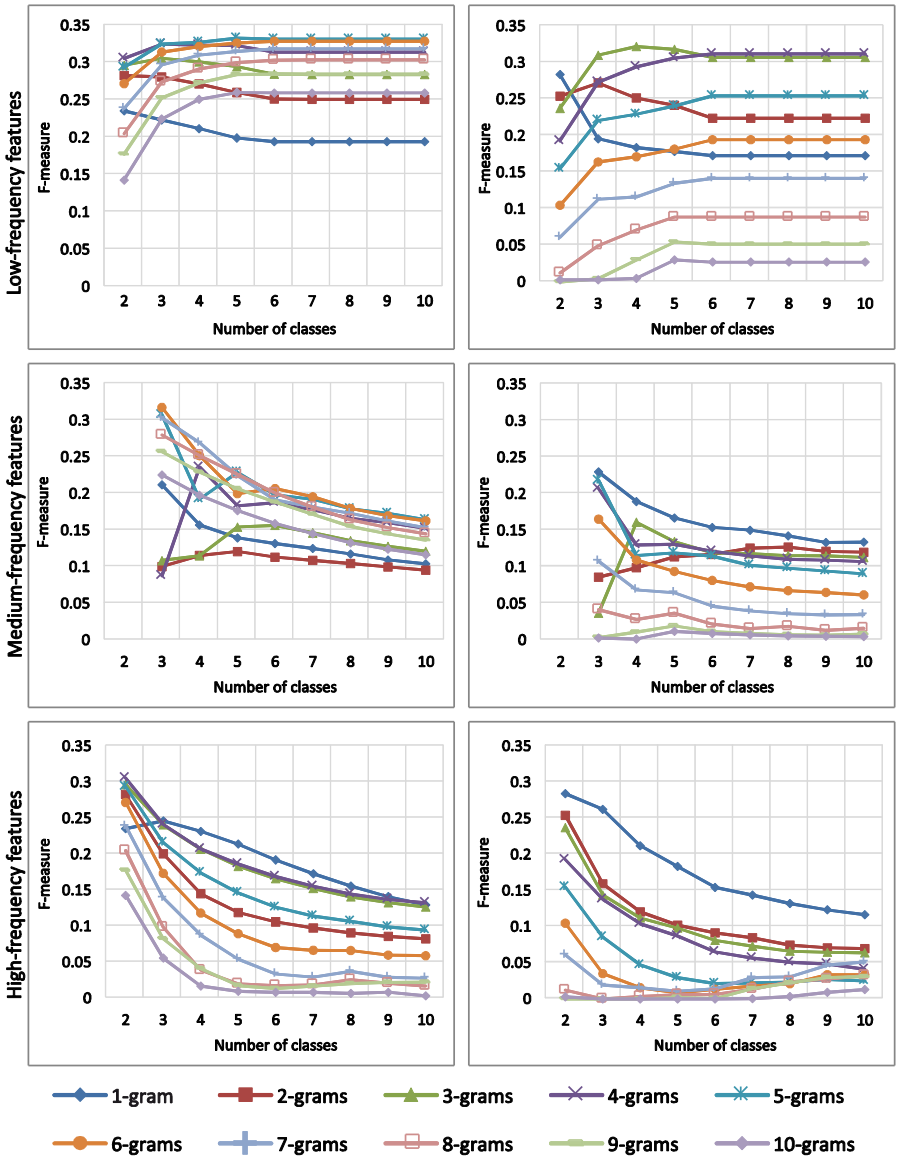


Figure V-10. Sensitivity of NFCP features performance to the number of classes on English (left) and Arabic (right)

grams into four classes produces two medium-frequency features. Therefore, what is plotted, in this case, is the average performance of the classes labelled 1 and 2.

The graphs become easier to interpret by keeping in mind that the parameter *number of classes* (*m*) controls the number of n-grams in the obtained classes. Therefore, the increase in *m* on the y-axis of Figure V-10 can be interpreted as a reduction in the number of n-grams

from which the NFCP feature is extracted.

It can be seen from the graphs that the performance increases or decreases between 2 classes and 6 classes, then it stabilises (with the low-frequency features) or continues to change slowly (with the medium- and high-frequency features) when *m* is above 6.

A more in-depth observation of the graphs reveals that the sensitivity of performance to the number of classes varies according to the length of n-grams. For instance, to obtain the best low-frequency features, we need to classify n-grams into *few classes* ($m \leq 4$) if the n-grams are relatively *short* ($n \leq 4$ for English and $n \leq 3$ for Arabic), but *m* shall be *equal or greater than 6* if the n-grams are *longer*. Another example could be observed in the medium-frequency classes, where 2- and 3-grams in Arabic in addition to 4-grams in English are not following exactly the general patterns.

From the above comments, we can conclude that whatever the length of n-grams, there is no benefit from classifying them into more than 6 classes because this generates NFCP features that are either similar to or worse performing than the ones extracted from a smaller number of classes. Nonetheless, the optimal size of a class (which is controlled by the chosen number of classes) depends on the frequency of n-grams as well as their length. In detail, to obtain the best low-frequency features, we recommend classifying n-grams into 6 classes, except for the short n-grams as explained in the previous paragraph. On the other hand, we obtain the best medium- and high-frequency features by classifying n-grams into 3 or 2 classes, respectively (with some exceptions as stated in the previous paragraph). Note that when n-grams are classified into only two classes – which is the configuration that produces the best NFCP features extracted from the high-frequency n-grams – the generated NFCP features from these two classes will be similar since the proportion of the high-frequency n-grams in a fragment is one minus the proportion of the low-frequency n-grams. All the above remarks are applicable for both Arabic and English.

To elucidate the findings above using more-general words, let us recall again that the *number of classes* is a parameter specific to our method that allows controlling the number of n-grams in each class, which is in turn related to the frequency range of n-grams in this class (see Section 3.1.1). Based on that, the experiments described in this section are an attempt to understand the variation of the performance of the NFCP features according to the size and the frequency range of the selected subset of n-grams. The above findings recommend considering a large number of n-grams to extract the best NFCP features from the *high-frequency* n-gram (*regardless of their length*) or the *low-frequency short* n-grams[20]. In contrast, the frequency range of the *low-frequency long* n-grams producing the best NFCP features should be as small as possible (i.e., only the n-grams that occur once).

[20] As detailed in previous paragraphs, in this context, short n-grams means $n \leq 3$ or $n \leq 4$ for Arabic and English, respectively. The rest are called long n-grams.

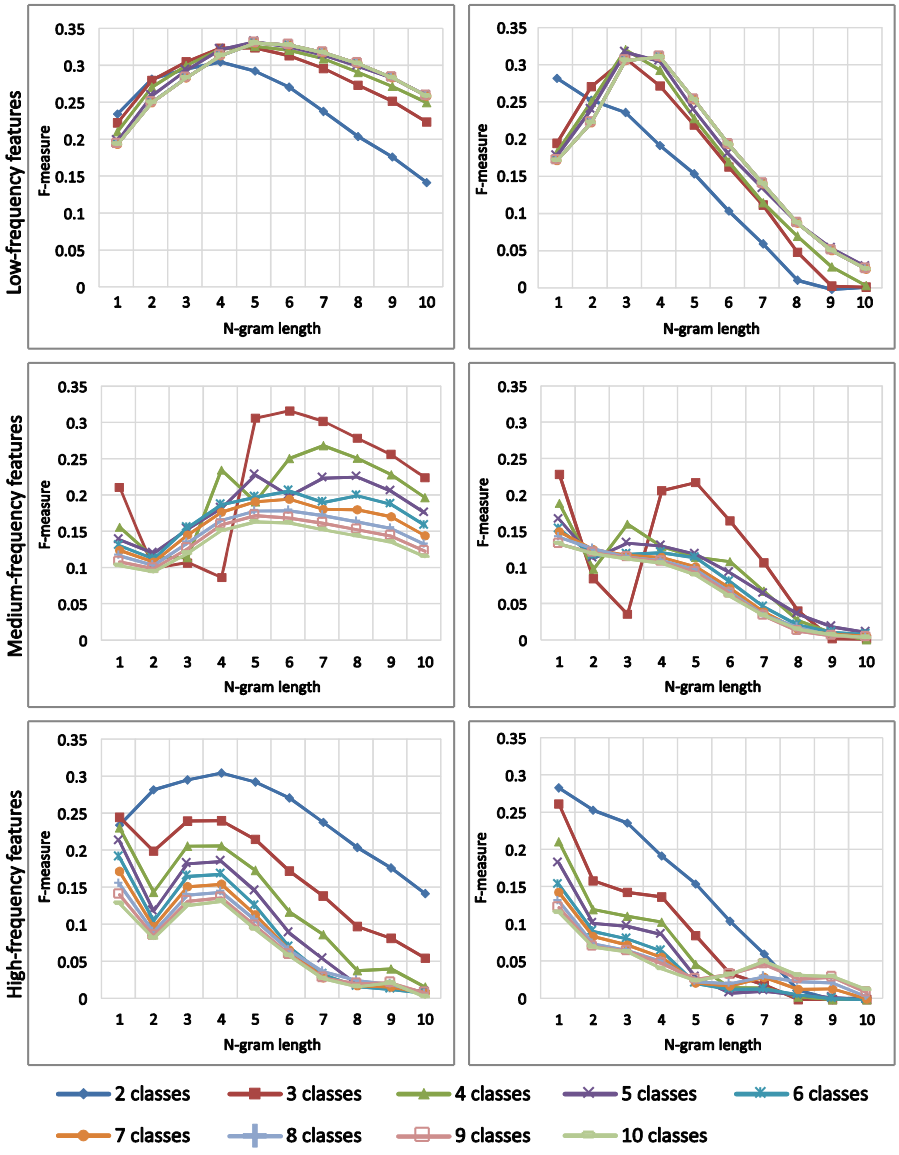


Figure V-11. Sensitivity of NFCP features performance to the n-gram length on English (left) and Arabic (right)

6.2.3 Sensitivity to N-gram Length

The graphs in Figure V-11 represent line charts of the NFCP features' performance as a function of the n-gram length. The F-measure plotted in this figure has been computed by applying the same averaging procedure as the one used for Figure V-10.

The graphs show that the middle-sized *low-frequency n-grams* outperform the short and the long n-grams. This remark is true in both English and Arabic corpora.

With regard to the *high-frequency n-grams*, generally speaking, the longer the n-grams, the smaller the performance of the related features in Arabic and English[21]. This observation means that representing the suspicious document fragments with the proportion of the high-frequency short n-grams is more helpful in detecting plagiarism than representing them with the proportion of the high-frequency long n-grams.

The best performance of the features based on *medium-frequency n-grams* regardless of the number of classes is reached with unigrams in Arabic (as for the high-frequency n-grams). In English, n-grams of 5 to 7 characters are the best.

A detailed observation reveals that the sensitivity to n-gram length is related to the language. That is, Arabic and English do not have exactly the same pattern of sensitivity to n-gram length. The best performing NFCP features are obtained with medium length n-grams in English (from 4 to 6). In Arabic, they have been obtained with even shorter n-grams (1-, 3- and 4-grams). Moreover, it seems that Arabic is more sensitive than English to the length of n-grams, for example, the long n-grams perform very poorly in Arabic: beyond 6-grams all the features have an F-measure under 0.2, which is not the case in English. Indeed, Arabic and English are different in terms of the distribution of word lengths. This distribution may have an impact on the meaningfulness of the linguistic information captured by the n-grams of a certain length. For example, most of the Arabic words are derived from roots of three characters. Consequently, many 3-grams represent word roots in Arabic, which is not the case in English. This fact, probably, explains the difference between the optimal parameters of the two languages.

6.3 Combining NFCP Features

The experiments described in this section investigate the best performing *subset* of the NFCP features. Indeed, we attempted to address this question in Section 5 by searching the optimal subset of the complementary NFCP features exclusively. In this section, however, the features to combine are selected either on the basis of their individual performance (reported in Section 6) or by applying some well-known filter or feature reduction methods. In detail, the experiments we conducted are:

A. Selecting the best feature of each n-gram super-class: In this experiment, we combined three features; each one is the best of the low-, the medium- and the high-frequency NFCP features, respectively.

B. Selecting the best feature of each n-gram length: In this experiment, we combined ten features; each one is the best NFCP feature extracted from n-grams of a particular length $n \in [1..10]$.

[21] There is an exception with features computed by classifying n-grams into 2 classes in English where peak performance has been reached with 4-grams.

Table V-5 The configurations that produce the best NFCP features

	n	*m*	n-gram class
English	5	5	o
Arabic	3	4	o

C. Using filter and feature reduction methods: More precisely, we used the principal component analysis (PCA), the correlation-based feature selection (Cbfs) and the information gain. We applied these techniques on 4 datasets where the text fragments are represented by different sets of the NFCP features, which are: (1) All the 540 NFCP features extracted by using the different configurations $\langle n, m\rangle \in [1..10] \times [2..10]$; (2) Only the high-frequency NFCP features (90 features); (3) Only the low-frequency NFCP features (90 features); (4) Only the medium-frequency NFCP features (360 features).

Since C involves several experiments, we will present only the results of the experiments that produce the best performance in each language, which are the PCA applied on the low-frequency NFCP features for English and the Cbfs applied on the low-frequency NFCP features for Arabic. A closer examination of the feature space resulted from using the above feature selection techniques revealed that the PCA reduced the 90 low-frequency features to one dimension, and the Cbfs retained only four low-frequency features.

In all the experiments, we trained and validated the classifiers on PAN-PC-10 and InAra-Training and tested them on PAN-PC-11, PAN-PC-09 and InAra-Test. Ultimately, we compared the results of the above feature selection experiments with the performance of the entire set of the 540 NFCP features, the best complementary NFCP features (reported in Section 5), and the best single NFCP feature for each language presented in Table V-5.

We observe from Figure V-12 that the feature selection has slightly improved our previous results reported in Section 5 (i.e. the best complementary features), notably in PAN-PC-11 and InAra-Test corpora where the performance increased from 0.30 to 0.34 and from 0.33 to 0.37, respectively. Interestingly, all the results obtained by feature selection, no matter which technique was used, are better than the results obtained by using the whole set of the 540 NFCP features without feature selection. This suggests that a solution based on NFCP features could be efficient since it is not necessary to use a broad set of features – which is computationally expensive – to achieve even better results.

Another interesting observation is that the best NFCP feature alone performs as good as some combination of features, notably in the English corpora. This could be attributed to the fact that an NFCP feature is a high-level feature extracted from many basic features and therefore, it is informative enough even when used alone.

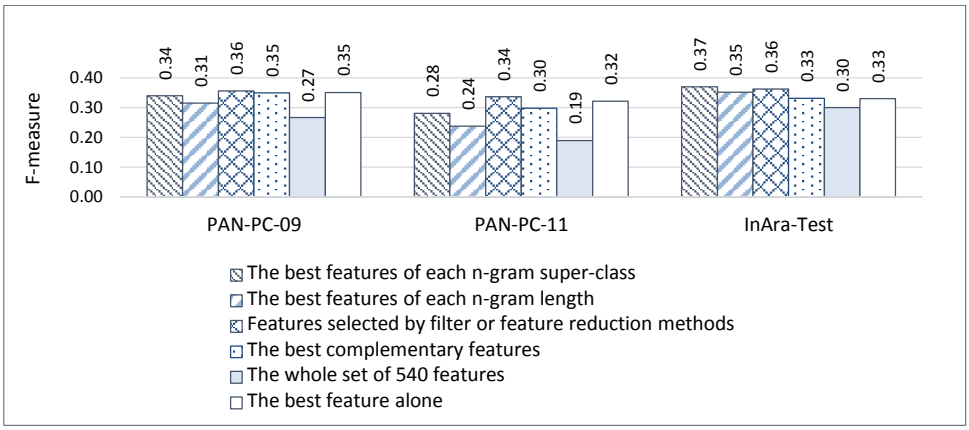


Figure V-12. Performance of combined NFCP features selected using different techniques

7 Sensitivity Analysis of Stamatatos' Method Performance to N-grams Frequency and Length

This section explores how selecting n-grams of a particular length according to their frequencies affects Stamatatos' (2009a) method performance. Thus, this exploration allows checking the possibility of improving the performance of the method by removing a subset of n-grams from the profile and/or changing the length of n-grams.

Before starting our analysis, let us remind the reader that our method and the one of Stamatatos utilise character n-grams to compute different high-level features: the proportion of the n-grams frequency classes and a dissimilarity measure, respectively. Therefore, comparing the analysis of this section with the one presented in Section 6 will enable us to discern whether the performance of a particular subset of n-grams is method-dependent or not. For instance, we showed that the least frequent n-grams produce the best NFCP features, but will they lead to optimal performance of Stamatatos' dissimilarity measure? Accordingly, addressing this question is another objective of this section.

7.1 Experimental Setup

Stamatatos' original method represents each document by almost all[22] its character n-grams regardless of their frequencies. Since we aim to analyse the effect of selecting n-grams on the performance of this method, we measured the variation of the F-measure according to the size of the selected set of n-grams. Therefore, we represented each document by sub-profiles of different sizes resulted from keeping only a proportion of the entire profile. Extracting the sub-

[22] Some non-alphabetic n-grams such as n-grams of numerals are discarded.

Table V-6. Cumulative percentages computed on the 3-grams of the suspicious-document01020 of PAN-PC-09

Cumulative percentage computed by starting from the least frequent n-grams			**Cumulative percentage computed by starting from the most frequent n-grams**		
N-gram's frequency *f*	# n-grams whose frequency = *f*	Cumulative percentage	N-gram's frequency *f*	# n-grams whose frequency = *f*	Cumulative percentage
1	412	69.36%	17	1	0.17%
2	101	86.36%	…	…	…
….	…	…	2	101	30.64%
17	1	100%	1	412	100%

profiles is based on the cumulative percentages that we computed on the frequency distribution table of the n-gram frequencies by starting once from the least frequent n-grams and once from the most frequent ones. See an example in Table V-6.

The size of the created sub-profiles is represented by a percentage X%, where X ∈ {10, 20, …, 90} (100% represents the full profile). A sub-profile is said to be of a size X% of the whole profile if the cumulative percentage of its n-grams belongs to the interval]X−10%, X%]. Note that if a document contains a large proportion of n-grams of a certain frequency, we cannot extract from it all the nine sub-profiles corresponding to the sizes indicated above. For example, in the document of Table V-6, the first sub-profile – created by starting the selection of n-grams from the least frequent ones – constitutes already almost 70% of the full profile. Therefore, the sub-profiles that comprise 10% to 60% of the n-grams are not created for this document because they will contain only a subset of n-grams whose occurrence is 1; however, we chose to create the sub-profiles by keeping (or discarding) all the n-grams of a particular frequency. Afterwards, if a sub-profile of a certain size could not be created for 25% or more of the total number of documents, we ignore the associated results.

We used the original implementation of the method[23] with the following modifications:

- We added a filter that cuts the profiles of the document and its fragments by taking into account the n-grams frequencies as explained above.
- We adjusted the size of the sliding window to be 1500 characters (instead of 1000)[24] in order to approximate to its size in our method, and so this parameter would less affect the analysis of the results.
- Since our experiments deal with Arabic in addition to English, and the original code supports only ASCII characters, we adapted the method to work with Arabic.

[23] We are so grateful to the author of the method Efstathios Stamatatos for sending us its code.

[24] We also adjusted another parameter of the method called *Real window length threshold* to 2250 instead of 1500 to make it appropriate to the new window size.

— For each experiment, we tuned the two parameters that control the plagiarism detection in the method (see the description of the method in Section 2.2) using around 200 documents from PAN09 competition training corpus for English texts (as done in the evaluation of the original method) and around 200 documents from InAra-Training for Arabic texts. We opted for the parameter-tuning phase instead of using the original parameters because preliminary experiments showed that the optimal parameters vary according to the sub-profile size and the length of n-grams. For instance, an experiment with the entire document's profile and another with 50% of it require the use of different parameters to achieve the best results. Likewise, the optimal thresholds used with 2-grams differ from those used with 4-grams. Hence, employing the same parameters for all the experiments may invalidate our analysis.

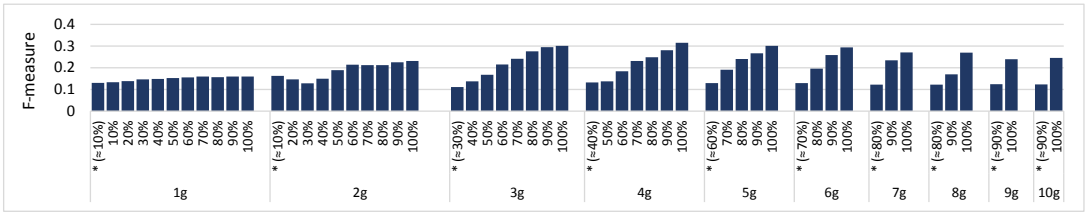
7.2 Results and Discussion

Each bar in Figure V-13 represents the average of the F-measure computed on the three PAN corpora for English and the two parts (training and test) of the InAra corpus for Arabic (as done in the previous experiments). Note that the performance associated with some sub-profile sizes is not depicted. For instance, there are no bars for some sub-profiles in the charts of 10-grams. This is because, for numerous documents, it was not applicable to create sub-profiles with these sizes as explained in the experimental setup.

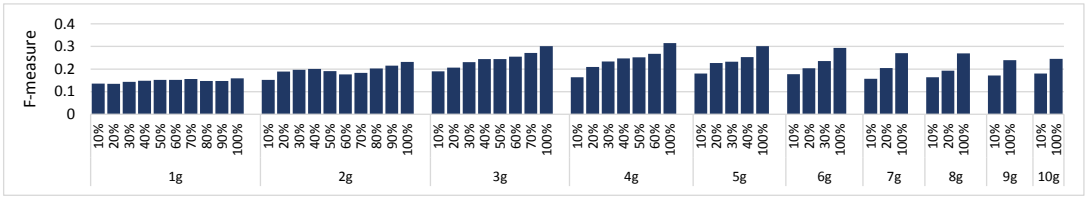
The charts show that the optimal performance of the method is attainable by representing the documents using all their n-grams. Accordingly, cutting the profile, either by keeping only the least or the most frequent n-grams, affects the performance negatively. To illustrate this fact, we compare the left bars of each n-gram length chart, which represent the performance of the least or the most frequent n-grams, with the extreme right bar, which depicts the performance of the full profile. Let us take the case of 4-grams on the English text. It can be seen that the F-measure obtained by using the full profile is 0.32, but it drops to 0.14 when keeping only the 50% least frequent n-grams (see the graph En-1) and to 0.25 when keeping only the 50% most frequent n-grams (see the graph En-2).

Despite the necessity to retain all the n-grams to reach optimal performance, it is worth mentioning that in this method the least frequent n-grams are less relevant than the most frequent ones. The following observations illustrate this statement:

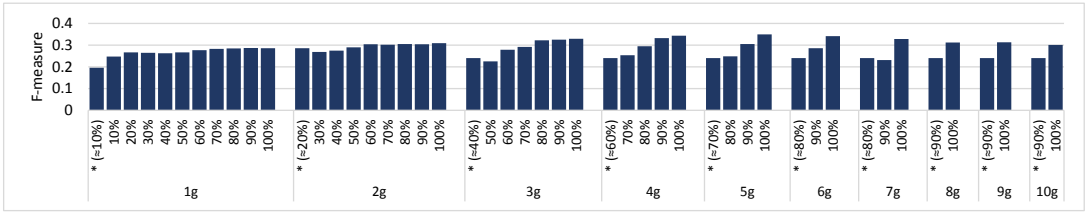
— For most n-gram lengths, the 50% most frequent n-grams outperform the 50% least frequent n-grams. For instance, see the charts of 3-grams for English and Arabic where this is readily noticeable (refer back also to the example mentioning 4-grams in the previous paragraph).
— The performance achieved by using only the 10% most frequent n-grams (see the extreme left bars of each chart in En-2 and Ar-2) is generally higher than the performance obtained by using the very least frequent n-grams (labelled with * in the charts En-1 and Ar-1),



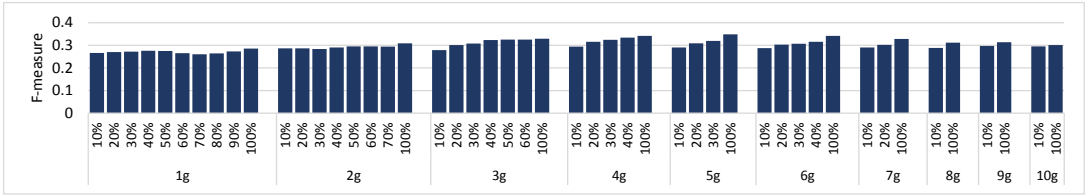
(En-1)



(En-2)



(Ar-1)



(Ar-2)

Figure V-13. Sensitivity of Stamatatos' method performance to the size of the selected subset of the n-grams (in percentage) and n-gram length. N-grams are selected from profiles sorted according to frequencies starting from the least frequent n-grams (En-1 and Ar-1) or the most frequent n-grams (En-2 and Ar-2). The performance is computed on English (En-1 and En-2) and Arabic (Ar-1 and Ar-2) documents. In the charts En-1 and Ar-1, the values of the x-axis labelled with an asterisk (*) represent the sizes of sub-profiles that contain only n-grams whose frequency = 1 whatever their proportion in the document's full profile.

which constitute a significant proportion of the profile, notably when $n \geq 3$.

- In Arabic documents specifically, it is obvious that the performance of a small subset of the most frequent n-grams (e.g., 10% of the full profile) is almost equal to the performance of the whole set of n-grams (see the graph Ar-2). Conversely, this is not the case for the least frequent n-grams (see the graph Ar-1).

Concerning the optimal n-grams' length, 4-grams and 5-grams yield the best performance

in English and Arabic, respectively.

Based on the above analysis, we recommend keeping in the profile all the n-grams regardless of their frequencies (as done in the original method) since they are all together essential to reach the optimal results of this method. If it is necessary to reduce the number of n-grams, then removing the least frequent n-grams would be less harmful than removing the most frequent ones, especially in Arabic documents.

A by-product of these experiments is the increase of the F-measure from 0.31 to 0.35 on PAN-PC-09, from 0.21 to 0.29 on PAN-PC-11 and from 0.26 to 0.33 on InAra-Test. These results are obtained by using 4-grams for English and 5-grams for Arabic and a window length of 1500 characters instead of the original configuration (3-grams with a window length of 1000 characters for both languages).

By comparing the behaviour of n-grams in this method and our method (described in Section 6), we can perceive that the n-grams that lead to the optimal results are not the same for the two methods. Below are some details:

- The least frequent n-grams alone produce the best NFCP features but a poor dissimilarity measure.
- It is not necessary to extract NFCP features from n-grams of different frequency ranges to attain competitive performance. However, achieving optimal performance of Stamatatos' method requires the use of all the n-grams regardless of their frequencies.
- The best n-gram length is specific to each method.

The conclusion we can draw from this comparison is that in the context of intrinsic plagiarism detection, the effectiveness of a subset of character n-grams in a method does not guarantee its effectiveness in other methods.

8 Conclusion

Although several papers have investigated the best ways of using character n-grams to solve diverse research problems, there is a lack of such studies in the context of intrinsic plagiarism detection. This chapter is an attempt to narrow this gap by examining the sensitivity of the intrinsic plagiarism detection performance to two factors: n-gram frequency and n-gram length. We conducted our study on five large collections of English and Arabic documents that have been used in the intrinsic plagiarism detection competitions of the PAN Lab.

Our experiments manipulated two intrinsic plagiarism detection methods that are based exclusively on character n-grams, although these low-level features are exploited in each method differently. The first method, which is the one we presented in this chapter, classifies the n-grams according to their frequencies in the given suspicious document. Then, it represents each fragment of the document by the proportion of its n-grams belonging to a particular class. We called this proportion the NFCP (N-gram Frequency Class Proportion)

feature. The second method (Stamatatos 2009a), which is a seminal state-of-the-art method, represents the suspicious document fragments by a dissimilarity measure between their n-grams and the n-grams of the entire document.

Concerning the first factor of our study, which is the n-grams frequency, our experiments showed that the best NFCP features are obtained from the least frequent n-grams. This means that the proportion of the least frequent n-grams (of a document) in its fragments is useful for marking the potentially plagiarised fragments. However, this class of n-grams (i.e., the least frequent ones) becomes less helpful in Stamatatos' method wherein the high-frequency n-grams contribute more, comparatively, to producing a discriminative dissimilarity measure. Besides, retaining all the n-grams, regardless of their frequencies, is the way to achieve the optimal performance of this method. Taken together, these results show that the relevance of a subset of character n-grams (selected based on their frequencies) to characterising plagiarism is not absolute. It is rather relative to how the n-grams are harnessed. In other words, the performance of a subset of character n-grams, selected according to their frequencies, in intrinsic plagiarism detection is method-dependent.

Concerning the second factor of our study, which is the n-grams length, our results are in line with the fact that the optimal length varies according to the language. Moreover, our experiments showed that this parameter is also method-dependent. That is, even in the same language, the optimal n-gram length varies for each method.

On the other hand, the experiments described in this chapter demonstrated the possibility to achieve state-of-the-art performance by using the character n-grams solely. Nevertheless, we believe that it would be beneficial to utilise them along with other features to capture further characteristics of plagiarism that might be missed when representing the text by the character n-grams alone. In this context, the NFCP features and Stamatatos' dissimilarity measure, being high-level features that encapsulate many n-grams in a single value, are well suited to be used alongside other features in machine learning-based methods while avoiding the curse of dimensionality. Thus, our study is a roadmap for researchers interested in including character n-grams into intrinsic plagiarism detection methods.

Finally, as future work, it would be interesting to apply the idea of the proportion of the frequency classes to word uni-grams[25] (or lemma uni-grams)[26], especially that the length of the character n-grams that leads to the best performing NFCP features (5-grams in English and 3-grams in Arabic –see Table V-5) is around the average word length in English and the length

[25] Thank you to the reviewer of the thesis Dr Alberto Barrón-Cedeño for this perspective linked to the findings of our experiments.

[26] Note that the proposed frequency classes of words should not be confused with those used to compute the *average word frequency class* (Meyer zu Eißen and Stein 2006), which is a vocabulary richness measure where the class of a word is computed differently based on its frequency in a corpus, and not in a single document.

of lemmas in Arabic[27]. Another future work is to employ the NFCP features in other tasks whose goal is the textual outlier detection such as authorship verification.

[27] The average word length in English is 4.79 (see English Letter Frequency Counts: Mayzner Revisited or ETAOIN SRHLDCU in http://norvig.com/mayzner.html). In Arabic, most of the words are derived from roots of 3 letters.

Chapter VI. Conclusions

> *If words were not repeated, they would have run out.*
>
> *Ali ibn Abi Talib (632–661)*[1]

Writing this thesis was a journey into two specific research areas within the main domain of plagiarism detection. The first area (which was the subject of Chapters II and III) is *Arabic plagiarism detection* and the second area (which was the subject of Chapters IV and V) is *intrinsic plagiarism detection*. This chapter summarises our contributions (in Section 1) and discusses some prospects and future works (in Sections 2 and 3).

1 Summary of the Contributions

1.1 Contributions in Arabic Plagiarism Detection

(i) *A shared task and evaluation datasets.* Our first contribution in the context of Arabic plagiarism detection is providing the research community with datasets of Arabic artificial suspicious documents to evaluate both the external and the intrinsic plagiarism detection tasks. We released these datasets through the PAN-AraPlagDet shared task that we organised at the FIRE 2015 conference. These datasets served to evaluate several works during and also after the shared task. Chapter III described in detail our experience of building the evaluation corpora and organising the shared task.

(ii) *Quality appraisal of the publications.* Quality is an important aspect of research works that we discussed in Chapter II. We did not plan to discuss this aspect in the thesis. However, the numerous low-quality papers we came across during the preparation of the literature survey

[1] Ali ibn Abi Talib is the cousin and son-in-law of Muhammad, the prophet of Islam (https://en.wikipedia.org/wiki/Ali). The original quotation in Arabic is "لولا أن الكلام يعاد لنفذ".

was indicative of a serious problem of quality and encouraged us to undertake a quality appraisal of the existing publications. We showed through this study that around 70% of the existing papers on Arabic plagiarism detection suffer from quality issues, most notably, issues related to the evaluation methodology of methods that render the concerned papers unreliable and unworthy for the advancement of the field. This result not only endorses the need for a trustworthy evaluation dataset, which is –as previously mentioned– one of our contributions, but also it is an alarm to the necessity to change the current research practices in the Arab region.

1.2 Contributions in Intrinsic Plagiarism Detection

(i) ***New IPD method.*** We developed an IPD method based on a novel way of using character n-grams. This method classifies the character n-grams according to their frequencies in the analysed document and uses the proportions of the obtained classes in each fragment as stylistic features. The underlying hypothesis behind the proposed features is that the proportion of the low- and the high-frequency n-grams in a text fragment can be indicative of a writing style change within the fragment, which means a potential plagiarism case according to the intrinsic approach assumption. The performance of the proposed method, which is comparable to the state of the art, confirmed our hypothesis.

The proposed features have two advantages. First, since they are based on the hypothesis above, it contributes to augmenting our understanding on what makes the originality of a text fragment suspect[2] without comparing it with external texts. This understanding facilitates the interpretation (and hence the communication) of the detection results in the absence of the strongest proof of plagiarism, which is the source[3]. Second, the proposed features, which are a reduced form of the common way of using n-grams, allow the exploitation of the character n-grams in the machine-learning approaches while avoiding the curse of dimensionally.

(ii) ***Investigating the best parameters of using character n-grams in IPD.*** We conducted a series of experiments that aim to answer the question: What are the best character n-grams in terms of their frequency and length for intrinsic plagiarism detection? The experiments were carried out using the proposed method and the seminal method of Stamatatos (2009a). The results show that there is no such thing as unique optimal frequency and length of character n-grams for the task of intrinsic plagiarism detection. Instead, the best values of these two parameters can be different across the methods and languages. Our work has practical

[2] Specifically in terms of character n-grams.

[3] To illustrate, in the absence of the source, a teacher should be able to explain the reasons for his doubts about the originality of a certain passage in a student essay. By using an IPD software based on the proposed features, the teacher can show the student that the suspicious passage contains, e.g., a high proportion of characters or character sequences (i.e., n-grams) that are infrequent in the rest of the essay. This facilitates the next step that aims to confirm or reject the doubts, which can be for example to ask the student to justify the presence of these n-grams (e.g., certain punctuation marks or words) specifically in that passage.

implications for researchers and practitioners interested in the use of character n-grams for IPD.

All our intrinsic plagiarism detection experiments, which are detailed in Chapter V, have been conducted on the proposed IPD Arabic corpus in addition to the PAN shared task English corpora. We are indeed the first who tested the intrinsic approach on Arabic documents.

(iii) *The first comprehensive survey on IPD.* Besides the experimental work, our third contribution (See Chapter IV) is providing the research community with the first survey of intrinsic plagiarism detection literature, which is a comprehensive repertoire of the building blocks of the current methods.

2 Future Work on Arabic Plagiarism Detection

The main takeaway from our research on Arabic plagiarism detection is the necessity to shift the focus from the development of prototypes[4] to the development of end-product software packages. In fact, despite the growing number of publications on Arabic plagiarism detection, the tools dedicated/adapted to support the Arabic language are very few. Therefore, future research in this area has to deal with questions that serve the practitioners. In the conclusion of Chapter III, we discussed this point. Below the questions we raised and that can be the subjects of future studies:

- To which extent can we rely on a detector designed for English to detect plagiarism in Arabic documents?
- What are the weaknesses of the well-known plagiarism detection tools (e.g., Turnitin[5]) when applied to Arabic texts?
- Is adapting an existing tool to Arabic peculiarities worth the effort in terms of improving significantly the performance?
- Do solutions specific to Arabic perform better than the language-independent ones?

In a more general context, which is the detection of text reuse, it would be interesting to identify the religious quotations and anecdotes used over the years in many ancient Arabic books. Detecting such text reuse cases is important not only in the context of plagiarism detection (to reduce the false positives) but also in the linguistic studies of the Arabic language history such as the work of (Belinkov et al. 2019)[6]. Thus, creating evaluation corpora where religious quotations and common anecdotes are annotated would be essential to evaluate the performance of models dedicated to detect them.

[4] This issue is also pointed out in Kahloula and Berri (2016) (the first author is the developer of almikshaf.com tool) who wrote : "Much of the research undertaken for the detection of plagiarism in Arabic documents has unfortunately led only to prototypes".

[5] https://www.turnitin.com

[6] Thank you to Dr Alberto Barrón-Cedeño for drawing our attention to (i) the application of text reuse detection in the context of the periodization of the Arabic language and (ii) the need to evaluate its models.

3 Intrinsic Plagiarism Detection: Current Challenges and Research Prospects

The main takeaway from our research on intrinsic plagiarism detection is our observation that the challenges of this task are caused by two kinds of barriers:

(i) the *inherent constraints* of IPD, which make this task a difficult research problem, and

(ii) a number of *assumptions* made in the current works, which limited the application cases of this approach.

In the rest of this chapter, we shed light into these *inherent constraints* and *assumptions*, and we provide our arguments that the prospects of this approach reside in overcoming these barriers.

3.1 IPD Beyond its Inherent Constraints

We called them inherent constraints because they are related to the nature of the problem. These inherent constraints are:

Constraint 1 **Uncertainty on the position of the writing style changes.** Plagiarism is mingled with the authentic text. In other words, the input of an IPD method is a document where the positions of the writing style shift are unknown.

Constraint 2 **Uncertainty on the length of plagiarism.** This constraint results from the previous one. The consequences of this constraint emerge notably when the plagiarism case is relatively short. In fact, it is difficult to model the writing style of short cases, which makes their detection challenging.

These two constraints made the IPD not only a *classification problem* (i.e., is a certain passage from the input document plagiarised or original?) but also a *segmentation problem* (i.e., what are the boundaries of the checked passages?)[7].

Since they are inherent to the problem, the above-mentioned constraints can be skipped by the transformation of the IPD problem to another one wherein dealing with the position and the length of the potentially plagiarised fragments is not required. More precisely, the traditional intrinsic plagiarism detection can be alleviated so that the expected output of the task becomes to determine whether the whole document is plagiarised or not (document-level plagiarism detection) instead of to uncover the position of the plagiarised fragments (fragment-level plagiarism detection). In fact, *style change detection*[8] is the name given by researchers

[7] In other words, what does represent a passage: a phrase, a sentence, a paragraph…, etc?

[8] This task has been first introduced in a shared task with the same name (see https://pan.webis.de/clef18/pan18-web/author-identification.html#style-change-detection) (last consultation 20/07/2019)

to this relaxed problem, which started to attract attention recently (Kestemont et al. 2018; Kopev et al. 2018; Zangerle et al. 2019) especially that, as expected, the performance of the dedicated methods is by far better than intrinsic plagiarism detection.

Another relaxed problem of IPD is a task whose input is an already segmented document according to the writing style changes. In this task, the uncertainty about the beginning and the length of plagiarism within the document is eliminated. This is because it is assumed that plagiarism may occur only at the beginning of a defined fragment (e.g., at the beginning of paragraphs), and its length cannot be shorter than one fragment. This problem was one of the sub-tasks of the PAN 2012 authorship clustering shared task (Juola 2012).

As clear from the above paragraphs, the existing attempts to eliminate the inherent constraints of IPD did not solve the problem but rather transformed it to other similar research problems. Thus, it would be interesting as future work to propose methods that tackle the constraints without transforming the problem.

Our perspective is to design IPD methods that are segmentation-free. In fact, there are heuristics used in external plagiarism detection (EPD) that identify the plagiarised passage without prior segmentation of the suspicious document. These heuristics consist of two mains operations: *seeding* and *extension*. The former identifies the smallest units of texts (e.g., words or phrases), called *seeds*, indicating the potential existence of plagiarism. Then, the latter operation merges the seeds that appear close together and the text in between to determine a plagiarism case. See (Potthast et al. 2013a) for further information on these heuristics.

It should be noted that, in EPD methods, seeds are pinpointed by applying text-matching techniques that detect the common substrings between the suspicious and the source document. However, in the envisaged IPD methods, we define a seed as a short substring that indicates the presence of a writing style anomaly[9]. The idea is to define the boundaries of a plagiarism case based on the positional distribution of the found seeds in the analysed document meaning that the part of the text which is dense of writing style anomalies (i.e., seeds) is marked as potentially plagiarised. Examples of such seeds may include unusual punctuation marks, rare words, infrequent n-grams, and high-level vocabularies in comparison with the rest of the document.

3.2 IPD Beyond the Current Assumptions

In addition to the barriers imposed by the nature of the problem, which affect the effectiveness of the IPD methods, there are other barriers that limit the cases to which this task can be applied. We argue that the latter barriers are caused by a number of assumptions, which are:

[9] We mean by writing style anomalies, odd writing styles or writing styles that are different from the one dominant in the analysed document.

Assumption 1 The suspicious document contains only verbatim plagiarism (not obfuscated).

Assumption 2 The suspicious document is mostly written by one author (the plagiarist) and the plagiarised text is only a small part of it (less than half of the document).

Assumption 3 The document is not 100% plagiarised. It should have an original part.

Assumption 4 The plagiarism is monolingual (i.e., not translated from another language).

Those assumptions are unrealistic. In reality, the plagiarised documents are various in terms of the type of plagiarism they contain and its proportion with regard to the rest of the text in the document. Moreover, unlike the inherent constraints, the assumptions above are extrinsic to the nature of the problem; they are made just to supposedly relax it. In fact, researchers made **Assumptions 1-3** when building the evaluation corpora (see Table VI-1). And consequently, these assumptions have been considered when developing the methods. Concerning **Assumption 4**, although some corpora involve cross-lingual plagiarism cases as shown in Table VI-1, no IPD method has yet been developed to specifically address this kind of plagiarism.

If these assumptions do not hold true – which is a conceivable scenario – it is still unknown how IPD will behave. And of course, IPD methods whose design relies on some of those assumptions are likely, in this case, to fail altogether to detect plagiarism.

In the next paragraphs, we argue the feasibility of IPD beyond its current limitations caused by the aforementioned assumptions. We analyse the different neglected[10] scenarios of plagiarism wherein an intrinsic-approach-based solution can be conceivable.

Conceivably, a plagiarism case becomes invisible for an intrinsic plagiarism detection method if the plagiarist succeeded to obfuscate it by rewriting it in her/his own writing style so that the contrast between it and the rest of the document fades away. On the other hand, a plagiarism case becomes invisible for an external plagiarism detection method if the plagiarist succeeded to obfuscate it so that the similarity with its source is concealed. Therefore, the obfuscations aiming to defeat the external plagiarism detection systems will not certainly

Table VI-1. Assumptions made when building the evaluation corpora of intrinsic plagiarism detection

	PAN-PC-09	PAN-PC-10	PAN-PC-11	PAN-PC-16	InAra
The plagiarism is verbatim	✓	✓	✓	✓	✓
The proportion of the plagiarism in document ≤ 50%	✓	✓	✓	✓	×
No document is 100% plagiarised	✓	✓	✓	✓	✓
The plagiarism is monolingual	×	×	×	✓	✓

[10] They have been neglected in the context of IPD. However, they have been addressed in the context of EPD.

defeat the intrinsic systems. In other words, we hypothesise that the writing style contrast between the stolen passage and the rest of the suspicious document may persist after an obfuscation oriented to defeat the external plagiarism detector.

Based on the above perceptions, we argue that it is more realistic to describe the plagiarism cases detectable by IPD as writing style irregularities without speculation as to whether the plagiarised texts have been obfuscated or not. Intuitively, some of the writing style inconsistencies could be a result of a failed attempt by the plagiarist to rewrite the stolen text in her/his own style or a result of an obfuscation that is rather intended to circumvent matching plagiarism with their sources. Again, this obfuscation does not necessarily produce a text that is stylistically consistent with the rest of the document. Then, there are no solid arguments behind the assumption that only the verbatim plagiarism is detectable by IPD (**Assumption 1**).

In addition to the above intuitions, the good news is that obfuscating the text can be even counterproductive with regard its goal, i.e., it makes plagiarism easy to notice instead of hiding it. Recently, it has been shown in the context of content spamming detection that the *automatically obfuscated* text using *article spinning* software[11] (with the aim to hide the similarity with its source) could be identified based on its conspicuous style (Shahid et al. 2017). A study in (Prentice and Kinden 2018) shows that automatic obfuscation of text is a phenomenon that starts to take place as a mean of hiding plagiarism. The authors of that study reported that they encountered unidiomatic texts in students' assignments, which are characterised by the use of odd synonyms instead of the standard terminology, and it turns out that this text is the result of paraphrasing tools. In another study (Rogerson and McCarthy 2017), it has been shown that plagiarism resulted from two paraphrasing tools is not detectable by a widely-used plagiarism detector based on text matching.

In another context, which is the detection of scientific fraud, it has been shown that the deception in academic writing could be characterised stylistically (Braud and Søgaard 2017; Markowitz and Hancock 2016). Markowitz and Hancock reported that fraudulent papers have low readability and high rate of jargon (i.e., specialised and uncommon terms), which are, interestingly, similar features to those of the automatically obfuscated text.

Concerning the detection of cross-lingual plagiarism based on its style, as we have said previously, there is no IPD method oriented to this kind of plagiarism (**Assumptions 4)**. However, this research path has been already suggested by Barrón-Cedeño (2012, Chapter 10: Conclusions) and Clough et al. (2015). While Barrón-Cedeño supports his suggestion by references that prove the style of human translation is distinctive (Baroni and Bernardini 2006; Koppel and Ordan 2011), Clough et al. suggest profiling the style of the commonly-used online

[11] Article spinning is a technique used in the context of search engine optimisation that consists in creating new versions of a webpage by paraphrasing its textual content. Its goal is to avoid creating duplicate web content, which is penalised by the search engines' algorithms. The text resulted from the article spinning software is called *spun text*.

translators[12]. There are, indeed, works on the identification of the machine translation (Aharoni et al. 2014; Arase and Zhou 2013) [13], which has proved to be an easy task.

Our insight is that features used to detect the scientific fraud, spun text, and translated text might pave the way for a *new generation of intrinsic plagiarism detection methods* that are able to discern plagiarism based not only on the writing style changes but also on the traces left on the text by translating it or trying to obscure its origin. This means that we have to envisage methods that are able *to profile plagiarism* (or at least some kinds of it, such as the automatically obfuscated or translated) without the need to compare its style with that of the host document.

From the technical viewpoint, we are suggesting a new perception of the IPD problem: from a style change (or anomaly detection) problem to a profiling problem. In the style change perception, plagiarism is detected based on the features that distinguish it from the dominant writing style in the suspicious document. However, by perceiving IPD as a profiling problem, we have to seek the peculiar features of plagiarism that distinguish it from any original text in general (and not only from the dominant style in the given suspicious document). The second perception allows overcoming the limitations imposed by **Assumptions 2-3**, meaning that it renders feasible to detect intrinsically the fully plagiarised documents or to spot the plagiarised fragments even if they constitute the majority of the suspicious document.

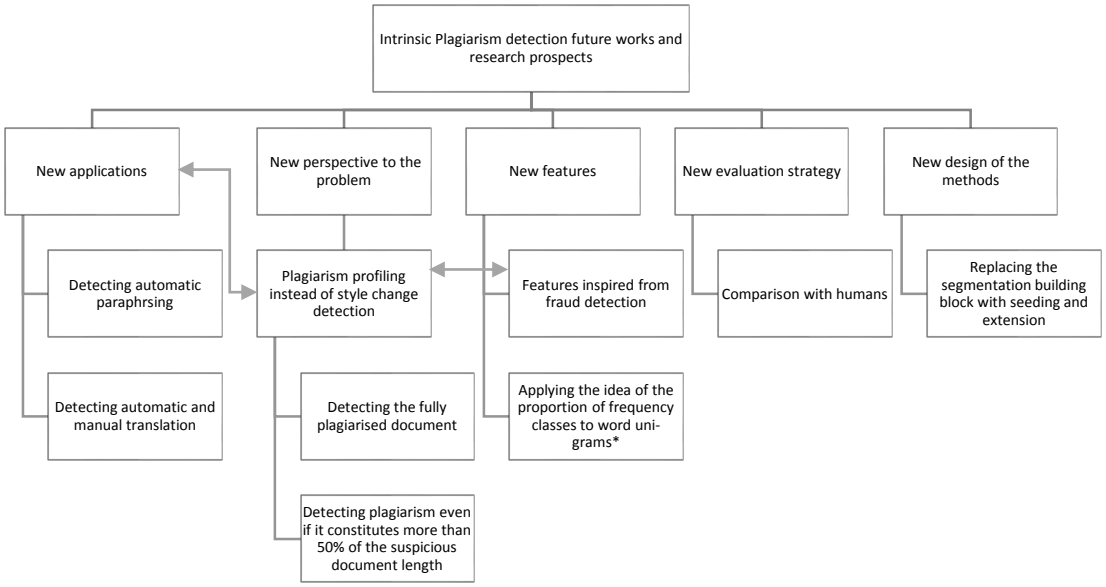
It remains to say that based on the research works discussed above, which demonstrate the feasibility of automatically identifying the obfuscated and translated texts based on their linguistic peculiarities, it appears that detecting plagiarism intrinsically has reasonable potential to succeed with the obfuscated (most notably the texts paraphrased automatically) and the translated text regardless of the proportion of plagiarism in the text. Therefore, dropping the assumptions imposed by the current perspective to the problem is perhaps what will open the door to advance research on the intrinsic approach and vary its applications.

3.3 Humans vs. Machine IPD

According to a study in (Bull et al. 2001), academics consider the writing style inconsistencies as the most common trigger of suspicion on the originality of the students' work. Although this study was one of the earliest motivations of intrinsic plagiarism detection (Clough 2003), important questions in this connection are still unanswered, which are:

- To which extent humans are able to detect plagiarism on the basis of the writing style inconsistencies? and
- How does humans' performance in this task compare to that of the existing intrinsic

[12] Clough (2015) suggested also to profile the style of the services that sell essays to the students. This kind of plagiarism is termed *contract cheating* and it starts to concern the scientific community (Rogerson 2017).
[13] Our awareness of these references is thanks to (Rabinovich et al. 2017).



* See the conclusion of Chapter V (pp. 122-123) for further details on this future work.

Figure VI-1. Summary of the discussed future works and research prospects. The arrow between some future works means that each one of them implies the other.

plagiarism detection methods?

It would be of much importance as future work to try to answer these questions to better evaluate the intrinsic plagiarism detection task. In other words, we suggest using the comparison with humans as an evaluation strategy. Indeed, our judgment of the (poor) performance of the algorithms on this task using the current evaluation strategy lacks fairness since the comparison with humans' performance may change our current conclusions. To illustrate, it is still unknown whether the plagiarism cases detectable automatically by the intrinsic approach are also detectable by humans or vice versa. Knowing that will help not only to better judge the performance of the automatic methods but also to spot their weaknesses and strengths in relation with humans' capabilities, which may inspire us to new directions to solve the problem.

To address the above questions, one approach would be to evaluate the performance of the intrinsic plagiarism detection methods on corpora where humans have annotated plagiarism on the basis of their observations of the writing style anomalies. It is worth to note that those experiments can be expensive because of the involvement of human resources. However, the reward would be to gain knowledge that is unobtainable with the current evaluation strategy.

Figure VI-1 summarises the discussed future works and prospects.

References

Abbasi, A., & Chen, H. (2005). Applying Authorship Analysis to Extremist- Group Web Forum Messages. *IEEE Intelligent Systems*, *20*(5), 67–75.

Abbasi, A., & Chen, H. (2008). Writeprints: A Stylometric Approach to Identity-Level Identification and Similarity Detection in Cyberspace. *ACM Transactions on Information Systems*, *26*(2), 1–29. https://doi.org/10.1145/1344411.1344413

Abdulhayoglu, M. A., Thijs, B., & Jeuris, W. (2016). Using Character N-Grams to Match a List of Publications to References in Bibliographic Databases. *Scientometrics*, *109*(3), 1525–1546. https://doi.org/10.1007/s11192-016-2066-3

Abouenour, L., Bouzoubaa, K., & Rosso, P. (2013). On the Evaluation and Improvement of Arabic WordNet Coverage and Usability. *Language Resources and Evaluation*, *47*(2013), 891–917. https://doi.org/10.1007/s10579-013-9237-0

Abouzakhar, N., Allison, B., & Guthrie, L. (2008). Unsupervised Learning-Based Anomalous Arabic Text Detection. In N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odjik, S. Piperidis, & D. Tapias (Eds.), *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC'08)* (pp. 291–296). Marrakech, Morocco: ELRA.

Agirre, E., Banea, C., Cer, D., Diab, M., Gonzalez-Agirre, A., Mihalcea, R., Rigau, G., & Wiebe, J. (2016). SemEval-2016 Task 1: Semantic Textual Similarity, Monolingual and Cross-Lingual Evaluation. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)* (pp. 497–511). https://doi.org/10.18653/v1/s16-1081

Aharoni, R., Koppel, M., & Goldberg, Y. (2014). Automatic Detection of Machine Translated Text and Translation Quality Estimation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)* (pp. 289–295). Stroudsburg, PA, USA: Association for Computational Linguistics. https://doi.org/10.3115/v1/P14-2048

Akiva, N. (2011). Using Clustering to Identify Outlier Chunks of Text - Notebook for PAN at CLEF 2011. In V. Petras, P. Forner, P. Clough, & N. Ferro (Eds.), *CLEF 2011 Evaluation Labs and Workshops –Working Notes papers, CEUR proceedings vol. 1177* (pp. 5–7). CEUR-WS.org.

Akiva, N. (2012). Authorship and Plagiarism Detection Using Binary BOW Features. In P. Forner, J. Karlgren, C. Womser-Hacker, & N. Ferro (Eds.), *CLEF 2012 Evaluation Labs and Workshops –Working Notes Papers, CEUR proceedings vol. 1178*. CEUR-WS.org.

Akiva, N., & Koppel, M. (2012). Identifying Distinct Components of a Multi-Author Document. In *European Intelligence and Security Informatics Conference (EISIC) August 22-24, Odense, Denmark* (pp. 205–209). IEEE. https://doi.org/10.1109/EISIC.2012.16

Akiva, N., & Koppel, M. (2013). A Generic Unsupervised Method for Decomposing Multi-Author Documents. *Journal of the American Society for Information Science and Technology*, *64*(11), 2256–2264. https://doi.org/10.1002/asi.22924

Elgendy, M. A. K. (2014). Plagiarism Detection Software in the Digital Environment Available across the Web: An Evaluation Study (in Arabic). *International Journal of Library and Information Sciences*, *1*(2), 34–93. https://doi.org/10.12816/0010485

AL-Smadi, M., Jaradat, Z., AL-Ayyoub, M., & Jararweh, Y. (2017). Paraphrase Identification and Semantic Text Similarity Analysis in Arabic News Tweets Using Lexical, Syntactic, and Semantic Features. *Information Processing and Management*, *53*(3), 640–652. https://doi.org/10.1016/j.ipm.2017.01.002

Al-Sulaiti, L., & Atwell, E. S. (2006). The Design of a Corpus of Contemporary Arabic. *International Journal of Corpus Linguistics*, *11*(2), 135–171. https://doi.org/10.1075/ijcl.11.2.02als

Al Manasrah, E. (2009). Intertextuality and Plagiarism in Critical Heritage (in Arabic). *Nizwa journal*, (56), 59–78.

Aldebei, K., He, X., Jia, W., & Yang, J. (2016). Unsupervised Multi-Author Document Decomposition Based on Hidden Markov Model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)* (pp. 706–714).

Aldebei, K., He, X., & Yang, J. (2015). Unsupervised Decomposition of a Multi-Author Document Based on Naive-Bayesian Model. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)* (pp. 501–505).

Alotaiby, F. A., & Alkharashi, I. A. (2007). Processing Large Arabic Text Corpora : Preliminary Analysis and Results, (Ldc), 78–82.

Alrabiah, M., Al-Salman, A., & Leeds, E. A. (2013). The Design and Construction of the 50 Million Words KSUCCA. In *Proceedings of WACL'2 Second Workshop on Arabic Corpus Linguistics* (pp. 5–8). https://doi.org/https://doi.org/10.1016/j.iac.2007.09.004

Alzahrani, S. (2015). Arabic Plagiarism Detection Using Word Correlation in N-Grams with K-Overlapping Approach, Working Notes for PAN-AraPlagDet at FIRE 2015. In P. Majumder, M. Mitra, M. Agrawal, & P. Mehta (Eds.), *Post Proceedings of the Workshops at the 7th Forum for Information Retrieval Evaluation (FIRE 2015), Gandhinagar, India, December 4-6, CEUR proceedings vol. 1587* (pp. 123–125). CEUR-WS.org.

Alzahrani, S. (2016). Cross-Language Semantic Similarity of Arabic- English Short Phrases and Sentences. *Journal of Computer Science*, (January 2016). https://doi.org/10.3844/jcssp.2016.1.18

Alzahrani, S., & Salim, N. (2008). Plagiarism Detection in Arabic Scripts Using Fuzzy Information Retrieval. In *Proceedings of 2008 Student Conference on Research and Development (SCOReD 2008), 26-27 Nov. 2008, Johor, Malaysia* (pp. 1–4).

Alzahrani, S., & Salim, N. (2009). Statement-Based Fuzzy-Set Information Retrieval versus Fingerprints Matching for Plagiarism Detection in Arabic Documents. In *5th Postgraduate Annual Research Seminar (PARS '09), Johor Bahru, Malaysia* (pp. 267–268).

Alzahrani, S., Salim, N., & Alsofyani, M. M. (2009). Work in Progress: Developing Arabic Plagiarism Detection Tool for E-Learning Systems. In *2009 International Association of Computer Science and Information Technology - Spring Conference (IACSIT-SC 2009)* (pp. 105–109). Singapore: IEEE. https://doi.org/10.1109/IACSIT-SC.2009.22

Anguera, X., Bozonnet, S., Evans, N., Fredouille, C., Friedland, G., & Vinyals, O. (2012). Speaker Diarization: A Review of Recent Research. *IEEE Transactions on Audio, Speech, and Language Processing*, *20*(2), 356–370.

Arase, Y., & Zhou, M. (2013). Machine Translation Detection from Monolingual Web-Text. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 1597–1607). Association for Computational Linguistics.

Argamon, S., Koppel, M., & Avneri, G. (1998). Routing Documents According to Style. In *First International workshop on innovative information systems* (pp. 85–92).

Augsten, N., Ohlen, M. B., & Gamper, J. (2010). The Pq-Gram Distance between Ordered Labeled Trees. *ACM Transactions on Database Systems (TODS)*, *35*(1), 4:1-4:36. https://doi.org/10.1145/1670243.1670247

Baayen, H., van Halteren, H., & Tweedie, F. (1996). Outside the Cave of Shadows : Using Syntactic Annotation to Enhance Authorship Attribution. *Literary and Linguistic Computing*, *11*(3), 121–131. https://doi.org/10.1093/llc/11.3.121

Baroni, M., & Bernardini, S. (2006). A New Approach to the Study of Translationese: Machine-Learning the

Difference between Original and Translated Text. *Literary and Linguistic Computing*, *21*(3), 259–274. https://doi.org/10.1093/llc/fqi039

Barrón-Cedeño, A. (2012). On the Mono- and Cross-Language Detection of Text Re-Use and Plagiarism (PhD thesis). Universidad Politécnica de Valencia, Spain.

Barrón-Cedeño, A., Vallés-Balaguer, E., & Rosso, P. (2012). Stylysis. Online tool. http://memex2.dsic.upv.es:8080/StylisticAnalysis/en/index.jsp

Barrón-Cedeño, A., Rosso, P., Lalitha Devi, S., Clough, P., & Stevenson, M. (2013a). PAN@FIRE: Overview of the Cross-Language !Ndian Text Re-Use Detection Competition. In *Proceedings of the Forum for Information Retrieval Evaluation (FIRE 2011)*. Bombay, India.

Barrón-Cedeño, A., Vila, M., AntòniaMartí, M., & Rosso, P. (2013b). Plagiarism Meets Paraphrasing : Insights for the Next Generation in Automatic Plagiarism Detection. *Computational Linguistics*, *39*(4), 917–947. https://doi.org/10.1162/COLI_a_00153

Baysen, E., Hošková-Mayerová, Š., Çakmak, N., & Baysen, F. (2017). Misconceptions Regarding Providing Citations: To Neglect Means to Take Risk for Future Scientific Research. In Š. Hošková-Mayerová, F. Maturo, & J. Kacprzyk (Eds.), *Mathematical-Statistical Models and Qualitative Theories for Economic and Social Sciences. Studies in Systems, Decision and Control* (Vol. 104, pp. 177–186). Springer. https://doi.org/10.1007/978-3-319-54819-7_12

Belinkov, Y., Magidow, A., Barrón-Cedeño, A., Shmidman, A., & Romanov, M. (2019). Studying the History of the Arabic Language: Language Technology and a Large-Scale Historical Corpus. *Language Resources and Evaluation*, *53*(December), 771–805. https://doi.org/10.1007/s10579-019-09460-w

Bensalem, I., Rosso, P., & Chikhi, S. (2012). Intrinsic Plagiarism Detection in Arabic Text : Preliminary Experiments. In R. Berlanga & P. Rosso (Eds.), *2nd Spanish Conference on Information Retrieval (CERI 2012)* (pp. 325–329). Valencia, Spain.

Bensalem, I., Rosso, P., & Chikhi, S. (2013a). A New Corpus for the Evaluation of Arabic Intrinsic Plagiarism Detection. In P. Forner, H. Müller, R. Paredes, P. Rosso, & B. Stein (Eds.), *CLEF 2013, LNCS, vol. 8138* (pp. 53–58). Heidelberg: Springer. https://doi.org/10.1007/978-3-642-40802-1_6

Bensalem, I., Rosso, P., & Chikhi, S. (2013b). Building Arabic Corpora from Wikisource. In *2013 ACS International Conference on Computer Systems and Applications (AICCSA), Fes/Ifran, Morocco* (pp. 1–2). IEEE. https://doi.org/10.1109/AICCSA.2013.6616474

Bensalem, I., Rosso, P., & Chikhi, S. (2014a). Intrinsic Plagiarism Detection Using N-Gram Classes. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, October 25-29* (pp. 1459–1464). Association for Computational Linguistics.

Bensalem, I., Rosso, P., & Chikhi, S. (2014b). Intrinsic Plagiarism Detection Using N-Grams Frequency Classes. In *Accepted in CICling 2014 (unpublished)*.

Bensalem, I., Boukhalfa, I., Rosso, P., Abouenour, L., Darwish, K., & Chikhi, S. (2015). Overview of the AraPlagDet PAN@FIRE2015 Shared Task on Arabic Plagiarism Detection. In P. Majumder, M. Mitra, M. Agrawal, & P. Mehta (Eds.), *Post Proceedings of the Workshops at the 7th Forum for Information Retrieval Evaluation (FIRE 2015), Gandhinagar, India, December 4-6, CEUR proceedings vol. 1587* (pp. 111–122). CEUR-WS.org.

Bensalem, I., Rosso, P., & Chikhi, S. (2019). On the Use of Character N-Grams as the Only Intrinsic Evidence of Plagiarism. *Language Resources and Evaluation*, *53*(3), 363–396. https://doi.org/10.1007/s10579-019-09444-w

Bonsall, B. (2004). The Automatic Detection of Plagiarism (Bachelor of Engineering report). University of Sheffield.

Boukhalfa, I., Mostefai, S., & Chekkai, N. (2018). A Study of Graph Based Stemmer in Arabic Extrinsic Plagiarism Detection. In *MedPRAI '18 Proceedings of the 2nd Mediterranean Conference on Pattern Recognition and Artificial Intelligence* (pp. 27–32). Morocco: ACM. https://doi.org/10.1145/3177148.3180089

Braud, C., & Søgaard, A. (2017). Is Writing Style Predictive of Scientific Fraud? In *Proceedings of the Workshop on Stylistic Variation, EMNLP 2017* (pp. 37–42). Association for Computational Linguistics. https://doi.org/10.18653/v1/W17-4905

Bretag, T. (2016). Defining Academic Integrity: International Perspectives – Introduction. In *Handbook of Academic Integrity* (pp. 3–5). https://doi.org/10.1007/978-981-287-098-8

Broder, A. Z. (2000). Identifying and Filtering Near-Duplicate Documents. In *Combinatorial Pattern Matching. CPM 2000. LNCS, vol 1848* (pp. 1–10). https://doi.org/10.1007/3-540-45123-4_1

Brooke, J., & Hirst, G. (2012). Paragraph Clustering for Intrinsic Plagiarism Detection Using a Stylistic Vector-Space Model with Extrinsic Features - Notebook for PAN at CLEF 2012. In P. Forner, J. Karlgren, C. Womser-Hacker, & N. Ferro (Eds.), *CLEF 2012 Evaluation Labs and Workshops –Working Notes Papers, CEUR proceedings vol. 1178*. CEUR-WS.org.

Bru, J. R., Martínez-Barco, P., & Muñoz, R. (2011). Hybrid System for Plagiarism Detection. In *Proceedings of Recent Advances in Natural Language Processing, RANLP 2011, 12-14 September, Hissar, Bulgaria* (pp. 527–532).

Buckwalter, T. (2000). Arabic Buckwalter Transliteration. http://www.qamus.org/transliteration.htm

Bull, J., Collins, C., Coughlin, E., & Sharp, D. (2001). Technical Review of Plagiarism Detection Software Report (Report prepared for the Joint Information System Committee). Computer Assisted Assessment Centre & University of Luton.

Burn-Thornton, K., & Burman, T. (2015). A Novel Approach for Analysis of 'Real World' Data: A Data Mining Engine for Identification of Multi-Author Student Document Submission. In M. Abou-Nasr, S. Lessmann, R. Stahlbock, & G. M. Weiss (Eds.), *Real World Data Mining Applications* (Vol. 17, pp. 203–219). Springer International Publishing. https://doi.org/10.1007/978-3-319-07812-0_11

Carnahan, N., Huderle, M., Jones, N., Stephan, C., Tran, T., & Wood-Doughty, Z. (2014). Plagiarism Detection (Project report). USA: Carleton College.

Cavnar, W. B., & Trenkle, J. M. (1994). N-Gram-Based Text Categorization. In *Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval* (pp. 161–175). https://doi.org/10.1.1.53.9367

Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, N., & Specia, L. (2017). SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Cross-Lingual Focused Evaluation. In *Proceedings ofthe 11th International Workshop on Semantic Evaluations (SemEval-2017)* (pp. 1–14).

Chaski, C. (1999). Linguistic Authentication and Reliability. National Criminal Justice Reference Service.

Clark, J. H., & Hannon, C. J. (2007). A Classifier System for Author Recognition Using Synonym-Based Features. In *MICAI 2007: Advances in Artificial Intelligence. LNCS vol. 4827* (pp. 839–849). Springer. https://doi.org/10.1007/978-3-540-76631-5_80

Clough, P. (2000). Plagiarism in Natural and Programming Languages : An Overview of Current Tools and Technologies (Internal Report CS-00-05). UK: University of Sheffield.

Clough, P. (2003). Old and New Challenges in Automatic Plagiarism Detection (Report). National UK Plagiarism Advisory Service.

Clough, P., & Stevenson, M. (2011). Developing a Corpus of Plagiarised Short Answers. *Language Resources and Evaluation*, *45*(March), 5–24. https://doi.org/10.1007/s10579-009-9112-1

Clough, P., Willett, P., & Lim, J. (2015). Unfair Means: Use Cases Beyond Plagiarism. In J. Mothe, J. Savoy, J. Kamps, K. Pinel-Sauvagnat, G. Jones, E. San Juan, L. Capellato, & N. Ferro (Eds.), *Experimental IR Meets Multilinguality, Multimodality, and Interaction. CLEF 2015. LNCS vol 9283* (Vol. 9283, pp. 229–234). Springer International Publishing. https://doi.org/10.1007/978-3-319-24027-5_21

Darwish, K., & Magdy, W. (2013). Arabic Information Retrieval. *Foundations and Trends® in Information Retrieval*, *7*(4), 239–342. https://doi.org/10.1561/1500000031

Drinan, P. (2016). Getting Political: What Institutions and Governments Need to Do. *Handbook of Academic Integrity*, 1075–1087. https://doi.org/10.1007/978-981-287-098-8

Faour, M. (2012). Religious Education and Pluralism in Egypt and Tunisia. *Carnegie Papers*, (August), 1–28.

Farghaly, A., & Shaalan, K. (2009). Arabic Natural Language Processing: Challenges and Solutions. *ACM Transactions on Asian Language Information Processing (TALIP)*, *8*(4), 14:1-14:22. https://doi.org/10.1145/1644879.1644881.http

Fayyad, U. M., & Irani, K. B. (1993). Multi-Interval Discretization of Continuos-Valued Attributes for Classification Learning. In *Thirteenth International Joint Conference on Artificial Intelligence* (pp. 1022–1027).

Feng, S., Banerjee, R., & Choi, Y. (2012). Characterizing Stylistic Elements in Syntactic Structure. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* (pp. 1522–1533). Association for Computational Linguistics.

Ferrero, J., Agnes, F., Besacier, L., & Schwab, D. (2017). CompiLIG at SemEval-2017 Task 1: Cross-Language Plagiarism Detection Methods for Semantic Textual Similarity. In *Proceedings ofthe 11th International Workshop on Semantic Evaluations (SemEval-2017)* (pp. 109–114). Association for Computational Linguistics.

Ferro, N., & Peters, C. (2019). From Multilingual to Multimodal : The Evolution of CLEF over Two Decades. In N. Ferro & C. Peters (Eds.), *Information Retrieval Evaluation in a Changing World - Lessons Learned from 20 Years of CLEF, volume 41 of The Information Retrieval Series* (pp. 3–44). Springer.

Fishman, T. A. (Ed.). (2013). The Fundamental Values of Academic Integrity (Second edi., Vol. 2). International Center for Academic Integrity.

Foltýnek, T., Meuschke, N., & Gipp, B. (2019). Academic Plagiarism Detection: A Systematic Literature Review. *ACM Computing Surveys*, *52*(6), Article 112. https://doi.org/10.1145/3345317

Franco-Salvador, M., Bensalem, I., Flores, E., Gupta, P., & Rosso, P. (2015). PAN 2015 Shared Task on Plagiarism Detection : Evaluation of Corpora for Text Alignment -Notebook for PAN at CLEF 2015. In L. Cappellato, N. Ferro, G. J. F. Jones, & E. S. Juan (Eds.), *Working Notes of CLEF 2015 - Conference and Labs of the Evaluation forum, CEUR proceedings vol. 1391*. CEUR-WS.org.

Franco-Salvador, M., Rosso, P., & Montes-y-Gómez, M. (2016). A Systematic Study of Knowledge Graph Analysis for Cross-Language Plagiarism Detection. *Information Processing and Management*, *52*(4), 550–570. https://doi.org/10.1016/j.ipm.2015.12.004

Gamon, M., & Grey, A. (2004). Linguistic Correlates of Style : Authorship Classification with Deep Linguistic Analysis Features. In *Proceedings of the 20th International Conference on Computational Linguistics* (pp. 611–617).

Gencosman, B. C., Ozmutlu, H. C., & Ozmutlu, S. (2014). Character N-Gram Application for Automatic New Topic Identification. *Information Processing and Management*, *50*(6), 821–856. https://doi.org/10.1016/j.ipm.2014.06.005

Ghanem, B., Arafeh, L., Rosso, P., & Sánchez-Vega, F. (2018). HYPLAG : Hybrid Arabic Text Plagiarism Detection System. In *NLDB 2018, LNCS vol. 10859* (pp. 315–323).

Giannella, C. (2016). An Improved Algorithm for Unsupervised Decomposition of a Multi-Author Document. *Journal of the Association for Information Science and Technology*, *67*(2), 400–411.

Gillam, L. (2013). Readability for Author Profiling ? Notebook for PAN at CLEF 2013. In P. Forner, R. Navigli, D. Tufis, & N. Ferro (Eds.), *CLEF 2013 Evaluation Labs and Workshops –Working Notes papers, CEUR proceedings vol. 1179*. CEUR-WS.org.

Gillam, L., Marinuzzi, J., & Ioannou, P. (2011). TurnItOff - Defeating Plagiarism Detection Systems. In *Proceedings of the 11th Higher Education Academy-ICS Annual Conference*. Higher Education Academy.

Gipp, B., Meuschke, N., & Beel, J. (2011). Comparative Evaluation of Text- and Citation-Based Plagiarism

Detection Approaches Using GuttenPlag. In *Proceeding of the 11th annual international ACM/IEEE joint conference on Digital libraries* (pp. 255–258).

Glover, A., & Hirst, G. (1996). Detecting Stylistic Inconsistencies in Collaborative Writing. In M. Sharples & T. van der Geest (Eds.), *The New Writing Environment* (pp. 147–168). London: Springer. https://doi.org/10.1007/978-1-4471-1482-6_12

Gomaa, W. H., & Fahmy, A. A. (2013). A Survey of Text Similarity Approaches. *International Journal of Computer Applications*, *68*(13), 975–8887.

Graham, N., Hirst, G., & Marthi, B. (2005). Segmenting Documents by Stylistic Character. *Natural Language Engineering*, *11*(04), 397–415. https://doi.org/10.1017/S1351324905003694

Grozea, C., & Popescu, M. (2010). Who ' s the Thief ? Automatic Detection of the Direction of Plagiarism. In *CICLing 2010, Iaşi, Romania, March 21-27, LNCS, vol. 6008* (pp. 700–710). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-12116-6_59

Guimeur, S. (2013). Le Recours Au (Copier-Coller) Sur La Déformation de l'information Scientifique: Cas Des Étudiants Universitaires de La Première Année Master Français Biskra. Université Mohamed Khider, Biskra, Algeria, Algeria.

Guthrie, D., Guthrie, L., Allison, B., & Wilks, Y. (2007). Unsupervised Anomaly Detection. In *IJCAI International Joint Conference on Artificial Intelligence* (pp. 1624–1628). Morgan Kaufmann Publishers.

Habash, N. (2010). Introduction to Arabic Natural Language Processing. Morgan & Claypool.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA Data Mining Software: An Update. *SIGKDD Explorations*, *11*(1), 10–18. https://doi.org/10.1145/1656274.1656278

Heather, J. (2010). Turnitoff: Identifying and Fixing a Hole in Current Plagiarism Detection Software. *Assessment & Evaluation in Higher Education*, *35*(6), 647–660. https://doi.org/10.1080/02602938.2010.486471

Hernández Fusilier, D., Montes-y-Gómez, M., Rosso, P., & Cabrera, R. G. (2015). Detection of Opinion Spam with Character N-Grams. In *CICLing 2015, Part II, LNCS 9042* (pp. 285–294). https://doi.org/10.1007/978-3-319-18117-2_21

Hersee, M. S. (2001). Automatic Detection of Plagiarism: An Approach Using the Qsum Method (Bachelor of Science Final Year Paper). University of Sheffield, UK.

Holmes, D. I. (1994). Authorship Attribution. *Computers and the Humanities*, *28*(2), 87–106. https://doi.org/10.1007/BF01830689

Holmes, D. I. (1998). The Evolution of Stylometry in Humanities Scholarship. *Literary and Linguistic Computing*, *13*(3).

Honoré, A. (1979). Some Simple Measures of Richness of Vocabulary. *Association for Literary and Linguistic Computing Bulletin*, *7*(2), 172–177.

Hoover, D. L. (2003). Another Perspective on Vocabulary Richness. *Computers and the Humanities*, *37*(2), 151–178. https://doi.org/10.1023/A:1022673822140

Hosny, M., & Fatima, S. (2014). Attitude of Students Towards Cheating and Plagiarism: University Case Study. *Journal of Applied Sciences*, *14*(8), 748–757. https://doi.org/10.3923/jas.2014.748.757

Houvardas, J., & Stamatatos, E. (2006). N-Gram Feature Selection for Authorship Identification. In *International Conference on Artificial Intelligence: Methodology, Systems, and Applications* (pp. 77–86).

Hussein, A. S. (2015). Arabic Document Similarity Analysis Using N-Grams and Singular Value Decomposition. In *2015 IEEE 9th International Conference on Research Challenges in Information Science (RCIS)* (pp. 445–455). IEEE. https://doi.org/10.1109/RCIS.2015.7128906

Hussein, A. S. (2016). Visualizing Document Similarity Using N-Grams and Latent Semantic Analysis. In *SAI Computing Conference 2016* (pp. 269–279).

Ivarsson, M., & Gorschek, T. (2011). A Method for Evaluating Rigor and Industrial Relevance of Technology Evaluations. *Empirical Software Engineering*, *16*(3), 365–395. https://doi.org/10.1007/s10664-010-9146-4

Jadalla, A., & Elnagar, A. (2012a). Iqtebas 1.0: A Fingerprinting-Based Plagiarism Detection System for Arabic Text-Based Documents. *International Journal on Data Mining and Intelligent Information Technology Applications*, *2*(2), 31–43. https://doi.org/10.4156/ijmia.vol2.issue2.4

Jadalla, A., & Elnagar, A. (2012b). A Plagiarism Detection System for Arabic Text-Based Documents. In M. Chau, G. A. Wang, W. T. Yue, & H. Chen (Eds.), *PAISI 2012. LNCS vol. 7299* (pp. 145–153). Springer, Heidelberg. https://doi.org/10.1007/978-3-642-30428-6_12

Jadalla, A., & Elnagar, A. (2012c). A Fingerprinting-Based Plagiarism Detection System for Arabic Text-Based Documents. In *8th International Conference on Computing Technology and Information Management (ICCM'12)* (pp. 477–482). Seoul: IEEE.

Jankowska, M., Milios, E., & Kešelj, V. (2014). Author Verification Using Common N-Gram Profiles of Text Documents. *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics*, 387–397.

Jaoua, M., Jaoua, F. K., Hadrich Belguith, L., & Ben Hamadou, A. (2011). Automatic Detection of Plagiarism in Arabic Documents Based on Lexical Chains (in Arabic). *Arab Computer Society Journal*, *4*(2), 1–11.

Juola, P. (2006). Authorship Attribution. *Foundations and Trends® in Information Retrieval*, *1*(3), 233–334. https://doi.org/10.1561/1500000005

Juola, P. (2012). An Overview of the Traditional Authorship Attribution Subtask, Notebook for PAN at CLEF 2012. In P. Forner, J. Karlgren, & C. Womser-Hacker (Eds.), *CLEF 2012 Evaluation Labs and Workshops –Working Notes Papers, CEUR proceedings vol. 1178* (pp. 1–7). CEUR-WS.org.

Kahloula, B., & Berri, J. (2016). Plagiarism Detection in Arabic Documents: Approaches, Architecture and Systems. *Journal of Digital Information Management*, *14*(2), 124–135.

Kanaris, I., Kanaris, K., Houvardas, I., & Stamatatos, E. (2007). Words vs. Character N-Grams for Anti-Spam Filtering. *International Journal on Artificial Intelligence Tools*, *16*(06), 1047–1067. https://doi.org/10.1142/S0218213007003692

Kasprzak, J., & Brandejs, M. (2010). Improving the Reliability of the Plagiarism Detection System Lab Report for PAN at CLEF 2010. In M. Braschler, D. Harman, E. Pianta, & N. Ferro (Eds.), *CLEF 2010 Evaluation Labs and Workshops –Working Notes papers, CEUR proceedings vol. 1176*. CEUR-WS.org.

Keogh, E., Chu, S., Hart, D., & Pazzani, M. (2004). Segmenting Time Series: A Survey and Novel Approach. In H. Bunke (Ed.), *Data Mining in Time Series Databases* (pp. 1–15). World Scientific Publishing.

Kern, R., & Granitzer, M. (2009). Efficient Linear Text Segmentation Based on Information Retrieval Techniques. In *Proceedings of the International Conference on Management of Emergent Digital EcoSystems - MEDES '09*. ACM Press. https://doi.org/10.1145/1643823.1643854

Kern, R., Klampfl, S., & Zechner, M. (2012). Vote/Veto Classification, Ensemble Clustering and Sequence Classification for Author Identification - Notebook of PAN at CLEF 2012. In P. Forner, J. Karlgren, C. Womser-Hacker, & N. Ferro (Eds.), *CLEF 2012 Evaluation Labs and Workshops –Working Notes Papers, CEUR proceedings vol. 1178* (pp. 1–15). CEUR-WS.org.

Kešelj, V., Peng, F., Cercone, N., & Thomas, C. (2003). N-Gram-Based Author Profiles For Authorship Attribution. In *Proceedings of the Conference Pacific Association for Computational Linguistics, PA-CLING'03* (pp. 255–264).

Kestemont, M., Luyckx, K., & Daelemans, W. (2011). Intrinsic Plagiarism Detection Using Character Trigram Distance Scores - Notebook for PAN at CLEF 2011. In V. Petras, P. Forner, P. Clough, & N. Ferro (Eds.), *CLEF 2011 Evaluation Labs and Workshops –Working Notes papers, CEUR proceedings vol. 1177*. CEUR-WS.org.

Kestemont, M., Tschuggnall, M., Stamatatos, E., Daelemans, W., Specht, G., Stein, B., & Potthast, M. (2018). Overview of the Author Identification Task at PAN-2018 Cross-Domain Authorship Attribution and Style

Change Detection. In L. Cappellato, N. Ferro, J.-Y. Nie, & L. Soulier (Eds.), *Working Notes of CLEF 2018 - Conference and Labs of the Evaluation Forum, CEUR proceedings vol. 2125*. Avignon, France: CEUR-WS.org.

Khan, I. H., Siddiqui, M. A., & Jambi, K. M. (2019). Towards Building an Arabic Plagiarism Detection System: Plagiarism Detection in Arabic. *International Journal of Information Retrieval Research*, *9*(3), 12–22. https://doi.org/10.4018/IJIRR.2019070102

Khan, I. H., Siddiqui, M. A., Mansoor Jambi, K., Imran, M., & Bagais, A. A. (2015). Query Optimization in Arabic Plagiarism Detection : An Empirical Study. *International Journal of Intelligent Systems and Applications*, *7*(1), 73–79. https://doi.org/10.5815/ijisa.2015.01.07

Khreisat, L. (2009). A Machine Learning Approach for Arabic Text Classification Using N-Gram Frequency Statistics. *Journal of Informetrics*, *3*(1), 72–77. https://doi.org/10.1016/j.joi.2008.11.005

Kimler, M. (2003, August). Using Style Markers for Detecting Plagiarism in Natural Language (Master thesis). Institutionen för datavetenskap, Sweden.

Knight, A., Almeroth, K., & Bimber, B. (2004). An Automated System for Plagiarism Detection Using the Internet. In *EdMedia: World Conference on Educational Media and Technology* (pp. 3619–3625). Association for the Advancement of Computing in Education (AACE).

Kopev, D., Zlatkova, D., Mitov, K., Atanasov, A., Hardalov, M., Koychev, I., & Nakov, P. (2018). Recursive Style Breach Detection with Multifaceted Ensemble Learning Daniel. In *AIMSA 2018, LNAI 11089* (Vol. 1, pp. 126–137). Springer International Publishing. https://doi.org/10.1007/978-3-319-99344-7

Koppel, M., Akiva, N., Dershowitz, I., & Dershowitz, N. (2011). Unsupervised Decomposition of a Document into Authorial Components. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics* (pp. 1356–1364). Association for Computational Linguistics.

Koppel, M., & Ordan, N. (2011). Translationese and Its Dialects, 1318–1326.

Koppel, M., & Schler, J. (2004). Authorship Verification as a One-Class Classification Problem. *Twenty-first international conference on Machine learning - ICML '04*, 62. https://doi.org/10.1145/1015330.1015448

Koppel, M., Schler, J., & Argamon, S. (2009). Computational Methods in Authorship Attribution. *Journal of the American Society for Information Science and Technology*, *60*(1), 9–26.

Koppel, M., & Seidman, S. (2013). Automatically Identifying Pseudepigraphic Texts. In *EMNLP 2013* (pp. 1449–1454). Seattle, Washington, USA: Association for Computational Linguistics.

Kulmizev, A., Blankers, B., Bjerva, J., Nissim, M., Van Noord, G., Plank, B., & Wieling, M. (2017). The Power of Character N-Grams in Native Language Identification. In *12th Workshop on Innovative Use of NLP for Building Educational Applications* (pp. 382–389).

Kuta, M., & Kitowski, J. (2014). Optimisation of Character N-Gram Profiles Method for Intrinsic Plagiarism Detection. In L. Rutkowski, M. Korytkowski, R. Scherer, R. Tadeusiewicz, L. A. Zadeh, & J. M. Zurada (Eds.), *ICAISC 2014, Part II, LNAI vol. 8468* (pp. 500–511). Springer. https://doi.org/10.1007/978-3-319-07176-3_44

Kuznetsov, M., Motrenko, A., Kuznetsova, R., & Strijov, V. (2016). Methods for Intrinsic Plagiarism Detection and Author Diarization Notebook for PAN at CLEF 2016. In K. Balog, L. Cappellato, N. Ferro, & C. Macdonald (Eds.), *Working Notes of CLEF 2016 - Conference and Labs of the Evaluation forum, CEUR proceedings vol. 1609* (pp. 912–919). CEUR-WS.org.

Larkey, L., Ballesteros, L., & Connell, M. (2007). Light Stemming for Arabic Information Retrieval. In *Arabic Computational Morphology* (pp. 221–243). Springer. https://doi.org/10.1145/564376.564425

López-Monroy, A. P., Escalante, H. J., Montes-y-Gómez, M., & Baró, X. (2020). Forensic Analysis Recognition. In D. Baneres, M. E. Rodríguez, & A. E. Guerrero-Roldán (Eds.), *Engineering Data-Driven Adaptive Trust-based e-Assessment Systems. Lecture Notes on Data Engineering and Communications Technologies, vol 34* (pp. 1–18). https://doi.org/10.1007/978-3-030-29326-0_1

Lulu, L., Belkhouche, B., & Harous, S. (2016). Candidate Document Retrieval for Arabic-Based Text Reuse Detection on the Web. In *IIT 2016 : The 12th International Conference on Innovations in Information Technology*. IEEE. https://doi.org/10.1109/INNOVATIONS.2016.7880048

Luyckx, K., & Daelemans, W. (2005). Shallow Text Analysis and Machine Learning for Authorship Attribution. In *Computational Linguistics in the Netherlands 2004: selected papers from the Fifteenth CLIN Meeting, LOT Occasional Series, vol. 4* (pp. 149–160).

Madkhali, M. M. (2017). Saudi Students' Perception of Plagiarism (Master thesis). St. Cloud State University, USA.

Magooda, A., Mahgoub, A. Y., Rashwan, M., Fayek, M. B., & Raafat, H. (2015). RDI System for Extrinsic Plagiarism Detection (RDI_RED), Working Notes for PAN-AraPlagDet at FIRE 2015. In P. Majumder, M. Mitra, M. Agrawal, & P. Mehta (Eds.), *Post Proceedings of the Workshops at the 7th Forum for Information Retrieval Evaluation (FIRE 2015), Gandhinagar, India, December 4-6, CEUR proceedings vol. 1587* (pp. 126–128). CEUR-WS.org.

Mahgoub, A. Y., Magooda, A., Rashwan, M., Fayek, M. B., & Raafat, H. (2015). RDI System for Intrinsic Plagiarism Detection (RDI_RID), Working Notes for PAN-AraPlagDet at FIRE 2015. In P. Majumder, M. Mitra, M. Agrawal, & P. Mehta (Eds.), *Post Proceedings of the Workshops at the 7th Forum for Information Retrieval Evaluation (FIRE 2015), Gandhinagar, India, December 4-6, CEUR proceedings vol. 1587* (pp. 129–130). CEUR-WS.org.

Malcolm, J. a., & Lane, P. C. R. (2009). Tackling the PAN'09 External Plagiarism Detection Corpus with a Desktop Plagiarism Detector. In *Proceedings of the SEPLN'09 Workshop on Uncovering Plagiarism, Authorship and Social Software Misuse (PAN 09)* (Vol. 502, pp. 29–33).

Manber, U. (1994). Finding Similar Files in a Large File System. In *Winter USENIX Technical Conference* (pp. 1–10).

Manning, C. D., & Schütze, H. (1999). Foundations of Statistical Natural Language Processing. Cambridge, Massachusetts: The MIT Press.

Markowitz, D. M., & Hancock, J. T. (2016). Linguistic Obfuscation in Fraudulent Science. *Journal of Language and Social Psychology*, *35*(4), 435–445. https://doi.org/10.1177/0261927X15614605

Martin, B. (2016). Plagiarism, Misrepresentation, and Exploitation by Established Professionals: Power and Tactics. In *Handbook of Academic Integrity* (pp. 913–927). https://doi.org/10.1007/978-981-287-098-8

Maurer, H., Kappe, F., & Zaka, B. (2006). Plagiarism-a Survey. *Journal of Universal Computer Science*, *12*(8), 1050–1084.

Mccabe, D. L. (2005). Cheating among College and University Students: A North American Perspective. *International Journal of Educational Integrity*, *1*(1), 1–11. https://doi.org/10.21913/IJEI.V1I1.14

McCabe, D. L., Feghali, T., & Abdallah, H. (2008). Academic Dishonesty in the Middle East: Individual and Contextual Factors. *Research in Higher Education*, *49*(5), 451–467. https://doi.org/10.1007/s11162-008-9092-9

McEnery, T., Xiao, R., & Tono, Y. (2006). Unit 9 Copyright. In *Corpus-based Language Studies: An Advanced Resource Book*. London: Routledge.

McNamee, P., & Mayfield, J. (2004). Character N-Gram Tokenization for European Language Text Retrieval. *Information Retrieval*, *7*(1–2), 73–97. https://doi.org/10.1023/B:INRT.0000009441.78971.be

Medhat, W., Hassan, A., & Korashy, H. (2014). Sentiment Analysis Algorithms and Applications: A Survey. *Ain Shams Engineering Journal*, *5*(4), 1093–1113. https://doi.org/10.1016/j.asej.2014.04.011

Menai, M. E. B. (2012). Detection of Plagiarism in Arabic Documents. *International Journal of Information Technology and Computer Science*, *4*(10), 80–89. https://doi.org/10.5815/ijitcs.2012.10.10

Menai, M. E. B., & Bagais, M. (2011). APlag: A Plagiarism Checker for Arabic Texts. In *Proceedings of the 6th IEEE International Conference on Computer Science and Education (ICCSE'11)* (pp. 1379–1383).

Singapore: IEEE. https://doi.org/10.1109/ICCSE.2011.6028888

Menzies, T., & Shepperd, M. (2019). Bad Smells in Software Analytics Papers. *Information and Software Technology*, *112*, 35–47.

Meskaldji, K., Chikhi, S., & Bensalem, I. (2018). A New Multi-Varied Arabic Corpus. *The International Conference on Pattern Analysis and Intelligent Systems (PAIS 2018)*, 1–5. https://doi.org/10.1109/PAIS.2018.8598524

Meuschke, N., & Gipp, B. (2013). State-of-the-Art in Detecting Academic Plagiarism. *International Journal for Educational Integrity*, *9*(1), 50–71.

Meyer zu Eißen, S., & Stein, B. (2006). Intrinsic Plagiarism Detection. In M. Lalmas, A. MacFarlane, S. Rüger, A. Tombros, T. Tsikrika, & A. Yavlinsky (Eds.), *Advances in Information Retrieval, Proceedings of the 28th European Conference on IR Research, ECIR 2006, London, LNCS vol. 3936* (Vol. 3936, pp. 565–569). Berlin, Heidelberg: Springer. https://doi.org/10.1007/11735106

Meyer zu Eißen, S., Stein, B., & Kulig, M. (2007). Plagiarism Detection without Reference Collections. In R. Decker & H.-J. Lenz (Eds.), *Advances in data analysis, Selected Papers from the 30th Annual Conference of the German Classification Society (GfKl), Berlin*, (pp. 359–366). Heidelberg: Springer. https://doi.org/10.1007/978-3-540-70981-7_40

Miao, Y., Eselj, V., & Milios, E. (2005). Document Clustering Using Character N-Grams: A Comparative Evaluation with Term-Based and Word-Based Clustering. In *Proceedings of the 14th ACM international conference on Information and knowledge management (CIKM'05)* (pp. 357–358). Bremen, Germany: ACM.

Miranda-García, a., & Calle-MartÍn, J. (2006). Yule's Characteristic K Revisited. *Language Resources and Evaluation*, *39*(4), 287–294. https://doi.org/10.1007/s10579-005-8622-8

Muhr, M., Kern, R., Zechner, M., & Granitzer, M. (2010). External and Intrinsic Plagiarism Detection Using a Cross-Lingual Retrieval and Segmentation System - Lab Report for PAN at CLEF 2010. In M. Braschler, D. Harman, E. Pianta, & N. Ferro (Eds.), *CLEF 2010 Evaluation Labs and Workshops –Working Notes papers, CEUR proceedings vol. 1176*. CEUR-WS.org.

Nagoudi, E. M. B., Cherroun, H., & Alshehri, A. (2018a). Disguised Plagiarism Detection in Arabic Text Documents. In *2018 2nd International Conference on Natural Language and Speech Processing (ICNLSP)*. IEEE. https://doi.org/10.1109/ICNLSP.2018.8374395

Nagoudi, E. M. B., Ferrero, J., Schwab, D., & Cherroun, H. (2018b). Word Embedding-Based Approaches for Measuring Semantic Similarity of Arabic-English Sentences. In A. Lachkar, K. Bouzoubaa, A. Mazroui, A. Hamdani, & A. Lekhouaja (Eds.), *Arabic Language Processing: From Theory to Practice. Proceedings of the 6th International Conference, ICALP 2017, Fez, Morocco, October 11–12*. Springer. https://doi.org/10.1007/978-3-319-73500-9_2

Nagoudi, E. M. B., Khorsi, A., Cherroun, H., & Schwab, D. (2018c). 2L-APD: A Two-Level Plagiarism Detection System for Arabic Documents. *Cybernetics and Information Technologies*, *18*(1), 124–138. https://doi.org/10.2478/cait-2018-0011

Oberreuter, G., Huillier, G. L., & R, A. (2011a). Outlier-Based Approaches for Intrinsic and External Plagiarism Detection. In *KES 2011, Part II, LNAI 6882* (pp. 11–20). Springer.

Oberreuter, G., L'Huillier, G., Ríos, S. A., & Velásquez, J. D. (2011b). Approaches for Intrinsic and External Plagiarism Detection - Notebook for PAN at CLEF 2011. In V. Petras, P. Forner, P. Clough, & N. Ferro (Eds.), *CLEF 2011 Evaluation Labs and Workshops –Working Notes papers, CEUR proceedings vol. 1177* (pp. 1–10). CEUR-WS.org.

Oberreuter, G., L'Huillier, G., Ríos, S. A., & Velásquez, J. D. (2011c). Outlier-Based Approaches for Intrinsic and External Plagiarism Detection. In *KES 2011, Part II, LNAI 6882* (pp. 11–20). Heidelberg: Springer.

Oberreuter, G., & Velásquez, J. D. (2013). Text Mining Applied to Plagiarism Detection: The Use of Words for Detecting Deviations in the Writing Style. *Expert Systems with Applications*, *40*(9), 3756–3763.

https://doi.org/10.1016/j.eswa.2012.12.082

Ottenstein, K. J. (1976). An Algorithmic Approach to the Detection and Prevention of Plagiarism. *ACM SIGCSE Bulletin*, *8*(4), 30–41. https://doi.org/10.1145/382222.382462

Pearce, C., & Nicholas, C. (1996). TELLTALE: Experiments in a Dynamic Hypertext Environment for Degraded and Multilingual Data. *Journal of the American Society for Information Science*, *47*(4), 263–275. https://doi.org/10.1002/(SICI)1097-4571(199604)47:4<263::AID-ASI2>3.0.CO;2-V

Peng, F., & Schuurmans, D. (2003). Combining Naive Bayes and N-Gram Language Models for Text Classification. In *Advances in Information Retrieval. ECIR 2003. Lecture Notes in Computer Science, vol 2633* (pp. 335–350). https://doi.org/10.1007/3-540-36618-0_24

Pertile, S. de L., Moreira, V. P., & Rosso, P. (2015). Comparing and Combining Content- and Citation-Based Approaches for Plagiarism Detection. *Journal of the Association for Information Science and Technology*, *67*(10), 2511–2526. https://doi.org/10.1002/asi.23593

Polydouri, A., Siolas, G., & Stafylopatis, A. (2017). Intrinsic Plagiarism Detection with Feature-Rich Imbalanced Dataset Learning. In G. Boracchi, L. Iliadis, C. Jayne, & A. Likas (Eds.), *Engineering Applications of Neural Networks. EANN 2017. Communications in Computer and Information Science, vol 744* (pp. 99–110). Springer. https://doi.org/10.1007/978-3-319-65172-9_9

Polydouri, A., Vathi, E., Siolas, G., & Stafylopatis, A. (2018). An Efficient Classification Approach in Imbalanced Datasets for Intrinsic Plagiarism Detection. *Evolving Systems*, 1–13. https://doi.org/10.1007/s12530-018-9232-1

Potthast, M., Stein, B., Eiselt, A., Barrón-Cedeño, A., & Rosso, P. (2009). Overview of the 1st International Competition on Plagiarism Detection. In B. Stein, P. Rosso, E. Stamatatos, M. Koppel, & E. Agirre (Eds.), *Proceedings of the SEPLN'09 Workshop on Uncovering Plagiarism, Authorship and Social Software Misuse (PAN 09), CEUR proceedings vol. 502* (pp. 1–9). CEUR-WS.org.

Potthast, M., Barrón-Cedeño, A., Eiselt, A., Stein, B., & Rosso, P. (2010a). Overview of the 2nd International Competition on Plagiarism Detection. In M. Braschler, D. Harman, E. Pianta, & N. Ferro (Eds.), *CLEF 2010 Evaluation Labs and Workshops –Working Notes papers, CEUR proceedings vol. 1176*. CEUR-WS.org.

Potthast, M., Barrón-Cedeño, A., Stein, B., & Rosso, P. (2010b). Cross-Language Plagiarism Detection. *Language Resources and Evaluation*, *45*(1), 45–62. https://doi.org/10.1007/s10579-009-9114-z

Potthast, M., Stein, B., Barrón-Cedeño, A., & Rosso, P. (2010c). An Evaluation Framework for Plagiarism Detection. In C.-R. Huang & D. Jurafsky (Eds.), *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)* (pp. 997–1005). Stroudsburg, USA: Association for Computational Linguistics.

Potthast, M., Eiselt, A., Barrón-Cedeño, A., Stein, B., & Rosso, P. (2011). Overview of the 3rd International Competition on Plagiarism Detection. In V. Petras, P. Forner, P. Clough, & N. Ferro (Eds.), *CLEF 2011 Evaluation Labs and Workshops –Working Notes papers, CEUR proceedings vol. 1177*. Amsterdam, The Netherland: CEUR-WS.org.

Potthast, M., Gollub, T., Hagen, M., Graßegger, J., Kiesel, J., Michel, M., Oberländer, A., Tippmann, M., Barrón-Cedeño, A., Gupta, P., Rosso, P., & Stein, B. (2012). Overview of the 4th International Competition on Plagiarism Detection. In P. Forner, J. Karlgren, C. Womser-Hacker, & N. Ferro (Eds.), *CLEF 2012 Evaluation Labs and Workshops –Working Notes Papers, CEUR proceedings vol. 1178*. CEUR-WS.org.

Potthast, M., Hagen, M., Gollub, T., Tippmann, M., Kiesel, J., Rosso, P., Stamatatos, E., & Stein, B. (2013a). Overview of the 5th International Competition on Plagiarism Detection. In P. Forner, R. Navigli, D. Tufis, & N. Ferro (Eds.), *CLEF 2013 Evaluation Labs and Workshops –Working Notes papers, CEUR proceedings vol. 1179*. CEUR-WS.org.

Potthast, M., Hagen, M., Völske, M., & Stein, B. (2013b). Crowdsourcing Interaction Logs to Understand Text Reuse from the Web. In *51st Annual Meeting of the Association of Computational Linguistics (ACL 2013)* (pp. 1212–1221). Association for Computational Linguistics.

Potthast, M., Gollub, T., Rangel, F., Rosso, P., Stamatatos, E., & Stein, B. (2014a). Improving the Reproducibility of PAN's Shared Tasks: Plagiarism Detection, Author Identification, and Author Profiling. In E. Kanoulas, M. Lupu, P. Clough, M. Sanderson, M. Hall, A. Hanbury, & E. Toms (Eds.), *Information Access Evaluation. Multilinguality, Multimodality, and Interaction. CLEF 2014. LNCS vol. 8685* (pp. 268–299). Springer. https://doi.org/10.1007/978-3-319-11382-1_22

Potthast, M., Hagen, M., Beyer, A., Busse, M., Tippmann, M., Rosso, P., & Stein, B. (2014b). Overview of the 6th International Competition on Plagiarism Detection. In L. Cappellato, N. Ferro, M. Halvey, & W. Kraaij (Eds.), *CLEF 2014 Evaluation Labs and Workshops –Working Notes papers, CEUR proceedings vol. 1180*. CEUR-WS.org.

Potthast, M., Göring, S., Rosso, P., & Stein, B. (2015). Towards Data Submissions for Shared Tasks : First Experiences for the Task of Text Alignment. In L. Cappellato, N. Ferro, G. J. F. Jones, & E. S. Juan (Eds.), *Working Notes of CLEF 2015 - Conference and Labs of the Evaluation forum, CEUR proceedings vol. 1391* (pp. 8–11). CEUR-WS.org.

Potthast, M., Rosso, P., Stamatatos, E., & Stein, B. (2019). A Decade of Shared Tasks in Digital Text Forensics at PAN. In *ECIR 2019, LNCS vol 11438* (pp. 291–300). Springer International Publishing. https://doi.org/10.1108/jd.2012.27868eaa.002

Prentice, F. M., & Kinden, C. E. (2018). Paraphrasing Tools, Language Translation Tools and Plagiarism: An Exploratory Study. *International Journal for Educational Integrity*, *14*(1), 1–16. https://doi.org/10.1007/s40979-018-0036-7

Rabinovich, E., Patel, R. N., Mirkin, S., Specia, L., & Wintner, S. (2017). Personalized Machine Translation: Preserving Original Author Traits, *1*, 1074–1084. https://doi.org/10.18653/v1/e17-1101

Rangel, F., Rosso, P., Koppel, M., Stamatatos, E., & Inches, G. (2013). Overview of the Author Profiling Task at PAN 2013. In P. Forner, R. Navigli, D. Tufis, & N. Ferro (Eds.), *CLEF 2013 Evaluation Labs and Workshops –Working Notes papers, CEUR proceedings vol. 1179*. CEUR-WS.org.

Rao, S., Gupta, P., Singhal, K., & Majumder, P. (2011). External & Intrinsic Plagiarism Detection : VSM & Discourse Markers Based Approach - Notebook for PAN at CLEF 2011. In V. Petras, P. Forner, P. Clough, & N. Ferro (Eds.), *CLEF 2011 Evaluation Labs and Workshops –Working Notes papers, CEUR proceedings vol. 1177* (pp. 2–6). CEUR-WS.org.

Rennie, J. D. M., Shih, L., Teevan, J., & Karger, D. R. (2003). Tackling the Poor Assumptions of Naive Bayes Text Classifiers. In *Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003)*. Washington DC.

Rogerson, A. M. (2017). Detecting Contract Cheating in Essay and Report Submissions: Process, Patterns, Clues and Conversations. *International Journal for Educational Integrity*, *13*(1). https://doi.org/10.1007/s40979-017-0021-6

Rogerson, A. M., & McCarthy, G. (2017). Using Internet Based Paraphrasing Tools: Original Work, Patchwriting or Facilitated Plagiarism? *International Journal for Educational Integrity*, *13*(1). https://doi.org/10.1007/s40979-016-0013-y

Rosso, P. (2015). Author Profiling and Plagiarism Detection. In *Information Retrieval: 8th Russian Summer School, RuSSIR 2014, Nizhniy, Novgorod, Russia, August 18-22, 2014, Revised Selected Papers* (pp. 229–250). Springer.

Rosso, P., Rangel, F., Potthast, M., Stamatatos, E., Tschuggnall, M., & Stein, B. (2016). Overview of PAN'16: New Challenges for Authorship Analysis: Cross-Genre Profiling, Clustering, Diarization, and Obfuscation. In N. Fuhr, P. Quaresma, T. Gonçalves, B. Larsen, K. Balog, C. Macdonald, L. Cappellato, & N. Ferro (Eds.), *Experimental IR Meets Multilinguality, Multimodality, and Interaction. CLEF 2016, LNCS 9822* (pp. 332–350). Springer. https://doi.org/10.1007/978-3-319-44564-9_28

Rosso, P., Potthast, M., Stein, B., Stamatatos, E., Rangel, F., & Daelemans, W. (2019). Evolution of the PAN Lab on Digital Text Forensics. In N. Ferro & C. Peters (Eds.), *Information Retrieval Evaluation in a Changing World. The Information Retrieval Series, vol 41* (pp. 461–485). Springer. https://doi.org/10.1007/978-3-030-22948-1_19

Rousseau, R. (2002). Claude Shannon: Scientist-Engineer. *Journal of Henan Normal University*, *30*(4), 1–13.

Sanchez-Perez, M. A., Sidorov, G., & Gelbukh, A. (2014). The Winning Approach to Text Alignment for Text Reuse Detection at PAN 2014 -Notebook for PAN at CLEF 2014. In L. Cappellato, N. Ferro, M. Halvey, & W. Kraaij (Eds.), *CLEF 2014 Evaluation Labs and Workshops –Working Notes papers, CEUR proceedings vol. 1180* (pp. 1004–1011). CEUR-WS.org.

Sánchez-Vega, F., Villatoro-Tello, E., Montes-y-Gómez, M., Rosso, P., Stamatatos, E., & Villaseñor-Pineda, L. (2019). Paraphrase Plagiarism Identification with Character-Level Features. *Pattern Analysis and Applications*, *22*(2), 669–681. https://doi.org/10.1007/s10044-017-0674-z

Sapkota, U., Bethard, S., y Gómez, M. M., & Solorio, T. (2015). Not All Character N-Grams Are Created Equal: A Study in Authorship Attribution. In *2015 Conference of the North American Chapter of the Association for Computational Linguistics – Human Language Technologies (NAACL HLT 2015)* (pp. 93–102). https://doi.org/10.3115/v1/N15-1010

Schleimer, S., Wilkerson, D. S., & Aiken, A. (2003). Winnowing: Local Algorithms for Document Fingerprinting. *Proceedings of the 2003 ACM SIGMOD international conference on on Management of data - SIGMOD '03*, 76–85. https://doi.org/10.1145/872757.872770

Seaward, L., & Matwin, S. (2009). Intrinsic Plagiarism Detection Using Complexity Analysis. In B. Stein, P. Rosso, E. Stamatatos, M. Koppel, & E. Agirre (Eds.), *Proceedings of the SEPLN'09 Workshop on Uncovering Plagiarism, Authorship and Social Software Misuse (PAN 09)* (pp. 56–61). CEUR-WS.org.

Shahid, U., Farooqi, S., Ahmad, R., Shafiq, Z., Srinivasan, P., & Zaffar, F. (2017). Accurate Detection of Automatically Spun Content via Stylometric Analysis. In *Proceedings - IEEE International Conference on Data Mining, ICDM* (Vol. 2017-Novem, pp. 425–434). https://doi.org/10.1109/ICDM.2017.52

Shannon, C. E. (1951). Prediction and Entropy of Printed English. *Bell System Technical Journal*, *30*(1), 50–64. https://doi.org/10.1002/j.1538-7305.1951.tb01366.x

Shrestha, P., & Solorio, T. (2015). Identification of Original Document by Using Textual Similarities. In A. Gelbukh (Ed.), *CICLing 2015, Part II, LNCS 9042* (pp. 643–654). Springer. https://doi.org/10.1007/978-3-319-18117-2_48

Siddiqui, M. A., Khan, I. H., Jambi, K. M., Elhaj, S. O., & Bagais, A. (2014). Developing an Arabic Plagiarism Detection Corpus. In *The Sixth International Conference on Wireless & Mobile Networks (WiMoNe - 2014), December 27 ~ 28 - 2014, Sydney, Australia* (pp. 261–269). Academy & Industry Research Collaboration Center (AIRCC). https://doi.org/10.5121/csit.2014.41221

Simpson, E. H. (1949). Measurement of Diversity. *Nature*, *163*, 688–688. https://doi.org/10.1038/163688a0

Sittar, A., Iqbal, H. R., Muhammad, R., & Nawab, A. (2016). Author Diarization Using Cluster-Distance Approach. In K. Balog, L. Cappellato, N. Ferro, & C. Macdonald (Eds.), *Working Notes of CLEF 2016 - Conference and Labs of the Evaluation forum, CEUR proceedings vol. 1609* (pp. 1000–1007). CEUR-WS.org.

Soori, H., Prilepok, M., Platos, J., Berhan, E., & Snasel, V. (2014). Text Similarity Based on Data Compression in Arabic. In I. Zelinka, V. H. Duy, & J. Cha (Eds.), *AETA 2013: Recent Advances in Electrical Engineering and Related Sciences* (Vol. 282, pp. 211–220). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-41968-3

Stamatatos, E. (2009a). Intrinsic Plagiarism Detection Using Character N-Gram Profiles. In B. Stein, P. Rosso, E. Stamatatos, M. Koppel, & E. Agirre (Eds.), *Proceedings of the SEPLN'09 Workshop on Uncovering Plagiarism, Authorship and Social Software Misuse (PAN 09)* (pp. 38–46). CEUR-WS.org.

Stamatatos, E. (2009b). A Survey of Modern Authorship Attribution Methods. *Journal of the American Society for Information Science*, *60*(3), 538–556. https://doi.org/10.1002/asi

Stamatatos, E. (2013). On the Robustness of Authorship Attribution Based on Character N-Gram Features. *Journal of Law & Policy*, *21*(2), 421–439.

Stamatatos, E. (2016). Universality of Stylistic Traits in Texts. In M. D. Esposti, E. G. Altmann, & F. Pachet (Eds.), *Creativity and Universality in Language* (pp. 143–155). Springer. https://doi.org/10.1007/978-3-319-24403-7_9

Stamatatos, E. (2018). Masking Topic-Related Information to Enhance Authorship Attribution. *Journal of the Association for Information Science and Technology*, *69*(3), 461–473. https://doi.org/10.1002/asi.23968

Stamatatos, E., Fakotakis, N., & Kokkinakis, G. (2001). Computer-Based Authorship Attribution without Lexical Measures. *Computers and the Humanities*, *35*(2), 193–214. https://doi.org/10.1023/A:1002681919510

Stamatatos, E., Tschuggnall, M., Verhoeven, B., Daelemans, W., Specht, G., Stein, B., & Potthast, M. (2016). Clustering by Authorship Within and Across Documents. In K. Balog, L. Cappellato, N. Ferro, & C. Macdonald (Eds.), *Working Notes of CLEF 2016 - Conference and Labs of the Evaluation forum, CEUR proceedings vol. 1609* (pp. 691–715). CEUR-WS.org.

Stein, B., Koppel, M., & Stamatatos, E. (2007). Plagiarism Analysis, Authorship Identification, and near-Duplicate Detection PAN'07. *ACM SIGIR Forum*, *41*(2), 68–71. https://doi.org/10.1145/1328964.1328976

Stein, B., Lipka, N., & Prettenhofer, P. (2011). Intrinsic Plagiarism Analysis. *Language Resources and Evaluation*, *45*(1), 63–82. https://doi.org/10.1007/s10579-010-9115-y

Stein, B., & Meyer zu Eißen, S. (2007). Intrinsic Plagiarism Analysis with Meta Learning. In B. Stein, M. Koppel, & E. Stamatatos (Eds.), *Proceedings of the SIGIR'07 International Workshop on Plagiarism Analysis, Authorship Identification, and Near-Duplicate Detection (PAN 2007), Amsterdam, Netherlands* (pp. 45–50). CEUR-WS.org.

Suárez, P., González, J. C., & Villena-Román, J. (2010). A Plagiarism Detector for Intrinsic Plagiarism - Lab Report for PAN at CLEF 2010. In M. Braschler, D. Harman, E. Pianta, & N. Ferro (Eds.), *CLEF 2010 Evaluation Labs and Workshops –Working Notes papers, CEUR proceedings vol. 1176*. CEUR-WS.org.

Suchomel, Š., Kasprzak, J., & Brandejs, M. (2012). Three Way Search Engine Queries with Multi-Feature Document Comparison for Plagiarism Detection - Notebook for PAN at CLEF 2012. In P. Forner, J. Karlgren, C. Womser-Hacker, & N. Ferro (Eds.), *CLEF 2012 Evaluation Labs and Workshops –Working Notes Papers, CEUR proceedings vol. 1178*. CEUR-WS.org.

Tabana, B. (1956). Literary Thefts: A Study in the Literary Works' Creation and Imitation (in Arabic). Cairo: Nahdhat Misr.

Tschuggnall, M. (2014). Intrinsic Plagiarism Detection and Author Analysis By Utilizing Grammar (PhD thesis). University of Innsbruck, Austria.

Tschuggnall, M., & Specht, G. (2012). Plag-Inn: Intrinsic Plagiarism Detection Using Grammar Trees. In *NLDB 2012, LNCS, vol. 7337* (pp. 284–289). Heidelberg: Springer.

Tschuggnall, M., & Specht, G. (2013a). Countering Plagiarism by Exposing Irregularities in Authors' Grammar. *2013 European Intelligence and Security Informatics Conference*, 15–22. https://doi.org/10.1109/EISIC.2013.10

Tschuggnall, M., & Specht, G. (2013b). Detecting Plagiarism in Text Documents through Grammar-Analysis of Authors. In V. Markl, G. Saake, K.-U. Sattler, G. Hackenbroich, B. Mitschang, T. Härder, & V. Köppen (Eds.), *15. GI-Fachtagung Datenbanksysteme für Business, Technologie und Web (BTW 2013), Magdeburg, Germany* (pp. 241–259). GI.

Tschuggnall, M., & Specht, G. (2013c). Using Grammar-Profiles to Intrinsically Expose Plagiarism in Text Documents. In *NLDB 2013, LNCS, vol. 7934* (pp. 297–302). Springer.

Tschuggnall, M., & Specht, G. (2014). Automatic Decomposition of Multi-Author Documents Using Grammar Analysis. In *Proceedings of the 26th GI-Workshop on Foundations of Databases (Grundlagen von Datenbanken)* (pp. 17–22). CEUR-WS.org.

Tschuggnall, M., Stamatatos, E., Verhoeven, B., Daelemans, W., Specht, G., Stein, B., & Potthast, M. (2017). Overview of the Author Identification Task at PAN-2017: Style Breach Detection and Author Clustering. In L. Cappellato, N. Ferro, L. Goeuriot, & T. Mandl (Eds.), *Working Notes of CLEF 2017 - Conference and Labs of the Evaluation Forum, CEUR proceedings vol. 1866*. CLEF and CEUR-WS.org.

Tweedie, F. J., & Baayen, R. H. (1998). How Variable May a Constant Be ? Measures of Lexical Richness in Perspective. *Computers and the Humanities*, *32*(5), 323–352. https://doi.org/10.1023/A:1001749303137

van Halteren, H. (2003). Detection of Plagiarism in Student Essays. In *Computational linguistics in the Netherlands 2003 : selected papers from the fourteenth CLIN meeting* (pp. 157–169).

van Halteren, H. (2004). Linguistic Profiling for Author Recognition And Verification. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics* (Article No. 199). Association for Computational Linguistics. https://doi.org/10.3115/1218955.1218981

Vaux, D. L. (2016). Scientific Misconduct: Falsification, Fabrication, and Misappropriation of Credit. In T. Bretag (Ed.), *Handbook of Academic Integrity* (pp. 895–911). Singapore: Springer. https://doi.org/10.1007/978-981-287-098-8

von Grunebaum, G. E. (1944). The Concept of Plagiarism in Arabic Theory. *Journal of Near Eastern Studies*, *3*(4), 234–253. https://doi.org/10.1086/370723

Weber-Wulff, D. (2015). Plagiarism Detection Software: Promises, Pitfalls, and Practices. In T. Bretag (Ed.), *Handbook of Academic Integrity* (pp. 1–10). Singapore: Springer Singapore. https://doi.org/10.1007/978-981-287-079-7_19-1

Yerra, R., & Ng, Y.-K. (2005). A Sentence-Based Copy Detection Approach for Web Documents. In *Fuzzy Systems and Knowledge Discovery (FSKD 2005), LNCS vol 3613* (pp. 557–570). Springer. https://doi.org/10.1007/11539506_70

Yule, G. U. (1944). The Statistical Study of Literary Vocabulary. Cambridge University Press.

Zangerle, E., Tschuggnall, M., Specht, G., Stein, B., & Potthast, M. (2019). Overview of the Style Change Detection Task at PAN 2019. In L. Cappellato, N. Ferro, D. E. Losada, & H. Müller (Eds.), *Working Notes of CLEF 2019 - Conference and Labs of the Evaluation Forum, CEUR proceedings vol. 2380*. CEUR-WS.org.

Zečević, A. (2011). N-Gram Based Text Classification According To Authorship. In *Proceedings of the Student Research Workshop associated with RANLP 2011* (pp. 145–149). Hissar, Bulgaria: Association for Computational Linguistics.

Zechner, M., Muhr, M., Kern, R., & Granitzer, M. (2009). External and Intrinsic Plagiarism Detection Using Vector Space Models. In B. Stein, P. Rosso, E. Stamatatos, M. Koppel, & E. Agirre (Eds.), *Proceedings of the SEPLN'09 Workshop on Uncovering Plagiarism, Authorship and Social Software Misuse (PAN 09)* (pp. 47–55). CEUR-WS.org.

Zhang, X., Zhao, J., & LeCun, Y. (2015). Character-Level Convolutional Networks for Text Classification. In *NIPS'15 Proceedings of the 28th International Conference on Neural Information Processing Systems* (pp. 649–657). MIT press. https://doi.org/10.1063/1.4906785