Text Alignment Module in CoReMo 2.1 Plagiarism Detector

Notebook for PAN at CLEF 2013

Diego A. Rodríguez Torrejón, José Manuel Martín Ramos
Universidad de Huelva
dartsystems@gmail.com, jmmartin@dti.uhu.es

Abstract. This paper describes the process and basics of the Text Alignment Module into the *CoReMo 2.1* Plagiarism Detector, which has won the Plagiarism Detection Text Alignment task in PAN-2013 edition, for both evaluation criteria of efficacy and efficiency, achieving the best detections and the best runtime too. Its high detection efficacy is mainly due to the special features of the contextual n-grams, evolved to surrounding context and odd-even skip n-grams. When combined all together, the matching opportunity increases, especially when translations or paraphrases happen, but keeping its highly discriminative feature that simplifies the accurate location for plagiarized sections. The optimized process by high performance C/C++ multi-core programming techniques, has yielded the best speed, but the tests were arranged in single core machines, so you can expect much better runtime.

1 Introduction

Plagiarism Detection is one of the fields that is awakening interest in the areas of Natural Language Processing and Information Retrieval. The various PAN[1] editions are continuously enforcing the improvement of existing techniques, compiling corpus with cases more realistic and difficult to detect, and developing systems, work plans and tasks to design and analyze the individual impact of proposals for the different subtasks about the performance obtained, the necessary hardware resources and time spent, thus facilitating the subsequent combination and improvement proposals in search of the ultimate plagiarism detector. *CoReMo* [1], [2], [3], [4], [12] is a Plagiarism Detection System that was initially designed to take part in PAN issues, which has achieved the highest performance results, and also highlighted hardware requirements and processing speed (one of the main goals for its developers). However, *CoReMo* uses pruning techniques to avoid the comparison of the full suspicious document with any source document if not detected evidence of plagiarism by its *High Accuracy Information Retrieval System (HAIRS)* and the *Reference Monotony Pruning strategy (RM)*, delimiting the suspected plagiarized section before making any comparisons with the suspicious document. *CoReMo* started to perform

1 http://pan.webis.de

exhaustive full documents pair comparisons in the *PAN-12* issue as a new feature. The detection capability, when compared to previous edition, was then greatly improved by extending the n-grams model used (*Contextual N-grams CTnG*) to *Surrounding Context N-grams* (*SCnG*) [4],[12], and the use of a post-processing to join closed detections (*Granularity Filter*). The new Text Alignment capability design looks for the maximal computational efficiency, usual in former *CoReMo* versions.

New improvements were arranged this year, extending again the model by *Odd-Even N-grams (OEnG)*, a best self-adaptive parameters tunning and multi-core redesign (the competition doesn't take advantage of that feature). A latter bug fixing achieved a better *Plagdet* (0.82827) than the official one (0,82220).

2 Surrounding Context N-grams and Odd-Even N-Grams

One of the most important innovations in the *CoReMo* last year's version was that the documents were modeled by extending the concept of former *Contextual N-grams* [1-2] (*CTnG*: case folding, stopwords and short length words removal, stemming and internal sort of n-gram components) to the *Surrounding Context N-grams* (*SCnG*) [4, 12], a special type of *skip n-grams* obtained by excluding the second or the last but one from a group of n+1 relevant terms joined to the previously explained CTnG process. The new *CoReMo* 2.1 also includes in the model another skip n-grams type, from odd or even relevant words (*OEnG*) processed in same *CTnG* way.

For instance, new modeling for "The ***quick brown fox jumps*** over the ***lazy dog***":

1. quick brown fox → brown_fox_quick (1^{st} direct *CT3G* way)
2. quick brown jumps → brown_jump_quick (1^{st} left-hand *SC3G* way)
3. quick fox jumps → fox_jump_quick (1^{st} right-hand *SC3G* way)
4. quick fox lazy → laz_fox_quick (1^{st} *OE3G* way)
5. brown fox jumps → brown_fox_jump (2^{nd} direct *CT3G* way)
6. brown fox lazy → brown_fox_laz (2^{nd} left-hand *SC3G* way)
7. brown jumps lazy → brown_jump_laz (2^{nd} right-hand *SC3G* way)
8. brown jumps dog → brown_dog_jumps (2^{nd} *OE3G* way)
9. fox jumps lazy → fox_jump_laz (3^{th} direct *CT3G* way)
10. fox jumps dog → dog_fox_jump (3^{nd} left-hand *SC3G* way)
11. fox lazy dog → dog_laz_fox (3^{th} right-hand *SC3G* way)
12. jumps lazy dog → dog_jump_laz (4^{th} direct *CT3G* way)

The including of *SCnG* and *OEnG* gets four times as many n-grmas (and matching chances) than the original *CTnG* method. It offers more possibilities to tackle obfuscation cases with almost the same practical high precision in the process. Once again, a higher n-grams quantity obtained acts as a magnifier effect in the analysis.

The memory requirements and processing time have obviously increased, but it improves dramatically the performance. Including these skip n-grams almost doesn't decreases the precision. N-gram *idf* studies on PAN-PC-2009 / 2010 / 2011 (table 1) corpora [5] show its exclusivity ratio almost unaltered.

Table 1. n-gram frequency study on PAN-PC-2011 only english source documents subcorpus

idf	quantity	ratio	quantity	ratio	quantity	ratio
	CT3G only		CT3G + SC3G		CT3G + SC3G + OE3G	
--	144426869	1.0000	408447501	1.0000	537613396	1.0000
01	132790997	0.9194	367321473	0.8993	481407991	0.8955
02	7559052	0.0523	25496723	0.0624	34537949	0.0642
03	1977892	0.0137	7253659	0.0178	9974359	0.0186
04	811445	0.0056	3120363	0.0076	4327470	0.0080
...						
97	43	0.0000	215	0.0000	265	0.0000
98	32	0.0000	184	0.0000	260	0.0000
99	45	0.0000	179	0.0000	261	0.0000
> 99	1663	0.0000	6379	0.0000	8626	0.0000

All n-grams are compared without a difference in the way they are created. The *SCnG* and *OEnG* are especially useful to improve the *CTnG* effectiveness when words changes (synonyms, negated antonyms, given names, translation or orthographic errors, characters changed by other UTF code having the same aspect, ...), new word insertions (enriched sentences) or removal (summarized sentences). The sentence reordering due to translation or changing from passive to active forms or vice versa are also supported.

This way gets more matching, especially for paraphrased or translated cases, to identify a possible plagiarism (almost as when using lower grade n-grams, but with more precise disambiguation instead). However, it gets more unconnected short detections which require to be joined. A distance joining step, named *Granularity*

Filter (*GF*) gets improved scores. Both *SCnG* and *GF* modes combined achieves about 45% best Plagdet score than when using direct *CTnG mode*. The inclusion of *OEnG* in the model gets a small but welcome improvement (+0,005). In order to facilitate the n-grams *location*, its modeling includes *offset and length* recording. The benefit of using this extended n-gram modeling compared to the former, based only in Contextual N-grams was shown in [4], improving the performance in a former *CoReMo* version, as can be seen in fig. 1 and fig. 2.

3 Detailed Comparison

Since by using this extended n-gram model, the matching is highly discriminative and more frequent, it's possible to get enough matching n-grams with very low noise, making the comparison tasks easier. For this detailed pair comparison task, alphabetically ordered versions of both *n-gram* modeled documents, with inner matching annotations and linking, are compared in the way of a modified "mergesort" [6] algorithm to speed up the job, linking every *SCnG* to an external matching list.

Minimum length and maximum distances between matches (for same detection) are adjusted, on bases of document length, number of n-grams and user settings for minimal monotony and n-grams *chunk* length (the basics classical adjustments in *CoReMo*), which differ for cross-lingual (not used this PAN issue) and monolingual comparison.

The distances are n-grams for suspicious documents and characters for the sources:

$$maxNgramDist = 2 \cdot chunkLength \quad (1)$$

$$maxCharDist = chunkLength \cdot wordLengthAverage \quad (2)$$

$$minNgramLength = (monotony - 1.5) \cdot chunkLength \quad (3)$$

$$minCharLenght = minNgramLength \cdot wordLengthAverage \quad (4)$$

The reliability of the matching n-grams is pondered by its inner matching frequency in both suspicious and source documents, to determine or reject the detected continuous matching sections and to create preliminary XML files (direct detection). After the end of a detection, a roll-back to the next n-gram happens starting the next possible detection (have in mind that a detection finishes when no new reliable match has been found after several n-grams).

The direct detections are post-processed by the *Granularity Filter* to join simultaneously nearby detections (4000 chars) in both suspicious and source sections, getting final XML detection files. Both XML files could be combined to create a best comparison readable HTML coloured document to emphasize direct detections within the final zones.

4 Self-Adaptive Tuning Parameters

The amount of false positives obtained in the no-plagiarism sub-corpus highly increases when the parameters are adjusted to improve the most difficult detections (shorter ones or the summarized ones). By tuning the chunk length to 3 as in former version, it was achieved a *Plagdet* performance of 0.4388 for the summary obfuscation, but it penalizes too much for the global results due to false positives when no plagiarism happens. It's necessary to discern when there is low obfuscation, high, none or no plagiarism. In the aim of auto select a best tuning that would affect minimally for false positives, the information obtained in the matching annotation process was taken into account about the inner and external matching rate (*imr* and *emr*) for both suspicious and source modeled documents. This analysis is yet in its infancy, but by the moment it gets to adapt the chunk lenght (*cl*) to different regions depending of the external matching rate (*emr*) for both documents:

- base case: cl = 8 * multiplicty factor
- *emr1* > 4% & *emr2* < 15% → *cl* = 3 *cl* / 7
- *emr1* > 30% & *emr2* >= 15% → *cl* = 2 *cl* / 3

5 Speed up Methodology

As one of the main goals for *CoReMo* is the high speed to obtain reliable detection results (and also to get the fastest and widest experimental process), the execution environment and the programming techniques focused on getting a maximal computational efficiency were used from the early design:

- C++ 64 bits programming, now powered by OpenMP 3.0
- GNU Linux 64bits OS and ext4 file system platform.
- Internal sort of n-grams is made by bubble sort algorithm.
- Quick sort algorithm is used to order n-grams into the modeled document.
- N-gram comparison between both documents is arranged by a modified *mergesort* algorithm [6]
- Local translation when cross-lingual comparisons happens.
- When comparing pairs lists, ordered by suspicious documents (the most usual case after locating source documents candidates), it is taken the advantage of n-grams modeling and inner matching in the suspicious one for consecutive comparisons.

It made it possible to achieve an average analysis time of 14 milliseconds per pair: 4 times faster than the second fastest algorithm, and 5 times faster than next one for effectiveness. However, the competition test uses single core virtual machines, and

the advantage of our multi-core optimized software is not taken. A real performance instance using a PC with AMD-FX8120@4.00GHz, 4 GB PC1600 DDR RAM and a SATA3 SSD: the runtime for the 2013 training corpus is only 4790 ms (0,923 ms/pair) instead of 13610 ms got when single core mode is used.

6 Tuning Parameters and Evaluation

The detailed results of the training (plagdet 0.8272) are displayed and compared to the ones achieved in the phase of competition (0.8222) in table 2. The best parameters settings were experimentally obtained by using the *PAN-PC-TA-2013* training corpus:

- n-gram grade: 3
- chunk length: 8 n-grams[2] (internally it changes to 32 when using *SCnG + OEnG*).
- minimum monotony: 2 chunks (same for monolingual or crosslingual modes).

The results obtained in training phase by both, buggy and bug fixed versions, were highly similar to the achieved in evaluation.

7 Conclusions and Future Work

Nowadays *CoReMo* is the fastest detector, but it's now optimized to take the opportunity of multi-core systems advantage. The lack of possibility of using a multi-core system in competition doesn't show the real system possibilities in the current machines, and the runtime is penalized due to the use of unnecessary concurrent programming techniques when only a single core is going to be used. We are planning to adapt our software to use GPU processors, with thousands of available cores. Those versions couldn't take part in next issues, if we don't provide a specially adapted version with a runtime power much lower than the real production systems.

Mixing this n-gram modeling with other NLP resources (as WordNet synsets) could improve detections when hardest obfuscation conditions happen.

The comparison of the *Plagdet* progress regarding the *PAN2012* must be done with caution, since not being necessary translation for any case in 2013.

Acknowledgments. To the PAN team, as their development aids, hard job and encourage have been crucial for our work, and to all the PAN competitors teams, as their effort and papers has always been for us a motivational challenge and a source of new ideas to improve our detection system. To the Elias/ESF for the last edition grant to assist to CLEF 2012, a great opportunity and incentive for our job.

2 For 3th grade n-grams, chunk length of 8 n-grams is equivalent to 10 relevant words.

Fig. 1. Plagdet/chunk_length comparative of CoReMo 1.6 using CT3N or SC3N w/wo Granularity Filter on PAN-PC-2011 only English subcorpus [4]

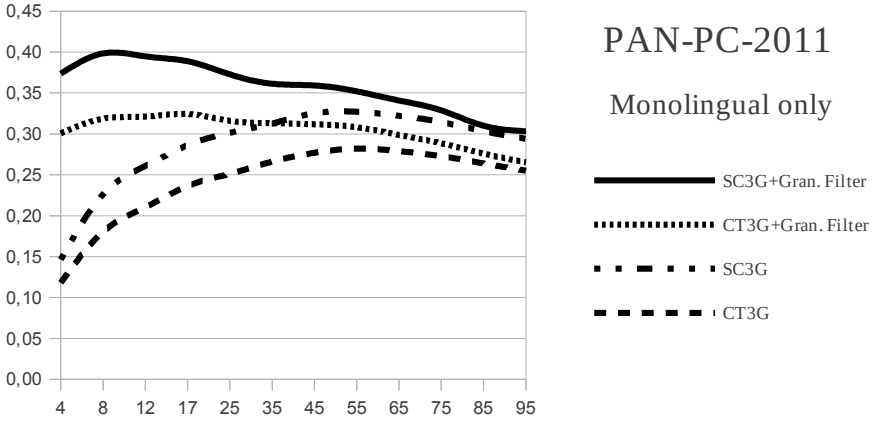

Fig. 2. Plagdet/chunk_length comparative of CoReMo 1.6 using CT3N or SC3N w/wo Granularity Filter on PAN-PC-2011 non-English subcorpus [4]

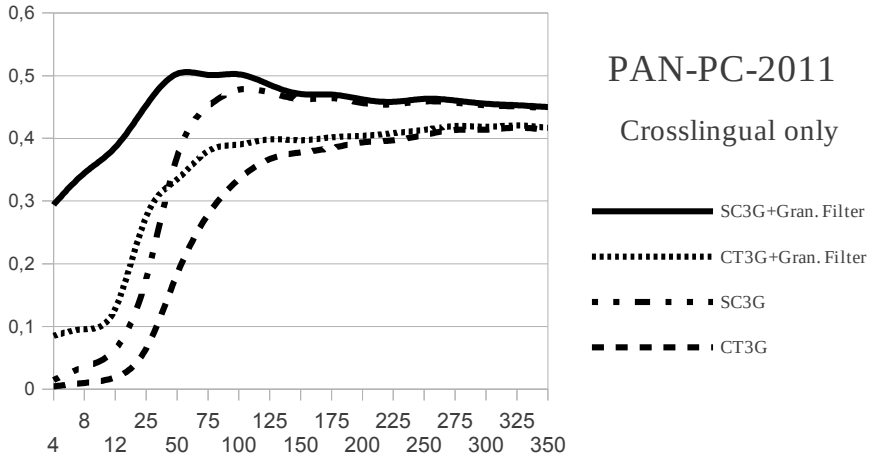

Table 2. ***CoReMo*** **21 achieved scores in training and competition phases with same tuned parameters: chunklength = 8 , monotony = 2 , adaptive mode on and filter distance = 4000 chars.**

	PAN-PC-2012 Training Corpus				**PAN-PC-2012 Competition Corpus**				
	Plagdet	***Recall***	***Precision***	***Granularity***	***Plagdet***	***Recall***	***Precision***	***Granularity***	***runtime (ms)***
No obfuscation	0.92733	0.97326	0.88554	1.00000	0.92586	0.95256	0.90060	1.00000	
Random obfus.	0.75527	0.63388	0.93417	1.00000	0.74711	0.63370	0.90996	1.00000	
Translated obfus.	0.84683	0.79951	0.90001	1.00000	0.85113	0.81124	0.89514	1.00000	
Summary obfus.	0.35513	0.22973	0.87716	1.03529	0.34131	0.21593	0.90750	1.07742	
Global					**0.82220**	0.76190	0.89484	1.00141	**72508**
Global bug fixed[3]	**0.82722**	0.76758	0.89929	1.00169	**0.82827**	0.77177	0.89564	1.00140	79965

3 A bug fixing after submission deadline achieves best Plagdet (0.82827) than the official one (0,82220). Results in training corpus are due to bug fixed version. Competion ones are by buggy version, and only Global bug fixed are shown.

References

1. Rodríguez-Torrejón, D.A., Martín-Ramos, J.M.: “Detección de plagio en documentos: sistema externo monolingüe de altas prestaciones basado en n-gramas contextuales” (Plagiarism Detection in Documents: High Performance Monolingual External Plagiarism Detector System Based on Contextual N-grams). Procesamiento del Lenguaje Natural. N. 45 (2010).
2. Rodríguez-Torrejón D.A., Martín-Ramos J.M.: CoReMo System (Contextual Reference Monotony) A Fast, Low Cost and High Performance Plagiarism Analyzer System: Lab Report for PAN at CLEF 2010. In Braschler M., Harman D., Pianta E., editors. Notebook Papers of CLEF 2010 LABs and Workshops, 22-23 September, Padua, Italy, 2010.
3. Rodríguez-Torrejón, D.A., Martín-Ramos, J.M.: Crosslingual CoReMo System: Notebook for PAN at CLEF 2011. In [10].
4. Rodríguez-Torrejón, D.A., Martín-Ramos, J.M.: N-gramas de contexto cercano para mejorar la detección de plagio (Surrounding Context N-grams to Improve the Plagiarism Detection) In [11]
5. Martin Potthast, Benno Stein, Alberto Barrón-Cedeño, and Paolo Rosso. An Evaluation Framework for Plagiarism Detection. In 23rd International Conference on Computational Linguistics (COLING 10), August 2010. Association for Computational Linguistics.
6. Chiara Basile, Dario Benedetto, Giampaolo Caglioti, and Mirko Degli Esposti. 2009. A Plagiarism Detection Procedure in Three Steps: Selection, Matches and Squares. In SEPLN 2009 Workshop on Uncovering Plagiarism, Authorship, and Social Software Misuse (PAN 09) (pan, 2009), pages 19–23.
7. Rodríguez-Torrejón, D.A., Barrón-Cedeño, A., Sidorov, G., Martín-Ramos, J.M., Rosso, P.: “Influencia del diccionario en la traducción para la detección de plagio translingüe”. (Dictionary Influence in Crosslingual Plagiarism Detection). in [11]
8. Rodríguez-Torrejón, D.A., Martín-Ramos, J.M.: “LEAP: una referencia para la evaluación de sistemas de detección de plagio con enfoque intrínseco” (LEAP: a Baseline for Intrinsic Focusing Plagiarism Detectors). In [11]
9. Potthast, M., Eiselt, A., Barrón-Cedeño, A., Stein, B., Rosso P.: Overview of the 3rd International Competition on Plagiarism Detection. In [10]
10. Vivien Petras and Paul Clough (Eds.): Notebook Papers of CLEF 2011 Labs and Workshops, 19-22 September, Amsterdam, The Netherlands (2011).
11. II Congreso Español de Recuperación de Información (CERI 2012). 17-18 June, Valencia (2012). http://users.dsic.upv.es/grupos/nle/ceri/index.html
12. Rodríguez-Torrejón. D.A. and Martín-Ramos, J.M.: “Detailed Comparison Module In CoReMo 1.9 Plagiarism Detector”—Notebook for PAN at CLEF 2012. In Forner et al. [13]. URL http://www.clef-initiative.eu/publication/working-notes
13. Pamela Forner, Jussi Karlgren, and Christa Womser-Hacker, editors. CLEF 2012 Evaluation Labs and Workshop – Working Notes Papers, 17-20 September, Rome, Italy, 2012. URL http://www.clef-initiative.eu/publication/working-notes