Detailed Comparison Module In CoReMo 1.9 Plagiarism Detector

Notebook for PAN at CLEF 2012

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 Detailed Comparison Module into the *CoReMo* 1.9 Plagiarism Detector, which has got a highlighted mention in the PAN2012 edition due to its running speed (at least 10 times faster than any other competitor) achieving very good detections. Its high detection efficacy is due to the special features of the contextual and surrounding context n-grams, which working together, increase the opportunity to match, especially when translations or paraphrases happen, but keeping a highly discriminative feature that simplifies the accurate location for plagiarized sections. The independence of external translation systems coupled to its optimized process by high performance C/C++ programming techniques, have yielded its high speed even when it isn't yet multi-core systems optimized.

1 Introduction

Plagiarism Detection is one of the fields that are awakening interest in the areas of Natural Language Processing and Information Retrieval. The various PAN[1] editions are continuously promoting 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] is a Plagiarism Detection System that was initially designed for participation in PAN issues, obtaining very acceptable performance results, but highlighted for hardware requirements and processing speed (one of the main goals for its developers), which this year has had the opportunity to demonstrate. However, *CoReMo* uses pruning techniques to avoid the comparison of the suspicious document with any source document if not detected evidence of plagiarism by its High Precision Information Retrieval System (HAIRS) and the Reference Monotony Pruning strategy (RM) delimiting the suspected plagiarized section before making any comparisons with the

1 http://pan.webis.de

suspicious document. Therefore, *CoReMo* did not performed exhaustive documents pair comparisons until the proposal for this PAN issue, forcing to change the design to meet the characteristics of the new edition, which seeks comprehensive comparison for pairs of documents. However, this is not the only new feature included in this *CoReMo* release, as the detection capability, when compared to the previous edition, was greatly improved by extending the n-grams model used (*Contextual N-grams CTnG*) to *Surrounding Context N-grams* (*SCnG*) [4] and the use of a post-processing to join closed detections (*Granularity Filter*).

The new Detailed Comparison capability design was arranged looking for the maximal computational efficiency, usual in former *CoReMo* versions, by using maximal efficiency programming techniques for the the new task algorithms.

Furthermore, it was found that earlier *CoReMo* versions generated the XML detection files delimiting the *offset* and *length* of detections by bytes instead of UTF8 characters, which caused discrepancies between the detection and annotation used in the gold standard corpus, which negatively affected the evaluation, up to 10%. This new version allows detections annotation by either Byte (faster) or UTF8 modes.

2 Surrounding Context N-grams

One of the most important innovations in *CoReMo* as regards last year's version is that the documents are modeled by an extended 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], which in addition to the former, triple them by including 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 previous to all the previously explained for CTnG process.

For instance, modeling "The ***quick brown fox jumps*** over the ***lazy dog***" to *SC3G*:

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. brown fox jumps → brown_fox_jump (2^{nd} direct *CT3G* way)
5. brown fox lazy → brown_fox_laz (2^{nd} left-hand *SC3G* way)
6. brown jumps lazy → brown_jump_laz (2^{nd} right-hand *SC3G* way)
7. fox jumps lazy → fox_jump_laz (3^{th} direct *CT3G* way)
8. fox jumps dog → dog_fox_jump (3^{nd} left-hand *SC3G* way)
9. fox lazy dog → dog_laz_fox (3^{th} right-hand *SC3G* way)
10. jumps lazy dog → dog_jump_laz (4^{th} direct *CT3G* way)

The use of *SCnG* finally gets 3 times as many n-grams than only using *CTnG*, and it supposes more possibilities to tackle obfuscation cases with almost the same practical high precision in the process. The biggest number of terms obtained acts as a magnifier effect in the analysis. The memory requirements are obviously tripled and processing time almost doubled, but it improves dramatically the performance. Including these skip n-grams almost doesn't decrease the precision. An n-gram frequency study on PAN-PC-2009/2010 (table 1) / 2011 corpora [5] shows its exclusivity ratio almost unaltered.

Table 1. n-gram frequency study on PAN-PC-2010 only english source documents subcorpus

idf	n-grams quantity	ratio	n-grams quantity	ratio
	CT3N only		CT3N + SC3N	
--	105692331	1.0000	300970577	1.0000
01	97978896	0.9270	273382406	0.9083
02	5118576	0.0484	17527916	0.0582
03	1290009	0.0122	4809681	0.0160
04	517621	0.0049	2016842	0.0067
05	260442	0.0025	1042349	0.0035
06	148766	0.0014	609152	0.0020
...				
95	24	0.0000	115	0.0000
96	23	0.0000	129	0.0000
97	25	0.0000	97	0.0000
98	35	0.0000	105	0.0000
99	15	0.0000	106	0.0000
> 99	1010	0.0000	3666	0.0000

All n-grams are compared without a difference in the way they are created. The *SCnG* 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 higher precision 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*. In order to facilitate the *CTnGs* or *SCnGs 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

As by using the extended *SCnG* 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 *SCnG* 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 of 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 crosslingual 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 documents (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 in both suspicious and source sections, getting final XML detection documents. Both XML documents could be combined to create a best comparison readable HTML color document to emphasize direct detections within the final zones.

4 Crosslingual Detection

The use of external translation systems (as i.e. Google Translator[2]) is a drawback for low response timing, availability and economy goals. Because of that, *CoReMo* performs its own translations locally when it detects a non English language document. The crosslingual analysis is locally arranged after a direct mapping from every non English word (or its stem) to its translated English stemmed word, using two special dictionaries [3][7]: direct2stem (first chance) and stem2stem (second chance, when it first fails). If no logical translation is found, then the non-English word is replaced by its English stem.

For every new English n-gram, the original *offset* and *length* are registered from the non-English document to get an easy and precise source plagiarized sections location.

When using the crosslingual training subcorpus, the Plagdet score achieved by *CoReMo* was 0.70176: good results having in mind that they are not being biased by the same Google Translator process in both (obfuscation and detection) phases. The lower results obtained in the test phase (0.2577) are due to the fact that only human translated simulated cases were then used. It is in the expected line after last year's report [9]. The *CoReMo* mapped translation process is however in its childhood, and it is expected to be improved in newer versions by several modifications and using better crosslingual stem dictionaries versions.

5 Speed up Methodology

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

- C++ 64 bits programming
- 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 list, 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 frequency in the suspicious document for consecutive comparisons.

It enabled to achieve an average analysis time of 0.19 seconds per pair: 13.6 times faster than the second fastest algorithm, and 31 times faster than the winner one. However, for this version none optimization was arranged to take advantage of

2 http://translate.google.com

multicore features of current processors, but it's expected to be included in the next version.

6 Tuning Parameters and Evaluation

The best parameters settings were obtained by using the PAN-PC-2012 training corpus. The results of the training (plagdet 0.6754) are displayed and compared to the ones achieved in the phase of competition (0,6252) in table 2. For both cases, these parameters were:

- chunk length: 4 n-grams (internally changes to 12 itself when using SCnGs).
- Cross-lingual chunk length: 47 n-grams (also 3 times bigger when using *SCnGs*).
- minimum monotony: 2 chunks (same for monolingual and crosslingual modes).

7 Conclusions and Future Work

Nowadays *CoReMo* is the fastest detector, but it should be optimized to take the opportunity of multi-core systems advantage.

The translated subcorpus analysis achieved better results than last year's (comparing human translation only) due to the newest n-gram modeling, but it is still using the same old dictionaries, and only about 50% of the words are translated. Larger and better dictionaries [7] would benefit this local technique. Other local translation methods could be explored.

Mixing this n-gram modeling with other NLP resources (WordNet synsets, odd/even skip n-grams, ...) could improve detections when hard obfuscation conditions happen.

The detailed comparison method got better Plagdet performance for the same corpus than the former method used in *CoReMo*. This suggests a new change for the traditional full process for local source collections.

The comparison of the Plagdet progress regarding the PAN2011 must be done with caution, since not being necessary prior source document detection, by using a LEAP[3] detector, could directly get reasonable results, as shown in [8] for the former Intrinsic Detection task.

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.

3 LEAP = Labeling Everything As Plagiarized

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]

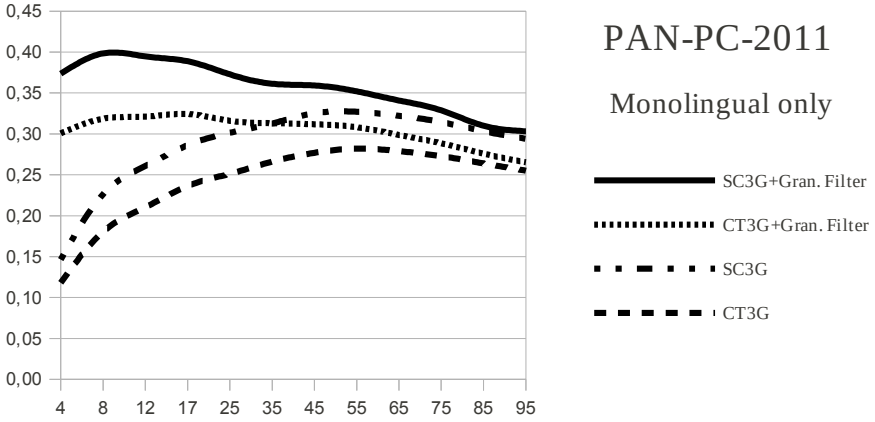

Fig. 2. Plagdet/chunk_length comparative of CoReMo 1.6 using CT3N or SC3N w/wo Granularity Filter on PAN-PC-2011 non-English subcorpus [4]

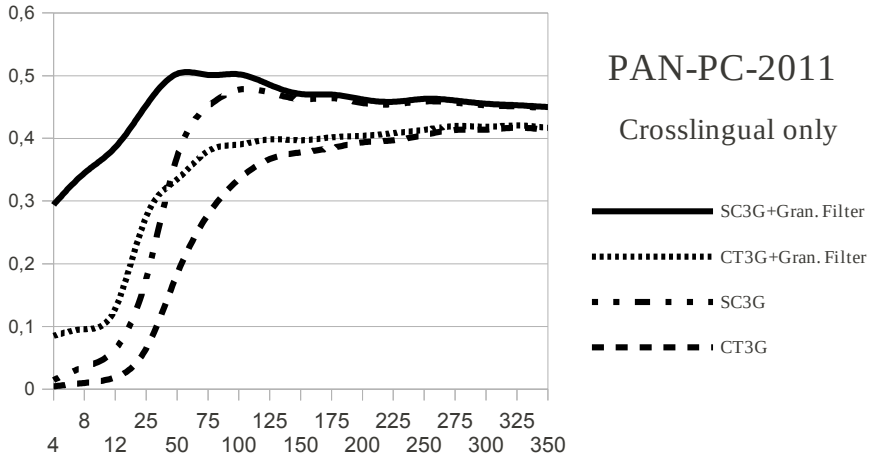

Table 2. ***CoReMo*** **1.9UTF achieved scores in training and competition phases with same tuned parameters: chunklength = 4 and monotony = 2**

	PAN-PC-2012 Training Corpus				**PAN-PC-2012 Competition Corpus**				
	Plagdet	***Recall***	***Precision***	***Granularity***	***Plagdet***	***Recall***	***Precision***	***Granularity***	***Avg. time (s)***
No plagiarism	1.0	1.0	1.0	1.0	0	0	0	1.0	0.16112
No obfuscation	0.87187735	0.82362082	0.92744216	1.0009165	0.93247701	0.96882651	0.89875647	1.0	0.15786
Artificial Low	0.77468919	0.73718058	0.81621938	1.0	0.71393468	0.59334680	0.89603990	1.0	0.15278
Artificial High	0.33936519	0.21152910	0.85772527	1.0	0.12760741	0.06961406	0.76443402	1.0	0.14492
Translation	0.70176951	0.59298909	0.85942631	1.0	0.25774484	0.14825321	0.98580851	1.0	0.09754
Human simulated	0.68898017	0.53438117	0.96944527	1.0	0.67366948	0.57757699	0.81161164	1.0024937	0.43346
Real cases	---	---	---	---	0.74061972	0.66846300	0.83023925	1.0	0.08212
Global	**0.67536855**	0.55155054	0.87106899	1.00012215	**0.62520246**	0.50042086	0.83442275	1.0009596	**0.19009**

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