Identifying free text plagiarism based on semantic similarity

George Tsatsaronis
Norwegian University of Science and Technology
Department of Computer and Information Science
Trondheim, Norway
gbt@idi.ntnu.no

Iraklis Varlamis
Harokopio University of Athens,
Department of Informatics and Telematics
Athens, Greece
varlamis@hua.gr

Andreas Giannakoulopoulos
Ionian University,
Department of Audio and Visual Arts
Corfu, Greece
agiannak@ionio.gr

Nikolaos Kanellopoulos
Ionian University,
Department of Audio and Visual Arts
Corfu, Greece
kane@ionio.gr

Contents

- Plagiarism types
- Anti-Plagiarism Methods and Tools
- Text plagiarism detection process
- Our approach
  - Semantic Relatedness for text
  - Threshold for plagiarism detection
- Experiments
- Results

Classification of Plagiarism

- Domain based
  - Academia
  - Journalism
  - Online
- Content type based
  - Text plagiarism
  - Code/program plagiarism
  - Image plagiarism
- Degree based
  - direct plagiarism: copy & paste
  - mosaic plagiarism: word switch
  - paraphrase plagiarism: summarizing and paraphrasing
  - plagiarism of ideas
  - insufficient acknowledgment

Linda N. Edwards, Matthew G. Schoengood. 2005. Avoiding and Detecting Plagiarism - A Guide for Graduate Students and Faculty *With Examples.* The Graduate School and University Center. The City University of New York: http://web.gc.cuny.edu/provost/pdf/AvoidingPlagiarism.pdf

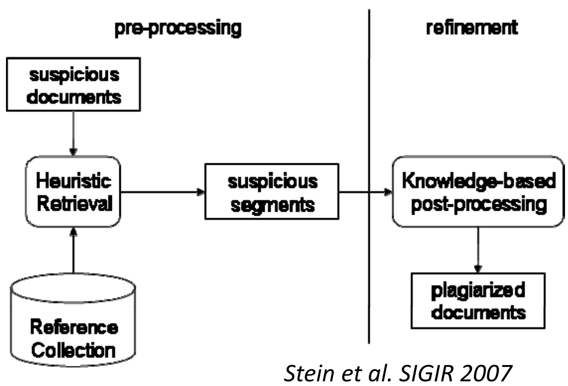
Available solutions

- Anti-Plagiarism Methods and Tools
  - Prevention:
    - Watermarking, Copy protection
    - Tools: iThenticate
  - Detection:
    - Document source comparison
    - Free online tools: Google, Duplichecker, Copyscape
    - Code: Jplag (Univ.Karlsruhe), MOSS (Stanford)
    - Commercial: Turnitin, Plagiarism Detector

Working with text

- Text similarity: easily detects direct plagiarism or mosaic plagiarism
  - Similarity (d1,d2) = f(common words/stems between d1,d2)
  - The ***more important*** words (weights are based on TF or TF/IDF) contribute more to the similarity score
- Text relatedness: better in detecting paraphrasing and plagiarism of ideas
  - Keywords carry several meanings (senses), texts are bags-of-meanings
  - The type of relation (hypernyms, hyponyms, meronyms etc.) between meanings affects the relatedness between words, e.g. cat-feline (synonym), cat-dog (siblings)
  - Slower than text similarity methods

The methodology

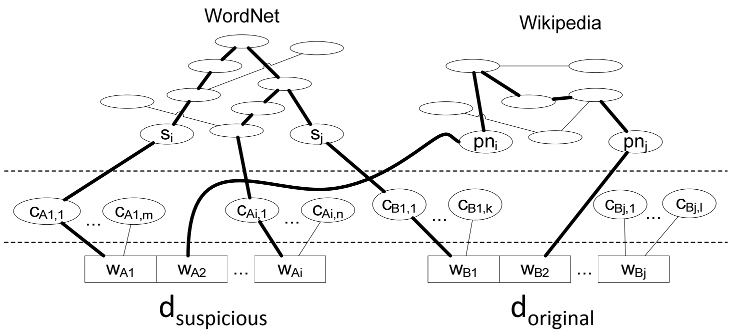


Stein et al. SIGIR 2007

- The suspicious text is compared against several reference texts.
- Texts are considered as bags-of-words or word-chains. Each text has a ***fingerprint***. Fingerprints are compared and a resemblance score is generated for each text.
- Comparison can be applied in document or text segment level.
- Suspicious documents that surpass a resemblance threshold are potentially plagiarized.

Semantic Relatedness and Omiotis

- Each text (suspicious or original) is converted to a word vector (lemmatization, stop-word removal, tf/idf weighting)
- The semantic relatedness between each pair ($d_{suspicious}$, $d_{original}$) is measured using Omiotis



$$Omiotis(A,B) = \frac{[\zeta(A,B) + \zeta(B,A)]}{2}$$

where $\zeta(A,B)=f(SR(w_{Ai}, w_{Bj}))$
and $SR(w_{Ai}, w_{Bj})=g(\text{path connecting } w_{Ai}\ \&\ w_{Bj}$

Relatedness score & plagiarism

- Problem
  - Given that we know a relatedness score between a suspicious item A and its potential source B, how do we decide that A is a plagiarism?
- Solution
  - Use a threshold
- Question
  - How do we define the relatedness threshold
- Solutions

Setting the threshold

- Define a cut-off threshold for a set of N candidate pairs
- Unsupervised methods
  - Use the ***mean*** relatedness value or order values and use the ***median*** value
  - Combine rankings: Produce k different rankings of the pairs using k different measures of relatedness. Combine rankings using a random weight assignment for each method
  - Iteratively adjust the 3 weights and re-aggregate the different rankings until the aggregated ranking is stabilized *(Klementiev, ECML 2007)*
- Supervised methods
  - Anti-plagiarism detection is addressed as a classification problem
  - Use predefined cases of plagiarism and no-plagiarism to train the classification algorithm (to decide on the best threshold value)
  - Evaluate using unclassified cases

Experiments

- Dataset
  - PAN Plagiarism Corpus: 1st International Competition on Plagiarism Detection 2009
  - Synthetic dataset comprising 20612 source and 20611 suspicious documents
- We employed
  - 11.000 text segment pairs (in English) annotated as plagiarism or non plagiarism cases in the competition results
    - 3400 pairs with high obfuscation, which are difficult to detect, 3400 pairs with low obfuscation and 4200 pairs with no obfuscation at all
  - another 11.000 text segment pairs, as negative (plagiarism) cases, since they are selected randomly from the same documents.

Metrics employed

- Three similarity values for each pair
  - cosine measure for textual similarity, using the TF-IDF weighting scheme for terms and the vector space representation (*Cosine*)
  - Omiotis and conceptual similarity using WordNet only as a knowledge base (*Omi*)
  - Omiotis and conceptual similarity using WordNet and Wikipedia as knowledge bases (*OmiWiki*)
- Results in the whole dataset and in the three obfuscation groups (*none*, *low*, *high*)
- Evaluation metrics: Precision (P), Recall (R), F-measure (F1)

Results

- Unsupervised methods
  - Mean similarity as cut-off

	All			None			Low			High		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Cosine	0.99	0.82	0.90	0.98	0.75	0.85	0.97	0.94	0.95	0.97	0.94	0.95
Omi	**0.99**	**0.85**	**0.92**	0.96	0.77	0.86	0.96	0.94	0.95	0.94	0.95	0.94
OmiWiki	0.99	0.84	0.91	**0.98**	**0.76**	**0.87**	**0.98**	**0.94**	**0.96**	**0.97**	**0.95**	**0.96**

  - Median similarity as cut-off

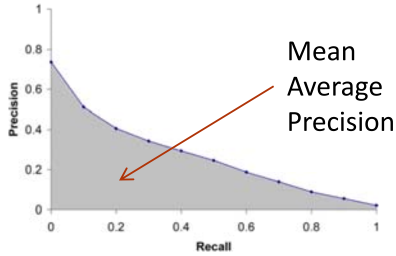
	All			None			Low			High		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Cosine	0.92	0.89	0.90	0.29	1	0.45	0.24	1	0.39	0.25	1	0.40
Omi	**0.93**	**0.89**	**0.91**	0.49	0.84	0.62	**0.48**	**0.98**	**0.65**	**0.48**	**0.98**	**0.65**
OmiWiki	**0.93**	**0.89**	**0.91**	**0.51**	**0.85**	**0.63**	0.48	0.97	0.64	0.48	0.97	0.64

- The distribution of values is right-skewed: many small values and few large values for all measures
- Mean is a better cut-off value in the unsupervised case

Results

- Unsupervised methods
  - Evaluate individual rankings using each method
  - This is an IR problem, so Mean Average Precision at the 11 standard recall points was used

	Cosine	Omiotis	OmiWiki
ALL	**0,948216**	0,94637	0,946629
None	**0,875089**	0,852222	0,853184
Low	0,930807	**0,931353**	**0,931341**
High	0,929725	**0,93113**	**0,930869**



- Omiotis and OmiWiki provide a better ranking in the cases where there is some type of obfuscation, either low, or high.
- Simple keyword similarity measures (e.g. cosine) cannot detect paraphrase plagiarism (keywords replaced by synonyms)

Results

- Unsupervised methods
  - Aggregate individual rankings produced by each method
  - Evaluate results when mean is used as a cut-off

	All			None			Low			High		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Aggregation	0.999	0.69	0.81	0.988	0.76	0.86	0.99	0.94	0.97	0.98	0.95	0.97

- Aggregating the values of the three measures may not improve the performance in the non obfuscated cases, but improves the performance in all other cases.

Results

- Supervised methods
  - 4 feature sets: {cosine}, {omi}, {omiwiki}, {cosine,omi,omiwiki}
  - 10-fold cross validation for the evaluation of performance
  - Two classification algorithms
    - logistic regression

	All			None			Low			High		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Cosine	0.987	0.88	0.93	0.983	0.758	0.855	0.988	0.935	0.96	0.963	0.934	0.948
Omi	0.988	0.878	0.929	0.99	0.758	0.858	0.991	0.934	0.961	0.988	0.936	**0.961**
OmiWiki	0.992	0.878	0.931	0.991	0.759	0.859	0.993	0.934	**0.962**	0.989	0.934	0.96
All Features	0.989	0.879	0.93	0.987	0.758	0.857	0.992	0.935	**0.962**	0.989	0.938	**0.962**

    - support vector machines

	All			None			Low			High		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Cosine	0.992	0.871	0.932	0.992	0.745	0.851	0.99	0.932	0.96	0.984	0.932	0.957
Omi	0.995	0.871	0.933	0.996	0.744	0.852	0.994	0.929	0.96	0.991	0.931	0.96
OmiWiki	0.996	0.87	0.933	0.997	0.744	0.852	0.995	0.928	0.96	0.992	0.932	0.961
All Features	0.995	0.873	**0.934**	0.995	0.775	**0.871**	0.993	0.933	**0.962**	0.991	0.937	**0.963**

Conclusions

- Omiotis using WordNet and Wikipedia resources showed improved performance against baseline statistical methods (stemming, tf/idf weighting and cosine), either supervised or unsupervised approaches are employed for determining the appropriate similarity thresholds.
- The use of semantics increases complexity but is necessary to decide on ambiguous plagiarism cases.
- Preprocessing is important: Using only the textual information and occurrence statistics is the first step in detecting plagiarism suspects.
- Traditional matching techniques can be used to locate suspect fragments in the first step and our semantic method can be subsequently applied to refine results at sentence level.
- Next step: Plug our semantic-based plagiarism detection module in an open source plagiarism detection software
- A demo of Omiotis is available at: http://omiotis.hua.gr

Thank you!

Questions?

varlamis@hua.gr
http://www.dit.hua.gr/~varlamis/