Comparing Medline citations using modified N-grams

Rao Muhammad Adeel Nawab,[1] Mark Stevenson,[2] Paul Clough[3]

[1]Department of Computer Science, COMSATS Institute of Information Technology, Lahore, Pakistan
[2]Department of Computer Science, University of Sheffield, Sheffield, UK
[3]Information School, University of Sheffield, Sheffield, UK

Correspondence to
Dr Mark Stevenson, Department of Computer Science, University of Sheffield, Regent Court, 211 Portobello, Sheffield S1 4DP, UK; m.stevenson@dcs.shef.ac.uk

Received 5 December 2012
Revised 24 April 2013
Accepted 4 May 2013
Published Online First 28 May 2013

ABSTRACT

Objective We aim to identify duplicate pairs of Medline citations, particularly when the documents are not identical but contain similar information.

Materials and methods Duplicate pairs of citations are identified by comparing word n-grams in pairs of documents. N-grams are modified using two approaches which take account of the fact that the document may have been altered. These are: (1) deletion, an item in the n-gram is removed; and (2) substitution, an item in the n-gram is substituted with a similar term obtained from the Unified Medical Language System Metathesaurus. N-grams are also weighted using a score derived from a language model. Evaluation is carried out using a set of 520 Medline citation pairs, including a set of 260 manually verified duplicate pairs obtained from the Deja Vu database.

Results The approach accurately detects duplicate Medline document pairs with an F_1 measure score of 0.99. Allowing for word deletions and substitution improves performance. The best results are obtained by combining scores for n-grams of length 1–5 words.

Discussion Results show that the detection of duplicate Medline citations can be improved by modifying n-grams and that high performance can also be obtained using only unigrams (F_1=0.959), particularly when allowing for substitutions of alternative phrases.

INTRODUCTION

This paper deals with the problem of identifying duplicate citations from Medline, particularly when the documents contain the same (or highly similar) information but are not identical. This situation occurs when a document has been altered (eg, by paraphrasing) to disguise the fact that it is a duplicate. Identifying duplicate (and near-duplicate) documents has been widely researched within a variety of domains.[1–3]

There are a number of reasons why duplicate citations may be found in Medline. Some documents in Medline are corrections or errata to previous publications and differ from the original document only in minor (but often very important) ways. Duplicate citations may also be added to increase the dissemination of important documents, such as the publication of clinical practice guidelines or policy statements in multiple journals. These duplicate documents are introduced to the database for good reason and add to the scientific knowledge it contains. However, duplicate documents are added to Medline for the less virtuous goal of increasing an author's publications by reporting the same work multiple times in different venues, or even reproducing another piece of work without acknowledgement (ie, plagiarism).

The sheer number of citations in Medline and the rate at which they are being added makes it difficult for any individual or group to have an overview of the information it contains. Consequently, it is possible to report work that reproduces research that has already been reported in another community without the connection being noticed. Previous studies have identified duplicate publications in the literature associated with several areas of medicine, including bone and joint surgery,[4] head and neck surgery,[5] and plastic surgery.[6]

BACKGROUND AND SIGNIFICANCE

An n-gram is a string of n adjacent words that occur within a text. Comparing n-grams has proven to be an effective method for detecting duplicate or similar documents that has been applied to a range of problems, including the identification of text reuse in journalism[7 8] and plagiarism detection.[9–12]

Errami *et al*[13] describe a study on duplicate citations in Medline. A set of over 62 000 abstracts was examined and the eTBlast tool[14] used to identify the most similar in Medline. They found that a small portion of the abstracts (1.39%) were highly similar. The majority of these (1.35%) had shared authors and were similar enough to be considered as duplicate publications, but the remainder (0.04%) did not have a shared author and are potential cases of plagiarism. The duplicate documents identified in this study are only a small portion of the documents examined but, given the size of Medline, suggest that as many as 3500 citations are potentially plagiarized and 117 500 are duplicate publications. These figures were reported in 2007 and are likely to be higher now.

Errami *et al*[15] reported an improvement on the techniques used by eTBlast. Their approach was based on determining the number of 'statistically improbable phrases' (SIP), essentially word 6-g, shared by a pair of documents. The count of SIPs found in both documents is divided by the number of SIPs in the shorter of the two documents, a widely used technique in document comparison.[16] However, words and phrases do not occur in documents with the same probabilities, making some SIPs more likely to be found in two citations than others. Errami *et al*[15] account for this by assigning scores to each SIP and using them when computing the similarity between citations. The scores are based on the probability of the SIP occurring in Medline. However, data sparsity makes it difficult to obtain accurate probability estimates for 6-g even in a corpus the size of Medline and instead a language model is used. The final score for each SIP is computed as $-\log(p)$ where p is the probability the language model assigned to the SIP.[17]

To cite: Nawab RMA, Stevenson M, Clough P. *J Am Med Inform Assoc* 2014;**21**:105–110.

Errami *et al*[15] reported that the SIP approach was able to identify duplicate publications with a precision of 84% and recall of 78.9%, outperforming eTBlast which achieved a precision of 87.8% and recall of 50.3% on the same dataset.

A limitation of both eTBlast and using SIPs is that they are unable to identify duplicate citations when the original text has been highly modified, such as by paraphrasing or replacing words with synonyms.[13 15] The authors acknowledge the need for text comparison techniques that can identify 'smart duplication',[13] as well as to 'analyse grammar and extract meaning from sentences rather than rely on word comparisons only'.[15] Previous approaches have attempted to take account of alterations to documents for duplicate detection.[8 18 19] However, these have not been applied to Medline citations. The approach most similar to the one presented here[7] was used to identify text reuse in journalism, an area in which creating documents by modifying another is standard practice.

The problem of identifying duplicate documents in a large collection such as Medline is often treated as a two-stage process.[20–22] The first stage, referred to as *Candidate Document Selection*, involves taking a document and identifying a set of potential duplicates from the collection. This is normally achieved using techniques from Information Retrieval, which are efficient enough to identify similar documents quickly, but not accurate enough to determine whether they are actual duplicates.[12 23] This is followed by a second stage, called *Detailed Analysis*, which compares the original document with each of the ones returned by the candidate document selection stage to determine which of them (if any) is in fact a duplicate. Using this two-stage approach has been shown to improve the speed and efficiency of plagiarism detection systems.[21] The focus of this paper is on the second stage of the duplicate detection process (ie, Detailed Analysis). We present an approach to this problem which compares pairs of Medline citations and takes account of the fact that they may not be exact copies. We have previously shown how candidate documents can be selected from a large collection such as Medline,[22 24] but our approach could be combined with any method for Candidate Document Selection.

MATERIALS AND METHODS

Document comparison

Our approach assumes that duplicate citations are likely to have more n-grams in common than non-duplicates. A number of measures for comparing n-grams in a pair of documents have been proposed, all of which are based around counting the number of n-grams that are common to a pair of documents and normalizing by some factor that takes account of the number of n-grams in one or both of the documents.[16] For these experiments the overlap between a pair of documents, A and B, is computed using the containment measure:

$$\text{score}_n(A, B) = \frac{\sum_{\text{ngram} \in B} \text{count}(\text{ngram}, A)}{\sum_{\text{ngram} \in B} \text{count}(\text{ngram}, B)} \quad (1)$$

where *count* (*ngram*, A) is the number of times *ngram* appears in A.

The containment measure has previously been found to be a useful measure for document comparison which has been applied to detecting near duplicate web pages,[1] identifying text reuse in journalism,[7] and plagiarism detection in student assignments.[25] It is useful for identifying when one document (B) is contained within, or the same as, another document which could be longer (A). A score of 1 indicates that document B is contained within A, since all of the n-grams in B are found in A, while a score of 0 suggests there is no connection between the documents.

In the normal application of the containment measure for duplicate detection, A is considered to be the document that is the duplicate of another (or contains the duplicate), while B is treated as the original document. We also adopt this approach and treat one citation as the potential duplicate of another. However, there are not major differences between the lengths of Medline citations (at least compared to other situations in which the containment measure has been applied) and the choice of which citation in a pair to treat as the original and which to treat as the duplicate is unlikely to have a major effect.

We cast the problem of duplicate detection as a supervised classification task in which the aim is to distinguish between two classes: duplicate and non-duplicate. The similarity scores between pairs of citations generated using the containment measure are used as features. We use the Weka[26] V.3.6.1 implementation of the C4.5 decision tree algorithm, J48, to classify citation pairs. (We tested various machine learning algorithms within Weka and J48 gave the highest results.) For all experiments, 10-fold cross-validation with randomized folds is carried out and repeated 10 times. Results reported are the average across the 10 runs of 10-fold cross-validation.

Citations are preprocessed using MetaMap,[27] which tokenizes the text and identifies possible phrases. The phrases identified by MetaMap are treated as single tokens during the creation of n-grams since we found that doing so was beneficial for duplicate detection and is also convenient for the techniques that are applied to modify n-grams.

MODIFIED N-GRAMS

A limitation of using n-gram overlap to determine document similarity is that it does not perform well when the original document has been modified (eg, by paraphrasing). In fact, techniques based on n-gram overlap can be fooled using very simple changes to a document. Insertion, deletion, or substitution of even a single token in a text results in the mismatch of at least one n-gram.[28] If every nth word in a text is altered in some way, then the two documents would have no n-grams of length n in common and metrics such as the containment measure would fail to identify the similarity between them.

To avoid this problem we make use of modified n-grams. These are n-grams which are derived from those found in one of the documents and are intended to reflect the changes that might occur if a document was altered in order to disguise the fact that it is a duplicate. Two methods for modifying the n-grams were explored: substitution and deletion.

Substitution

The first type of modified n-grams is created by substituting one of the words in the n-gram with one of its synonyms from the Unified Medical Language System (UMLS) Metathesaurus.[29] The citation is first run through MetaMap[27] and each term mapped onto a set of potential Concept Unique Identifiers (CUIs). The MRCONSO table in the UMLS Metathesaurus lists various ways in which each CUI is described in the various vocabularies that are used to create the UMLS Metathesaurus. This table is used to generate a set of alternative terms that can be substituted for each term in the n-gram. A set of modified n-grams is then created by choosing one of the terms in the

n-gram and substituting it with one of the alternative terms from the table.

Deletion

A second type of modified n-grams is generated by simply deleting words from the n-gram. If $w_1, w_2, \ldots, w_n$ is an n-gram, then n−2 modified n-grams can be created by removing one of $w_2, w_3 \ldots w_{n-1}$. Modified n-grams are not created by removing the first or last word since doing so would simply duplicate existing n-grams of order n−1. Unigrams only consist of a single word and are too short for deleted n-grams to be created from them.

Examples

Consider the phrase '*Contribution of the TGFB1_Gene to Myocardial_Infarction susceptibility*', which is the title of the Medline citation with PMID 22872813. Preprocessing this phrase with MetaMap identifies the terms '*TGFB1 Gene*' and '*Myocardial Infarction*' which are each treated as a single token.Box 1 shows some of the modified n-grams that are created for one of the 4-g contained within this phrase: tgfb1_gene to myocardial_infarction susceptibility. The top part of the box shows four modified n-grams which are created using the substitution approach. The first two of these are created by substituting the term tgfb1_gene with synonyms found in the UMLS Metathesaurus. MetaMap maps the terms 'TGFB1 Gene' onto the CUI C1366557 and the MRCONSO table in the UMLS Metathesaurus lists *transforming_growth_factor_1* and *tgfb1* as alternative terms associated with that CUI. The second pair is created by substituting *myocardial_infarction* with synonyms in the same way.

The lower part of the box contains examples of modified 4-g generated using the deletion approach. These are created by deleting terms from the two 5-g that contain that 4-g. (In general modified versions of an n-gram are created using (n+1)-grams and deleting terms from them.) Note that some of the modified n-grams are ungrammatical (eg, 'of the to myocardial_infarction'). This is not a problem for the approach since these n-grams are very unlikely to be seen in Medline citations, so they do not contribute to the duplicate detection score.

COMPARING MODIFIED N-GRAMS

To detect duplicate citations, modified n-grams are created for the document that is suspected of being a duplicate (A in equation (1)). Comparison between the documents is then carried out by determining the proportion of n-grams in B which also occur as n-grams in A or as modified n-grams generated from A.

For each n-gram in A, the set of possible modified n-grams is created, denoted as *mod* (*ngram*). The original n-gram *ngram* is also included in *mod* (*ngram*). The modified count for the number of occurrences of an n-gram in A, *mod_count* (*ngram*, A), is then computed as the number of times it appears in *mod* (*ngrams*), that is,

$$\text{mod\_count}(\text{ngram}, A) = \sum_{\text{ngram}' \in \text{mod}(\text{ngram})} \text{count}(\text{ngram}', A) \quad (2)$$

The deletion and substitution approaches generate large numbers of modified n-grams. This means that the number of shared n-grams can exceed the total number of n-grams in B, leading to a score greater than 1. To avoid this, the overlap counts are bounded by the number of times that n-gram appears in B. Consequently, the text reuse detection score, score_n (A, B), is computed as:

$$\text{score}_n(A,B) = \frac{\sum_{\text{ngram} \in B} \min(\text{mod\_count}(\text{ngram}, A), \text{count}(\text{ngram}, B))}{\sum_{\text{ngram} \in B} \text{count}(\text{ngram}, B)} \quad (3)$$

The way in which n-gram overlap counts are bounded is similar to the approaches used by the BLEU[30] and ROUGE[31] systems for automatic evaluation of Machine Translation and summarization output which also compare documents using n-gram overlap.

Weighting n-grams

N-grams do not occur with equal frequency in Medline: the fact that a pair of citations has a rare n-gram in common is much stronger evidence that they are duplicates than if they shared an n-gram that occurs in many citations. Previous methods for detection of duplicate citations in Medline weighted n-grams based on their frequency,[15] with more importance being assigned to rarer n-grams (see *Background and significance*). We employ a similar approach.

A bi-gram language model was created using the SRILM language modeling toolkit[32] with Good-Turing smoothing.[33] The model was trained using a corpus of 344 000 citations randomly selected from the 2011 Medline/PubMed Baseline Repository. The language model is used to estimate the probability of each n-gram in the comparison of citation pairs. In order to increase the significance of rare n-grams, the Information Content value of each n-gram is computed as −log(p), where p is the probability assigned to that n-gram by the language model. When the language-model is used to weight n-grams, the Information Content values are used in equation (3) rather than simple counts.

Evaluation

Evaluation is carried out using the Deja Vu database (http://dejavu.vbi.vt.edu/dejavu/) of highly similar citations in Medline. The pairs of citations in Deja Vu were identified using an automatic text comparison tool, eTBlast.[14 34] A small subset of these pairs have been manually examined and verified as true duplicates. A set of 260 of these with different authors were randomly selected. Pairs with different authors were used since

Box 1 Example modified n-grams

- *Original 4-g*
- the tgfb1_gene to myocardial_infarction

Substitutions

- the transforming_growth_factor_β_1 to myocardial_infarction
- the tgfb1 to myocardial_infarction
- the tgfb1_gene to heart_attack
- the tgfb1_gene to coronary_attack

Deletions

- of tgfb1_gene to myocardial_infarction
- of the to myocardial_infarction
- the to myocardial_infarction susceptibility
- the tgfb1_gene myocardial_infarction susceptibility

they cannot be identified by matching author names, making it more difficult to detect duplicates. We also created an equal number of citation pairs that are not similar by choosing pairs of Medline citations at random since the probability of two randomly selected citations being duplicates is almost 0.[13] In total, the dataset consists of 520 citation pairs (half duplicates).

Performance is computed using precision, recall, and F_1-measure. Precision is the proportion of pairs that are identified as duplicates that actually are, while recall is the proportion of duplicate pairs which are identified as being so. F_1-measure is the harmonic mean of these two measures. Precision (P), recall (R), and F_1-measure (F_1) are computed as follows:

$$P = \frac{TP}{TP + FP} \quad R = \frac{TP}{TP + FN} \quad F_1 - measure = \frac{2 \times P \times R}{P + R} \qquad (4)$$

where TP are true positives (ie, citation pairs identified as being duplicates which are in fact true duplicates), FP are false positives (pairs identified as being duplicates that are not duplicates), and FN are false negatives (pairs identified as not being duplicates which are really duplicates). Figures for all three measures are macro-averaged across both classes (duplicate and non-duplicate).

eTBlast

To provide comparison with an existing approach we also report results for eTBlast[14 34] on the same dataset. Each pair of citations is supplied to the online version (http://etest.vbi.vt.edu/etblast3/index/compare) and the similarity score it generates used as a feature for the learning algorithm.

eTBlast is designed to carry out both stages of the duplicate detection problem (see *Background and significance*), while the approach presented in this paper deals with only the Detailed Analysis stage. Although the evaluation described here concerns the Detailed Analysis stage, it provides comparison with a state-of-the-art approach to duplicate detection in Medline.

RESULTS AND DISCUSSION

Table 1 shows results of the evaluation using various lengths of n-grams and their combination. 'NG' is the basic containment measure which does not make use of modified n-grams. 'Del', 'Sub', and 'Del+Sub' indicate that modified n-grams are included, generated using deletion, substitution, or by combining both approaches. (Note that unigrams are too short for modified n-grams to be generated using the Del approach and consequently there are no figures reported in the table.) The prefix 'LM-' indicates that the n-grams are weighted using the language model.

In the table, 'Unigrams', 'Bigrams', 'Trigrams' etc. list the performance using n-grams of a single length. In these cases the learning algorithm is provided with a single feature, the score generated for that length of n-gram. 'Combined' lists performance when the scores are generated using all five lengths of n-gram and in this case the learning algorithm is provided with five features, the scores obtained for each length of n-gram. However, unigram features are not available for approaches that make use of the Del modification and in these cases the feature is obtained from the undeleted version (ie, the unigram features for 'Del', 'LM-Del', 'Del+Sub', and 'LM-Del+Sub' are obtained from 'NG', 'LM-NG', 'Sub', and 'LM-Sub', respectively).

The best overall performance (F_1 score of 0.99) is obtained when n-grams of different lengths are combined and modified n-grams created using the 'Sub' approach included. This compares well against the performance obtained using eTBlast under the same conditions (F_1 score of 0.84), which demonstrates the usefulness of combining together information derived from different lengths of n-grams and their modifications.

Improvements are often observed when modified n-grams are used. Including n-grams generated using the 'Sub' approach consistently leads to improvements in performance, many of which are significant. However, including n-grams generated using the 'Del' approach does not always improve performance and is actually harmful in some cases, for example, using trigrams. The biggest improvement in performance is observed when both types of modified n-grams are used. (This is not the case when different lengths of n-grams are combined, however the results obtained using 'Sub' and 'Del+Sub' are extremely close.) The success of these approaches, particularly 'Sub', shows that duplicate citations are not always verbatim copies and it is important to take account of this when comparing pairs of documents. The 'Sub' approach effectively replaces words and phrases with synonyms, a convenient way of quickly changing documents when they are duplicated. The success of this approach shows that the citations have been altered this way when duplicated.

The best result for single length n-grams is obtained with unigrams. Using n-grams generated using the 'Sub' approach weighted with the language model achieves an F_1 score of 0.96, which is a significant improvement on the performance when modified n-grams are not included. The strong performance of unigrams is likely to have a significant influence on the success achieved when the n-grams of different lengths are combined.

Longer n-grams do not perform as well as unigrams and do not benefit as much from the addition of n-grams generated using 'Sub'. The difference between the performance of unmodified unigrams and longer n-grams is relatively small.

Table 1 Performance on duplicate citation detection using various lengths of n-grams and their combination

	Unigrams			Bigrams			Trigrams		
	p	R	F1	p	R	F1	p	R	F1
NG	0.85	0.84	0.84	0.83	0.81	0.81	0.84	0.82	0.82
LM-NG	0.84	0.83	0.83	0.84	0.83	0.82	0.85	0.84	0.84
Del	–	–	–	0.83	0.81	0.81	0.83	0.81	0.81
LM-Del	–	–	–	0.84	0.82	0.82	0.84	0.83	0.83
Sub	0.96*	0.96*	0.96*	0.85	0.84	0.83	0.86	0.84	0.84
LM-Sub	0.96*	0.96*	0.96*	0.86*	0.86*	0.86*	0.86	0.84	0.84
Del+Sub	–	–	–	0.86*	0.85*	0.85*	0.87*	0.85*	0.85*
LM+Del+Sub	–	–	–	0.87*	0.87*	0.87*	0.86*	0.85*	0.85*
	Fourgrams			Fivegrams			Combined		
	p	R	F_1	p	R	F_1	p	R	F_1
NG	0.83	0.82	0.82	0.84	0.83	0.83	0.92	0.91	0.91
LM-NG	0.84	0.83	0.82	0.84	0.84	0.84	0.93	0.93	0.93
Del	0.84	0.83	0.83	0.85	0.84	0.84	0.88	0.87	0.87
LM-Del	0.85*	0.84*	0.84*	0.85*	0.85*	0.85*	0.94	0.94	0.94
Sub	0.84	0.83	0.83	0.85	0.85	0.85	0.99*	0.99*	0.99*
LM-Sub	0.86*	0.86*	0.86*	0.83	0.83	0.83	0.99*	0.99*	0.99*
Del+Sub	0.86*	0.85*	0.85*	0.86*	0.86*	0.86*	0.99*	0.99*	0.99*
LM–Del+Sub	0.88*	0.87*	0.87*	0.85	0.84	0.84	0.99*	0.99*	0.99*
	eTBlast			0.87	0.84	0.84			

*Statistically significant improvement over the baseline approach (ie, NG) (Wilcoxon signed-rank test, $p<0.05$).
NG is the basic containment measure which does not make use of modified n-grams. Del, Sub, and Del+Sub indicate that modified n-grams are included, generated using deletion, substitution, or by combining both approaches. LM indicates that the n-grams are weighted using the language model.

However, performance of unigrams when n-grams generated using 'Sub' are included is noticeably higher than the results for any other single length n-gram, with or without the inclusion of modified n-grams. Previous work[15] used 6-g for duplicate citation detection (see *Background and significance*). The results reported here cannot be compared directly with that approach due to the different experimental settings. However, they do show that there is a benefit to using shorter sequences of text that take less account of the structure of the text. In fact, comparing documents using unigrams is effectively a bag of words approach that ignores the order in which the terms appear.

The effect of weighting n-grams using the language model is inconsistent. Performance varies according to the length of n-grams and inclusion or otherwise of modified n-grams. Previous work on the detection of duplicate Medline citations[15] also weighted n-grams but did not report performance figures for the un-weighted case. These results suggest that there is little to be gained from weighting n-grams.

Although results vary across the different approaches, the precision and recall figures are roughly equal for each individual approach. The learning algorithm that was used to determine whether pairs of documents were duplicates or not was set to optimize F_1-measure. It would be possible to alter this and, for example, create a classifier that was biased towards identifying all possible instances of duplicates at the expense of returning false positives (ie, high recall and lower precision).

Results of these experiments show that comparison of n-grams is a useful approach for the Detailed Analysis stage of duplicate detection and that modifying n-grams improves performance. However, this approach is not suitable for the Candidate Document Selection stage due to the number of pairwise comparisons that would be required. It is also likely that performance would not be as high as reported here since the distribution of duplicate and non-duplicate pairs would be different in this scenario.

CONCLUSION

Duplicate citations can appear in Medline for a number of reasons, including multiple publication of the same work and plagiarism of work carried out by other researchers. This paper reported an approach to the detection of duplicate Medline citations designed to identify cases where the original citation has been altered. The approach is based on the comparison of n-grams in the citations. In order to identify non-identical copies, the n-grams are modified by deleting terms or substituting with synonyms from the UMLS Metathesaurus. Evaluation was carried out using an existing database of duplicate Medline citations and the proposed approach compared well to an existing method that carries out both stages of duplicate detection.

The results presented here show that duplicate Medline citation can be accurately detected by comparing n-grams. Modifying n-grams to take account of the fact the duplicates may not be verbatim copies improves performance. The best results were obtained by combining information from n-grams of different lengths. Unigrams were found to perform particularly well.

Previous studies on duplicate detection in Medline highlighted the need to identify 'smart duplication',[15] and the results presented here show that taking account of ways in which a document can be modified improves performance. In future we would like to explore other approaches to modifying n-grams that capture other types of text editing operations that might be carried out when documents are reused, such as reordering and paraphrasing. We would also like to create a system that carries out both stages of duplicate detection by combining our approach with ones for Candidate Document Selection.

Contributors All authors contributed to the design of the study, interpretation of results, and approval of final manuscript. RN carried out the experiments. The first draft of the paper was written by RN and subsequently edited by MS and PC.

Funding The COMSATS Institute of Information Technology, Islamabad, Pakistan funded this work under the Faculty Development Program.

Competing interests None.

Provenance and peer review Not commissioned; externally peer reviewed.

Data sharing statement The data used to evaluate this work is already publicly available (via the Deja Vu database and Medline) and there is no additional data to provide.

REFERENCES

1 Broder A. On the resemblance and containment of documents. *Proceedings of the Conference on Compression and Complexity of Sequences*; 1997, IEEE Computer Society, 21–9.
2 Hoad TC, Zobel J. Methods for identifying versioned and plagiarized documents. *J Am Soc Info Sci Technol* 2003;54:203–15.
3 Kim J, Candan KS, Tatemura J. Efficient overlap and content reuse detection in blogs and online news articles. In: *Proceedings of the 18th International Conference on World Wide Web WWW-09*; 2009, ACM, 81–90.
4 Gwilym S, Swan M, Giele H. One in 13 'original' articles in the journal of bone and joint surgery are duplicate or fragmented publications. *J Bone Joint Surg* 2004;86-B:743–5.
5 Bailey B. Duplicate publication in the field of otolaryngology—head and neck surgery. *Otolaryngol Head Neck Surg* 2002;126:211–16.
6 Durani P. Duplicate publications: redundancy in plastic surgery literature. *J Plast Reconstr Aesthet Surg* 2006;59:975–7.
7 Clough P, Gaizauskas R, Piao S, *et al*. Measuring text reuse. *Proceedings of the 40th Annual Meeting of the Association of Computational Linguistics*; 2002:152–9.
8 Nawab R, Stevenson M, Clough P. Detecting text reuse with modified and weighted n-grams. **SEM 2012:The First Joint Conference on Lexical and Computational Semantics—Volume 1: Proceedings of the main conference and the shared task*; 2012:54–8.
9 Lyon C, Malcolm J, Dickerson B. Detecting short passages of similar text in large document collections. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*; 2001:118–25.
10 Stein B, Rosso P, Stamatatos E, *et al*. 3rd PAN workshop on uncovering plagiarism, authorship and social software misuse. *25th Annual Conference of the Spanish Society for Natural Language Processing (SEPLN)*; 2009:1–77.
11 Potthast M, Stein B, Eiselt A, *et al*. Overview of the 2nd International Competition on Plagiarism Detection. *Proceedings of the CLEF10 Workshop on Uncovering Plagiarism, Authorship and Social Software Misuse*; 2010.
12 Potthast M, Eiselt A, Barrón-Cedeño A. Overview of the 3rd International Competition on Plagiarism Detection. *Notebook Papers of CLEF 11 Labs and Workshops*; 2011.
13 Errami M, Hicks JM, Fisher W, *et al*. Déjà vu: a study of duplicate citations in MEDLINE. *Bioinformatics* 2008;24:243–9.
14 Lewis J, Ossowski S, Hicks J, *et al*. Text similarity: an alternative way to search Medline. *Bioinformatics* 2006;22:2298–304.
15 Errami M, Sun Z, George AC, *et al*. Identifying duplicate content using statistically improbable phrases. *Bioinformatics* 2010;26:1453–7.
16 Manning H, Schütze H. *Foundations of statistical natural language processing*. MIT Press: Cambridge, 1999.
17 Cover T, Thomas J. *Elements of information theory*. New York: Wiley, 1991.
18 Chen C, Yeh J, Ke H. Plagiarism Detection using ROUGE and WordNet. *In J Comput* 2010;2:34–44.
19 Chong M, Specia L. Lexical generalisation for word-level matching in plagiarism detection. *Proceedings of the recent advances in natural language processing (RANLP)*; 2011:704–9.
20 Stein B, Eissen SM, Potthast M. Strategies for retrieving plagiarized documents. *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*; 2007:825–6.
21 Barron-Cedeño A, Rosso P, Benedi J. Reducing the plagiarism detection search space on the basis of the Kullback-Leibler distance. *Proceedings of 10th International Conference on Computational Linguistics and Intelligent Text Processing*; 2009, Springer, 523–34.
22 Nawab RMA, Stevenson M, Clough P. Retrieving candidate plagiarised documents using query expansion. *Proceedings of the 34th European Conference on Information Retrieval (ECIR)*; 2012, Springer, 207–18.
23 Potthast M, Gollub T, Hagen M, *et al*. Overview of the 4th International Competition on Plagiarism Detection. *CLEF 2012 Evaluation Labs and Workshop—Working Notes Papers*, 2012.

24 Nawab RMA. *Mono-lingual paraphrased text reuse and plagiarism detection*. [PhD thesis] Department of Computer Science, University of Sheffield, 2012.
25 Chong M, Specia L, Mitkov R. Using natural language processing for automatic detection of plagiarism. *Proceedings of the 4th International Plagiarism Conference (IPC-2010)*, 2010.
26 Witten L, Frank E, Hall M. *Data mining: practical machine learning tools and techniques*. Morgan Kaufmann, 2011.
27 Aronson A, Lang F. An overview of Metamap: historical perspective and recent advances. *J Am Med Inform Assoc* 2010;17:229–36.
28 Ceska Z. *Automatic plagiarism detection based on latent semantic analysis*. [PhD thesis] Czech Republic, University of West Bohemia, 2009.
29 Nelson S, Powell T, Humphreys B. The Unified Medical Language System (UMLS) project. In: Kent A, Hall CM *Encyclopedia of library and information science*. Marcel Dekker, 2002:369.
30 Papineni K, Roukos S, Ward T, *et al*. BLEU: a method for automatic evaluation of machine translation. *Proceedings of 40th Annual Meeting of the Association of Computational Linguistics*; 2002:311–8.
31 Lin CY. ROUGE: a package for automatic evaluation of summaries. *Proceedings of Workshop on Text Summarization, Post-Conference Workshop of ACL*; 2004:74–81.
32 Stolcke A. SRILM: an extensible language modeling toolkit. *Proceedings of the International Conference on Spoken Language Processing*; 2002:901–4.
33 Wang W, Stolcke A, Zheng J. Reranking machine translation hypotheses with structured and web-based language models. *IEEE Workshop on Automatic Speech Recognition & Understanding*; 2007, IEEE, 159–64.
34 Errami M, Wren JD, Hicks JM, *et al*. eTBLAST: a web server to identify expert reviewers, appropriate journals and similar publications. *Nucleic Acids Res* 2007;35: W12–15.