IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Acronyms as an integral part of multi–word term recognition – A token of appreciation

Irena Spasić
School of Computer Science & Informatics, Cardiff University, Cardiff, CF24 3AA, UK

Corresponding author: Irena Spasić (e-mail: spasici@cardiff.ac.uk).

ABSTRACT Term conflation is the process of linking together different variants of the same term. In automatic term recognition approaches, all term variants should be aggregated into a single normalized term representative, which is associated with a single domain–specific concept as a latent variable. In a previous study, we described FlexiTerm, an unsupervised method for recognition of multi–word terms from a domain–specific corpus. It uses a range of methods to normalize three types of term variation – orthographic, morphological and syntactic variation. Acronyms, which represent a highly productive type of term variation, were not supported. In this study, we describe how the functionality of FlexiTerm has been extended to recognize acronyms and incorporate them into the term conflation process. The main contribution of this study is not acronym recognition per se, but rather its integration with other types of term variation into the term conflation process. We evaluated the effects of term conflation in the context of information retrieval as one of its most prominent applications. On average, relative recall increased by 32 percent points, whereas index compression factor increased by 7 percent points. Therefore, evidence suggests that integration of acronyms provides non–trivial improvement of term conflation.

INDEX TERMS text mining, natural language processing, terminology, information retrieval

I. INTRODUCTION

Terms are linguistic representations of domain–specific concepts [1, 2]. For practical purposes, terms are often defined as noun phrases that frequently are mentioned in a domain–specific discourse [3, 4]. They are distinguished from other salient phrases by the measures of their unithood and termhood [4]. Unithood is defined as the degree of collocational stability, while termhood refers to relevance to the domain. Termhood implies that terms carry heavier information load compared to other phrases used in a sublanguage, and as such they can be used to index and retrieve domain–specific documents, model domain–specific topics, identify text phrases useful for automatic summarization of domain–specific documents, identify slot fillers in information extraction, etc. It is, thus, essential to build and maintain terminologies in order to enhance the performance of many text mining applications [5]. Therefore, automatic term recognition (ATR) methods are needed to efficiently annotate electronic documents with a set of terms they mention. One such method is FlexiTerm, which implements an unsupervised approach to extraction of multi–word terms from a domain–specific corpus [6]. When originally evaluated on five biomedical corpora, the best results achieved were as follows: precision (94.56%), recall (71.31%) and F–measure (81.31%). Obviously, recall has considerable room for improvement. In relation to relatively poor recall, we focus on a specific methodological issue, which is related to the way (or lack) of processing acronyms. To highlight the issue and illustrate it with practical examples, we hereby provide a brief overview of the FlexiTerm method. It performs term recognition in three steps:

1. Lexico–syntactic filtering is used to select multi–word term candidates.
2. Term candidates are normalized to neutralize term variation.
3. A statistical measure of termhood is calculated in order to rank normalized term candidates.

A. STEP 1: LEXICO–SYNTACTIC FILTERING

Once input documents have been pre–processed [7, 8], including segmentation and part–of–speech (POS) tagging, term candidates are extracted by matching lexico–syntactic

patterns that specify the structure of the targeted noun phrases (NPs):

1. (JJ | NN)$^+$ NN, e.g. *congestive heart failure*
2. (NN | JJ)* NN POS (NN | JJ)* NN, e.g. *Hoffa's fat pad*
3. (NN | JJ)* NN IN (NN | JJ)* NN, e.g. *acute exacerbation of chronic bronchitis*

We used the Penn Treebank tag set [9] throughout this article (e.g. *NN*, *JJ*, *NP*, etc.).

B. STEP 2: TERM VARIANT NORMALISATION

Ideally, all term variants should be aggregated into a single normalized representative that would represent a term associated with a single domain–specific concept as a latent variable whose statistical properties we aim to measure [10]. Term candidates identified in Step 1 are normalized by addressing three types of term variation:

1. morphological variation, where the transformation of the content words involves inflection (e.g. *lateral meniscus* vs. *lateral menisci*) or derivation (e.g. *meniscal tear* vs. *meniscus tear*),
2. orthographic variation, where different conventions are used with respect to spelling (e.g. *Streptococcus pneumoniae* vs. *Streptococcus pneumonia*) and hyphenation (e.g. *posterolateral tibial plateau* vs. *postero–lateral tibial plateau*), and
3. syntactic variation, where the content words are re–arranged in terms of the overall phrase structure (e.g. *inhalation of thermal water* vs. *thermal water inhalation*).

The normalization process is similar to the one described in [11] and consists of the following steps:

1. Remove punctuation (e.g. ' in possessives), numbers and stop words including prepositions.
2. Remove any lowercase tokens with ≤2 characters.
3. Stem all remaining tokens and group them into a set.
4. For each stem, use approximate string matching to find similar stems in other term candidates and add them to the set.
5. The resulting set of stemmed tokens is the normalized term form.

For example, this process would map term candidates such as episodes of *presyncope* and *presyncopal episode* to the same normalized form {episod, presyncop}, thus neutralising both morphological and syntactic variation. Similarly, *posterolateral corner* and *postero–lateral corner* would be represented by {*postero–later*, *posterolater*, *corner*}. In this approach, morphological variation is neutralized by stemming [7, 8], orthographic variation is neutralized by approximate string matching [12-14], whereas syntactic variation is neutralized by representing term candidates as sets, in which particular order of the corresponding content words is no longer relevant.

C. STEP 3: TERMHOOD CALCULATION

Each term candidate is quantified by its termhood following the idea of cost criteria originally introduced for automatic collocation extraction [15]. Formally, the termhood of a normalized term representative t is calculated as follows:

$$C-value(t)=\begin{cases}\ln|t|\cdot f(t) & , \text{ if } S(t)=\varnothing \\ \ln|t|\cdot\left(f(t)-\frac{1}{|S(t)|}\sum_{s\in S(t)} f(s)\right) & , \text{ if } S(t)\neq\varnothing\end{cases} \quad (1)$$

In this formula, $|t|$ represents the number of non–equivalent tokens in *t*, e.g. |{*postero–later*, *posterolater*, *corner*}| = 2 because *postero–later* is equivalent to *posterolater* based on approximate matching. Effectively, this number corresponds to the length of the corresponding term variants not counting the tokens removed by Steps 1 and 2 of the normalization process. Further, $f(t)$ is the frequency with which any of the corresponding term variants occurred in the corpus, e.g. f({*postero–later*, *posterolater*, *corner*}) would add up the frequencies of *posterolateral corner* and *postero–lateral corner*. Finally, $S(t)$ is a set of all other term representatives that are proper supersets of *t*, e.g. it would contain a normalized form of the term candidate *posterolateral corner injury*, which would be {*postero–later*, *posterolater*, *corner*, *injuri*}. This *C–value* formula is equivalent to the one proposed to rank term candidates represented by strings [16]. It favors longer multi–word term candidates that occur more frequently and independently (i.e. not embedded in other term candidates).

D. ISSUES RELATED TO ACRONYMS

As described above, FlexiTerm will successfully neutralize three major sources of term variation. For example, it will correctly identify that term variants *exacerbation of chronic obstructive pulmonary disease* and *chronic obstructive pulmonary disease exacerbation* are equivalent to each other. Similarly, it will correctly identify that term variants *exacerbation of COPD* and *COPD exacerbation* are also equivalent to each other. However, it will fail to identify that all four term variants are equivalent to one another. This issue is related to a type of variation associated with multi–word terms, where multiple words are blended into a single token called an acronym, typically by taking the initial letters of salient words (e.g. *COPD* is an acronym of *chronic obstructive pulmonary disease*) or, in some cases, their morphemes (e.g. *ICS* is an acronym of *inhaled corticosteroids*). In particular, biomedical literature is associated with the widespread use and frequent coinage of acronyms [17]. Back in 2002, it was estimated that the number of unique acronyms in PubMed was increasing by approximately 11,000 per annum, whereas the number of the corresponding terms was growing at four times that rate [18]. The main purpose of introducing acronyms is to facilitate the use of frequently referenced multi–word terms in a domain–

specific discourse. In effect, acronyms are handy proxies for multi–word terms and, therefore, should be treated as multi–word terms themselves in term recognition approaches.

TABLE I
A SUMMARY OF ACRONYM RECOGNITION APPROACHES

Citation	System	Approach	Acronym definition	Domain	Documents	Evaluation (%)
[19]	Acrophile	pattern matching, heuristic	explicit	general	web pages	P =87, R= 88, F = 87
[20]		pattern matching, text compression	explicit	general	technical reports	P=90, R = 80, F=85
[17]	ACROMED	pattern matching, context–free parsing, heuristic	explicit	biomedical	PubMed abstracts	P=98, R=72, F=83
[21]		maximum entropy	implicit in text, explicit in a dictionary	clinical	clinical notes	A = 90
[18]	ARGH	heuristic	explicit	biomedical	PubMed abstracts	P=96, R=93
[22]	AbbRE	pattern matching, heuristic	explicit and implicit	biomedical	full–text articles	P=95, R=70
[23]		pattern matching, longest common sequence, logistic regression	explicit	biomedical	PubMed abstracts	P=80, R=83, F=82
[24]		heuristic	explicit	biomedical	PubMed abstracts	P=96, R=82, F=88
[25]		pattern matching, collocation analysis	explicit	biomedical	PubMed abstracts	P=96, R=88
[26]		dictionary matching for abbreviations, SVM for disambiguation against the full forms	does not matter	biomedical	PubMed abstracts	A=84
[27]	ALICE	heuristic	explicit	biomedical	PubMed abstracts	P=97, R=95
[28]		heuristic with syntactic constraints and pattern matching, several supervised learning methods	explicit	biomedical	PubMed abstracts	P=93, R=84, F=88
[29]		pattern matching, semi-supervised learning	implicit in text, explicit in a local dictionary	clinical	clinical notes	A = 68
[30]		dictionary extracted from literature, SVM for disambiguation against full forms	explicit and implicit	biomedical	PubMed abstracts	P=99, R=98, A = 98.5
[31]	ADAM	pattern matching, collocation analysis of context (*n*–grams)	explicit	biomedical	PubMed abstracts	P=97
[32]		ATR to identify terms appearing frequently in the proximity of an acronym, likelihood scores of being their full forms	explicit	biomedical	PubMed abstracts	P=78, R=85
[33]		supervised learning (naive Bayes, SVM and C4.5 decision trees)	implicit in text, explicit in a dictionary	clinical	clinical notes	A>90
[34]		dictionary extracted from PubMed, supervised learning for disambiguation (naive Bayes and SVM)	explicit and implicit	biomedical	full-text articles	P=92, R=91
[35]	AB3P	heuristic	explicit	biomedical	PubMed abstracts	P=97, R=85, F=91
[36]		HMM	explicit	biomedical	full-text articles	P=95, R=91, F=93
[37]		existing algorithms adapted to read and produce a specific format	explicit	biomedical		
[38]		distributional semantics (random indexing and random permutation)		medical / clinical	full-text articles / health records	R=39 to full form, R=33 from full form

P, R, F and A stand for precision, recall, F–measure and accuracy respectively.

Unfortunately, in its current form FlexiTerm will only extract acronyms when they are embedded in other terms (e.g. *exacerbation of COPD*), but not their standalone occurrences (e.g. *COPD*). This will skew the termhood calculation according to formula (1), because the frequency *f*(*t*) of a multi–word term (e.g. *chronic obstructive pulmonary disease*) will not take into account its mentions as an acronym (e.g. *COPD*), which by all intents and purposes is likely to be used more often than the original term. Another anomaly associated termhood calculation is that two term variants *exacerbation of chronic obstructive pulmonary disease* and *exacerbation of COPD* differ in length (five vs. two content words), which favors the longer variant. Moreover, both variants are disadvantaged in terms of their frequencies, which are calculated separately and, therefore, are practically halved in comparison to the joint frequency.

These facts imply that multi–word terms that have their own acronyms or embed references to other acronyms are statistically disadvantaged by the C–value formula and as such may remain unrecognized, thereby negatively affecting the recall of the method. Mapping acronyms to their full forms would resolve these issues. However, this cannot be done by post–processing FlexiTerm results. Acronym recognition and mapping to the corresponding full forms need to be fully integrated into the multi–word term recognition process after the initial selection of multi–word term candidates, but prior to termhood calculation. In this study, we describe the modification to the original FlexiTerm method that addresses this goal. The first prerequisite to attaining this goal is an acronym recognition method, which would extract acronym–definition pairs from a domain–specific corpus. In the following section, we provide an overview of such methods.

II. RELATED WORK

Acronyms are a highly productive type of term variation [39]. In particular, the prevalence of acronyms in biomedical domains [40] gave rise to proliferation of acronym disambiguation methods that extract acronyms and map them to their sense encoded explicitly in the full form. Table I provides a summary of such methods. Most of these methods focus on extracting acronyms from biomedical literature, and have been evaluated on either abstracts (e.g. [17, 18, 23-28, 30, 32, 34, 35]) or full–text articles (e.g. [22, 36]). These approaches rely on scientific writing conventions according to which acronyms should be defined the first time they are used in a document by first writing the full form followed by the acronym, written in uppercase, within parentheses [41]. With some exceptions (e.g. see [42]), general compliance with these conventions is exploited by the aforementioned methods, which typically apply pattern matching to identify potential acronym–definition pairs followed by heuristic alignment of the two (e.g. [17-19, 22, 24, 27, 35]). This alignment can be posed as the longest common subsequence problem, in which case dynamic programming can be used as an alternative to heuristic approaches to find an optimal alignment [23]. An early approach used text compression to match acronyms and potential definitions [20]. Several supervised learning methods were used to learn how to select acronym–definition pairs, out of which support vector machines (SVM) provided the best results [28]. More recently, hidden Markov models (HMM) have been used to support the alignment of acronyms and their definitions [36]. In a large corpus, where there are multiple long–form candidates for a given acronym, statistical analysis can be used to support mapping of the acronym to the most likely definition. Examples of statistical approaches include logistic regression [23], collocation analysis [25, 31] and termhood [32].

So far we discussed recognition of acronyms as local abbreviations, whose long form is explicitly stated in a document [30]. By contrast, global abbreviations appear in a document without their definitions. They are commonly found in clinical narratives and to a lesser extent in scientific literature. These are usually common abbreviations, which are widely accepted as preferred synonyms of prominent domain–specific concepts (e.g. DNA and deoxyribonucleic acid) [22]. As such, they are described in relevant domain dictionaries, e.g. [43] and [44]. However, shorter acronyms tend to be ambiguous [39, 45], and, therefore, they may have multiple entries in such dictionaries (e.g. *diabetes mellitus*, *dystrophia myotonica*, *doctor of medicine*, *dextromethorphan* and *Drosophila melanogaster* share the same acronym, *DM*). Automatic recognition of global acronyms usually entails their mapping to a correct entry in an external dictionary and this may be viewed as a word sense disambiguation problem [46]. Supervised learning approaches have been most commonly used to classify acronyms with respect to their sense, e.g. SVM, naive Bayes classification and C4.5 decision trees [26, 30, 33, 34]. Semi–supervised methods based on maximum entropy [21] and cosine similarity [29] applied to acronym's context have also been tried. More recently, models of distributional semantics, which are based on the assumption that linguistic items with similar distributions in a large corpus tend to have similar meanings, have been used to pair up acronyms and their long forms [38]. This approach represents an unsupervised approach, which has got the advantage of being inherently portable.

The goal of our study was not to implement a new acronym recognition approach per se, but rather to integrate such functionality with that of FlexiTerm. The following section describes how we implemented such integration.

III. METHODS

A. PROBLEM SPECIFICATION

We have previously differentiated between two types of acronyms – local and global. Local acronyms are explicitly defined in a document following scientific writing conventions, which prescribe that the first mention of an acronym is accompanied with its full form, either of which is specified within parentheses, e.g.

> *The nuclear factor kappaB (**NF–kappaB**) is thought to be crucially involved in the gene activation of several cytokines, including tumor necrosis factor (**TNF**).*
>
> *Glucocorticoid receptors are also able to interact with transcriptional factors such as **AP–1** (activator protein–1) of **NF–kappaB** (nuclear factor–kappaB).*

By contrast, global acronyms appear in a document without their definitions. They are commonly found in clinical narratives and to a lesser extent in scientific literature, e.g.

***MRI** RIGHT KNEE – Normal meniscus and collateral ligaments. Normal postero–lateral corner structures. **ACL** is slightly ill–defined and has intrasubstance high signal, which I think is most likely to be due to mucoid degeneration, but **ACL** and **PCL** are intact.*

In this paper, we will refer to these two types of acronyms as explicit and implicit acronyms respectively. Their use is associated with different types of discourse, e.g. acronyms are explicitly defined in scientific literature, but not necessarily in clinical notes or patient narratives. The original FlexiTerm method proved to be more robust than the baseline against less formally structured texts, such as those found in patient blogs or clinical notes. To integrate acronyms into multi–word term recognition while preserving the generality of the method, both types of acronyms need to be supported.

B. EXPLICIT ACRONYM RECOGNITION

We have previously discussed a range of methods that support explicit acronym recognition, most of which implement heuristic approaches with no significant differences in performance. Typically, the precision of these methods is in the 90s and F–measure is in the 80s. Recall tends to vary more, but usually it the ranges from 70s to 90s. This is measured against all instances of acronym–definition pairs. For FlexiTerm to incorporate acronyms into multi–word term recognition, they need to be correctly interpreted, i.e. mapped to the corresponding full forms. For a given acronym we do not need to extract every instance of acronym–definition pairs. In fact, a single acronym–definition pair would suffice. In this respect, recall is not an essential criterion for our choice of an acronym recognition method. Precision, on the other hand, is an essential requirement. Given that most of the considered methods have got the precision well over 90%, our decision was based on two relevant criteria: (1) generality of the method, and (2) its ease of use. In terms of generality, heuristic approaches are preferred to machine learning ones as they are readily portable between domains and require no training. As for the ease of use, source code should be readily available to enable necessary modifications and incur as little re–implementation as possible. A simple algorithm for identifying abbreviation by Schwartz and Hearst [24] is by far the most referenced method of its kind and it does satisfy both criteria. It performs at 96% precision, is available under an open source license and is written in the same programming language as FlexiTerm. As such, it was a natural choice to support explicit acronym recognition in FlexiTerm.

Originally, Schwartz and Hearst algorithm operates at a document level, i.e. it systematically scans the document for potential acronym–definition pairs, followed by extraction of the full forms, which do not cross heuristically determined sentence/clause boundaries. At the very start, FlexiTerm performs linguistic pre–processing of input documents. This process involves sentence splitting, tokenization, lemmatization and stemming. The pre–processing results are stored in a relational database for easy access and retrieval. To take advantage of this fact and make better use of available computational resources, we modified the original Schwartz and Hearst algorithm to operate at a sentence level. Only those sentences that contain potential acronyms, identified by the presence of parentheses, are retrieved from the database and passed on to the acronym recognition module.

All instances of automatically identified acronym–definition pairs are also stored in a database for further analysis by FlexiTerm. Assuming that acronyms are synonyms of multi–word terms, we compare their automatically extracted definitions against term candidates already identified by FlexiTerm using lexico–syntactic filtering. In this manner, we constrain the results of acronym recognition using lexico–syntactic information and, thereby, reduce occasional false positives [17, 28].

In addition to improving the precision of acronym recognition, this step is important for the term normalization process. FlexiTerm aims to maintain a single normalized representative for all term variants, which is associated with a single domain–specific concept, as a latent variable whose statistical properties we aim to measure. For an acronym, as a single token, to fit into this normalization scheme, it needs to be normalized to the same representative as its full form. At this stage, multi–word term candidates have already been normalized. By matching the acronym's full form to an existing term candidate, we can simply re–use its normalized form.

Acronyms, like other words, tend to have only one sense per discourse [47]. However, an acronym's full form may be matched to multiple normalized term representatives, in which case we need to perform disambiguation in order to add acronym as a variant to one and only one term representative. The same disambiguation approach is applied to both explicit and implicit acronyms, thus, we will re–visit this issue once we have described our approach to implicit acronym recognition.

C. IMPLICIT ACRONYM RECOGNITION

Implicit acronyms are not explicitly defined in a document. They are commonly found in clinical narratives as widely accepted synonyms of the corresponding domain–specific terms (e.g. *STD* and *sexually transmitted disease*). Such acronyms are known globally and, hence, their usage is prescribed in relevant dictionaries. Few methods summarized in Table I that focus on implicit acronym recognition in clinical narratives incorporate such dictionaries as local lexical resources in their methods [21, 29, 33]. FlexiTerm, however, is a data–driven, domain–independent method and we would like to preserve these features in its new version that incorporates acronym recognition. To achieve this,

implicit acronyms need to be recognized dynamically without resorting to static lexical resources.

We implemented a simple heuristic approach that first identifies potential acronyms using their orthographic properties and frequency of occurrence. Recall that all input documents undergo linguistic pre–processing, including tokenization and lemmatization, whose results are stored in a relational database for easy access and retrieval. A single query is used to retrieve potential acronyms using the following criteria on their lemmas: (1) It must start with an uppercase letter. (2) It must not contain a lowercase letter. (3) It must not end with a period. (4) It has to be at least three characters long. (5) Its frequency of occurrence must be above a certain threshold.

Proper English words get lowercased as part of the lemmatization process. Therefore, performing the given query against lemmas will only focus on words where uppercase format is their distinct characteristic rather than a consequence of syntax (e.g. starting a sentence with a capital letter) or formatting conventions (e.g. uppercasing section titles in clinical narratives). For example, in the following section title *MRI RIGHT KNEE* of an imaging report, the last two words would get lemmatized to *right* and *knee* respectively, which, therefore, would not be considered as acronyms despite their frequent uppercased use in a corpus of imaging reports.

The first two criteria combined allow for some types of punctuation, e.g. *PAPP–A* (*pregnancy-associated plasma protein A*) and *PM&R* (*physical medicine and rehabilitation*). According to these two criteria, numbers are also allowed, e.g. *PAI1* (*plasminogen activator inhibitor 1*), but lowercased letters are not. Therefore, instances such as *NF–kappaB* (*nuclear factor–kappa B*) would not be considered. Unlike explicit acronyms, whose recognition exploits their proximity to the corresponding full forms, the selection of implicit acronym candidates relies solely on their surface forms, which are subsequently matched to phrases found elsewhere in the corpus. Therefore, to reduce the number of false positives, stricter selection criteria need to apply. In the wider context of ATR and specifically the role of acronyms in term conflation, the precision of acronym recognition outweighs the concerns related to its recall.

The third criterion has been introduced to prevent selection of abbreviations other than acronyms, e.g. contractions such as *DR.* and *MRS.*, which are frequently found in clinical narratives. Note that this will also prevent selection of punctuated versions of acronyms (e.g. *M.R.I.* vs. *MRI*). Although there are exceptions, a prevalent rule is to omit the periods in acronyms [48]. Therefore, this constraint is not expected to affect the recall significantly. In a further attempt to prioritize precision over recall, we do not attempt to extract two–letter acronyms, because shorter acronyms tend to be ambiguous [39]. Finally, we assume that important acronyms are frequently used in a domain–specific corpus. Omission of rare acronyms would not have a significant effect on termhood calculation based on the C–value formula, which provides further justification for introducing a frequency threshold.

Once potential acronyms have been identified, the next step is to map them to their full forms, which are supposed to be terms themselves. Therefore, we compare acronyms against term candidates already identified by FlexiTerm using lexico–syntactic filtering. Given a potential acronym as a sequence of k characters $L_1L_2...L_k$, a single query is used to retrieve term candidates that consist of k tokens that start with the given characters (irrespective of their case) in the given order. For example, *ACL* would match *anterior cruciate ligament*, but not *articular cartilage*. By focusing on initialisms only, this approach is purposefully strict in an attempt to reduce the search space and false positives, and thereby improve the performance in terms of efficiency and precision.

As before, by matching the acronym to an existing term candidate, we can simply re–use its normalized form. A potential problem is that an acronym may be matched to multiple normalized term representatives, in which case we need to perform disambiguation in order to add acronym as a variant to one and only one term representative. The same disambiguation approach is applied to both explicit and implicit acronyms, which is described in the following section.

D. ACRONYM SENSE DISAMBIGUATION

We implemented a heuristic approach to acronym disambiguation. In the first step, we compare potential normalized term representatives with respect to their frequency of occurrence in the corpus. We select the most frequent one as the most plausible full form based on a hypothesis that acronyms are introduced to facilitate the use of frequently referenced multi–word terms in a domain–specific discourse.

In case of a tie, we compare potential normalized term representatives using their length measured by the number of tokens. We select the longest one in order to prevent selecting full forms that embed other acronyms. For example, *AECOPD* can be introduced as an acronym for either *acute exacerbation of chronic obstructive pulmonary disease* or *acute exacerbations of COPD*. In our experiments, both definitions were extracted as multi–word term candidates and were normalized to {*acut*, *exacerb*, *chronic*, *obstruct*, *pulmonary*, *diseas*} and {*acut*, *exacerb*, *copd*} respectively. Eventually, both of these variants will be merged, but in this manner *AOCOPD* will be mapped directly to the full form without having to expand the embedded acronym.

Finally, in an unlikely event that an acronym still remains ambiguous, we use a brute–force strategy and select the first normalized term representative in alphabetical order. This step is used only as the last resort to guarantee one–to–one mapping from acronyms to normalized term representatives (in a deterministic fashion) so that FlexiTerm may proceed

with termhood calculation without double counting the acronyms.

E. MULTI-WORD TERM RECOGNITION

The following pseudocode provides a summary of the FlexiTerm method, which now fully integrates acronym recognition into the multi–word term recognition process after the initial selection of multi–word term candidates, but prior to termhood calculation:

1. Pre–process text to annotate it with lexico–syntactic information.
2. Select multi–word term candidates using pattern matching on POS tagged text.
3. Normalize multi–word term candidates by performing the following steps.
   a. Remove punctuation, numbers and stop words.
   b. Remove any lowercase tokens with ≤2 characters.
   c. Stem each remaining token.
4. Map acronyms to their full forms (one–to–one).
   a. Recognize acronyms and their potential full forms.
   b. Remove full forms that do not have a match amongst multi–word term candidates.
   c. Normalize acronyms' full forms (see Step 3).
   d. Disambiguate acronyms with multiple (normalized) full forms.
      i. Remove less frequent full forms.
      ii. Remove shorter full forms.
      iii. Remove alphabetically descendant full forms.
5. Add acronyms to the list of multi–word term candidates, which are normalized using their full forms.
6. Process acronyms nested within multi–word term candidates.
   a. Replace acronym with its full form.
   b. Re–normalize multi–word term candidate (see Step 3).
7. Extract distinct token stems from normalized multi–word term candidates.
8. Compare token stems using lexical and phonetic similarity.
9. Expand normalized term candidates by adding similar token stems (see Step 5).
10. For each normalized multi–word term candidate *t*:
    a. Determine set *S*(*t*) of all normalized term candidates that contain *t* as a subset.
    b. Calculate *C–value*(*t*) according to formula (1).
11. Rank normalized term candidates using their C–value.

Steps 4–6 summarize modifications to the original FlexiTerm method. Once the acronyms have been recognized as described in the preceding sections, they are added to the list of multi–word term candidates as variants of their full forms. Both acronym and its full form will have the same normalized representative, which means that they will be treated as a single term candidate for the purpose of termhood calculation. Once stand–alone acronyms have been added to the list of multi–word term candidates, all other normalized term candidates are searched for nested occurrences of newly added acronyms, which are then replaced by their normalized representatives. The updated term candidates are then re–normalized to restore alphabetical order of individual tokens in their normalized forms. Once all acronyms have been processed, the termhood calculation proceeds as prescribed in the original method.

IV. RESULTS

A. APPLICATION CONTEXT

The main goal of integrating acronym recognition into the multi–word term recognition process is to neutralize this type of term variation and its effects on term recognition. Specifically, by addressing this type of term variation in addition to morphological, orthographic and syntactic variation, we are looking to further improve term conflation, i.e. grouping all variants of the same term together [49-56]. One of the most prominent applications of term conflation is information retrieval (IR) [57-60], a process of selecting documents relevant to a user's information need expressed using a search query. In the context of IR, term conflation can support query expansion, whose goal is to automatically expand the query by adding synonyms and other closely related words [61]. In particular, matching acronyms to their long forms is often quoted as an important step for improving the performance of IR systems in terms of precision and recall [21, 26, 28, 30, 33, 35], which is further emphasized by the fact that the use of acronyms in search queries is frequent [62]. We will, therefore, evaluate the new version of FlexiTerm in the context of IR as one of its immediate applications.

B. EVALUATION MEASURES

Given a fixed document collection and a user's information need expressed as a search query, a document retrieved by a system is classified either as a true positive (TP) if it is relevant to the given information need or as a false positive (FP) if it is not. Conversely, a relevant document is classified as a false negative (FN) if it is not retrieved by the system. Given the total numbers of TPs, FPs and FNs, precision (P) and recall (R) are calculated as the following ratios on a scale from 0 to 1:

$$P = \frac{TP}{TP + FP} \quad R = \frac{TP}{TP + FN} \qquad (2)$$

In other words, precision represents the proportion of correctly retrieved documents, while recall represents the proportion of relevant documents that are retrieved by the system. For the precision to be calculated it suffices to manually inspect the retrieved documents with respect to their relevance to the search query. Calculating recall, on the other hand, requires manually annotating the whole document collection, which is potentially large, thus

rendering this measure impractical in many cases. If we focus on recall as a way of comparing multiple systems against one another, then it is worth noting that its denominator, i.e. the sum of TPs and FNs, which equals the number of relevant document, is independent of the system and as such will remain constant across all systems. Therefore, when comparing the recall of two systems, their ratio will match that of their numerators, i.e. TPs, which would already be calculated for the precision by manually inspecting the retrieved documents, therefore eliminating the need for manually annotating the whole document collection. Nonetheless, it is still useful to normalize the value of TPs on a scale from 0 and 1. Relative recall achieves this by dividing the number of relevant documents retrieved by a given system (i.e. TPs) by the total number of relevant documents retrieved by any of the considered systems [63]. In our experiments, we will be reporting precision and relative recall values.

In the context of IR, we can also measure the extent to what a term–based index would be compressed by conflation of term variants. This is analogous to the idea of index compression factor (ICF), which represents the fractional reduction in index size achieved through stemming and is calculated according to the following formula:

$$ICF = \frac{w - s}{w} \qquad (3)$$

where w is the number of distinct words before stemming and s is the number of distinct stems [64]. We adapted this formula by calculating w as the number of distinct term variants and s as the number of distinct terms (i.e. their normalized representatives). In this case, ICF represents the extent to which a list of terms is compressed by their normalization. Higher values of ICF indicate higher rate of term conflation.

C. EXPERIMENTS

We would like to compare how much term conflation as part of multi–word term recognition improves IR results. This comparison requires three prerequisites: (1) a baseline ATR system, (2) a document collection, and (3) a test set of term–based search queries.

Let us first discuss the choice of a baseline system. The main aim of our experiments is to measure the effect that the inclusion of acronyms has on the performance of multi–word term recognition, in particular their term conflation component. In other words, we want to conduct a controlled experiment in which acronym recognition represents an independent variable of otherwise fixed term recognition process. Therefore, to measure relative improvement of term conflation, the original FlexiTerm method represents a natural baseline. In our experiments, we will refer to the two versions of the system as FlexiTerm 1.0 and FlexiTerm 2.0. Similar relationship exists between FlexiTerm 1.0 and TerMine [65], a freely available service from the academic domain based on C–value [16]. FlexiTerm 1.0 extends the term conflation component of TerMine by addressing syntactic variation on top of orthographic and morphological variation.

In summary, we conducted a series of controlled experiments in which term conflation was treated as an independent variable of otherwise fixed term recognition process. The following types of term variation were considered in three experiments: (1) orthographic and morphological variation (TerMine), (2) orthographic, morphological and syntactic variation without acronyms (FlexiTerm 1.0), and (3) orthographic, morphological and syntactic variation with acronyms (FlexiTerm 2.0). In the context of controlled experiments with a focus on term variation, the use of any other external system as the baseline would be inappropriate.

TABLE II
DATA SETS USED IN EVALUATION

Data set	Topic	Document type	Source
D1	molecular biology	abstract	PubMed
D2	COPD	abstract	PubMed
D3	COPD	patient blogpost	Web
D4	obesity, diabetes	clinical narrative	i2b2
D5	knee MRI scan	clinical narrative	NHS

The next choice to be made in our experiments is that of a document collection to run the three systems on. We originally evaluated the performance of FlexiTerm 1.0 using five document collections from different biomedical subdomains (e.g. molecular biology, medical diagnostic imaging or respiratory diseases) as well as text written by different types of authors and/or aimed at different audience (e.g. scientists, healthcare professionals or patients). Table II describes the five collections consisting of 100 documents each, which we re–used in this study (see [6] for more details).

Finally, to create a test set of term–based search queries for each document collection, we re–used the ATR results of the two baseline systems from the previous study [6] and combined them with the ATR results from this study. We selected a subset of automatically recognized terms in a manner that does not favor any of the three systems. For each document collection, we started with an empty set of terms. In each iteration, three terms were added to the set. The highest ranked term by TerMine that was not already in the test set was added, followed by the highest ranked term by FlexiTerm 1.0 that was not already in the test set, followed by the highest ranked term by FlexiTerm 2.0 that was not already in the test set. The process was stopped after five iterations.

Having selected 15 terms per document collection, each term was converted into the corresponding search query by automatically expanding it with all its variants automatically recognized by the system considered. For example, let us consider *COPD exacerbation* as the search term and how it

would be automatically expanded using the output of the three systems. Using the TerMine results, a Boolean query would be expanded into: *"COPD exacerbation" OR "COPD exacerbations"*. Using FlexiTerm 1.0 results, the query would be expanded using two additional variants as follows: *"COPD exacerbation" OR "COPD exacerbations" OR "exacerbation of COPD" OR "exacerbations of COPD"*. Finally, using FlexiTerm 2.0 results, the query would include three additional variants: *"COPD exacerbation" OR "COPD exacerbations" OR "exacerbation of COPD" OR "exacerbations of COPD" OR "exacerbation of chronic obstructive pulmonary disease" OR "exacerbations of chronic obstructive pulmonary disease" OR "chronic obstructive pulmonary disease exacerbations"*.

The search queries (represented formally in SQL) were run against individual sentences in a relevant data collection, which was managed in a relational database. The retrieved sentences were inspected manually to differentiate between TPs and FPs. The only FP identified was related to a term variant *human cells*, which was incorrectly grouped with *human t cells* by both FlexiTerm versions. Such high precision throughout can be explained by the homogeneity of the test corpora and "one sense per discourse" hypothesis [47]. In reality (e.g. if running the same queries against PubMed), the precision would naturally be expected to be lower. Nonetheless, in the context of this study it provides evidence that most term variants were correctly conflated by all three systems considered.

To calculate relative recall, TPs were compared to the union of TPs retrieved by any of the three versions of the search query. Finally, the values of relative recall were micro–averaged to evaluate the overall performance (see Figure 1). The following trends can be observed. With one exception (D1), FlexiTerm 1.0 outperforms TerMine by 6 percent points on average. FlexiTerm 2.0 outperforms other two methods substantially. On average, it improves relative recall by 29 percent points. These values demonstrate the benefits of term conflation. In general, the larger the conflation classes (on average), the higher the relative recall. To measure the former, we used ICF (see Figure 2) – the bigger ICF, the better the conflation. By neutralizing morphological and orthographic variation, TerMine achieved ICF of 16% on average. By neutralizing syntactic variation in addition to these two types of variation, FlexiTerm 1.0 achieved ICF of 19% on average. By including acronyms on top of these three types of term variation, FlexiTerm 2.0 achieved ICF of 26% on average. The following example illustrates the added value that consideration of acronyms provides to term conflation. A single multi–word term candidate *health–related quality of life* was successfully matched to three other variants solely by the consideration of acronyms: *health–related QoL*, *HR–QoL* and *HRQL*.

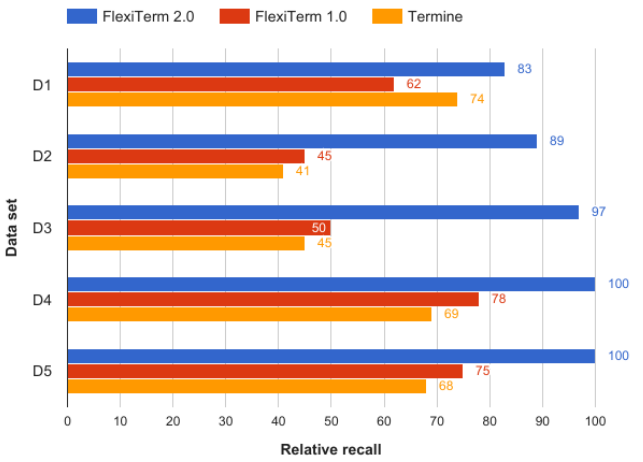

FIGURE 1. **The effects of term conflation on relative recall.**

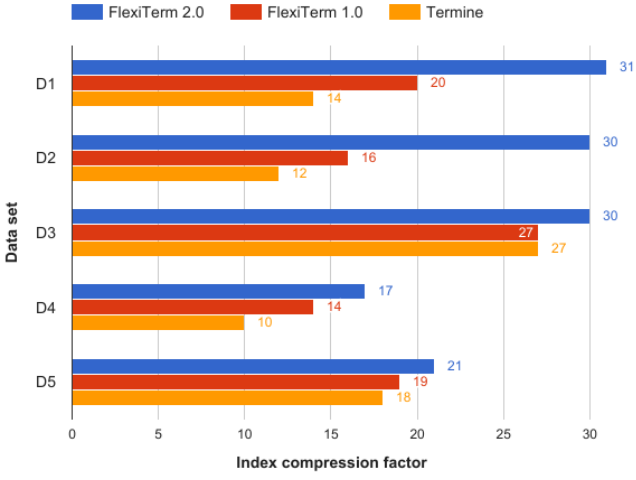

FIGURE 2. **The extent of term conflation measured by the ICF.**

D. DISCUSSION

In this section we discuss the results of acronym recognition. Under the "one sense per discourse" hypothesis [47], we evaluated the precision of acronym recognition by checking whether it matched the sense of a multi–word term candidate it was added to as a variant. In case of explicit acronym recognition, which was originally evaluated as an information extraction task, at 96% the precision of the chosen algorithm was very high to start with [24]. Our own algorithm for implicit acronym recognition was deliberately strict in order to achieve high precision. Overall, lexico–syntactic constraints applied to multi–word term candidates in combination with sense disambiguation (see Step 4 of the FlexiTerm algorithm) resulted in 100% precision. In other words, all automatically recognized acronyms were correctly interpreted. In addition to discussing the effects that addition of these acronyms had on overall term recognition, we also turn our attention to issues related to recall, i.e. those

acronyms that were not recognized. We discuss the results for each data set D1–D5 separately.

Coincidentally, a total of 57 explicit acronyms were extracted from both literature corpora D1 and D2. Tables III and IV provide top 10 most frequently mentioned acronyms mapped to their full forms, which were extracted automatically. The last two columns show the rank of the full form (together with all its variants) produced by FlexiTerm 2.0 and FlexiTerm 1.0 respectively. The given values illustrate that acronyms provide a strong boost in term candidate ranking. In particular, terms that were previously not recognized (indicated by the N/A value) benefited from aggregation with the corresponding acronyms, which enabled them to pass the termhood threshold.

Recognition of implicit acronyms in a collection of patient blog posts (D3) resulted in a total of only two acronyms (see Table V). Interestingly, the blog posts were written by patients with chronic obstructive pulmonary disease, but the term itself was previously not recognized due to patients' tendency to use the corresponding acronym *COPD*. Once the full form was mapped to the acronym and their numerical properties aggregated, *chronic obstructive pulmonary disease* became the highest ranked term. Overall, the use of acronyms in patient blogs was not frequent. Two other relevant acronyms, *MRSA* (*methicillin–resistant Staphylococcus aureus*) and *FEV* (*forced expiratory volume*), were not recognized because their full forms were not mentioned in the corpus. This did not have a negative impact on term recognition, because these acronyms were rarely used. The analysis of potential acronyms identified by the use of uppercase letters highlighted a potential problem with acronym recognition in patient narratives, which may be confused with the use of Internet slang, e.g. *LOL* (*laughing out loud*). Even though they are formed following the same principles as domain–specific acronyms, their full forms do not generally match the structure of terms and, therefore, would be filtered out during lexico–syntactic filtering. However, they could still be matched incorrectly to other term candidates, e.g. *lease of life*. In our experiments, the frequency threshold for potential acronyms prevented such errors.

TABLE III
TOP 10 MOST FREQUENT ACRONYMS IN DATA SET D1

Acronym	Full form	Frequency	Term rank	Previous term rank
NF–kappaB	nuclear factor–kappaB	36	3	31
TNF–alpha	tumor necrosis factor alpha	34	1	13
TNF	tumor necrosis factor alpha	24	1	13
CBF	core binding factor	19	10	N/A
GM–CSF	granulocyte–macrophage colony–stimulating factor	15	12	44
GR	glucocorticoid receptor	12	7	23
PMA	phorbol myristate acetate	12	19	57
AR	androgen receptor	11	36	58
HIV	human immunodeficiency virus	11	17	40
IFN–gamma	interferon gamma	11	47	N/A

TABLE IV
TOP 10 MOST FREQUENT ACRONYMS IN DATA SET D2

Acronym	Full form	Frequency	Term rank	Previous term rank
COPD	chronic obstructive pulmonary disease	406	1	1
PR	pulmonary rehabilitation	26	9	27
QoL	quality of life	15	8	9
AECOPD	acute exacerbations of chronic obstructive pulmonary disease	14	6	14
OR	odd ratio	13	15	N/A
ICS	inhaled corticosteroids	10	30	35
BAL	bronchial lavage	9	42	N/A
FRC	functional residual capacity	9	21	N/A
HI	high–intensity group	9	33	N/A
CB	chronic bronchitis	8	14	24

TABLE V
TOP 10 MOST FREQUENT ACRONYMS IN DATA SET D3

Acronym	Full form	Frequency	Term rank	Previous term rank
COPD	chronic obstructive pulmonary disease	103	1	N/A
UBE	upper body ergometer	3	13	N/A

TABLE VI
TOP 10 MOST FREQUENT ACRONYMS IN DATA SET D4

Acronym	Full form	Frequency	Term rank	Previous term rank
CHF	congestive heart failure	27	3	8
DVT	deep venous thrombosis	19	14	76
RCA	right coronary artery	15	10	18
PTCA	percutaneous transhepatic coronary angioplasty	11	24	73
ETT	exercise tolerance test	10	17	37
SVG	saphenous vein graft	9	25	56
PND	paroxysmal nocturnal dyspnea	7	30	56
CCU	cardiac care unit	6	36	60
COPD	chronic obstructive pulmonary disease	6	53	N/A
UTI	urinary tract infection	6	21	31

TABLE VII
TOP 10 MOST FREQUENT ACRONYMS IN DATA SET D5

Acronym	Full form	Frequency	Term rank	Previous term rank
ACL	anterior cruciate ligament	97	2	N/A
PCL	posterior cruciate ligament	57	5	N/A
MCL	medial collateral ligament	35	8	17
LCL	lateral collateral ligament	3	33	36

A total of 10 implicit acronyms (see Table VI) were correctly recognized from a collection of hospital discharge summaries (D4). A total of 8 acronyms were not recognized, because their full forms were not mentioned elsewhere in the corpus, e.g. *PICC* (*peripherally inserted central catheter*) and *PND* (*post nasal drip*). Because of a strict condition not to consider two–letter acronyms in an attempt to reduce false positives, two such acronyms were not recognized, *CP* (*chest pain*) and *EF* (*ejection fraction*). Interestingly, in these two cases full forms were used more frequently than the corresponding acronyms. For example, *chest pain* was used 98 times, whereas *CP* was used only 12 times. Similarly, *ejection fraction* was used 47 times, whereas *EF* was used 20 times. This phenomenon can be explained by the fact that (1) shorter acronyms tend to be ambiguous [39], so clinicians may be consciously avoiding their use, and (2) their full forms are shorter and, therefore, not as time consuming to write. Because of relatively frequent use of the full forms, these two terms were still highly ranked (2nd and 12th) and, therefore, less affected by ignoring their acronyms. This provides additional justification for setting a threshold for the length of implicit acronyms. In addition to length, we also imposed a strict condition that the number of letters in an implicit acronym has to match the number of tokens in the full form. Only one acronym was not recognized for this reason – *CXR* (*chest X–ray*). As before, the relatively short full form *chest X–ray* was used more frequently than the corresponding acronym *CXR* (31 times vs. 8 times). Again, the term itself was successfully recognized on its own and ranked 20th. Two three–letter acronyms whose full form consists of two tokens, *ASA* (*acetylsalicylic acid*) and *CPK* (*creatine phosphokinase*), would not be recognized anyway because their full forms were not mentioned elsewhere in the corpus. Two three–letter acronyms whose full form consists of a single word, *HTN* (*hypertension*) and *HCT* (*hematocrit*), are irrelevant in the context of multi–word term recognition.

Finally, there were only 6 implicit acronyms mentioned in a collection of imaging reports (D5). The most frequent acronym *MRI* was not recognized as such, because it is described in WordNet and is, therefore, treated as a regular English word and lowercased during the lemmatization process. Its full form was not mentioned either and its use was confined to the report title. We did not attempt to recognize two–letter acronyms such as *OA* (*osteoarthritis*). The full form of this particular acronym is a single–word term, which makes it irrelevant to our term recognition method. All remaining acronyms were correctly recognized (see Table VII). They provided a substantial boost to the calculation of termhood, based on which two previously unrecognized terms were ranked among top five.

V. CONCLUSIONS

In this study, we fully integrated acronym recognition and their mapping to the corresponding full forms into the multi–word term recognition process. Our approach supports two modes of acronym recognition: (1) explicit (or local) acronyms, which are defined in a text document following scientific writing conventions, and (2) implicit (or global) acronyms, which appear in a text document (e.g. clinical notes) without their definitions. While implicit acronym recognition in itself presents a novel approach, the main contribution of this study is not acronym recognition per se, but rather its integration with other types of term variation into the term conflation process. The novelty of this study lies in the use of acronym recognition to resolve a methodological issue concerning the way in which multi–word terms are processed statistically. In turn, by addressing acronyms in addition to morphological, orthographic and syntactic variation, we improved the conflation of term variants substantially across a wide range of biomedical discourse types, including scientific literature, clinical notes and patient narratives. The results demonstrate that the given methodological issue entailed practical implications in terms of performance.

We evaluated the effects of term conflation in the context of information retrieval as one of its most prominent applications. Specifically, term conflation was evaluated in relation to query expansion and index compression. By using term variants to automatically expand search queries, substantial improvement was made in terms of relative recall while maintaining the same precision. The addition of acronyms improved relative recall of the method by 32 percent points on average. This is substantially higher than the previous improvement (less than 3 percent points) made over the original baseline on account of syntactic variation.

REFERENCES

1. Jacquemin, C., *Spotting and Discovering Terms through Natural Language Processing*. 2001: MIT Press. 378.
2. Frantzi, K. and S. Ananiadou, *Automatic term recognition using contextual cues*, in *Proceedings of 3rd DELOS Workshop on Cross-Language Information Retrieval*. 1997: Zurich, Switzerland. p. 2155-2162.
3. Dailles, B., *Study and implementation of combined techniques for automatic extraction of terminology*, in *The Balancing Act - Combining Symbolic and Statistical Approaches to Language*, P. Resnik and J. Klavans, Editors. 1996, MIT Press. p. 49-66.
4. Kageura, K. and B. Umino, *Methods of automatic term recognition - A review*. Terminology, 1996. **3**(2): p. 259-289.
5. Feldman, R., et al., *Text mining at the term level*, in *Principles of Data Mining and Knowledge Discovery, Second European Symposium, PKDD '98, Nantes, France, Proceedings*, J. Zytkow and M. Quafafou, Editors. 1998, Springer-Verlag. p. 65-73.
6. Spasić, I., et al., *FlexiTerm: A flexible term recognition method.* Journal of Biomedical Semantics, 2013. **4**: p. 27.
7. The Stanford Natural Language Processing Group, *Stanford Log-linear Part-Of-Speech Tagger*, in *The Stanford NLP Group*. 2012.
8. Toutanova, K., et al. *Feature-rich part-of-speech tagging with a cyclic dependency network*. in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*. 2003. Edmonton, Canada.
9. Marcus, M.P., M.A. Marcinkiewicz, and B. Santorini, *Building a large annotated corpus of English: the Penn Treebank.* Computional Linguistics, 1993. **19**(2): p. 313-330.
10. Nenadic, G., I. Spasic, and S. Ananiadou, *Mining Term Similarities from Corpora*. Terminology, 2004. **10**(1): p. 55-80.
11. McCray, A., S. Srinivasan, and A. Browne. *Lexical methods for managing variation in biomedical terminologies*. in *18th Annual Symposium on Computer Applications in Medical Care*. 1994. Washington, USA.
12. Jazzy, *(Java spelling checker API).* 2012.
13. Damerau, F., *A technique for computer detection and correction of spelling errors.* Communications of the ACM, 1964. **7**(3): p. 171-176.
14. Philips, L., *Hanging on the Metaphone.* Computer Language, 1990. **7**(12): p. 39-43.
15. Kita, K., Y. Kato, T. Omoto and Y. Yano. , *A comparative study of automatic extraction of collocations from corpora: Mutual information vs. cost criteria.* Journal of Natural Language Processing, 1994. **1**(1): p. 21-33.
16. Frantzi, K. and S. Ananiadou, *The C-value/NC-value domain independent method for multiword term extraction.* Journal of Natural Language Processing, 1999. **6**(3): p. 145-180.
17. Pustejovsky, J., et al., *Automatic extraction of acronym-meaning pairs from MEDLINE databases.* Studies in Health Technology and Informatics, 2001. **84**(1): p. 371-375.
18. Wren, J.D. and H. Garner, *Heuristics for identification of acronym-definition patterns within text: Towards an automated construction of comprehensive acronym-definition dictionaries.* Methods of Information in Medicine, 2002. **41**(5): p. 426-434.
19. Larkey, L.S., et al. *Acrophile: an automated acronym extractor and server*. in *Proceedings of the 5th ACM Conference on Digital Libraries*. 2000. San Antonio, Texas, USA.
20. Yeates, S., D. Bainbridge, and I.H. Witten. *Using compression to identify acronyms in text*. in *Proceedings of the Data Compression Conference*. 2000. Snowbird, Utah, USA.
21. Pakhomov, S. *Semi-supervised maximum entropy based approach to acronym and abbreviation normalization in medical texts*. in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*. 2002. 160-167.
22. Yu, H., G. Hripcsak, and C. Friedman, *Mapping abbreviations to full forms in biomedical articles.* Journal of American Medical Informatics Association, 2002. **9**(3): p. 262–272.
23. Chang, J.T., H. Schütze, and R.B. Altman, *Creating an online dictionary of abbreviations from MEDLINE.* Journal of the American Medical Informatics Association, 2002. **9**(6): p. 612-620.
24. Schwartz, A. and M. Hearst. *A simple algorithm for identifying abbreviation definitions in biomedical text*. in *Proceedings of the 8th Pacific Symposium on Biocomputing*. 2003. Lihue, Hawaii, USA.
25. Liu, H. and C. Friedman. *Mining terminological knowledge in large biomedical corpora*. in *Proceedings of the Pacific Symposium on Biocomputing*. 2003. Lihue, Hawaii, USA.
26. Yu, Z., Y. Tsuroka, and J. Tsujii. *Automatic resolution of ambiguous abbreviations in biomedical texts using support vector machines and one sense per discourse hypothesis*. in *Proceedings of the SIGIR Workshop on Text Analysis and Search for Bioinformatics*. 2003. Toronto, Canada.
27. Ao, H. and T. Takagi, *ALICE: An algorithm to extract abbreviations from MEDLINE.* Journal of American Medical Informatics Association, 2005. **12**(5): p. 576-586.
28. Nadeau, D. and P.D. Turney, *A supervised learning approach to acronym identification*, in *Advances in Artificial Intelligence, Vol. 3501*, B. Kégl and G. Lapalme, Editors. 2005. p. 319-329.
29. Pakhomov, S., T. Pedersen, and C.G. Chute. *Abbreviation and acronym disambiguation in clinical discourse*. in *Proceedings of the Annual AMIA Symposium*. 2005. Washington, DC, USA.
30. Gaudan, S., H. Kirsch, and D. Rebholz-Schuhmann, *Resolving abbreviations to their senses in Medline.* Bioinformatics, 2005. **21**(18): p. 3658-3664.
31. Zhou, W., V.I. Torvik, and N.R. Smalheiser, *ADAM: Another database of abbreviations in MEDLINE.* Bioinformatics, 2006. **22**(22): p. 2813-2818.

32. Okazaki, N. and S. Ananiadou. *A term recognition approach to acronym recognition*. in *Proceedings of the COLING/ACL*. 2006. Sydney, Australia.
33. Pakhomov, M.J.S., T. Pedersen, and C.G. Chute. *A comparative study of supervised learning as applied to acronym expansion in clinical reports.* in *Proceedings of the Annual American Medical Informatics Association Symposium*. 2006. Washington, DC, USA.
34. Yu, H., et al., *Using MEDLINE as a knowledge source for disambiguating abbreviations and acronyms in full-text biomedical journal articles.* Journal of Biomedical Informatics, 2007. **40**(2): p. 150-159.
35. Sohn, S., et al., *Abbreviation definition identification based on automatic precision estimates.* BMC Bioinformatics, 2008. **9**(1): p. 402.
36. Movshovitz-Attias, D. and W.W. Cohen. *Alignment-HMM-based extraction of abbreviations from biomedical text*. in *Proceedings of the 2012 Workshop on Biomedical Natural Language Processing*. 2012. Montreal, Canada.
37. Doğan, R.I., et al., *Finding abbreviations in biomedical literature: three BioC-compatible modules and four BioC-formatted corpora.* Database, 2014: p. bau044.
38. Henriksson, A., et al., *Synonym extraction and abbreviation expansion with ensembles of semantic spaces.* Journal of Biomedical Semantics, 2014. **5**: p. 6.
39. Andersson, L., A. Hanbury, and A. Rauber, *The portability of three types of text mining techniques into the patent text genre*, in *Current Challenges in Patent Information Retrieval*, M. Lupu, K.M.N. Kando, and A.J. Trippe, Editors. 2017. p. 241-280.
40. Liu, H. and A.R.A.C. Friedman. *A study of abbreviations in MEDLINE abstracts*. in *Proceedings of the American Medical Informatics Association Symposium*. 2002. Washington, DC, USA.
41. Doumont, J.-L., *English Communication for Scientists*. 2010, Cambridge, MA, USA: NPG Education.
42. Cheng, T.O., *Acronym aggravation.* British Heart Journal, 1994. **7**(1): p. 107-109.
43. Stedman, *Stedman's Medical Dictionary, 29th edition*. 2016: Lippincott Williams & Wilkins.
44. Bodenreider, O., *The Unified Medical Language System (UMLS): integrating biomedical terminology*. Nucleic Acids Research, 2004. **32**(Database): p. D267–D270.
45. Moon, S., et al., *A sense inventory for clinical abbreviations and acronyms created using clinical notes and medical dictionary resources.* Journal of the American Medical Informatics, 2013. **21**(2): p. 299-307.
46. Agirre, E. and M. Stevenson, *Knowledge sources for WSD*, in *Word Sense Disambiguation*. 2006. p. 217-251.
47. Gale, W.A., K.W. Church, and D. Yarowsky. *One sense per discourse*. in *Proceedings of the HLT Workshop on Speech and Natural Language*. 1992. Harriman, New York, USA.
48. Robert Allen (Ed.), *Pocket Fowler's Modern English Usage (Second edition)*. 2008: Oxford University Press.
49. Jacquemin, C. and J. Royaute. *Retrieving terms and their variants in a lexicalized unification-based framework*. in *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1994. Dublin, Ireland.
50. Klavans, J.L., C. Jacquemin, and E. Tzoukermann. *A natural language approach to multi-word term conflation*. in *Proceedings of the DELOS Conference*. 1997.
51. Tzoukermann, E., J.L. Klavans, and C. Jacquemin. *Effective use of natural language processing techniques for automatic conflation of multi-word terms: the role of derivational morphology, part of speech tagging, and shallow parsing*. in *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1997. Philadelphia, Pennsylvania, USA.
52. Jacquemin, C. and E. Tzoukermann, *NLP for term variant extraction: Synergy between morphology, lexicon, and syntax.* Natural Language Information Retrieval, 1999. **7**: p. 25-74.
53. Savary, A. and C. Jacquemin, *Reducing information variation in text*, in *Text- and Speech-Triggered Information Access, Lecture Notes in Computer Science, Vol 2705*, R. S. and G. G., Editors. 2003, Springer: Berlin, Heidelberg. p. 145-181.
54. Gálvez, C., *Standardization of terms applying finite-state transducers (FST)*, in *Handbook of Research on Digital Libraries: Design, Development and Impact*. 2009. p. 102-112.
55. Jacquet, G., M. Ehrmann, and R. Steinberger. *Clustering of multi-word named entity variants: Multilingual evaluation*. in *Proceedings of the 9th International Conference on Language Resources and Evaluation*. 2014. Reykjavik, Iceland.
56. Tseytlin, E., et al., *NOBLE - Flexible concept recognition for large-scale biomedical natural language processing.* BMC Bioinformatics, 2016. **17**: p. 32.
57. Frakes, W.B. *Term conflation for information retrieval*. in *Proceedings of the 7th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1984. Cambridge, England.
58. Jacquemin, C., J.L. Klavans, and E. Tzoukermann. *Expansion of multi-word terms for indexing and retrieval using morphology and syntax*. in *Proceedings of the 8th Conference on European Chapter of the Association for Computational Linguistics*. 1997. Madrid, Spain.
59. Gálvez, C., F.d. Moya-Anegón, and V.H. Solana, *Term conflation methods in information retrieval: Non-linguistic and linguistic approaches.* Journal of Documentation, 2005. **61**(4): p. 520-547.
60. Pirkola, A., *Extracting variant forms of chemical names for information retrieval.* Information Research, 2008. **13**(3).
61. Manning, C.D., P. Raghavan, and H. Schütze, *Introduction to information retrieval*. 2008: Cambridge University Press.
62. Dogan, R.I., et al., *Understanding PubMed user search behavior through log analysis.* Database, 2009: p. bap018.
63. Clarke, S.J. and P. Willett, *Estimating the recall performance of Web search engines.* Aslib Proceedings, 1997. **49**(7): p. 184-189.
64. Frakes, W.B. and C.J. Fox, *Strength and similarity of affix removal stemming algorithms.* ACM SIGIR Forum, 2003. **37**(1): p. 26-30.
65. Termine. *http://www.nactem.ac.uk/software/termine/*. 2017 [cited 2013.

IRENA SPASIĆ received a PhD degree in computer science from the University of Salford, UK in 2004. Following posts at the Universities of Belgrade, Salford and Manchester, she joined Cardiff School of Computer Science & Informatics in 2010, and became full professor in 2016. Her research interests include text mining, knowledge representation, machine learning and information management with applications in healthcare, life sciences and social sciences. She leads the text and data mining research theme at Cardiff University and is a co–founder of the UK Healthcare Text Analytics Research Network (HealTex).