Idiom-Based Features in Sentiment Analysis: Cutting the Gordian Knot

Irena Spasić, Lowri Williams, and Andreas Buerki

Abstract—In this paper we describe an automated approach to enriching sentiment analysis with idiom-based features. Specifically, we automated the development of the supporting lexico-semantic resources, which include (1) a set of rules used to identify idioms in text and (2) their sentiment polarity classifications. Our method demonstrates how idiom dictionaries, which are readily available general pedagogical resources, can be adapted into purpose-specific computational resources automatically. These resources were then used to replace the manually engineered counterparts in an existing system, which originally outperformed the baseline sentiment analysis approaches by 17 percentage points on average, taking the F-measure from 40s into 60s. The new fully automated approach outperformed the baselines by 8 percentage points on average taking the F-measure from 40s into 50s. Although the latter improvement is not as high as the one achieved with the manually engineered features, it has got the advantage of being more general in a sense that it can readily utilize an arbitrary list of idioms without the knowledge acquisition overhead previously associated with this task, thereby fully automating the original approach.

Index Terms—Sentiment analysis, natural language processing, text mining, knowledge engineering, feature extraction

1 Introduction

Figurative language whose meaning differs from the literal interpretation poses significant challenges to natural language understanding. Idioms are considered to be one of the most prominent types of figurative language. However, there is considerable disagreement about what constitutes an idiom and how idioms might be categorized (for overviews see [1], [2]). From a semantic standpoint, it may be exceptionally difficult to make a conclusive assessment of idiomaticity, because such assessment depends on an often questionable abstraction of word senses away from contexts or on fine judgments of what is literal or metaphorical [3]. Nevertheless, semantic non-compositionality and a degree of fixedness are often taken as key markers of idioms, e.g., [4], [5], [6]. This definition chimes with the popular understanding of the term idiom and works reasonably well for prototypical cases such as *fly off the handle*. A distinction is often made between idioms of encoding (where idiomatic knowledge is mainly required to produce an idiom, e.g., *long time no see*) and idioms of decoding (where idiomatic knowledge is required to understand an idiom, e.g., *paint the town red*) [7], with the latter being of primary interest for natural language understanding.

In a previous study we investigated the role of idioms in sentiment analysis [8], an important subarea of natural language understanding whose aim is to automatically interpret opinions, sentiments, attitudes and emotions expressed in written text [9]. To estimate the degree to which the inclusion of idioms as features could improve the results of traditional sentiment analysis approaches, we compared our results to two such methods, SentiStrength [10], [11] and Stanford CoreNLP's sentiment annotator [12]. First, to support the use of idioms as features in sentiment analysis we collected a set of 580 emotionally charged idioms, which were then annotated with sentiment polarity using a web–based crowdsourcing approach. In addition, we manually defined a set of lexico-semantic pattern-matching rules to automate the recognition of idiom occurrences in text. Second, to evaluate the results of sentiment analysis enriched with idiom features, we assembled a corpus of sentences in which these idioms were expressed. Each sentence was annotated with sentiment polarity using the same crowdsourcing approach. These annotations formed the basis for the gold standard, which was used to compare our idiom-enriched sentiment analysis approach against the two baseline methods. The performance was evaluated in terms of precision, recall and F-measure. The relative improvement over the baseline results by 20 and 15 percentage points respectively was found to be statistically significant. While the results were improved significantly across all sentiment polarity classes (i.e., positive, negative and other), the most notable improvement was recorded in the classification of positive sentiments, where recall was improved by 45 percentage points in both experiments without compromising the precision.

Given the positive findings of the initial study, we are now looking to fully automate the originally proposed sentiment analysis approach enriched with idioms as features.

- *I. Spasić and L. Williams are with the School of Computer Science & Informatics, Cardiff University, 5 The Parade, Cardiff CF24 3AA, United Kingdom. E-mail: {spasici, williamsl10}@ cardiff.ac.uk.*
- *A. Buerki is with the School of English, Communication and Philosophy, Cardiff University, Colum Drive, Cardiff CF10 3EU, United Kingdom. E-mail: buerkia@cardiff.ac.uk.*

Manuscript received 19 May 2017; revised 15 Oct. 2017; accepted 21 Nov. 2017. Date of publication 19 Dec. 2017; date of current version 29 May 2020. (Corresponding author: Irena Spasić.)
Recommended for acceptance by F. Sebastiani.
Digital Object Identifier no. 10.1109/TAFFC.2017.2777842

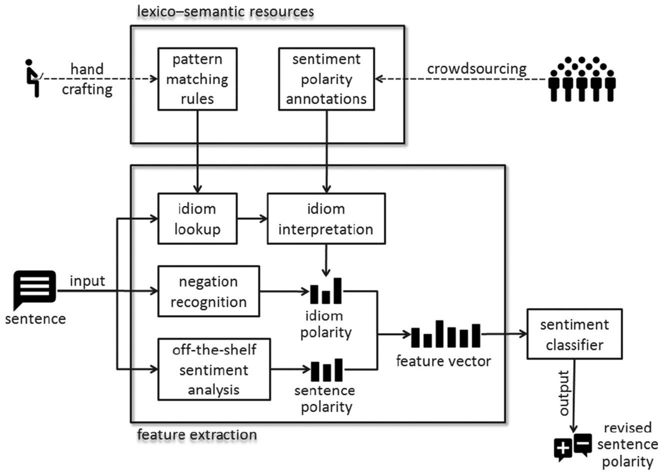

Fig. 1. The system architecture diagram.

The main limitation of the original approach is a significant knowledge-engineering overhead involved in handcrafting lexico-semantic patterns for recognition of idioms and annotation of their polarity. In this study, we describe how we addressed this bottleneck by automating two crucial steps: (1) encoding lexico-semantic patterns that enable idiom recognition in text, and (2) determining idiom polarity. As a result, we fully automated the use of idioms in sentiment analysis and minimized the knowledge engineering bottleneck associated with this task. To demonstrate the effectiveness of the newly proposed approach, we re-ran the experiments described in the original study in order to compare the two cases.

2 Related Work

Fig. 1 provides an overview of the originally proposed system, which incorporates idiom as features into sentiment analysis [8]. We will illustrate its functionality using a set of examples given in Table 1. We will also use this framework to make references to related work where appropriate.

2.1 Extraction of Idiom-Based Features

Given a sentence, the system will look up occurrences of idioms from a predefined list using a pattern-matching approach that accounts for their lexico–syntactic variations.

TABLE 1
A Sample of Input Sentences

ID	Sentence	SentiStrength	Stanford CoreNLP	Sentiment annotation
S1	This observation was to bear fruit in later years.	neg:-1 pos:1 neutral	neg:22 ∅:65 pos: 12 neutral	positive
S2	The noise must have been awful, but it was music to my ears.	neg:-4 pos: 1 negative	neg:60 ∅:31 pos:9 negative	positive
S3	You have what is called, I believe, a large chip on your shoulder.	neg:-1 pos:1 neutral	neg:23 ∅:53 pos:24 neutral	negative
S4	We watch as friendships come apart at the seams.	neg:-1 pos:2 positive	neg:49 ∅:34 pos: 17 negative	negative
S 5	They are not after an olive branch.	neg:-1 pos:1 neutral	neg:5S ∅:38 pos:4 negative	negative

TABLE 2
Lexico-Semantic Information About Idioms

ID	Idiom	Pattern	Polarity
I1	to bear fruit	<BEAR> fruit	neg:0 ∅:0 pos:100
I2	music to one's ears	music to < PRP$> ears	neg:0 ∅:0 pos:100
I3	a chip on one's shoulder	chip on <PRP$> shoulder	neg: 100 ∅:0 pos:0
I4	to come apart at the seams	<COME> apart at the seams	neg: 100 ∅:0 pos:0
I5	olive branch	olive branch	neg:0 ∅:0 pos:100

Non-terminal symbols <BEAR>, <COME> and <PRP$> can be replaced by any form of the verb to bear (i.e., bore, born, borne, bears and bearing), any form of the verb to come (i.e. come, came, comes and coming) and a possessive pronoun (i.e. my, your, his, her, its, our, their and one's) respectively.

For example, using the patterns associated with idioms given in Table 2, the system identifies the occurrences of idioms I1–I5 in sentences S1–S5 respectively and interprets them using the crowdsourced sentiment polarity annotations (the last column in Table 2). Based on the negation recognized in sentence S5, the sentiment polarity associated with idiom I5 is inverted from positive to negative.

2.2 Off-the-Shelf Sentiment Analysis

In parallel, the overall sentiment of the given sentences is calculated using an off–the–shelf approach to sentiment analysis. We provide results of two such systems: SentiStrength [10], [11] and Stanford CoreNLP's sentiment annotator [12]. If we compare the automatically predicted sentiment polarity to manual annotations (see Table 1), we can see that all but two predictions were incorrect. SentiStrength uses a rule-based approach to estimate the sentiment of individual words and combines these values to predict the overall sentiment. This approach is not suitable for analyzing idioms in terms of their sentiment, because their meaning, including the associated sentiment, cannot be entirely predicted from the constituent words considered independently [13]. For example, due to an absence of either positive or negative words in the lexical construction of idioms given in Table 2, all of them were classified as neutral by SentiStrength even though, as multi-word units, they are strongly polarized in terms of the underlying sentiment.

Stanford CoreNLP's sentiment annotator, however, uses a deep neural network (DNN) approach to build up sentiment representation of a sentence on top of its grammatical structure. In other words, the sentiment is predicted based on the way in which the words are combined into phrases, which is one of the reasons why this method performed better than SentiStrength on a given set of sentences (see Table 1). Nonetheless, it still misclassified the sentiment of 3 out of 5 sentences, the main reason being that the use of idioms is relatively infrequent [14], [15], and, therefore, training data may not contain sufficient representation of idioms for them to be generalized into a sentiment classification model. Focusing on the DNN approach, it has been found that while some features are learnt repeatedly across multiple networks, rare features are not always learnt [16]. Still, it has been shown that rare features generally improve the quality of text classification [17], [18]. With respect to idioms, our previous study on sentiment analysis identified them as being very predictive but comparatively rare features [8]. These three facts combined imply that idioms need to be incorporated as features into sentiment analysis

TABLE 3
Sentences Represented by Feature Vectors

ID	Idiom			Sentence			Sentiment prediction
	neg	∅	pos	neg	∅	pos	
S1	0	0	100	22	65	12	positive
S2	0	0	100	60	31	9	positive
S3	100	0	0	23	53	24	negative
S4	100	0	0	49	34	17	negative
S5	100	0	0	58	38	4	negative

approaches, but that these features would be difficult to learn automatically using machine learning approaches such as DNNs. Therefore, an alternative unsupervised approach is needed in order to systematically incorporate idioms as features into sentiment analysis methods.

2.3 Feature Combination for Supervised Sentiment Classification

Having extracted two types of sentiment polarities, one related to idioms (see Section 2.1) and the other related to the overall sentiment of the sentence (see Section 2.2), the next step is to combine these features in order to re-calculate the overall sentiment of the sentence by taking idioms into account. Having had no prior knowledge of the significance of idioms for the sentiment classification task, we simply concatenated all features into a single vector (see Table 3 for examples). This approach does not guarantee an optimal performance [19], which means that there is further potential for improvement. Nonetheless, the initial results indicated that even this simple approach improved the results of sentiment analysis significantly.

A wide range of supervised learning approaches can be used to perform sentiment analysis over the combined feature vectors. We used Weka [20], a popular suite of machine learning software, to train a sentiment classification model. The trained model was embedded into the system shown in Fig. 1 as a sentiment classifier (see bottom right box), where it is used to classify combined feature vectors in terms of their sentiment polarity.

By this point, we did not refer to specific machine learning algorithms, because the “no free lunch” theorem suggests that there is no universally best learning algorithm [21]. In other words, the choice of an appropriate algorithm should be based on its performance for the particular problem at hand and the properties of data that characterize the problem. We based our choice on the results of cross-validation experiments on the training dataset, in which a Bayesian network classifier outperformed other methods available in Weka.

2.4 Addressing the Resource Bottleneck

The key functionality of the system, i.e., the extraction of idiom-based features, is supported by a set of lexico-semantic resources (see upper box in Fig. 1). Traditionally, such resources would be created manually by dedicated experts, but crowdsourcing emerged as a viable alternative for creating such resources on a much larger scale [22], [23], [24]. In our approach, we used a combination of the two approaches. Idiom formation patterns were hand-crafted by an expert in computational linguistics, whereas their sentiment polarities were crowdsourced from non-experts. The reliability of non-expert annotations has been identified as a risk associated with the use of crowdsourcing [25], [26]. The main strategy for improving the reliability of non-expert annotations appears to be increasing their number [27]. Combined with the overhead associated with handcrafting lexico-semantic patterns, the need to increase the number of manual annotations per each idiom creates a bottleneck in the acquisition of lexico-semantic resources (see dashed lines in Fig. 1).

In particular, lexicon acquisition is a major bottleneck for sentiment analysis. To address this issue, much of the work in sentiment analysis focused on automating the acquisition of sentiment lexicons. The suggested approaches can be divided into two basic categories—corpus-based and thesaurus-based approaches. Corpus–based approaches rely on a hypothesis that words with the same polarity co-occur in a corpus. Therefore, the polarity of words may be determined from their co-occurrence with the “seed” words of known polarity [28], [29], [30], [31]. Most of the corpus-based approaches focus on single words. However, the polarity of a phrase may differ from that of its words [32], [33], which makes these approaches unsuitable for the task of determining the polarity of idioms. To capture non-compositional semantics, a corpus-based approach has been generalized to n-grams [34]. This approach has been effective in modelling the sentiment of modifier-noun pairs and negations. However, it is not suitable for handling idioms due to their variability (in relation to the use of n-grams) and relative rarity (in relation to the use of distributional semantics).

Thesaurus-based methods typically explore the structure of a thesaurus (e.g., WordNet [35]) to determine polarity of unknown words by using their relationships to the “seed” words of known polarity [36], [37], [38], [39], [40]. These approaches rely on a hypothesis that synonyms (e.g., *excellent* and *splendid*) have the same polarity, whereas antonyms (e.g., *excellent* and *inferior*) have the opposite polarity. Starting with the “seed” set of words, the thesaurus network of lexical relationships is crawled to iteratively propagate the polarity in a rule-based approach. We too explored the use of WordNet for the task of determining the polarity of idioms. We randomly selected 10 percent out of our set of 580 idioms. We then searched WordNet for these idioms. Out of 58 idioms only 7 were found. Our finding concurs with a previous study that concluded that WordNet does not systematically include idiomatic expression [41]. Therefore, as it stands, the above approaches cannot be applied to idioms.

In the absence of explicit lexical relationships between some words, further sources of information contained in a thesaurus have been explored. For example, the glosses (i.e., textual definitions) of words have been explored based on a hypothesis that words that have similar glosses have similar polarity [42], [43]. However, these approaches fail to capture contextual polarity of words (e.g., *low risk* versus *low cost*). The gloss-based approach has been used in [44] to generalize their previous approach to determining contextual polarity [32], but it still remains limited to adjective-noun pairs. Nonetheless, the general idea of using glosses to determine the polarity of corresponding lexical items is applicable to idioms. We already concluded that we cannot use WordNet for this purpose. Fortunately, a plethora of readily available pedagogical resources dedicated specifically to the study of idioms can be used instead.

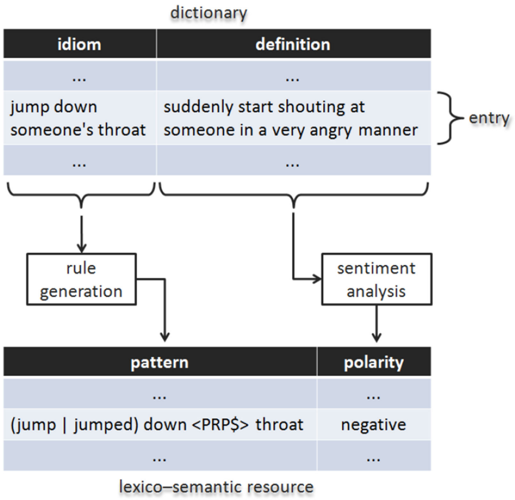

Fig. 2. Automated resource acquisition.

In the following section, we describe how we addressed the resource bottleneck by automating two crucial steps: (1) encoding lexico-semantic patterns that enable idiom recognition in text, and (2) determining idiom polarity. As a result, we fully automated the use of idioms in sentiment analysis and minimized the knowledge engineering bottleneck associated with this task. Consequently, this can scale up the semantic coverage of the original system beyond the limited set of 580 manually selected idioms.

3 Methods

The aim of this study was to determine to what extent we can automate the use of idioms in sentiment analysis. To address this aim we:

1. developed methods to automate resource acquisition (see Fig. 2),
2. applied these methods to generate lexico-semantic resources,
3. incorporated these resources into the sentiment analysis system by replacing the original ones, which were generated manually (see upper box in Fig. 1),
4. repeated the experiments described in the original study [8], and
5. evaluated the performance of the system using the original results as the baseline.

In this section, we provide details associated with the approaches used to support steps 1-3 above. The remaining steps, i.e., the experiments and their results, are reported in Section 4.

3.1 Pattern-Matching Rule Induction

When used in discourse, idioms may occur in different surface forms. Hence their occurrences in text cannot be identified using string matching. Much of the work on idiom recognition focused on distinguishing between literal and figurative meaning of idiomatic expressions [45], [46], [47], [48], [49], [50] using statistical approaches based on a hypothesis of lexical fixedness or lexical cohesion. These studies impose considerable syntactic restrictions, e.g., by focusing solely on verb+noun combinations. On the other side, studies that do not impose such restrictions concentrate on segmentation of corpora into multi-word lexical units [51], [52], a superclass of idioms, and as such are too generic.

In the original study, we defined a set of lexico-semantic pattern–matching rules to automate the recognition of idiom occurrences in text. The goal of this study is to use the canonical form of an idiom to derive its variations automatically, where the canonical form refers to the main form listed in an idiom dictionary. The difficulty associated with this task is the fact that idioms are very heterogeneous in terms of their transformational capacity [53].

Some idioms allow virtually no variation without the loss of the idiomatic sense while most allow (or in some cases require) various, often extensive, types of variation [54], [55], [56]. For the purposes of this study, we focused on generalizing over the following types of common variation: inflection, open slots, adjectival and adverbial modification, passivization and distribution over multiple clauses as they were described in [57].

3.1.1 *Inflection*

In terms of inflection of idiom constituents, in many cases verbs can be used in different tenses, whereas some nouns can be either singular or plural. For example, the verb in the idiom *stir a hornet's nest* is used in the present perfect tense in the following example:

> Forbes has *stirred* up a hornet's nest.

Similarly, the noun in the idiom *bone to pick* is used in plural the following example:

> He generously leaves us one or two *bones* to pick.

The problem of inflection can be solved by lemmatizing both the canonical form of an idiom and the text to be matched. For example, lemmatization leaves both idioms unchanged, but transforms the given sentences into forms in which the given idioms can be matched as strings:

> Forbes have *stir up a hornet's nest*.
> He generously leave us one or two *bone to pick*.

3.1.2 *Open Slots*

Many idioms contain open slots into which noun phrases can be inserted. For example, in the use idiom *send someone packing* the open slot, which is indicated by an indefinite pronoun in the citation form, is replaced by a two–word noun phrase in the following example:

> New rule could send *some insurers* packing.

The problem of open slots in idioms can be addressed by another type of linguistic processing—shallow parsing or chunking, which groups words into phrases. For example, the result of parsing the given sentence is as follows:

> [NP *New rule*] [VP *could send*] [NP *some insurers*] [VP *packing*].

The elements of the imposed shallow structure can then be used to generalize the search for idioms with open slots using a pattern *send* <NP> *packing* (or its lemmatized version—*send* <NP> *pack*) [58], where indefinite pronoun in the idiom's canonical form was replaced automatically by <NP> (a non-terminal symbol that can be replaced by any noun phrase) in the corresponding pattern matching rule. Even though state-of-the-art noun phrase chunking methods perform at F-measure of 94 percent [59], the problem of incorrectly parsed noun phrases remains a potential problem in this approach. Alternatively, one may choose to ignore the syntactic structure altogether and instead search for a flexgram [60], a sequence of tokens with one or more gaps of variable length, e.g., *send * packing*. This is a less accurate method (i.e., it may generate more false positives), but more robust in terms of recall.

3.1.3 Modification

The components of some idioms are modifiable, e.g., by using adjectives to modify nouns or adverbs to modify verbs. The following example of the idiom *grasp at straws* contains both types of modification:

> You seem to want to grasp *desperately* at *every single* straw.

Nouns and verbs as potentially modifiable components can be identified using part-of-speech tagging, e.g., *grasp*/VB *at*/IN *straws*/NN. The results of lemmatization and tagging can be combined to generate the corresponding flexgram automatically by inserting gaps before nouns and after verbs. In the previous example, the automatically generated flexgram *grasp * at * straw* would match the modified version of the idiom.

3.1.4 Passivization

In addition to inflection (see Section 3.1.1) verbs in many idioms may vary in terms of their voice too. Passive voice allows the object of an otherwise active sentence to become the subject of a passive sentence. In this process, the order between the verb and its object gets reversed with the original idiom components becoming separated. For example, compare an active form of the idiom *bury the hatchet*:

> Christmas looks to be a time for *burying* the *hatchet* or exhuming it for re–examination.

to a passive one:

> From the look of things, the *hatchet* has been long *buried*.

To account for the passivization of idioms, automatically acquired part–of–speech information can be used to identify non–auxiliary verbs at the beginning of an idiom and produce additional flexgram for its passive form, in which the verb should appear at the end with a gap inserted in front. For example, the tagged version of the given idiom, *bury*/VB *the*/DT *hatchet*/NN, can be used to identify *bury* as the leading verb and produce *the hatchet * bury* as the passive version of the matching flexgram. The lemmatized versions of the flexgram and the passive sentence now match:

> From the look of thing, *the hatchet have be long bury*.

3.1.5 Distribution Over Multiple Clauses

The components of some idioms may be distributed between a main clause and a subordinate one as is the case in the following example:

> You remember [NP *the hatchet*] [SBAR that we *buried* last year with such pomp and ceremony]?

The issue associated with this phenomenon is that idiom components become separated by the introduction of a subordinate clause. Most of the examples of this type variation are related to the use of the verb component of an idiom as the main verb of the subordinate clause [57] and can be effectively resolved by the pattern–matching rule generated previously to address passivization. For example, the same flexgram *the hatchet * bury* will also match the lemmatized version of the distributed idiom:

> You remember *the hatchet that we bury* last year with such pomp and ceremony?

Similarly, most other types of variations discussed in [57] can be recognized by the pattern–matching rules generated to address passivization.

3.1.6 Hand-Crafted versus Automatically Induced Patterns

We compared the performance of automatically induced pattern-matching rules against that of hand-crafted ones. The rules were applied against the test dataset of 500 sentences in which idiom occurrences were annotated manually. Hand-crafted rules retrieved all idiom occurrences, therefore achieving 100 percent recall while achieving 94.44 percent precision. The loss of precision was associated with the literal use of idiomatic expressions. On the other hand, automatically generated rules recorded 92.68 percent precision at 92.87 percent recall. While the precision was comparable down to 2 percent points, the recall dropped by 7 percent points, which would suggest that the flexibility of pattern matching rules was somehow affected. However, a closer inspection of the results revealed that the drop in recall was associated with incorrectly lemmatized words, which in turn was due to incorrectly determined part of speech. Most commonly, participles and adjectives were confused. For example, idiom *pleased as punch* was tagged as *pleased/VB as/IN punch/NN* and lemmatized accordingly as *please as punch*. However, its occurrence in the corpus was tagged as *pleased/JJ as/IN punch/NN* and lemmatized accordingly as *pleased as punch*, which caused the above rule to fail. Nonetheless, the overall performance was sufficient to proceed with further experiments.

3.2 Sentiment Polarity of Idioms

The main idea behind automatically interpreting the figurative meaning of an idiom is to instead interpret the literal meaning of its dictionary definition [61], [62]. For example, a dictionary definition of the idiom *live the life of Riley* is "a person who has a comfortable and enjoyable life, without having to make much effort." Most syllabi for English as a second language pay special attention to studying idioms [63], hence there is an abundance of teaching material, including dictionaries, dedicated specifically to the study of idioms. These readily available pedagogical resources can be utilized for the purpose of supporting automated interpretation of the figurative meaning of an idiom. In this study, we focus specifically on the interpretation of the underlying sentiment.

As reported in the previous sections, in our original study we collected a set of 580 emotionally charged idioms, including their definitions, from an educational web site—Learn English Today [64]. This resource was used to support the functionality described in this section. We originally obtained

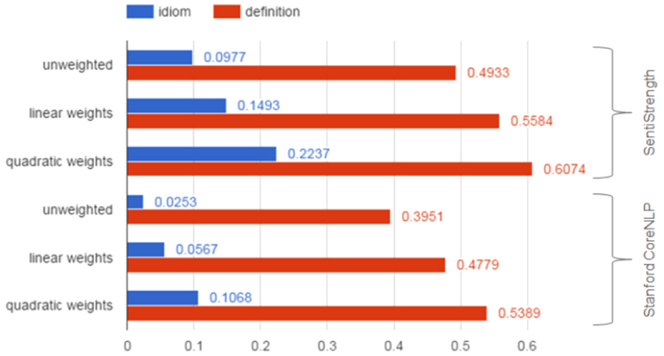

Fig. 3. Kappa agreement with the crowdsourced sentiment polarity annotations.

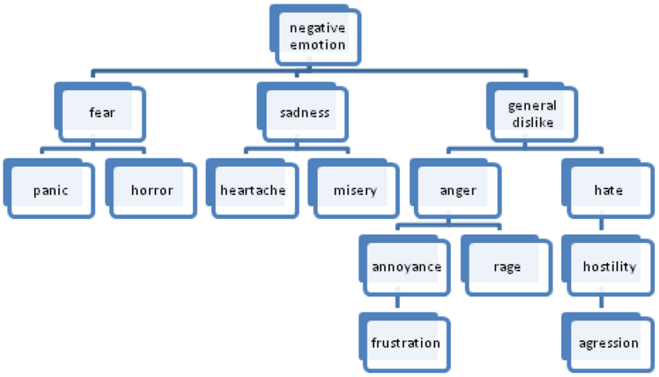

Fig. 4. An excerpt from the WordNet-Affect hierarchy.

sentiment polarities for the given set of idioms using a crowdsourcing approach. One of the goals of this study was to extract these polarities automatically from idiom definitions instead. We describe two approaches to this problem, one using off-the-shelf sentiment analysis tools and the other one based on mapping idiom definitions to WordNet-Affect, a hierarchy which includes a subset of WordNet synsets suitable to represent affective concepts such as moods and situations eliciting emotions or emotional responses [65].

3.2.1 Approach 1: Off-the-Shelf Sentiment Analysis

Off–the–shelf sentiment analysis tools struggle to identify sentiment conveyed by the figurative meaning of idioms. For example, in the absence of any positive or negative words in the idiom *live the life of Riley*, SentiStrength classifies its sentiment as neutral. However, if we apply the same sentiment analysis approach to its definition "a person who has a comfortable and enjoyable life, without having to make much effort," SentiStrength classifies its sentiment as positive based on the presence of two positive words, *comfortable* and *enjoyable*. Similarly, Stanford CoreNLP's sentiment annotator quantifies negative, neutral and positive sentiment of the idiom itself as 3, 77 and 20 respectively, but when applied to its definition the sentiment values change to 2, 5 and 93, thus correctly changing the sentiment classification from neutral to positive.

To reinforce the point about off–the–shelf sentiment analysis tools struggling to identify sentiment conveyed by the figurative meaning of idioms, we applied them to all 580 idioms and their definitions and compared the outcomes to the crowdsourced sentiment polarity annotations using inter–annotator agreement. The agreement was measured using three versions of Cohen's kappa coefficient [66]: simple unweighted, with linear weighting and with quadratic weighting. The kappa coefficient is calculated according to the following formula

$$\kappa = 1 - \frac{1 - p_o}{1 - p_e} \ , \tag{1}$$

where p_o is the observed agreement (i.e., the proportion of items on which both annotators agree) and p_e is the expected chance agreement calculated under the assumption that: (1) both annotators act independently, and (2) random assignment of annotation categories to items is governed by distribution of items across these categories. We report the values for the original kappa coefficient so that we can interpret the agreement on the following scale [67]: 0-0.20 (poor), 0.21-0.40 (fair), 0.41-0.60 (moderate), 0.61-0.80 (good), 0.81-1.00 (very good).

Cohen's kappa coefficient treats all disagreements equally, which is not suitable when the annotation categories are ordered as they indeed are: negative < neutral < positive. In such case, it is preferable to use weighted kappa coefficient [68], which accounts for the degree of disagreement by assigning different weights w_i to cases where annotations differ by i categories. If there are n categories, the weights can be calculated according to the following formulas for linear and quadratic weighting respectively

$$w_i = 1 - \frac{i}{n-1}, \quad w_i = 1 - \frac{i^2}{(n-1)^2}. \tag{2}$$

For example, for a total of 3 categories, linear weights would be set to 1, 0.5 and 0 when there is a difference of 0, 1 and 2 categories respectively, whereas the quadratic weights would be set to 1, 0.75 and 0. The weights are then used to multiply the corresponding proportion of disagreements in the observed matrix before calculating the kappa coefficient.

We provided the kappa values in Fig. 3, from which we can observe that the agreement with manual annotation increases by 0.4019 on average when sentiment analysis is applied to the definitions of the corresponding idioms. This improved the agreement from very poor to moderate. We can also notice that SentiStrength performed better than Stanford CoreNLP on this particular dataset.

3.2.2 Approach 2: Identifying Affective Concepts

WordNet is a lexical database of English nouns, verbs, adjectives and adverbs grouped together into sets of interlinked synonyms known as synsets [35]. WordNet-Affect [65] was created specifically as a lexical model for classifying affects, such as moods, situational emotions, or emotional responses, either directly (e.g., *joy*, *sad*, *happy*, etc.) or indirectly (e.g., *pleasure*, *hurt*, *sorry*, etc.). It was formed by aggregating a subset of WordNet synsets into an affect hierarchy (see Fig. 4). WordNet-Affect has been used as a lexical resource to support sentiment analysis studies, e.g., [69], [70], [71].

Our local version of the lexicon contains approximately 1,500 words including all derivational forms of the word senses originally found in WordNet-Affect. This resource enables a more sophisticated interpretation of the sentiment(s) associated with an idiom. In our approach, we represented

TABLE 4
Idioms Represented by Future Vectors

Idiom	negative emotion	sadness	misery	general dislike	anger	annoyance	hate	hostility
see red	1	0	0	1	1	1	0	0
bad blood	1	0	0	1	0	0	1	1
face like a wet weekend	1	1	1	0	0	0	0	0

each idiom using a vector whose features correspond to nodes in the WordNet-Affect hierarchy. For each non–negated mention of an affective word found in the idiom definition, the corresponding feature is set to 1 together with all other features that correspond to its ancestors. This approach ensures that hierarchical relationships between affects are translated into a flat vector representation.

For example, when interpreting the idiom *see red* using its definition "to suddenly become very angry or annoyed," two affective words are identified, *angry* and *annoyed*. As a result, the values corresponding to *negative emotion, general dislike, anger* and *annoyance* (see Fig. 4 for their hierarchical relationships) would be set to 1, whereas all other coordinates would remain zero. Similarly, when interpreting the idiom *bad blood* using its definition "intense hatred or hostility," two affective words are identified—*hatred* and *hostility*. As a result, the values corresponding to *negative emotion, general dislike, hate* and *hostility* (see Fig. 4 for their relationships) would be set to 1, whereas all other coordinates would remain zero. Finally, when interpreting the idiom *face like a wet weekend*, which based on its definition "to look sad and miserable," two affective words are identified—*sad* and *miserable*. As a result, the values corresponding to *negative emotion, sadness* and *misery* would be set to 1. We summarized these values in Table 4.

Generalization. Note that the vectors given in Table 4 are for illustrative purpose only and as such focus only on a small portion of the WordNet-Affect hierarchy. In practice, the length of the vector would match the size of the hierarchy, i.e., each coordinate would correspond to one of 278 nodes in the WordNet-Affect hierarchy. This leads to a relatively high dimensionality of the feature space, which may be associated with poorer classification performance. This problem is known as the curse of dimensionality (or Hughes effect) [72], where, given a fixed size of the training dataset, the predictive power of a machine learning algorithm reduces as the dimensionality increases. In order to reduce the number of features, we can exploit the structure of the WordNet-Affect hierarchy by simply projecting the original vectors onto a subspace that corresponds to the upper levels of the hierarchy, thereby selecting more general features. For example, focusing on two upper levels of the hierarchy shown in Fig. 4, we can simply remove the remaining features (shaded cells in Table 4) from the original vectors. One problem associated with this approach is that the WordNet-Affect hierarchy is unbalanced in the sense that the nodes at the same level may not be of the same generality, which may introduce issues of biased representation.

Clustering. Alternatively, we can use a data-driven approach to reduce the number of affective features. The given vector representation allows us to compare idioms to one another in terms of their affective content, e.g., by using cosine similarity measure

$$similarity(x,\ y) = \cos\theta = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2} \cdot \sqrt{\sum_{i=1}^{n} y_i^2}}, \tag{3}$$

where x and y are two non-zero vectors of dimensionality n and $\boldsymbol{\theta}$ is the angle between them. In general, the cosine similarity values range from -1 (corresponds to 180°, thus indicating opposite direction) to 1 (corresponds to 0°, thus indicating the same direction), where 0 indicates that the given vectors are orthogonal. In case of our vector representation, all vector components are always non–negative and so are the corresponding cosine similarity values. Therefore, in this special case the cosine similarity will range from 0 to 1 with higher values indicating higher similarity. In this representation, positive and negative affects will be orthogonal to one another. Going back to our examples (see Table 4), we can establish that *see red* and *bad blood* are more similar to each other ($similarity = 0.50$) than they are to *face like a wet weekend* ($similarity = 0.29$), because they share two features as a direct consequence of encoding hierarchical relationships from WordNet-Affect in a flat vector representation.

To visualize the similarity of affect between idioms, we applied multidimensional scaling to a distance matrix based on cosine similarity. The scatter plot shown in Fig. 5 shows a clear separation between idioms in terms of their affect. The first direction (along the x-axis) separates positive affects (to the left) from negative ones (to the right). The second direction (along the y-axis) separates anger (at the bottom) from anxiety (at the top). A clustering algorithm can be used to identify clusters of related idioms. Table 5 illustrates the results of applying k-means clustering ($k = 10$). In principle, these clusters can be mapped to affects as we indicated in Table 5. Nonetheless, uncategorized clusters can still be used as affective features to support sentiment analysis. The dimensionality of the problem can be controlled by limiting the number of clusters.

Mapping Affects to Sentiment Polarities. Either of the two approaches, generalization or clustering, can be used to support the extraction of affective aspects of idiom-based features. However, to support compatibility with the original study so that its results can be used as a baseline, we need to map affects to sentiment polarities and for this we used the former approach. A total of 421 idiom definitions were successfully mapped onto affects and then generalized into sentiment polarities. The results were then compared to crowdsourced annotations and the following values were recorded for the three versions of kappa coefficient: 0.5851 (unweighted), 0.6770 (with linear weights) and 0.7450 (with quadratics weights). These values are much higher than the ones achieved by off–the–shelf sentiment analysis tools (see Fig. 3). However, 159 out of 580 idioms definitions (i.e., 27 percent) remained unclassified as they did not contain non–negated mentions of affective words listed in WordNet-Affect. To take advantage of both approaches to sentiment polarity classification (described in Sections 3.2.1 and 3.2.2 respectively), their results were combined (see Fig. 6). We applied SentiStrength, which performed better than Stanford CoreNLP's sentiment annotator on these data (see Fig. 3), to the set of 159 idiom definitions that remained

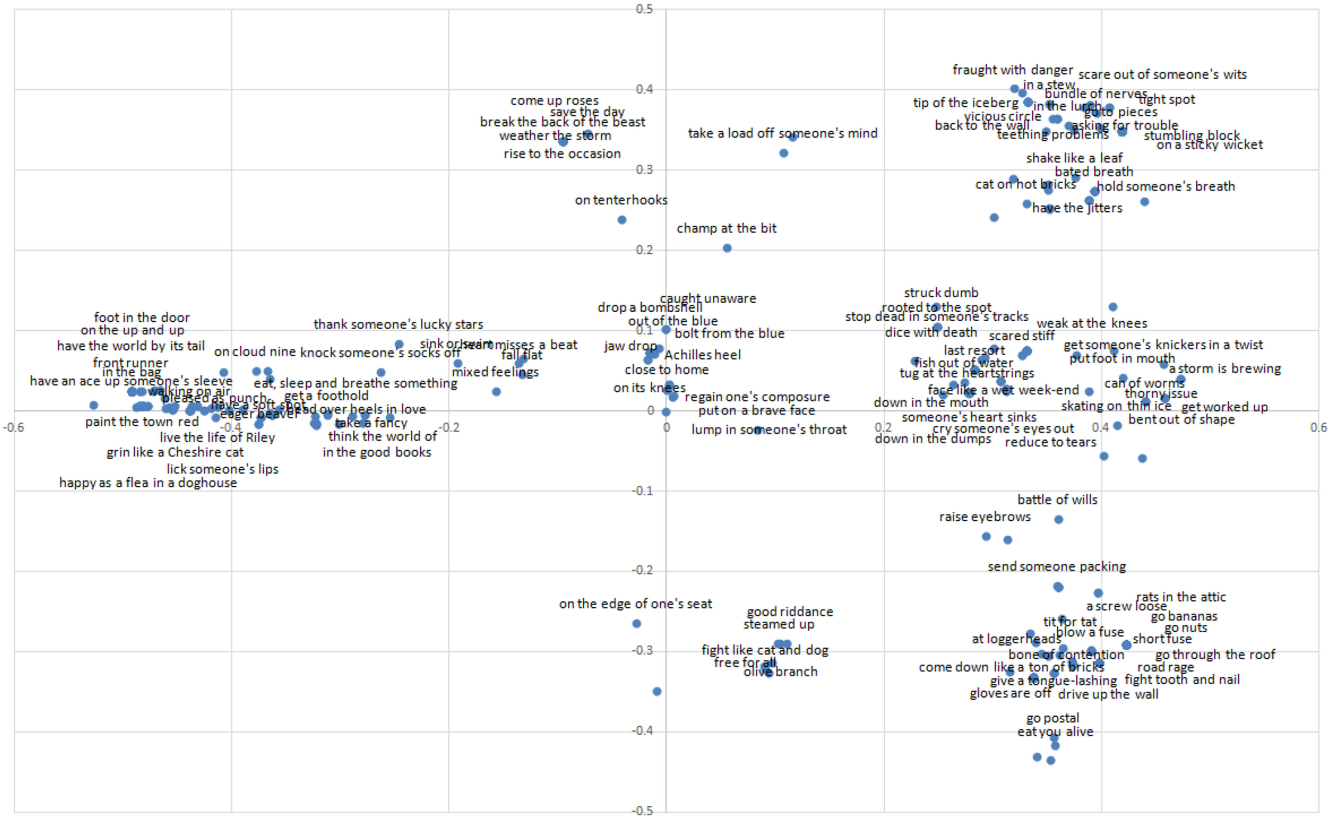

Fig. 5. Multidimensional scaling results.

TABLE 5
Clustering Results

Cluster	Interpretation	Members	Size
1	surprise	bolt from the blue; drop a bombshell; jaw drop; mixed feelings; knock down with feather	28
2	frustration	get someone's knickers in a twist; fish out of water; groan inwardly; put foot in mouth; slip through fingers	81
3	relief	take a load off someone's mind; break the back of the beast; come up roses; save the day; weather the storm	11
4	affection	eat, sleep and breathe something; have a soft spot; on cloud nine; knock someone's socks off; in the good books	30
5	anxiety	break out in a cold sweat; cat on hot bricks; shake like a leaf; alarm bells ringing; cloud on the horizon	89
6	happiness	lick someone's lips; pleased as punch; live the life of Riley; grin like a Cheshire cat; walking on air	24
7	excitement	have a ball; have a whale of a time; paint the town red; over the moon; in seventh heaven	22
8	anger	come down like a ton of bricks; fly off the handle; go through the roof; hot under the collar: see red	100
9	satisfaction	bear fruit, reach first base, foot in the door, place in the sun, have the world by its tail	42
10	contempt	fight like cat and dog, good riddance, steamed up, fit of pique, free for all	10

unclassified after using the WordNet-Affect approach. The overall results were compared to crowdsourced annotations and the following values were recorded for the three versions of kappa coefficient: 0.5062 (unweighted), 0.5802 (with linear weights) and 0.6409 (with quadratics weights). The resulting sentiment polarity values were used to replace crowdsourced sentiment polarity annotations in the original sentiment analysis system described in Fig. 1.

4 RESULTS

We re-used the gold standard dataset from the original study [8] to perform evaluation experiments. Table 6 summarizes the methods M1–M6 whose performance we wanted to compare. The main goal of this study was to

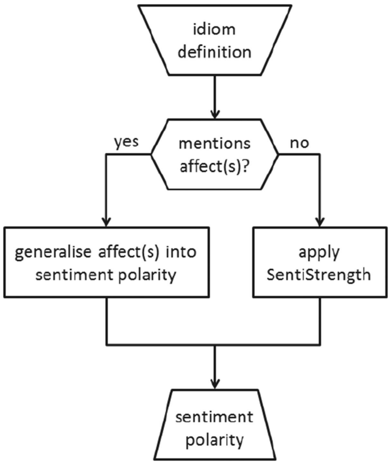

Fig. 6. A combined approach to sentiment polarity classification.

TABLE 6
Sentiment Analysis Methods

ID	Method	Features	Supporting resources
M1	SentiStrength [10], [11]	words	sentiment lexicon
M2	supervised learning [8]	sentiment polarities idioms	SentiStrength crowdsourcing
M3	supervised learning (this study)	sentiment polarities idioms	SentiStrength idiom dictionary
M4	Stanford CoreNLP [12]	phrases of variable length	sentiment treebank
M5	supervised learning [8]	sentiment polarities idioms	Stanford CoreNLP crowdsourcing
M6	supervised learning (this study)	sentiment polarities idioms	Stanford CoreNLP idiom dictionary

TABLE 7
Confusion Matrices

M1	pos	∅	neg	M2	pos	∅	neg
pos	40	62	36	pos	102	18	13
∅	22	72	30	∅	39	49	36
neg	31	96	111	neg	24	44	170
M3	pos	∅	neg	M4	pos	∅	neg
pos	75	18	45	pos	41	23	74
∅	41	26	57	∅	19	19	86
neg	32	38	168	neg	25	43	170
M5	pos	∅	neg	M6	pos	∅	neg
pos	104	10	24	pos	64	13	61
∅	46	20	58	∅	24	13	87
neg	35	22	181	neg	33	15	190

Rows and columns correspond to actual and predicted sentiment respectively.

investigate whether the results of sentiment analysis enriched with idiom-based features are comparable when crowdsourcing of the supporting lexico-semantic resources is replaced by a fully automated approach (by comparing M2 versus M3 and M5 versus M6). In expectation that a fully automated approach may underperform in comparison to manually crafted features, we also wanted to investigate whether the idiom-based approach would still outperform the original baseline methods, which do not incorporate idioms as features (by comparing M1 versus M3 and M4 versus M6). The classification performance was evaluated in terms of F-measure (see Fig. 7) based on confusion matrices given in Table 7. As expected, when manually crafted lexico-semantic resources were replaced by automatically generated ones, the performance dropped by 10.6 (M2 versus M3) and 7.6 (M5 versus M6) percentage points. However, the use of automatically generated lexico-semantic resources still improves the performance of the original sentiment analysis methods by 9.0 (M1 versus M3) and 7.4 (M4 versus M6) percentage points.

A closer inspection of the confusion matrices (see Table 7) reveals that the use of idioms as features, either manually or automatically engineered, improves the sensitivity with respect to positive and negative polarities. However, automatically engineered features are more biased towards negative polarities (see the last column in Table 7). This may be explained by the way in which the idiom polarities were encoded. The crowdsourced idiom polarities allowed for fuzzy representation by means of distributing the number of annotations across the available options: positive, negative and other. For example, the idiom *mind someone's own business* was originally annotated as pos:0 ∅:60 neg:40, thereby allowing different interpretations of the given idiom. On the other hand, the automatically extracted idiom polarities do not support fuzzy representation. For example, the same idiom was represented as pos:0 ∅:0 neg:100, which indicates that the given idiom is strictly negative.

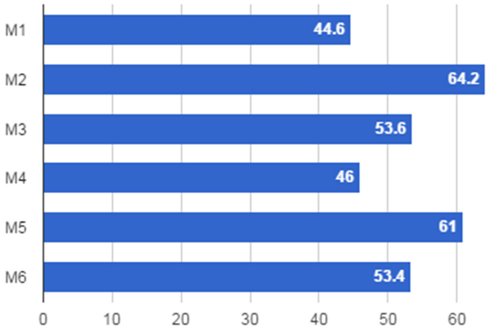

Fig. 7. Evaluation results using F-measure.

This may be remedied by incorporating the notion of ambiguity and/or intensity into idiom polarity representation. Off–the–shelf sentiment analysis tools used in Section 3.2.1 output the strength of both positive and negative sentiment, which can be used to support fuzzy representation of automatically extracted sentiment polarities. To what extent this could improve the performance of methods M3 and M6 will be the subject of future work. Nonetheless, the experiments conducted in this study confirm its main hypothesis that automatically engineered idiom-based features do improve the results of sentiment analysis.

5 Conclusions

We demonstrated that automatically engineered idiom-based features improve sentiment analysis results. The overall performance in terms of F-measure was improved from 45 to 54 percent in one experiment and from 46 to 53 percent in the other. The results are still poorer than the ones achieved using manually engineered features, which improved the baseline sentiment analysis results from 45 to 64 percent in one experiment and from 46 to 61 percent in the other. However, the advantage of the new approach is that it is more general in a sense that it can readily utilize an arbitrary list of idioms in sentiment analyses. It can also be used to support sentiment analysis that focuses on a full range of emotions (see Section 3.2.2) and not merely sentiment polarity. More importantly, the performance of a fully automated approach to using idioms in sentiment analysis can still be improved.

First, as we pointed out in Section 4, the fuzzy representation of idiom polarities may reduce misclassification of less strongly polarized idiom examples. Second, to support compatibility with the original study [8] we re-used the same supervised learning method—a Bayesian network classifier, which outperformed alternative machine learning algorithms in cross-validation experiments performed on the training data in which the idiom-based features were encoded based on manually engineered lexico-semantic resources. According to the "no free lunch" theorem, any two learning algorithms are equivalent when their performance is averaged across all possible problems [21]. In other words, there is no universally best learning algorithm, which suggests that the choice of an appropriate algorithm should be based on its performance for the particular problem at hand and the properties of data that characterize the problem. The fact that the distribution of the training data has changed by replacing manually engineered features with automatically engineered ones opens a possibility of another machine learning algorithm producing a better classification model. These two hypotheses are outside of the scope of this study and will be the subject of future work. In this study we demonstrated that we can (1) fully automate the use of idioms in sentiment analysis, (2) minimize the knowledge engineering bottleneck associated with this task and (3) still improve the performance of the baseline sentiment analysis approaches.

Acknowledgments

LW gratefully acknowledges the support of the Engineering and Physical Sciences Research Council (ref: 1511905). We are thankful to Kathleen Beke, the creator of the web site Learn English Today, for kindly letting us use their data. We would also like to thank David Bevan for proof reading the manuscript.

References

[1] A. Wray, *Formulaic Language and the Lexicon*. Cambridge, U.K.: Cambridge Univ. Press, 2002.

[2] A. Buerki, "Formulaic sequences: A drop in the ocean of constructions or something more significant?" *Eur. J. English Studies*, vol. 20, pp. 15–34, 2016.

[3] J. Bybee, "From usage to grammar: The mind's response to repetition," *Language*, vol. 82, pp. 711–733, 2006.

[4] C. Manning and H. Schütze, *Foundations of Statistical Natural Language Processing*. Cambridge, MA, USA: MIT Press, 1999.

[5] I. A. Sag, T. Baldwin, F. Bond, A. Copestake, and D. Flickinger, "Multiword expressions: A pain in the neck for NLP," in *Proc. 3rd Int. Conf. Comput. Linguistics Intell. Text Process.*, 2002, pp 1–15.

[6] A. Villavicencio, A. Copestake, B. Waldron, and F. Lambeau, "Lexical encoding of MWEs," in *Proc. Workshop Multiword Expressions: Integrating Process.*, 2004, pp. 80–87.

[7] A. Makkai, *Idiom Structure in English*. The Hague, The Netherlands: Mouton, 1972.

[8] L. Williams, C. Bannister, M. Arribas-Ayllon, A. Preece, and I. Spasić, "The role of idioms in sentiment analysis," *Expert Syst. Appl.*, vol. 42, pp. 7375–7385, 2015.

[9] B. Liu, *Sentiment Analysis: Mining Opinions, Sentiments and Emotions*. Cambridge, U.K.: Cambridge Univ. Press, 2015.

[10] M. Thelwall, "SentiStrength," 2014. [Online]. Available: http://sentistrength.wlv.ac.uk/

[11] M. Thelwall, K. Buckley, G. Paltoglou, D. Cai, and A. Kappas, "Sentiment strength detection in short informal text," *J. Am. Soc. Inf. Sci. Technol.*, vol. 61, pp. 2544–2558, 2010.

[12] R. Socher, et al., "Recursive deep models for semantic compositionality over a Sentiment Treebank," in *Proc. Conf. Empirical Methods Natural Language Process.*, 2013, pp. 1631–1642.

[13] G. Nunberg, I. A. Sag, and T. Wasow, "Idioms," *Language*, vol. 70, pp. 491–538, 1994.

[14] R. Moon, "Frequencies and forms of phrasal lexemes in English," in *Phraseology: Theory, Analysis and Applications*, A. P. Cowie, Ed. Oxford, U.K.: Oxford Univ. Press, 1998, pp. 79–100.

[15] L. E. Grant, "Frequency of 'core idioms' in the British National Corpus (BNC)," *Int. J. Corpus Linguistics*, vol. 10, pp. 429–451, 2005.

[16] Y. Li, J. Yosinski, J. Clune, H. Lipson, and J. Hopcroft, "Convergent learning: Do different neural networks learn the same representations?" in *Proc. 1st NIPS Int. Workshop Feature Extraction: Modern Questions Challenges*, 2015, pp. 196–212.

[17] L. Price and M. Thelwall, "The clustering power of low frequency words in academic Webs," *J. Assoc. Inf. Sci. Technol.*, vol. 56, pp. 883–888, 2005.

[18] P. Schönhofen and A. A. Benczúr, "Exploiting extremely rare features in text categorization," in *Proc. 17th Eur. Conf. Mach. Learn.*, 2006, pp. 759–766.

[19] T. Damoulas and M. A. Girolami, "Combining feature spaces for classification," *Pattern Recognit.*, vol. 42, pp. 2671–2683, 2009.

[20] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The Weka data mining software: An update," *ACM SIGKDD Explorations Newsletter*, vol. 11, pp. 10–18, 2009.

[21] D. H. Wolpert, "The lack of a priori distinctions between learning algorithms," *Neural Comput.*, vol. 8, pp. 1341–1390, 1996.

[22] D. Feng, S. Besana, and R. Zajac, "Acquiring high quality non-expert knowledge from on-demand workforce," in *Proc. Workshop People's Web Meets NLP: Collaboratively Constructed Semantic Resources*, 2009, pp. 51–56.

[23] M. Poesio, J. Chamberlain, U. Kruschwitz, L. Robaldo, and L. Ducceschi, "Phrase detectives: Utilizing collective intelligence for internet-scale language resource creation," *ACM Trans. Interactive Intell. Syst.*, vol. 3, 2013, Art. no. 3.

[24] S. M. Mohammad and P. D. Turney, "Crowdsourcing a word–emotion association lexicon," *Comput. Intell.*, vol. 29, pp. 436–465, 2013.

[25] O. Alonso, D. E. Rose, and B. Stewart, "Crowdsourcing for relevance evaluation," *ACM SIGIR Forum*, vol. 42, pp. 9–15, 2008.

[26] G. Kazai, N. Milic-Frayling, and J. Costello, "Towards methods for the collective gathering and quality control of relevance assessments," in *Proc. 32nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2009, pp. 452–459.

[27] R. Snow, B. O'Connor, D. Jurafsky, and A. Y. Ng, "Cheap and fast - but is it good? Evaluating non-expert annotations for natural language tasks," in *Proc. Conf. Empirical Methods Natural Language Process.*, 2008, pp. 254–263.

[28] V. Hatzivassiloglou and K. R. McKeown, "Predicting the semantic orientation of adjectives," in *Proc. 35th Annu. Meeting Assoc. Comput. Linguistics Eighth Conf. Eur. Chapter Assoc. Comput. Linguistics*, 1997, pp. 174–181.

[29] P. Turney and M. Littman, "Measuring praise and criticism: Inference of semantic orientation from association," *ACM Trans. Inf. Syst.*, vol. 21, pp. 315–346, 2003.

[30] M. Taboada, C. Anthony, and K. Voll, "Methods for creating semantic orientation dictionaries," in *Proc. 5th Int. Conf. Language Resources Eval.*, 2006, pp. 427–432.

[31] W. Du, S. Tan, X. Cheng, and X. Yun, "Adapting information bottleneck method for automatic construction of domain-oriented sentiment lexicon," in *Proc. 3rd ACM Int. Conf. Web Search Data Mining*, 2010, pp. 111–120.

[32] H. Takamura, T. Inui, and M. Okumura, "Latent variable models for semantic orientations of phrases," in *Proc. 11th Conf. Eur. Chapter Assoc. Comput. Linguistics*, 2006, pp. 201–208.

[33] T. Wilson, J. Wiebe, and P. Hoffmann, "Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis," *Comput. Linguistics*, vol. 35, pp. 399–433, 2009.

[34] N. Malandrakis, A. Potamianos, E. Iosif, and S. Narayanan, "Distributional semantic models for affective text analysis," *IEEE Trans. Audio Speech Language Process.*, vol. 21, no. 11, pp. 2379–2392, Nov. 2013.

[35] G. A. Miller, "WordNet: A lexical database for English," *Commun. ACM*, vol. 38, pp. 39–41, 1995.

[36] S.-M. Kim and E. Hovy, "Determining the sentiment of opinions," in *Proc. 20th Int. Conf. Comput. Linguistics*, 2004, Art. no. 1367.

[37] J. Kamps, M. Marx, R. J. Mokken, and M. D. Rijke, "Using WordNet to measure semantic orientations of adjectives," in *Proc. 4th Int. Conf. Language Resources Eval.*, 2004, pp. 1115–1118.

[38] A. Hassan and D. Radev, "Identifying text polarity using random walks," in *Proc. 48th Annu. Meeting Assoc. Comput. Linguistics*, 2010, pp. 395–403.

[39] E. C. Dragut, C. Yu, P. Sistla, and W. Meng, "Construction of a sentimental word dictionary," in *Proc. 19th ACM Int. Conf. Inf. Knowl. Manag.*, 2010, pp. 1761–1764.

[40] Y. Lu, M. Castellanos, U. Dayal, and C. Zhai, "Automatic construction of a context-aware sentiment lexicon: An optimization approach," in *Proc. 20th Int. Conf. World Wide Web*, 2011, pp. 347–356.

[41] A. Osherson and C. Fellbaum, "The representation of idioms in WordNet," in *Proc. 5th Global WordNet Conf.*, 2010.

[42] A. Andreevskaia and S. Bergler, "Mining WordNet for a fuzzy sentiment: Sentiment tag extraction from WordNet glosses," in *Proc. 11th Conf. Eur. Chapter Assoc. Comput. Linguistics*, 2006, pp. 209–216.

[43] A. Esuli and F. Sebastiani "Determining the semantic orientation of terms through gloss classification," in *Proc. 14th ACM Int. Conf. Inf. Knowl. Manag.*, 2005, pp. 617–624.

[44] H. Takamura, T. Inui, and M. Okumura, "Extracting semantic orientations of phrases from dictionary," in *Proc. Annu. Conf. North American Chapter Assoc. Comput. Linguistics*, 2007, pp. 292–299.

[45] A. Fazly, P. Cook, and S. Stevenson, "Unsupervised type and token identification of idiomatic expressions," *Comput. Linguistics*, vol. 35, pp. 61–103, 2009.

[46] C. Sporleder and L. Li, "Unsupervised recognition of literal and non-literal use of idiomatic expressions," in *Proc. 12th Conf. Eur. Chapter Assoc. Comput. Linguistics*, 2009, pp. 754–762.

[47] L. Li and C. Sporleder, "Using Gaussian Mixture models to detect figurative language in context," in *Proc. Annu. Conf. North American Chapter Assoc. Comput. Linguistics*, 2010, pp. 297–300.

[48] A. Feldman and J. Peng, "Automatic detection of idiomatic clauses," in *Proc. 14th Int. Conf. Comput. Linguistics Intell. Text Process.*, 2013, pp. 435–446.

[49] J. Peng and A. Feldman, "Automatic idiom recognition with word embeddings," in *Proc. Annu. Int. Symp. Inf. Manage. Big Data*, 2016, pp. 17–29.

[50] G. D. Salton, R. J. Ross, and J. D. Kelleher, "'Idiom token classification using sentential distributed semantics," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, 2016, pp. 194–204.

[51] J. Brooke, V. Tsang, and G. H. F. Shein, "Unsupervised multiword segmentation of large corpora using prediction-driven decomposition of n-grams," in *Proc. 25th Int. Conf. Comput. Linguistics*, 2014, pp. 753–761.

[52] N. Schneider, E. Danchik, C. Dyer, and N. A. Smith, "Discriminative lexical semantic segmentation with gaps: Running the MWE gamut," *Trans. Assoc. Comput. Linguistics*, vol. 2, pp. 193–206, 2014.

[53] R. E. Vega-Moreno, *Creativity and Convention: The Pragmatics of Everyday Figurative Speech*. Amsterdam, The Netherlands: John Benjamins Publishing Company, 2007.

[54] R. Moon, *Fixed Expressions and Idioms in English: A Corpus-Based Approach*. Oxford, U. K.: OUP, 1998.

[55] A. Langlotz, *Idiomatic Creativity: A Cognitive-Linguistic Model of Idiom-Representation and Idiom-Variation in English*. Amsterdam, The Netherlands: John Benjamins, 2006.

[56] K. Dutton, *Exploring the Boundaries of Formulaic Sequences: A Corpus-Based Study of Lexical Substitution and Insertion in Contemporary British English*. Saarbrücken, Germany: VDM Verlag, 2009.

[57] S. Z. Riehemann, "A constructional approach to idioms and word formation," PhD thesis, Dept. Linguistics, Stanford Univ., Stanford, CA, 2001.

[58] S. Sekine and K. Dalwani, "Ngram search engine with patterns combining token, POS, chunk and NE information," in *Proc. 7th Int. Conf. Language Res. Eval.*, 2010.

[59] Z. Huang, W. Xu, and K. Yu, "Bidirectional LSTM-CRF models for sequence tagging," arXiv preprint arXiv:1508.01991, 2015.

[60] M. van Gompel and A. van den Bosch, "Efficient *n*-gram, skipgram and flexgram modelling with Colibri Core," *J. Open Res. Softw.*, vol. 4, 2016, Art. no. e30.

[61] R. Verma and V. Vuppuluri, "A new approach for idiom identification using meanings and the Web," in *Proc. 10th Int. Conf. Recent Advances Natural Language Process.*, 2015, pp. 681–687.

[62] C. Liu and R. Hwa, "Phrasal substitution of idiomatic expressions," in *Proc. 15th Annu. Conf. North American Chapter Assoc. Comput. Linguistics: Human Language Technol.*, 2016, pp. 363–373.

[63] D. Liu, "The most frequently used spoken American English idioms: A corpus analysis and its implications," *TESOL Quarterly*, vol. 37, pp. 671–700, 2003.

[64] Learn English Today, 2013. [Online]. Available: http://www.learn-english-today.com/idioms/idioms_proverbs.html

[65] A. Valitutti, C. Strapparava, and O. Stock, "Developing affective lexical resources," *Psychology*, vol. 2, pp. 61–83, 2004.

[66] J. Cohen, "A coefficient of agreement for nominal scales," *Edu. Psychological Meas.*, vol. 20, pp. 37–46, 1960.

[67] D. G. Altman, *Practical Statistics for Medical Research*. London, U.K.: Chapman and Hall/CRC, 1990.

[68] J. Cohen, "Weighted Kappa: Nominal scale agreement with provision for scaled disagreement or partial credit," *Psychological Bulletin*, vol. 70, pp. 213–220, 1968.

[69] C. Strapparava and R. Mihalcea, "Learning to identify emotions in text," in *Proc. ACM Symp. Applied Comput.*, 2008, pp. 1556–1560.

[70] A. Balahur, et al., "Sentiment analysis in the news," in *Proc. Int. Conf. Language, Resources Eval.*, 2010.

[71] I. Spasić, P. Burnap, M. Greenwood, and M. Arribas-Ayllon, "A naïve Bayes approach to topic classification in suicide notes," *Biomed. Inf. Insights*, vol. 5, pp. 87–97, 2012.

[72] G. F. Hughes, "On the mean accuracy of statistical pattern recognizers," *IEEE Trans. Inf. Theory*, vol. 14, no. 1, pp. 55–63, Jan. 1968.

Irena Spasić received the PhD degree in computer science from the University of Salford, United Kingdom, in 2004. Following posts with 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 with Cardiff University and is a co-founder of the UK Healthcare Text Analytics Research Network (HealTex).

Lowri Williams received the BSc degree in business information systems, in 2013. She is currently working toward the PhD degree in computer science at Cardiff University, United Kingdom. Her research revolves around the topic of sentiment analysis, paying particular attention to the effects of idioms on such technologies.

Andreas Buerki received the PhD degree in general linguistics from the University of Basel, in 2013. He is currently a lecturer in linguistics with Cardiff University, specialising in phraseology and corpus linguistics as well as quantitative approaches to linguistic structure and language change. He is a member of the advisory council of the European Society of Phraseology.