Diego Reforgiato Recupero
Department of Computer Science, University of Cagliari, Cagliari, ITALY

Mehwish Alam
FIZ Karlsruhe—Leibniz Institute for Information Infrastructure, Karlsruhe, GERMANY

Davide Buscaldi
Laboratoire d'Informatique de Paris-Nord, Université Paris 13, Paris, FRANCE

Aude Grezka
Laboratoire d'Informatique de Paris-Nord, Université Paris 13, Paris, FRANCE

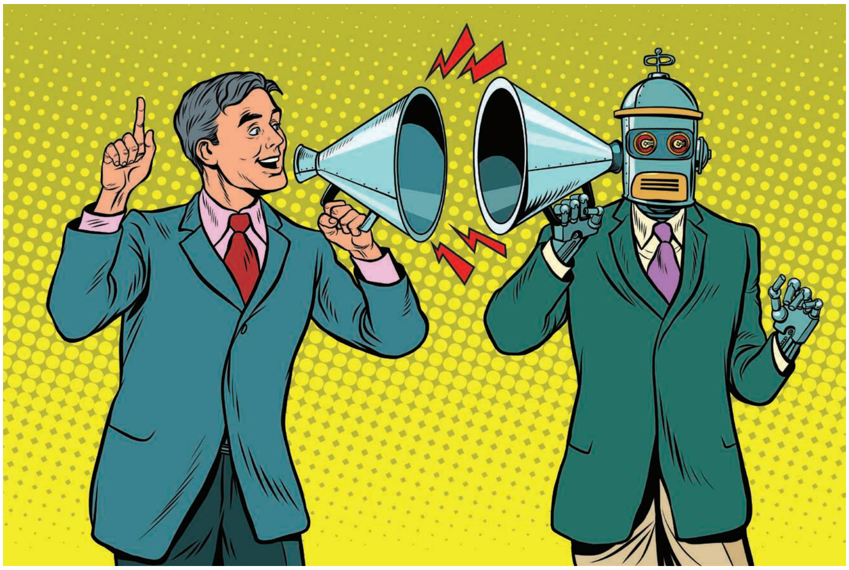
Farideh Tavazoee
Department of Economic and Business Science, University of Cagliari, Cagliari, ITALY

Frame-Based Detection of Figurative Language in Tweets

Abstract

This paper analyzes the problem of figurative language detection on social media, with a focus on the use of semantic features for identifying irony and sarcasm. Framester, a novel resource that acts as a hub between FrameNet, WordNet, VerbNet, BabelNet, DBpedia, Yago, DOLCE-Zero and others, has been used to extract semantic features from text. These semantic features are used to enrich the representations of tweets with event information using frames and word senses in addition to lexical units. The data set used for experimentation purposes contains tweets taken from different corpora including both figurative (containing irony and sarcasm) and non-figurative language. Two major tasks were performed: (i) detecting figurative language in tweets in a dataset containing both figurative and non-figurative tweets, (ii) classifying tweets containing irony and sarcasm. A 10-fold cross-validation experiment shows that the obtained accuracy for both tasks increases significantly when the semantic features such as linguistic frames and word senses are used in addition to lexical units, indicating that they may be important clues for figurative language. The approach was developed on top of Apache Spark so that it is easily scalable to much higher volumes of data, allowing for real-time analysis.

Digital Object Identifier 10.1109/MCI.2019.2937614
Date of current version: 14 October 2019



1. Introduction

The use of figurative language is rapidly growing in online textual resources including product reviews and social media, making Sentiment Analysis (SA) [1]–[4] one of the most important and trending research problems. SA often involves assessing the polarity (positive or negative) of an opinion holder on a particular topic within some text. Furthermore, sentiment polarity may not be able to fully express the affective meaning (e.g., emotion) that writers want to give to an object or to any of its related features: therefore, there is a need for a stronger and more effective evaluation that is able to show writers' emotions in text. The task of emotion detection is to identify the emotion conveyed within a text (e.g., happiness, sadness, anger, etc.). Opinion mining and emotion detection are important for a variety of applications and industries, from helping

Corresponding Author: Diego Reforgiato Recupero (diego.reforgiato@unica.it)

advertisers and content creators in selling their products more effectively to gauge the emotions of autistic children. Among these tools, those specialized in automatic language processing, and in particular the analysis of feelings and emotions, have been developed to quickly identify the feelings arising from the huge number of Internet users. These comments freely produced are strategic for companies and represent a true source of information.

However, within the area of SA and emotion detection, there are still unresolved challenges. The presence of indirect language containing figurative expressions (such as irony or sarcasm), especially on social media, complicates the task, where a positive statement is usually used to convey negative meanings [5].

Irony is a heterogeneous concept, which complicates the study of the processes necessary for detection. This heterogeneity is illustrated by the difficulty of differentiating irony from sarcasm: irony is often conflated with sarcasm. Verbal irony expresses a contradiction between the speaker's thought and its expression, i.e., an utterance in which what is said is different from what is signified. Irony can be produced in different ways, some of which correspond to figures of speech (antiphrasis, hyperbole, understatement, etc.). The content of the ironic message can be positive or negative. Moreover, the prosody can also modulate this message.

Sarcasm is a form of verbal irony, but it is insulting, intentionally. Unlike irony, sarcasm is more difficult to detect. Indeed, to be sarcastic is to say the opposite of what one thinks. Although sarcasm is considered a "biting irony", it has several predicates of definition that are not present in irony, such as insult, malice, cruelty or aggression.

Figurative language is one of the most difficult topics within Natural Language Processing (NLP). It takes the advantage of linguistic expressions such as metaphor, analogy, ambiguity, irony in order to project more complex meaning which usually represents a real challenge even for humans [6], [7]. Such language patterns make the current techniques for SA obsolete by creating an obstacle for these systems using direct approaches (making use of statistical classifiers and lexical and semantic resources) which are not adequate for indirect meanings of certain statements.

The following example sentence from Twitter illustrates the type of linguistic elements that underlie irony:

The irony of Chelsea fans moaning
about China & all their money buying
all those players and ruining football. (1)

In the above sentence, there is a strong ironic flavor for Chelsea fans complaining for China investing a lot in European soccer when Chelsea, a famous English soccer team, yearly spends a lot to buy very expensive top players. The reason why current SA methods are not appropriate for this kind of statements is because the occurrences of words such as *moan* are used in the opposite meaning. Without the background knowledge, the topic *China* would have been negative and *Chelsea fans* the holder of the opinion. If a system successfully detects the irony within that sentence, the polarity changes and the topic of the new negative opinion becomes *Chelsea fans*. If some words express a positive feeling (*great, good, nice,* etc.) or negative (*unacceptable, disappointed, irritated,* etc.), identifying their presence in a statement is not enough to define the tone of this statement.

Language is comprised of different levels of articulations (lexical, syntactic, semantic, pragmatic), and each level comes with its own difficulties. On the lexical level, the textual data are subjected to particular orthographic forms. Spelling mistakes or typing errors, which are frequent in forums and social media, complicate the automatic analysis of a text. This multiplication of orthographic forms makes recognition of lexical units for the analysis of feelings more difficult. At the syntactic level, information is in the form of natural language, so the analyzer can be confronted with heterogeneous syntactic forms. These forms do not always correspond to the usual grammatical norms. The language used in tweets is spontaneous. Many times the structure of sentence is also modified (absence of verbs, incomplete sentences, etc.). At the semantic level, the main difficulty related to semantics is the polysemy of words, which can make any analysis of meaning ambiguous and create misunderstandings. Finally, the pragmatic level implies general knowledge of the context of the situation.

This paper deals with the problem of detecting tweets containing figurative language expressions from a wide set of tweets (both figurative and non-figurative) by using a supervised approach leveraging their semantic information. We focus on tweets written in English. More particularly, we employ the following:

- ❑ **Frame Detection** was performed for extracting event based information using a recently published framework Word Frame Disambiguation[1] [8], which uses a newly published linguistic linked data resource Framester using frame semantics as its core,
- ❑ **Word Sense Disambiguation** is performed using Babelfy [9] for overcoming the problem of polysemic words (i.e., one word having different meanings).

The data set used for experimentation purposes contains tweets taken from *SemEval-2015 Task 10: SA in Twitter* and *SemEval-2015 Task 11: SA of Figurative Language in Twitter* which we have merged together for creating a dataset consisting of figurative tweets (containing irony and sarcasm) and non-figurative tweets. Irony Sarcasm Analysis Corpus [10] has been used for performing another set of experiments. As a first step, the input datasets were extended by including Frames and BabelNet synsets (BN). As a second step, two major tasks were performed: (i) figurative language detection, (ii) irony or sarcasm detection. A 10-fold cross-validation shows that the obtained accuracy for both tasks increases significantly when using semantic features. Results thus show that the proposed approach considering words' semantics for this task of SA surpasses syntactic approaches, which form the baseline for the considered datasets. One of the major problems encountered while

[1]http://lipn.univ-paris13.fr/framester/en/wfd_html/

dealing with tweets is the enormity of the data leading to the Big Data solutions. For solving this issue, our approach uses Apache Spark and its Machine learning Library (MLlib).

The remainder of this paper is organized as follows: Section 2 describes state-of-the-art work within the domain of irony, sarcasm, and figurative language detection. Section 3 gives a brief introduction to the methodologies used by the proposed approach for extracting semantic features. Section 4 presents the proposed framework. Section 5 introduces the datasets used for tests and how they have been obtained and processed, while, Section 6 shows the experimental setup and the obtained results. Finally, Section 7 concludes the paper and discusses future directions.

Figurative language such as irony complicates opinion mining, attributing a negative meaning to positive statements. This work shows that semantic frames may be important clues in determining the subjectivity of a text, important for figurative language detection.

2. Related Work

With the adoption of social networks, the users post their comments, opinions and emotions on-line. These trends breed new challenges [11], [12] and opportunities for analyzing their text in order to detect sentiments and emotions [1]–[4], [13].

2.1. Hybrid Approaches

These approaches include statistical methods combined with knowledge bases. [1] is one of the first methods to target the problem of SA using statistical approaches on top of pre-processed textual data. Later on, the rapid growth of Semantic Web techniques has greatly affected the SA methods by improving the results over classical statistical approaches taking into account the semantics [14]. For example, in [2], [3], [15] the authors proposed an approach based on the neo-Davidsonian assumption that events and situations are the primary entities for contextualizing opinions. This allowed distinguishing holders, main topics, and sub-topics of an opinion by employing a machine reader tool that leverages NLP and Knowledge Representation components jointly with cognitively-inspired frames. Another example that leverages frame semantics and lexical resources within the SA was published in [16], where authors employed frame semantics and conceptual information (BabelNet synsets) detected by Framester to extract semantic features from social media for polarity detection showing a remarkable improvement in F-measure when using semantics. This paper extends the previous study for the figurative language detection problem within social media. The above mentioned studies employing semantics use *Sentic Computing* [17] techniques to bridge the gap between statistical NLP and linguistics, common sense reasoning, and affective computing; furthermore, they enable the analysis of text not only at document, page or paragraph level, but also at sentence, clause and concept level.

2.2. Methods for Irony Detection

Irony and its theory about its usage in human interaction have been discussed in detail as a sophisticated and complex mode of communication [18], where irony markers and motivations that speakers have for using irony are indicated. Usually, reasons for using irony lie in its social and rhetorical functions whereas the function of markers of irony is to make its processing simpler. Several attempts for targeting this problem have been made and, in particular, a challenge, related to the polarity detection of tweets containing figurative language has been presented at the prestigious SemEval 2015 workshop[2]. 15 teams participated in the task and a total of 35 runs have been submitted. The best reported system achieved the score of 0.758 using the Cosine Similarity measure, and a score of 2.117 using the MSE. The score of each system ranged from 0.059 to 0.758 using cosine similarity and from 11.274 to 2.117 using MSE.

A similar challenge for irony polarity detection has been proposed for the Italian language at SENTIPOLC[3], indicating a growing interest in irony detection in the international NLP community. Similar challenges, not involving directly an irony detection task, but in which irony detection may prove useful, have been organized also for French (DEFT2015[4]) and Spanish (TASS2015[5]).

It can be noticed that the use of figurative language can be peculiar for each language. Authors in [19] investigated the automatic detection of irony and humor in social networks. They proposed a rich set of features for text interpretation and representation to train classification procedures. Decision trees have been employed focusing on lexical and semantic information that characterize each word, rather than the words themselves as features. The used features took into account frequency, written/spoken differences, sentiments, ambiguity, intensity, and synonymy.

Other methods to try to automatically detect irony and humor are discussed in [20], [21], where authors identify figurative uses of language. In particular, authors have considered features to represent a different type of patterns from a text such as ambiguity, polarity, unexpectedness, and emotional content: they represent low and high level properties of figurative language based on formal linguistic elements. Patterns have been evaluated on a corpus of 50k tweets. The research has shown that all the features together provide a useful linguistic inventory for detecting humor and irony at textual level.

ContextuAl SarCasm DEtector (CASCADE) [22] performs context and

[2] http://alt.qcri.org/semeval2015/task11/

[3] http://www.di.unito.it/~tutreeb/sentipolc-evalita14/

[4] https://deft.limsi.fr/2015/

[5] https://gplsi.dlsi.ua.es/sepln15/en/node/36

content driven sarcasm detection on social media. It also considers the variation in the nature of sarcasm from person to person. To solve this issue the study uses user embeddings and personality traits along with the content based features extracted using Convolutional Neural Networks. [23] uses multi-task learning for sentiment and sarcasm classification. In contrast to these approaches, our approach is mainly solving a part of the problem, namely, figurative language detection and tries to experimentally evaluate the advantages that are offered by semantic features as well as linguistic frames.

Finally, [24] employs semantics for irony detection. The authors propose an algorithm to automatically determine whether a word expressed irony or not through semantic similarity and using a dataset including documents where the same word can be used to express irony or not.

3. Framester

Framester [25] is a wide coverage hub of linguistic linked data standardized in the form of a knowledge graph based on *Frame Semantics*. It uses FrameNet [26], WordNet [27] word senses and BabelNet [28] word senses as the starting point, where several strategies are used to improve the limited coverage of FrameNet. It is bootstrapped from the RDF versions of FrameNet [29], OntoWordNet [30], VerbNet [31], and BabelNet[6] using the underlying semantics. Further connections to other data sources such as ontological resources (e.g., DBpedia Ontology, DolceZero [32]), factual resources (DBpedia and Yago), sentimental resources (i.e., SentiWordNet [33], DepecheMood [34]) and topical resources (DeepKnowNet [35]) are also defined using logical rules.

Framester follows the formalization of D&S (Descriptions and Situations [36]) knowledge pattern, where a frame class is a sub-class of description and a frame occurrence, i.e., an occurrence of a frame in text, is a sub-class of a situation. In the context of SA, the situations in the text can be described by evoking particular frames.

The basic FrameNet coverage enhancements introduced in Framester are constituted of three subsets referred to as Framester Base, Direct eXtensions (DirectX) and Transitive eXtensions (TransX) (named as profile b, d and t in Section 4, respectively and explained further below). It uses the predicate skos[7]:closeMatch for defining the mappings between the BabelNet Synsets and FrameNet frames (Listing 1 some of the triples related to the frame `Complaining`). The datasets are available on-line[8].

[6] http://babelnet.org/

LISTING 1 **Sample triples depicting the mappings between FrameNet frames and BabelNet Synsets**

```
frame:Complaining  skos:closeMatch
                     bn:s00089067v,
                     bn:s00089056v,
                     bn:s00085471v,
                     bn:s00083397v,
                     bn:s00089058v,
                     bn:s00009545n.
```

3.1. Framester Base

It contains manually curated mappings created using existing FrameNet-WordNet mappings (eXtended WordFrameNet [37], FrameBase [38], and other existing sources). For example, the sentence 1 annotated with Framester Base extensions is shown in Figure 1.

3.2 DirectX

It extends the Base profile using semantic relations from WordNet (i) direct hyponyms of the noun/verb synsets, (ii) "Instance-of" relations between noun synsets, (iii) adjective synset similarity, (iv) same verb groups including verb synsets, (v) pertainymy relations between adverb synsets and noun/adjective synsets, (vi) participle relations between adjective and verb synsets and finally, (vii) morphosemantic links between

[7] prefix skos: <http://www.w3.org/2004/02/skos/core#>
[8] https://github.com/framester/Framester

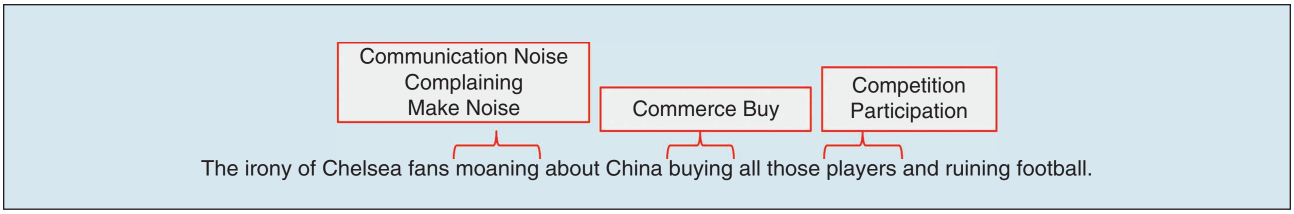


FIGURE 1 Sentence (1) annotated with Framester Base mappings.

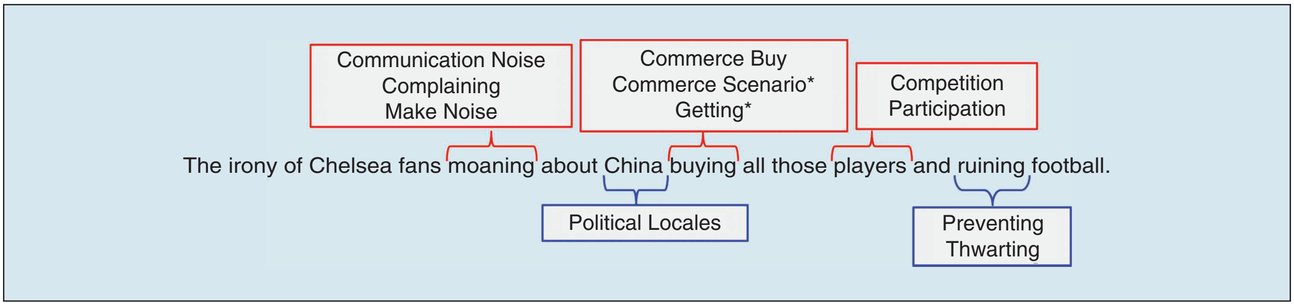


FIGURE 2 Sentence (1) annotated with mappings from DirectX. The blue boxed frames represent the new annotations obtained by DirectX mappings. The symbol (*) in front of Frame labels represents new profile specific frames on existing annotations.

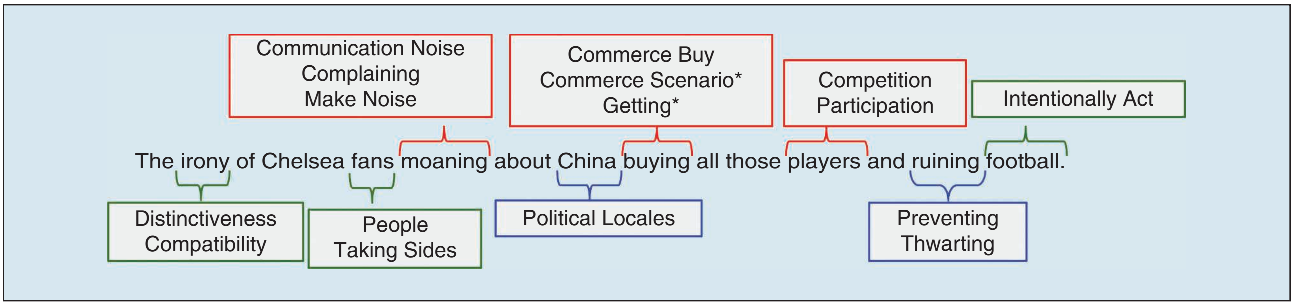


FIGURE 3 Sentence (1) annotated with mappings from TransX. The green boxed frames represent the new annotations obtained by TransX mappings. The symbol (*) in front of Frame labels represents new profile specific frames on existing annotations.

adjective and verb synsets. Figure 2 shows the running example annotated using DirectX.

3.3. TransX

It further extends DirectX using (i) transitive hyponymy relations, (ii) unmapped siblings of mapped noun/verb synsets and finally, (iii) derivational links. Figure 3 shows the running example annotated using TransX.

The above examples are annotated using Word Frame Disambiguation (WFD) framework which is a detour based approach to frame detection. It can be accessed through an API available online[9].

It is implemented as a pipeline including tokenization, POS tagging, lemmatization, word sense disambiguation (using UKB [39] and Babelfy [9]), and finally frame detection by detour using the WFD profiles described above.

4. Leveraging Frame Semantics for Extracting Semantic Features

To extract certain semantic features of each tweet, Word Frame Disambiguation (WFD) API (see Section 3 for details) was used for improving the results. WFD-API takes text as an input and generates an output with the corresponding BabelNet synsets and FrameNet frames along with the position of the lexical unit evoking the frame as an output. In our experiments we did not apply any normalization to the tweets, i.e. we did not try to transform hashtags into lemmas and we did not correct the spelling of the tweets. Even working on the raw tweets, we were able to cover 96% of tweets with extended semantic information. Semantic frames and BabelNet synsets have been extracted using the Base and TransX profiles as detailed in Section 3. Figure 4 shows the output of the WFD API for the tweet *Oh, you're 14 and quit smoking? How inspiring!*

We considered all the frames evoked by each word in a tweet and created several other new datasets from the combinations of the following features: tweet, BabelNet synsets and semantic frames. We augmented the original tweets with the semantic features using the following strategies:

1) only tweets (our baseline)
   Oh, you're 14 and quit smoking? How inspiring!
2) frame annotations only
   Process_stop Activity_stop Capacity Ingest_substance Ingestion Breathing
3) BabelNet synsets only (using Babelfy for Word Sense Disambiguation) (replacement),
   s13696060v s00084710v s00093939v s00084027v
4) both frame annotations and BN synsets (replacement),
   Process_stop Activity_stop Capacity Ingest_substance Ingestion Breathing s13696060v s00084710v s00093939v s00084027v
5) tweets along with their frame annotations (augmentation)
   Oh, you're 14 and quit smoking? How inspiring! Process_stop Activity_stop Capacity Ingest_substance Ingestion Breathing
6) tweets with their BN sysnets (augmentation)
   Oh, you're 14 and quit smoking? How inspiring! s13696060v s00084710v s00093939v s00084027v
7) data set containing three layers i.e., words, BN synsets and the frame annotations (augmentation)
   Oh, you're 14 and quit smoking? How inspiring! Process_stop Activity_stop Capacity Ingest_substance Ingestion Breathing s13696060v s00084710v s00093939v s00084027v

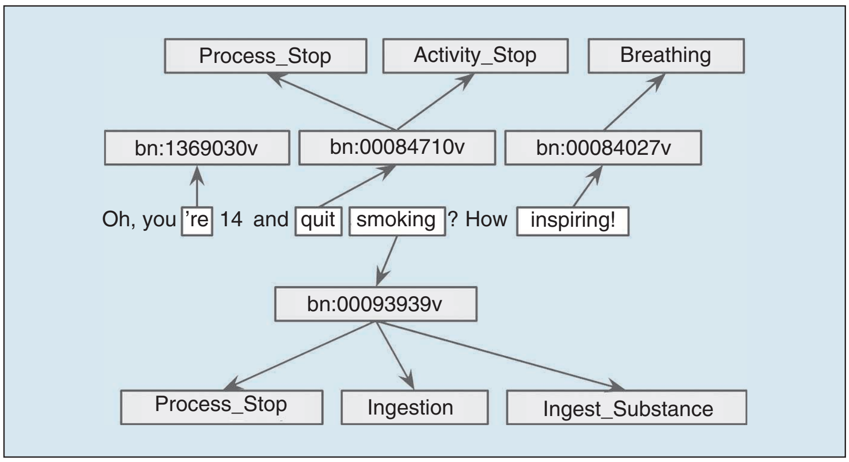


FIGURE 4 Output of the Framester APIs with the tweet *Oh, you're 14 and quit smoking? How inspiring!*

[9] http://lipn.univ-paris13.fr/framester/en/wfd_html/

5. Datasets

Recently, some datasets containing ironic and non-ironic tweets have been released for Italian and French in Evalita (2014 and 2016) and DEFT2017 campaigns, respectively. These datasets are annotated with different labels, such as subjectivity, polarity and irony. Several SemEval campaigns, starting in 2013, proposed a SA task, but in all these campaigns the data were annotated almost exclusively by polarity. Notably, the task-11 proposed in 2015 was focused on figurative language, and all the tweets selected for the task contained irony or sarcasm. In order to achieve this result, the task organizers harvested tweets that have been explicitly marked as ironic and sarcastic by their authors using hashtags such as *#not*, *#irony*, *#sarcasm*, *#yeahright* and similar ones [40]. Here are some examples of the tweets contained in the task-11 SemEval 2015 collection:

- *I love it when kids scream from being so tired instead of just going to sleep #myfavorite #sarcasm*
- *Oh yea, tomorrow should be so much fun! #Not*

For our objective, which is to test whether our approach is able to distinguish figurative from non-figurative language, we balanced the dataset by performing an under-sampling of the majority class of our binary classification problem. Moreover, the idea of having a balanced dataset of figurative and non-figurative is that an equal distribution allows us to focus on the problem of discovering what are the most characteristic features which are representing figurative language, leaving behind the problem of learning on unbalanced datasets. We chose to use the test dataset of the SemEval 2016 task 4 on SA. Here are some examples of the tweets contained in the task-4 SemEval 2016 collection:

- *Haven't read To Kill a Mockingbird in years. That may be a good thing for when I read Go Set a Watchman. Might make it less heartbreaking.*
- *Trump said June 30th that he'd be at Miss USA pageant in Baton Rouge. Organizers say he's not coming. No word yet from his camp on his plans*

The dataset contains 7,825 figurative and 8,959 non-figurative tweets. The dataset is slightly different from the original one because of the way the data were distributed in SemEval, by providing only the tweet ID. The tweets were then retrieved using these IDs through Twitter API. However, since 2015 some tweets have been deleted, so the size of the dataset is smaller than the original one. The figurative dataset was further processed in order to remove the distinctive hashtags: keeping them would have made the task trivial, reducing it to the problem of finding hashtags like *#not*, *#irony*, *#sarcasm* in the tweet.

Figure 5(a) illustrates the most frequent words in each category, with their relative probability with respect to the other category (we only considered the words occurring in both categories). It is evident that subjectivity plays an important role in figurative tweets, due to the prominence of pronouns 'I' and 'me', in particular. On the other hand, temporal markers seem important for non-figurative tweets ('tomorrow', 'Sunday', '1st'), indicating that they tend to be in relation to events. The dataset for the experiments concerning only sarcasm VS irony detection was obtained from the figurative one, by selecting those tweets that were explicitly tagged as #irony or #sarcasm, and subsequently removing the hashtags. The result of this process is 1,363 ironic and 2,081 sarcastic tweets.

For this dataset, Figure 6 depicts that the terms characterizing the tweets labelled as *#sarcasm* may be related to

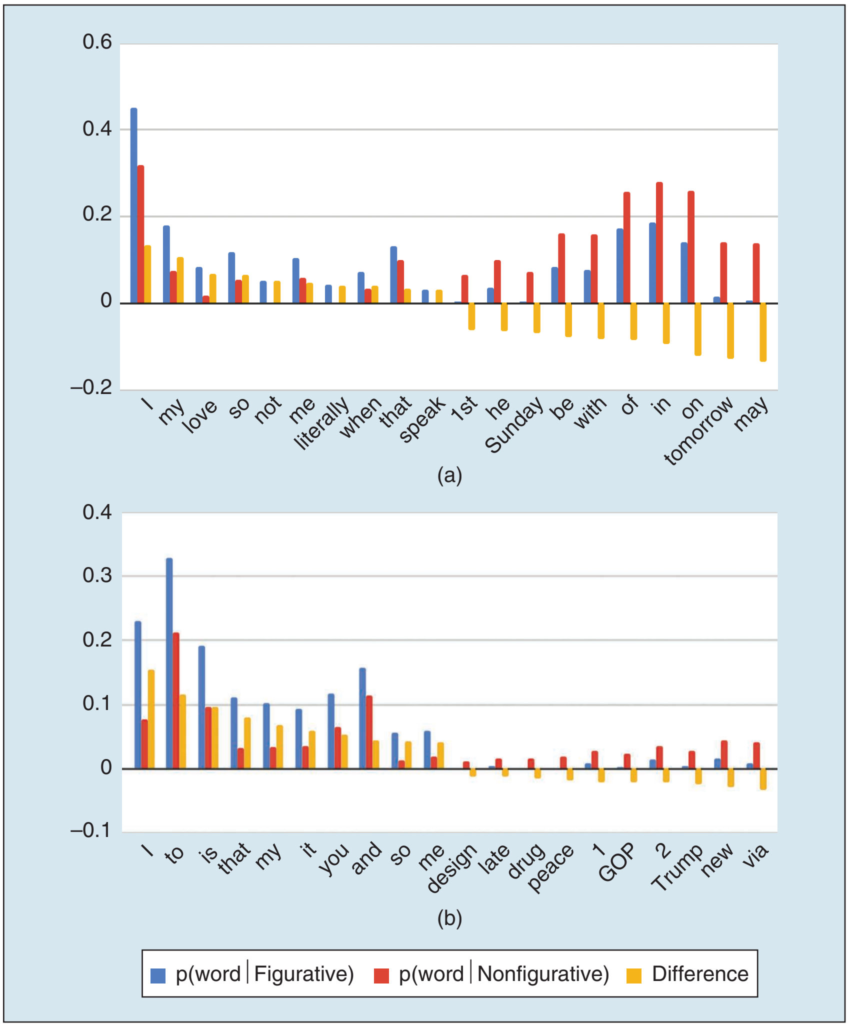


FIGURE 5 Top 10 words for each category in the figurative vs. non-figurative dataset for SemEval (a) and Klinger (b), ranked by their relative probability. The leftmost (rightmost) 10 words are the most likely to occur in the figurative (non-figurative) part of the underlying dataset.

hyperboles ('so…', 'great…', 'really…'). This is in line with the analysis carried out by [41], which correlated the presence of linguistic markers such as exclamations and intensifiers to sarcasm. Our curated version of the SEMEVAL dataset (IDs of the tweets and the combinations of unigrams and semantic frames) is available on-line[10].

To verify our results on a different dataset, we chose the recently published Irony Sarcasm Analysis Corpus [10], publicly available for download[11], which consists of four sub-corpora: irony, sarcasm, regular and figurative (the last is ironic and sarcastic, but it has been subsampled to obtain a balanced corpus). Each original training sub-corpus consisted of 30,000 tweets whereas each test sub-corpus included 3,000 tweets. The method of collecting these data is similar to the one used for the task-11 SemEval 2015, that is, irony and sarcasm hashtags are considered as self-annotations. As the dataset contained tweet IDs only, a script was developed to download each related tweet content. Because of the same issues as encountered in the first dataset, since 2016 some tweets have been deleted and, thus, the size of the sub-corpus is smaller than the original ones. Each downloaded training sub-corpus includes around 20k tweets whereas each downloaded test sub-corpus includes around 2,400 tweets.

As it can be observed in Figure 5(b), the terms characterizing the figurative tweets are indicative of subjectivity, with a prominence of personal pronouns, like in the SemEval corpus. In the non-figurative set, the keywords seem more related to factual information, events and the presence of http links ('via'). We will refer to the two datasets described above as the SEMEVAL and the KLINGER datasets and we have evaluated our approach on each of them.

6. Experiments

This section details different experiments conducted representing the content of tweets by using each of the seven features models detailed in Section 4. The main research questions that we raised and wanted to answer with these experiments were: how useful are semantic features for figurative language and irony/sarcasm detection? In the positive case, how much gain are we able to achieve from within the figurative language detection tasks? As already mentioned in Section 1, the conducted experiments are related to the following two tasks:

- ❏ detecting tweets using figurative language expressions out of a dataset which also includes regular tweets;
- ❏ distinguishing ironic from sarcastic tweets out of a dataset which includes both.

As described in Section 5, each task above has been tested on two datasets. The seven features models were applied to each dataset. Each tweet was previously processed through a tokenization step to clean it up by breaking down the text by spaces and punctuation marks. In addition to the content representation strategies, we considered also two different token representations: *N-grams* and *TF-IDF*. To identify a set of useful n-grams we tokenized the text considering that short forms such as *doesn't*, *we'll*, *she's*, *etc.* have to be treated as one word. For the figurative language detection task, the number of words in the dictionary was around 55,000 for SEMEVAL and 70,000 for KLINGER, whereas for the irony/sarcasm detection task, the number of words was around 15,000 for SEMEVAL and 65,000 for KLINGER. For both tasks we used the total number of words in the dictionary as size of the representation model as we always noticed a deterioration of the accuracy when we tried to reduce the size of the dictionary. As far as the SEMEVAL dataset is concerned, for the figurative language dataset, we tried with the 5,000, 7,000, 15,000 and 55,000 most frequent words, corresponding, respectively, to more than 5, 3, 2 and 1 occurrences. For the ironic/sarcastic dataset we tried 1,000, 2,000, 3,000 and 15,000 most frequent words that correspond, respectively, to more than 5, 3, 2 and 1 occurrences. As far as the KLINGER dataset is concerned, for the figurative language dataset, we tried with the 5,000, 7,000, 15,000 and 70,000 most frequent words, corresponding, respectively, to more than 11, 8, 3 and 1 occurrences. For the ironic/sarcastic dataset we tried 5,000, 7,000, 15,000 and 65,000 most frequent words that correspond, respectively, to more than 9, 6, 3 and 1 occurrences. Tables 1 and 2 show the accuracy and F-measure for the baseline method in both tasks, using unigrams according to the bag of word model (BOW), for the SEMEVAL dataset, different sizes for the representation model and Naive Bayes classifier. The

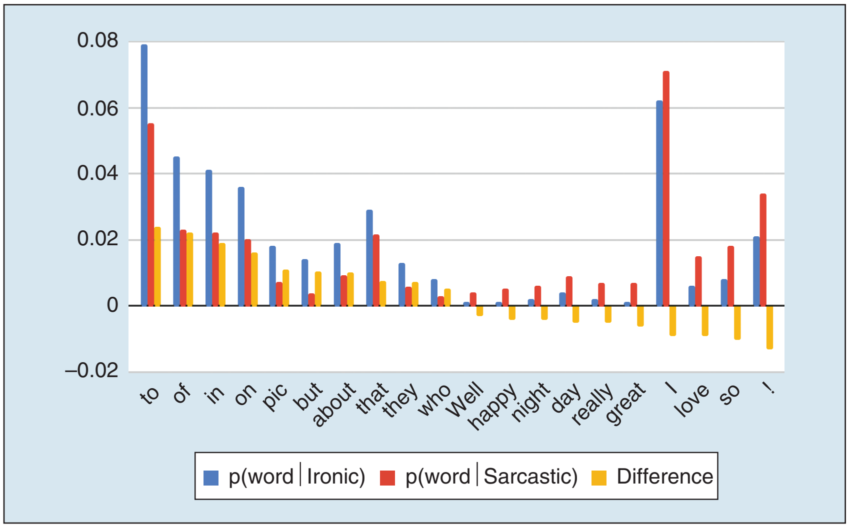


FIGURE 6 Top 10 words for each category in the irony vs. sarcasm dataset, ranked by their relative probability. The leftmost 10 words are the most likely to occur as ironic, the rightmost 10 words are the most likely to occur as sarcastic.

[10] http://lipn.univ-paris13.fr/~buscaldi/datasets.zip
[11] http://www.romanklinger.de/ironysarcasm/

experiment also served to justify the use of the full vocabulary size for the two different datasets. It is not surprising that the highest value for the vocabulary size produces the best results for both tasks, given the small size of tweets. Moreover, some hashtags, one of the distinctive features of tweets, may occur rarely but they are highly informative keywords.

Similarly, for KLINGER, Tables 3 and 4 indicate the accuracy and F-measure we obtained for the baseline method in both tasks. In particular, for the first task, we used the figurative (balanced) and the regular dataset whereas for the second task we used the irony and the sarcasm datasets. To note that irony and sarcasm tweets size of KLINGER are much higher than those of SEMEVAL; this explains why we had higher threshold values in Table 4.

TABLE 1 Setting the threshold for the optimal feature size for the figurative language dataset of SEMEVAL and baseline accuracy and F-measure comparison using unigrams.

	BOW – 5,000 (> 5 OCCURRENCES)	BOW – 7,000 (> 3 OCCURRENCES)	BOW – 15,000 (> 2 OCCURRENCES)	BOW – 55,000 (> 1 OCCURRENCE)
ACC.	0.85	0.86	0.87	0.90
F.	0.86	0.89	0.90	0.92

TABLE 2 Setting the threshold for the optimal feature size for the irony/sarcasm dataset of SEMEVAL and baseline accuracy and F-measure comparison using unigrams.

	BOW – 1,000 (> 5 OCCURRENCES)	BOW – 2,000 (> 3 OCCURRENCES)	BOW – 3,000 (> 2 OCCURRENCES)	BOW – 15,000 (> 1 OCCURRENCE)
ACC.	0.84	0.86	0.82	0.88
F.	0.88	0.89	0.85	0.93

TABLE 3 Setting the threshold for the optimal feature size for the figurative language dataset of KLINGER and baseline accuracy and F-measure comparison using unigrams.

	BOW – 5,000 (> 11 OCCURRENCES)	BOW – 7,000 (> 8 OCCURRENCES)	BOW – 15,000 (> 3 OCCURRENCES)	BOW – 70,000 (> 1 OCCURRENCE)
ACC.	0.80	0.85	0.87	0.89
F.	0.79	0.82	0.86	0.88

TABLE 4 Setting the threshold for the optimal feature size for the irony/sarcasm dataset of KLINGER and baseline accuracy and F-measure comparison using unigrams.

	BOW – 5,000 (> 9 OCCURRENCES)	BOW – 7,000 (> 6 OCCURRENCES)	BOW – 15,000 (> 3 OCCURRENCES)	BOW – 65,000 (> 1 OCCURRENCE)
ACC.	0.75	0.76	0.77	0.78
F.	0.76	0.77	0.78	0.79

TABLE 5 Accuracy and F-measure for the figurative language detection problem across the different features models on SEMEVAL.

FEATURES	ACC. TF.IDF	ACC. BOW	F. TF.IDF	F. BOW
UNIGRAMS	0.72	0.90	0.74	0.92
BABELNET SYNSETS	0.73	0.86	0.74	0.88
SEMANTIC FRAMES	0.56	0.68	0.59	0.72
BNS+SF	0.72	0.86	0.75	0.88
UNIGRAMS+BNS	**0.77**	**0.94**	**0.79**	**0.95**
UNIGRAMS+SF	0.76	0.92	0.78	0.93
UNIGRAMS+BNS+SF	0.74	0.92	0.75	0.94

For each dataset, two Naive Bayes classifiers were trained using unigram features and TF-IDF features. In the second case, each word was represented using its TF-IDF value instead of frequency. Tables 5 and 6 show the accuracy and F-measure obtained for each of the seven features models in the figurative language detection task for SEMEVAL whereas Tables 7 and 8 show results on KLINGER. The results have been calculated using 10-fold cross-validation and averaged over 10 runs (also for the baselines/unigrams).

The accuracy has been calculated as $accuracy = (TP + TN)/(TP + TN + FP + FN)$ and the F-measure as: $F\text{-}measure = (2 * TP)/(2 * TP + FP + FN)$, where TP, TN, FP and FN correspond, respectively, to true positives, true negatives, false positives and false negatives. Table 9 and Table 10 illustrate the process of computing TP, TN, FP and FN with the help of the confusion matrices for the two tasks with values for the baselines on SEMEVAL. It should be noticed that for the task 1 and 2, an under-sampling of the majority class (sarcastic and non-figurative tweets) was performed. Sarcastic tweets were reduced from 2,081 to 1,350 whereas non-figurative tweets were reduced from 8,959 and to 3,400. We thus obtained balanced datasets for the two classification tasks.

For task 1, using semantic features improves the classification with respect to unigrams (which represents our baseline) for all the combinations including both unigrams and the semantic features. The combinations that resulted in the lowest accuracy (lower than the baseline) were those employing semantic frames without unigrams (e.g., BabelNet synsets, semantic frames, BabelNet synsets + semantic frames). This suggests a connection between the style of text when adopting figurative language expression to denote either irony or sarcasm. Also, the adoption of semantics brings new information in helping the classifier to better distinguish figurative language from non-figurative one.

In order to verify the role of semantic frames in the classification when using SEMEVAL, we carried out an analysis to identify which frames were more characteristics of figurative language. To do that, we counted the occurrences of each frame F within the two classes (figurative language and non). Then we calculated the probability for each frame p_F to occur in each of the two classes (p_F^1 and p_F^2). For each frame, we then computed the differences of probabilities $p_F^1 - p_F^2$ to understand whether the frame should be assigned to the first class or to the second. Given a frame F, if $p_F^1 - p_F^2 > 0$, then we would assign F to the first class; otherwise, if $p_F^1 - p_F^2 < 0$, then we would assign F to the second class.

Figures 7(a) and 7(b) show, respectively, the frames assigned to the figurative language class and the frames assigned to the non-figurative language class for SEMEVAL and KLINGER.

The frames in Figure 7(a) reflect the subjective nature of ironic and sarcastic comments. In particular, *Experiencer_focus*, *Desirability*, *Statement*, *Expertise* are all frames that are related to a subjective view. On the other hand, Figure 7(b) shows frames that are more closely related to objective statements, such as *Leadership*, *Calendric_unit*, *Political_locales* (aka place names or toponyms). This analysis shows that frames may be important clues in determining the subjectivity of a text, which is, in turn, an important clue for figurative language. For the SEMEVAL dataset, there are also some frames that seem to be related to a specific event that occurred during the time window in which the non-figurative dataset was collected (the election of miss USA), such as *Finish_competition*, *finish_game* and *win_prize*. For task 2, using semantic features improves the classification in every combination except when using the semantic frames alone. The reason is that frames generalize too much and they are not detailed enough to convey the

TABLE 6 Accuracy and F-measure for the irony/sarcasm detection problem across the different features models on SEMEVAL.

FEATURES	ACC. TF.IDF	ACC. BOW	F. TF.IDF	F. BOW
UNIGRAMS	0.71	0.88	0.74	0.93
BABELNET SYNSETS	0.74	0.90	0.77	0.94
SEMANTIC FRAMES	0.70	0.82	0.73	0.85
BNS+SF	0.72	0.89	0.74	0.93
UNIGRAMS+BNS	**0.76**	**0.91**	**0.80**	**0.95**
UNIGRAMS+SF	0.75	0.89	0.78	0.92
UNIGRAMS+BNS+SF	0.73	0.89	0.77	0.94

TABLE 7 Accuracy and F-measure for the figurative language detection problem across the different features models on KLINGER.

FEATURES	ACC. TF.IDF	ACC. BOW	F. TF.IDF	F. BOW
UNIGRAMS	0.75	0.89	0.74	0.88
BABELNET SYNSETS	0.67	0.73	0.65	0.73
SEMANTIC FRAMES	0.58	0.64	0.59	0.65
BNS+SF	0.6	0.71	0.61	0.71
UNIGRAMS+BNS	**0.83**	**0.90**	**0.82**	**0.89**
UNIGRAMS+SF	0.73	0.89	0.74	0.88
UNIGRAMS+BNS+SF	0.77	0.88	0.76	0.87

TABLE 8 Accuracy and F-measure for the irony/sarcasm detection problem across the different features models on KLINGER.

FEATURES	ACC. TF.IDF	ACC. BOW	F. TF.IDF	F. BOW
UNIGRAMS	0.70	0.78	0.69	0.79
BABELNET SYNSETS	0.64	0.71	0.66	0.72
SEMANTIC FRAMES	0.55	0.61	0.60	0.67
BNS+SF	0.63	0.70	0.66	0.72
UNIGRAMS+BNS	**0.73**	**0.80**	**0.74**	**0.80**
UNIGRAMS+SF	0.73	0.80	0.71	0.79
UNIGRAMS+BNS+SF	0.70	0.79	0.71	0.80

TABLE 9 Confusion matrix for task 1 (figurative language detection) and values for the baseline on SEMEVAL.

		ACTUAL CLASS	
		TWEETS W/FIG. LANG.	**TWEETS W/OUT FIG. LANG.**
PREDICTED CLASS	*TWEETS W/FIG. LANG.*	TRUE POSITIVES (3,092)	FALSE POSITIVES (349)
	TWEETS W/OUT FIG. LANG.	FALSE NEGATIVES (343)	TRUE NEGATIVES (3,060)

TABLE 10 Confusion matrix for task 2 (irony vs sarcasm) and values for the baseline on SEMEVAL.

		ACTUAL CLASS	
		IRONIC TWEETS	**SARCASTIC TWEETS**
PREDICTED CLASS	*IRONIC TWEETS*	TRUE POSITIVES (1,198)	FALSE POSITIVES (161)
	SARCASTIC TWEETS	FALSE NEGATIVES (165)	TRUE NEGATIVES (1,189)

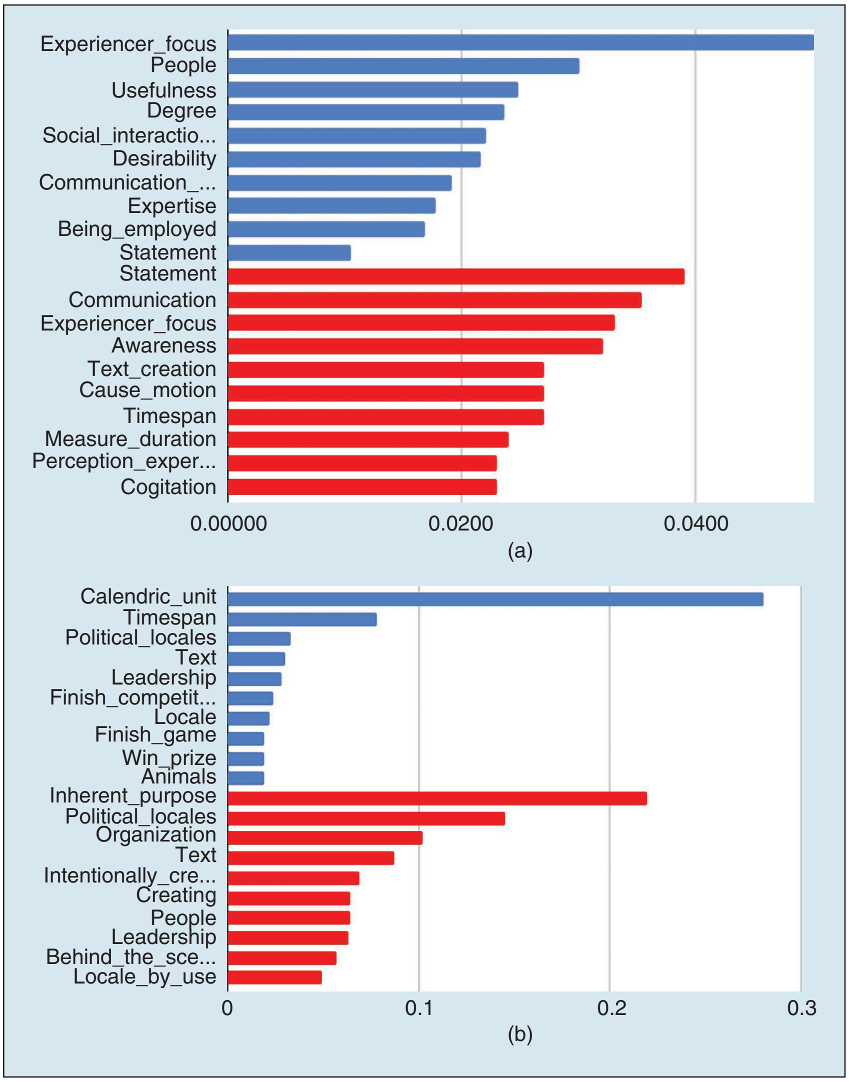


FIGURE 7 Most likely frames within the figurative language category (a) and non-figurative language category (b): these frames are more likely to be found within the tweets in SEMEVAL (blue) and KLINGER (red) containing figurative language expressions. X-axis values indicate the difference between the probability of observing the frame in the figurative category and the probability of observing the same frame in the non-figurative category.

TABLE 11 Accuracy and F-measure for the figurative language detection problem removing the objective tweets for SEMEVAL.

FEATURES	ACC. TF.IDF	ACC. BOW	F. TF.IDF	F. BOW
UNIGRAMS	0.70	0.90	0.68	0.89
BABELNET SYNSETS	0.67	0.87	0.68	0.86
SEMANTIC FRAMES	0.60	0.71	0.62	0.72
BNS+SF	0.66	0.86	0.68	0.87
UNIGRAMS+BNS	**0.75**	**0.93**	**0.72**	**0.92**
UNIGRAMS+SF	0.74	0.93	0.70	0.90
UNIGRAMS+BNS+SF	0.73	0.91	0.74	0.92

semantics of a sentence. The observed results, in particular the improvements obtained by augmentation using semantic frames and BabelNet synsets, agree with those previously obtained for the polarity detection problem [16]. In particular, for both tasks, the best results have been obtained when using the combination of unigram + BabelNet synsets.

As mentioned above, because figurative tweets are subjective in nature, in order to remove any potential biases from our collections, we have created two more datasets (one for SEMEVAL and one for KLINGER) by removing all the non-subjective tweets from the non-figurative dataset and made the new collection more uniform. Therefore, one final experiment we performed has been to eliminate the objective tweets from the non-figurative collection and to re-run the figurative vs. non-figurative experiment. Tables 11 and 12 show the accuracy and F-measure obtained for such a case for each model and features (for both datasets) and that further confirms our initial results. Again, the results show that the model with semantic frames and BabelNet synset improves the figurative language detection.

In order to filter out the objective tweets from the non-figurative collection, we had to build a subjectivity classifier, since the tweets in the SemEval-2016 task 4 collection and in the Klinger dataset did not include such information about the tweets. To achieve the aforementioned goal, 400 tweets were randomly chosen (dev set) and manually labeled by two annotators with subjective or objective labels. For both SEMEVAL and KLINGER, the inter-annotator agreement between the two annotators was *kappa* = 0.72. Next, we extracted several features that might help to distinguish the subjectivity or objectivity of a tweet. To this end, we extracted in total 5,610 and 5,627 features for SEMEVAL and KLINGER datasets, respectively, involving the frequency of 5,569 (SEMEVAL) and 5,627 (KLINGER) strong subjective clues[12], a set of sentiment features consists of 8 emotional features and 2 features describing the polarity (i.e., positive and negative), the frequency of 25 pronouns, the frequency of distinctive punctuation marks like @, ! and #, as well as three features from common emoticons. All the features were

[12] http://mpqa.cs.pitt.edu/lexicons/subj_lexicon

embedded in a feature vector to be used to train a Random Forests (RF) classifier. A 10-fold cross-validation test showed a 94% accuracy, which is expected to be lower on data outside the dev set. The classifier was applied to the SemEval-2016 non-figurative test set and Klinger regular test set to identify and remove the objective tweets: the result was a dataset composed of 5,529 subjective tweets (out of the 8,959 non-figurative dataset) for SEMEVAL and 15,935 subjective tweets (out of the 20,015 that compose the regular dataset) for KLINGER.

TABLE 12 **Accuracy and F-measure for the figurative language detection problem removing the objective tweets for KLINGER.**

FEATURES	ACC. TF.IDF	ACC. BOW	F. TF.IDF	F. BOW
UNIGRAMS	0.70	0.91	0.72	0.93
BABELNET SYNSETS	0.60	0.79	0.68	0.87
SEMANTIC FRAMES	0.58	0.73	0.66	0.84
BNS+SF	0.68	0.80	0.71	0.87
UNIGRAMS+BNS	**0.73**	**0.92**	**0.76**	**0.95**
UNIGRAMS+SF	0.71	0.92	0.74	0.95
UNIGRAMS+BNS+SF	0.70	0.91	0.72	0.94

Statistical significance of the results presented in this work was tested by computing a paired two-tailed *t-test*. We obtained *p-values* < 0.001 indicating that the differences between the best result and the chosen baseline are statistically significant.

Our code has been developed on top of Apache Spark 2.0 and run on a cluster of two Dell PCs with 6 cores each and 12GB of RAM each. For the Naive Bayes classifiers, we employed the Machine Learning Library (MLlib) of Apache Spark. Therefore it is well suited for big data domain as it can be easily run on datasets of huge dimensions by simply adding more nodes to the cluster.

7. Conclusions and Perspectives

This paper focuses on the problem of figurative language detection from social media (with a focus on the use of semantic features to identify irony and sarcasm) and how the augmentation and replacement techniques of semantic features improved the overall detection. More precisely, the tasks we addressed were: (i) the detection of figurative tweets out of a dataset with both figurative and non-figurative tweets and (ii) the detection of ironic tweets out of a dataset with both ironic and sarcastic tweets. We leveraged Framester to extract semantic frames and BabelNet synsets that augmented the vector space representation of unigrams that has been fed to Naive Bayes classifier and that performed better than the baseline (unigrams only without semantic frames). We have also found out that semantic frames may be important clues in determining the subjectivity of a text, which is, in turn, an important clue for figurative language. To avoid any potential biases due to the subjectivity, we have also shown that removing all the objective tweets among the non-figurative ones from the dataset does not change the accuracy of our method with respect to the baseline. In order to confirm these results, we will need to carry out further experiments, in particular by varying the collections and/or taking into account stylistic features. It is quite evident from the high baseline score that the two collections are easily recognizable from a thematic point of view. Our experiments on exclusively subjective tweets ruled out only one of the possible biases, but there may be others related to specific events and the time frame in which the tweets were collected. However, finding labelled data, especially with irony or figurative language markers, is not easy. The integration of stylistic features in the models may allow for a deeper understanding of how figurative language is expressed.

We are also aware that deep neural models are acquiring great importance in most NLP tasks. These models are able, and this is true especially in the case of convolutional networks, to automatically learn abstract features from text, which makes them particularly fit for the figurative language detection problem [5]. We are planning to build a NN model that is also able to take into account semantic frames, for instance using embeddings derived from knowledge bases such as RDF2Vec [42] and Frame2Vec [43], in the hope that these features may prove as useful as they proved for the more classic models that we tested in this paper.

The collections we used for our experiments are constituted by tweets, which have some very distinctive characteristics, both stylistic and in length. However, it would be difficult to find an appropriate source for long figurative texts. A crowdsourced corpus of ironic and sarcastic reviews compiled by [44] shows that the length of such reviews may vary but the ironic or sarcastic elements are expressed at utterance level. The presence of irony may also be supported by several text utterances that can be extracted from different parts of a long text, which would increase the difficulty of the task because of the need to localize these fragments within the text.

Last but not least, and in line with the manner social media is changing the way people interact with each other and to keep up with the rise of the five V's (volume, variety, velocity, veracity, and value), we developed our algorithms on top of Apache Spark and run it on a cluster of PCs that can be easily extended by adding more nodes to higher and higher efficiency. We plan on performing further comparisons by using lexicons such as SenticNet [45] and OntoSenticNet [46] to improve the overall accuracy.

References

[1] V. S. Subrahmanian and D. R. Recupero, "AVA: Adjective-verb-adverb combinations for sentiment analysis," *IEEE Intell. Syst.*, vol. 23, no. 4, pp. 43–50, July–Aug. 2008.

[2] A. Gangemi, V. Presutti, and D. R. Recupero, "Frame-based detection of opinion holders and topics: A model and a tool," *IEEE Comput. Intell. Mag.*, vol. 9, no. 1, pp. 20–30, Feb. 2014.

[3] D. R. Recupero, V. Presutti, S. Consoli, A. Gangemi, and A. G. Nuzzolese, "Sentilo: Frame-based sentiment analysis," *Cogn. Comput.*, vol. 7, no. 2, pp. 211–225, Apr. 2015.

[4] E. Cambria, "Affective computing and sentiment analysis," *IEEE Intell. Syst.*, vol. 31, no. 2, pp. 102–107, Mar. 2016.
[5] S. Poria, E. Cambria, D. Hazarika, and P. Vij, "A deeper look into sarcastic tweets using deep convolutional neural networks," in *Proc. 26th Int. Conf. Computational Linguistics: Technical Papers*, Dec. 2016, pp. 1601–1612.
[6] J. M. Pierrel, *Ingéniérie des Langues*. Hermès, 2000.
[7] P. Schoentjes, *Poetique de l'ironie*. Paris: Seuil, 2001.
[8] A. Gangemi, M. Alam, and V. Presutti, "Word frame disambiguation: Evaluating linguistic linked data on frame detection," in *Proc. 4th Int. Workshop on Linked Data for Information Extraction co-located with ISWC*, Oct. 2016, pp. 23–31.
[9] A. Moro, A. Raganato, and R. Navigli, "Entity linking meets word sense disambiguation: A unified approach," *Trans. Assoc. Comput. Linguist.*, vol. 2, pp. 231–244, Dec. 2014.
[10] J. Ling and R. Klinger, "An empirical, quantitative analysis of the differences between sarcasm and irony," in *Proc. 13th Extended Semantic Web Conf. 2016 Satellite Events*, 2016, pp. 203–216.
[11] D. Reforgiato and E. Cambria, "ESWC'14 challenge on concept-level sentiment analysis," *Commun. Comput. Inf. Sci.*, vol. 475, pp. 3–20, Oct. 2014.
[12] M. Dragoni and D. Reforgiato, "Challenge on fine-grained sentiment analysis within ESWC 2016," *Commun. Comput. Inf. Sci.*, vol. 641, pp. 79–94, Oct. 2016.
[13] M. Dragoni, A. G. B. Tettamanzi, and C. da Costa Pereira, "Propagating and aggregating fuzzy polarities for concept-level sentiment analysis," *Cogn. Comput.*, vol. 7, no. 2, pp. 186–197, Oct. 2015.
[14] H. Saif, Y. He, and H. Alani, "Semantic sentiment analysis of Twitter," in *Proc. Int. Semantic Web Conf.*, 2012.
[15] D. R. Recupero, S. Consoli, A. Gangemi, A. G. Nuzzolese, and D. Spampinato, "A semantic web based core engine to efficiently perform sentiment analysis," in *The Semantic Web: ESWC 2014 Satellite Events - ESWC 2014 Satellite Events*, 2014, pp. 245–248.
[16] A. Dridi and D. R. Recupero, "Leveraging semantics for sentiment polarity detection in social media," *Int. J. Mach. Learn. Cybern.*, vol. 10, no. 8, pp. 2199–2207, Sept. 2019.
[17] E. Cambria and A. Hussain, *Sentic Computing: A Common-Sense-Based Framework for Concept-Level Sentiment Analysis*, 1st ed. New York: Springer-Verlag, 2015.
[18] S. Attardo, "Irony markers and functions: Towards a goal-oriented theory of irony and its processing," *Rask*, vol. 12, no. 1, pp. 3–20, Jan. 2000.
[19] F. Barbieri and H. Saggion, "Automatic detection of irony and humour in Twitter," in *Proc. 5th Int. Conf. Computational Creativity*, June 2014, pp. 155–162.
[20] A. Reyes, P. Rosso, and D. Buscaldi, "From humor recognition to irony detection: The figurative language of social media," *Data Knowl. Eng.*, vol. 74, pp. 1–12, Apr. 2012.
[21] A. Reyes, P. Rosso, and T. Veale, "A multidimensional approach for detecting irony in Twitter," *Lang. Resour. Eval.*, vol. 47, no. 1, pp. 239–268, Mar. 2013.
[22] D. Hazarika, S. Poria, S. Gorantla, E. Cambria, R. Zimmermann, and R. Mihalcea, "CASCADE: Contextual sarcasm detection in online discussion forums," in *Proc. 27th Int. Conf. Computational Linguistics*, 2018, pp. 1837–1848.
[23] N. Majumder, S. Poria, H. Peng, N. Chhaya, E. Cambria, and A. F. Gelbukh, "Sentiment and sarcasm classification with multitask learning," *IEEE Intell. Syst.*, vol. 34, no. 3, pp. 38–43, Jan. 2019.
[24] A. V. Toropova, "Irony detection based on semantic similarity," *Tr. SPIIRAN*, vol. 1, pp. 178–193, Aug. 2014.
[25] A. Gangemi, M. Alam, L. Asprino, V. Presutti, and D. R. Recupero, "Framester: A wide coverage linguistic linked data hub," in *Proc. 20th Int. Conf. Knowledge Engineering and Knowledge Management*, Nov. 2016.
[26] C. F. Baker, C. J. Fillmore, and J. B. Lowe, "The Berkeley framenet project," in *Proc. 36th Annu. Meeting Association Computational Linguistics and 17th Int. Conf. Computational Linguistics*, Aug. 1998, vol. 1, pp. 86–90.
[27] C. Fellbaum, "Review of" WordNet: An electronic lexical database" by Christiane Fellbaum, The MIT Press 1998," *Comput. Linguist.*, vol. 25, no. 2, pp. 292–296, June 1999.
[28] R. Navigli and S. P. Ponzetto, "BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network," *Artif. Intell.*, vol. 193, pp. 217–250, Dec. 2012.
[29] A. G. Nuzzolese, A. Gangemi, and V. Presutti, "Gathering lexical linked data and knowledge patterns from FrameNet," in *Proc. 6th Int. Conf. Knowledge Capture (K-CAP)*, 2011, pp. 41–48.
[30] A. Gangemi, R. Navigli, and P. Velardi, "The OntoWordNet Project: Extension and axiomatization of conceptual relations in WordNet," in *Proc. Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE - OTM Confederated Int. Conf.,* Nov. 2003, pp. 820–838.
[31] K. K. Schuler, "VerbNet: A broad-coverage, comprehensive verb lexicon," Ph.D. dissertation, 2005.
[32] A. Gangemi, N. Guarino, C. Masolo, and A. Oltramari, "Sweetening WORDNET with DOLCE," *AI Mag.*, vol. 24, no. 3, pp. 13–24, Sept. 2003.
[33] S. Baccianella, A. Esuli, and F. Sebastiani, "SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining," in *Proc. Int. Conf. Language Resources and Evaluation, LREC*, May 2010, pp. 17–23.
[34] J. Staiano and M. Guerini, "Depeche mood: A lexicon for emotion analysis from crowd annotated news," in *Proc. 52nd Annu. Meeting Association Computational Linguistics*, June 2014, vol. 2, pp. 427–433.
[35] M. Cuadros, L. Padró, and G. Rigau, "Highlighting relevant concepts from topic signatures," in *Proc. 8th Int. Conf. Language Resources and Evaluation, LREC*, May 2012, pp. 3841–3848.
[36] A. Gangemi and P. Mika, "Understanding the semantic web through descriptions and situations," in *Proc. Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE - OTM Confederated Int. Conf.,* Nov. 2003, pp. 689–706.
[37] M. L. de Lacalle, E. Laparra, and G. Rigau, "Predicate matrix: Extending SemLink through WordNet mappings," in *Proc. 9th Int. Conf. Language Resources and Evaluation (LREC)*, May 2014, pp. 903–909.
[38] J. Rouces, G. de Melo, and K. Hose, "FrameBase: Representing n-ary relations using semantic frames," in *Proc. 12th Extended Semantic Web Conf.*, 2015, pp. 505–521.
[39] E. Agirre and A. Soroa, "Personalizing pagerank for word sense disambiguation," in *Proc. 12th Conf. European Chapter Association Computational Linguistics. The Association Computer Linguistics*, Mar. 30–Apr. 3, 2009, pp. 33–41.
[40] A. Ghosh et al., "Semeval-2015 task 11: Sentiment analysis of figurative language in Twitter," in *Proc. 9th Int. Workshop Semantic Evaluation (SemEval 2015)*, 2015, pp. 470–478.
[41] F. Kunneman, C. Liebrecht, M. van Mulken, and A. van den Bosch, "Signaling sarcasm: From hyperbole to hashtag," *Inform. Process. Manage.*, vol. 51, no. 4, pp. 500–509, July 2015.
[42] P. Ristoski and H. Paulheim, "RDF2vec: RDF graph embeddings for data mining," in *Proc. Int. Semantic Web Conf.*, 2016.
[43] M. Alam, D. R. Recupero, M. Mongiovì, A. Gangemi, and P. Ristoski, "Event-based knowledge reconciliation using frame embeddings and frame similarity," *Knowledge-Based Syst.*, vol. 135, pp. 192–203, Nov. 2017.
[44] E. Filatova, "Irony and sarcasm: Corpus generation and analysis using crowdsourcing," in *Proc. 8th Int. Conf. Language Resources and Evaluation, LREC*, May 2012, pp. 392–398.
[45] E. Cambria, S. Poria, D. Hazarika, and K. Kwok, "SenticNet 5: Discovering conceptual primitives for sentiment analysis by means of context embeddings," in *Proc. 32nd AAAI Conf. Artificial Intelligence*, Feb. 2018, pp. 1795–1802.
[46] M. Dragoni, S. Poria, and E. Cambria, "OntoSenticNet: A commonsense ontology for sentiment analysis," *IEEE Intell. Syst.*, vol. 33, no. 3, pp. 77–85, June 2018.

Guest Editorial

(continued from page 11)

is prototyped on low-cost and ubiquitous WiFi devices and evaluated in extensive real-world experiments. Experimental results verify the effectiveness of BeSense in recognizing user behaviors.

The last paper, "ADMM Empowered Distributed Computational Intelligence for Internet of Energy" by W. Zhong et al., proposes an approach that employs Alternating Direction Method of Multipliers (ADMM) as the theoretical framework for the design of distributed computational intelligence in Internet of Energy (IoE). The authors discuss the challenges of ADMM implementation in IoE and propose a joint computing and networking resources management architecture to meet the challenges. Numerical results show that this architecture could reduce the computing and communication costs of ADMM implementation.

The guest editors would like to thank all the authors for their contributions to the special issue, and all reviewers for their voluntary services. Special thanks go to the Editor-in-Chief, Prof. Hisao Ishibuchi, and all the members of the editorial team for their support during the editing process of this Special Issue.