A Hybrid Approach to the Sentiment Analysis Problem at the Sentence Level

Orestes Appel[a], Francisco Chiclana[a], Jenny Carter[a], Hamido Fujita[b]

[a] *Centre for Computational Intelligence, De Montfort University, Leicester, UK*
[b] *Intelligent Software Systems Laboratory, Iwate Prefectural University, Takizawa, Iwate, Japan*

Abstract

The objective of this article is to present a hybrid approach to the Sentiment Analysis problem at the sentence level. This new method uses natural language processing (NLP) essential techniques, a sentiment lexicon enhanced with the assistance of SentiWordNet, and fuzzy sets to estimate the semantic orientation polarity and its intensity for sentences, which provides a foundation for computing with sentiments. The proposed hybrid method is applied to three different data-sets and the results achieved are compared to those obtained using Naïve Bayes and Maximum Entropy techniques. It is demonstrated that the presented hybrid approach is more accurate and precise than both Naïve Bayes and Maximum Entropy techniques, when the latter are utilised in isolation. In addition, it is shown that when applied to datasets containing snippets, the proposed method performs similarly to state of the art techniques.

Keywords: Sentiment Analysis, Semantic rules, Fuzzy sets, Unsupervised machine learning, SentiWordNet, Naïve Bayes, Maximum Entropy, Computing with Sentiments.

1. Introduction

The human brain has an inherent ability to detect emotion or sentiment in written or spoken language. Social media and other tools related to today's world have increased the number of sources and volume of information dramatically, and the ability of people to process all that is being seriously compromised. Hence, the ability of having computers to go at high speed through the myriad of data available and extract sentiment and/or opinions would be greatly beneficial. *Sentiment Analysis* (SA) is one of the research areas of fastest growth in the last few years. A number of definitions about it are available. Typically, the main objective of SA is to establish the attitude of a given person with regard to some subject, paragraph or document.

Bing Liu [34] defines an opinion as follows: "In an opinion we find the following items: Opinion targets (entities and their features/aspects), Sentiments (positive or negative), Opinion holders (persons who hold the opinions) and Time (when opinions are expressed). Opinions then can be: (a) Direct opinions, (b) Indirect opinions, or Comparative Opinions. A regular opinion is defined as a quintuple $(e_j, a_{jk}, so_{ijkl}, h_i, t_l)$ where e_j is a target entity, a_{jk} is an aspect/feature of the entity e_j, so_{ijkl} is the sentiment value of the opinion from the opinion holder h_i on feature a_{jk} of entity e_j at time t_l." Usually, so_{ijkl} (semantic orientation) is *positive*, *negative* or *neutral*.

Email addresses: orestes.appel@email.dmu.ac.uk (Orestes Appel), chiclana@dmu.ac.uk (Francisco Chiclana), jennyc@dmu.ac.uk (Jenny Carter), issam@iwate-pu.ac.jp (Hamido Fujita)

Quite often, the most commonly applied techniques to address the SA problem belong either in the category of text classification Supervised Machine Learning (SML) (methods like Naïve Bayes, Maximum Entropy or Support Vector Machine (SVM)) or text classification Unsupervised Machine Learning (UML). However, it seems that fuzzy sets, considering their mathematical properties and their ability to deal with vagueness and uncertainty (characteristics present in Natural Languages) are well-equipped to model sentiment-related problems. As it is well known, fuzzy relations have been extensively used in disciplines as dissimilar as linguistics [15], clustering [46] and decision-making [62], among many others. Thus, a combination of techniques -which would fit the concept of *hybrid*- could be successful at addressing the SA challenges by exploiting the best in each technique. In the next paragraph we will address our motivation for exploring this realm of possibilities.

Dzogang et al. stated in [18] that usually authors refer mainly to psychological models when attacking the SA problem. However, other models may be successful as well in this domain. As per Dzogang et al. "it must be underlined that some appraisal based approaches make use of graduality through fuzzy inference and fuzzy aggregation for processing affective mechanisms ambiguity and imprecision." When dealing with SA, Bing Liu [35], one of the main world experts in this area, says that "we probably relied too much on Machine Learning". When it comes to discussing the progress in the SA discipline, Poria et al. [52] introduced a novel idea to concept-level sentiment analysis, which involves combining together linguistics, common-sense computing, and machine learning, aiming to improve precision on polarity detection. This approach of merging techniques is essentially a *hybrid style* of compounding the power of several tools. Considering all of the arguments above, we believe that the following concepts could be applied in combination:

- The concept of *graduality* expressed through fuzzy sets.
- The idea that other tools, together, besides Supervised Machine Learning in isolation, may be viable as well when extracting sentiment from text (especially, if combined with other techniques).
- The positive contribution that NLP tools, semantic rules and a solid opinion lexicon can have in identifying polarity.

Based on these arguments, our research hypothesis can be stated as follows:

Hypothesis 1. *A sentiment analysis method at the sentence level, using a combination of sentiment lexicons, NLP essential tools and fuzzy sets techniques, should perform same or better than today's accepted text classification supervised machine learning algorithms when the latter are utilised in isolation.*

We are establishing the aforementioned hypothesis as we are in search of a sentiment analysis method that closely resembles the way human beings deal with this topic. We expect in the future to be able to expand our method to deal with human-aspects like humour, irony and sarcasm, which most likely will require providing context. However, it is our belief that the sooner we get closer to the way humans process sentiment, the better positioned we will be to take the next step. We call our proposed system a *hybrid* one because of the fact that it uses a combination of methods and techniques that stem from different research disciplines: fuzzy set theory, natural language processing algorithms and linguistic systems.

The rest of this paper is organised as follows: Section 2 addresses related work. Section 3 presents the research methodology with a focus on three main components: the process to follow, the data to be used, and the performance measurement of the SA solution. Section 4 describes in detail the main components of the proposed Hybrid approach to the SA problem at the sentence level. This section culminates with the presentation of both the Hybrid Standard Classification (HSC) and the Hybrid Advanced Classification (HAC) methods, which adds graduality estimation to polarity identification (the latter being performed by HSC). Section 5 shows the experimental results, starting with the outcome of using two well known and accepted SA Machine Learning methods that originate in the text classification field: Naïve Bayes and Maximum Entropy. This section also includes a comparison analysis between the results obtained applying the proposed hybrid method against the aforementioned machine learning techniques. A comparison against the state of the art is shown as well, as a reference. In closing, some concluding remarks and future work plans are presented in Section 6.

For a summarised survey on Sentiment Analysis, please refer to the article by Appel et al. [2]. For a complete review of the evolution of the SA field, please refer to the thorough work of Ravi and Ravi [54]. For a focused account on recent advances in SA techniques [7] is recommended.

2. Related work

SA is a discipline that has seen a lot of activity since about 2000, when Rosalind Picard published her important book *Affective Computing* [51], i.e. “computing that relates to, arises from, or deliberately influences emotion or other affective phenomena”. When one reviews the most recent trends in the field, of which *Sentic Computing*, led by Erik Cambria [6, 10, 11] is a good example, it becomes evident the amount of effort that has gone into researching SA. As per their creators [6], sentic computing “relies on the ensemble application of common-sense computing and the psychology of emotions to infer the conceptual and affective information associated with natural language.” Other articles worth mentioning explore topics around sentiment lexicon-based techniques, like the contributions of Cho et al. [14] and Huang et al. [31]. The work by Bravo-Márquez et al. [5], on the use of multiple techniques and tools in SA, offers a complete study on how several resources that “are focused on different sentiment scopes” can complement each other. The authors focus the discussion on methods and lexical resources that aid in extracting sentiment indicators from natural languages in general. A comprehensive work on *semantic analysis* is Cambria et al. [8], while Schouten and Frasincar work [56] provides a complete survey specific to aspect-level sentiment analysis.

A number of researchers have explored the application of hybrid approaches by combining various techniques with the aim of achieving better results than a standard approach based on only one tool. Indeed, this has been done by Poria et al. in [52] where a novel framework for concept-level sentiment analysis, Sentic Pattern, is introduced by combining linguistics, common-sense computing, and machine learning for improving the accuracy of tasks such as polarity detection. The authors claim that “by allowing sentiments to flow from concept to concept based on the dependency relation of the input sentence, authors achieve a better understanding of the contextual role of each concept within the sentence and, hence, obtain a polarity detection engine that outperforms state-of-the-art statistical methods”. When no matching sentic pattern is found in SenticNet [9] they resort to Supervised Machine Learning. The hybrid approach put forward in the present article uses a dictionary of words frequencies and occurrences instead to address the case when a word is not found in its lexicon. An additional difference with regard to lexicon data centres around polarity ranges. SenticNet enables polarities to be measured in the interval $[-1, 1]$ while

SentiWordNet allows polarities to be in the range $[0, 1]$. As it will later be apparent, the hybrid method presented here creates the foundation for the introduction of the concept of *computing with sentiments*, which derives from Zadeh's innovative idea of computing with words [74].

Related to the use of lexicons in SA approaches, it is worth mentioning the following two research efforts. The first one is by Hajmohammadi et al. [23] on a novel learning model based on the combination of uncertainty-based active learning and semi-supervised self-training approaches, which also addressed the challenges associated with the construction of reliable annotated sentiment corpora for a new language. This research provided us with important lessons on the difficulties and potential pitfalls of embracing such a task and how to better deal with it. The other research effort is by Hogenboom et al. [27, 28] on the the use of emoticons as modifiers of the sentiment expressed in text and as vehicles of sentiment by themselves. According to the findings of the authors, the sentiment associated to emoticons dominates the sentiment conveyed by the text fragment in which these emoticons are embedded. In their work they introduce a sentiment lexicon, which is a point of commonality with the research presented here, as well as a cleverly designed emoticon lexicon.

In [12], Cambria et al. explore how the high generalisation performance of extreme learning machines (feed forward neural networks with a single layer of hidden nodes with the characteristic that the weights connecting inputs to hidden nodes are randomly assigned and are never updated) can be exploited to perform analogical reasoning "in a vector space model of affective common-sense knowledge". The authors also addressed issues related to the so-called 'concept-based approaches', which they have extensively researched, by focusing on a semantic analysis of text through the use of web semantic networks to properly manage "the conceptual and affective information associated with natural language opinions". Again, a point in common with the research approach proposed here is the heavy utilisation of semantic techniques.

As mentioned before, our overall aim is to develop a sentiment analysis method that closely resembles the way human beings deal with this topic, in order to better comprehend emotions and human traits, such as humour, irony and sarcasm (for which providing context will be a requirement). Previous research efforts on this area includes, on the one hand, Hogenboom et al. work [29] that focuses in using rhetorical structure in sentiment analysis, and utilises structural aspects of text as an aid to distinguish important segments from those less important, as far as contributing to the overall sentiment being communicated. As such, they put forward a hypothesis based on segments' rhetorical roles while accounting for the full hierarchical rhetorical structure in which these roles are defined. Heerschop et al. [26] propose a Rhetorical Structure Theory (RST) based approach [36], called *Pathos*, to perform document sentiment analysis partly based on the discourse structure of a document. Text is then classified into important and less important spans, and by weighting the sentiment conveyed by distinct text spans in accordance with their importance, the authors claim that they can improve the performance of a sentiment classifier. The idea of applying discourse analysis to determine the parts of the text that are most relevant to the overall document sentiment is obviously relevant and could be helpful in achieving our overall aim by extending the model proposed in this paper.

3. Research methodology

The research methodology that will be used is discussed from three different perspectives: the process that will be followed, the data that will be used and the description of the indicators that will be utilised for measuring the performance of the proposed SA solution.

3.1. The process

In order to measure success, the proposed method should perform *the same or better* than today's accepted supervised machine learning text classification solutions when utilised in isolation. In the specific case of the SA problem, the proposed solution is compared against two supervised machine learning methods that enjoy a high level of acceptance and credibility in the text classification research community and that are relatively easy to implement: Naïve Bayes (NB) and Maximum Entropy (ME). At this time, we have opted not to compare results against a popular classification technique, Support Vector Machine (SVM), because we are focusing our research hypothesis at performing sentiment analysis at the sentence level, and research from Wang and Manning [61] demonstrated that Naïve Bayes actually outperforms SVMs for 'snippets': "[...] for short snippet sentiment tasks, NB actually does better than SVMs (while for longer documents the opposite result holds)." The comparison will focus on sentiment/opinion polarity determination.

3.2. The data

A natural question to answer at this point is what data to use to benchmark our results. The following subsections will describe the details of the data sets utilised in this study.

3.2.1. Twitter dataset

Based on the terms and conditions for the utilisation of the data and because of privacy acts' related regulations, many Twitter datasets have been withdrawn from public access as a request from Twitter. However, despite the fact just mentioned, there are still a few Twitter datasets available publicly. We have chosen two of them. The first one we have utilised is *Sentiment140*, available at `http://help.sentiment140.com/for-students`. This dataset offers Twitter corpus data available at their site in CSV format and any trace of emoticons has been removed. We will call this data set Twitter A. The second dataset, provided by Twitter Sentiment Analysis Training Corpus, can be downloaded at `http://thinknook.com/twitter-sentiment-analysis-training-corpus-dataset-2012-09-22/`. It contains approximately a million and a half classified tweets, each row is marked as 1 for positive sentiment and 0 for negative sentiment. The dataset is based on data from two sources: University of Michigan Sentiment Analysis competition on Kaggle and the Twitter Sentiment Corpus by Niek Sanders. We have randomly chosen 1000 tweets of each type (negative and positive) that have been used in our experiments. We will call this dataset Twitter B. This dataset required much more cleansing effort when compared to the Twitter A data (there were numerous errors, mistypes, emoticons, strange characters, etc., that needed to be removed).

3.2.2. Movie Review dataset

Pang and Lee [44] published the datasets that were utilised in SA experiments and for which the results were addressed and discussed in [42, 43, 45]. The datasets are subdivided into categories, namely, sentiment polarity datasets, sentiment scale datasets and subjectivity datasets. As such, it seems adequate to use the Movie Review Dataset provided by Pang and Lee that is available at `http://www.cs.cornell.edu/people/pabo/movie-review-data/`. The fact that many articles in SA discuss this dataset and have used it to validate their own methods and approaches makes it an ideal candidate from the benchmarking angle.This dataset was first used in the experiments described in [43].

3.3. Most commonly used measurements in the evaluation of SA

It has become customary to evaluate the performance of sentiment classification systems utilising the following four indeces, as defined in [23, 55] (refer to the so-called *confusion matrix* given in Table 1):

- *Accuracy* – the portion of all true predicted instances against all predicted instances:
$$\frac{TP + TN}{TP + TN + FP + FN}$$
- *Precision* – the portion of true positive predicted instances against all positive predicted instances:
$$\frac{TP}{TP + FP}$$
- *Recall* – the portion of true positive predicted instances against all actual positive instances:
$$\frac{TP}{TP + FN}$$
- *F1-score* – a harmonic average of precision and recall:
$$\frac{2 \times Precision \times Recall}{Precision + Recall}$$

	Predicted Positives	**Predicted Negatives**
Actual Positive instances	# of True Positive (TP) instances	# of False Negative (FN) instances
Actual Negative instances	# of False Positive (FP) instances	# of True Negative (TN) instances

Table 1: Confusion Matrix

Readers are referred to the work by Sadegh et al. [23, 55] for more elaborated details on these performance indicators.

4. A hybrid approach to the SA problem at the sentence level

Let us clarify what we mean by utilising a 'hybrid approach' that is key to our proposed solution. Our intention is to manage *hybrid concepts* at two different levels: (a) the methods employed by the sentiment classifiers, and (b) the techniques used to build key components in our approach, like the creation and population of the sentiment/opinion lexicon, and the word dictionaries. The following paragraphs will discuss the different components of our proposed hybrid solution. For a graphic depiction of our proposed system, see Fig. 1.

4.1. Component 1: the sentiment/opinion lexicon

Liu compiled an opinion lexicon that "does include a list of positive and negative opinion words or sentiment words for English (around 6,800 words) [...] compiled over many years starting from [their] first paper [30]" (`http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html#lexicon`).

These opinion lexicon words will be used as a starting point and they will be enriched in a number of ways, including a new structure and organisation more adequate for the research

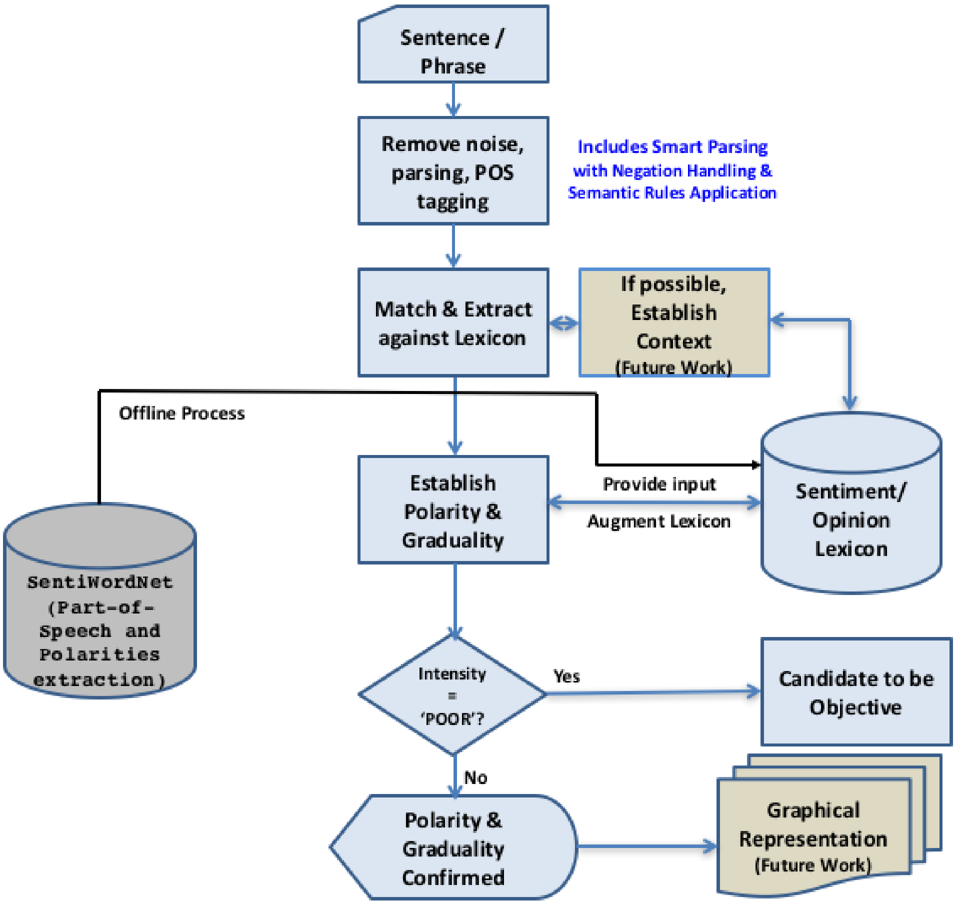

Sentence / Phrase
Remove noise, parsing, POS tagging
Includes Smart Parsing with Negation Handling & Semantic Rules Application
Match & Extract against Lexicon
If possible, Establish Context (Future Work)
Offline Process
Establish Polarity & Graduality
Provide input
Augment Lexicon
Sentiment/ Opinion Lexicon
SentiWordNet (Part-of-Speech and Polarities extraction)
Intensity = 'POOR'?
Yes
Candidate to be Objective
No
Polarity & Graduality Confirmed
Graphical Representation (Future Work)

Figure 1: View of our proposed hybrid approach

approach proposed here. Part of the reasoning behind using Liu's lexicon is to re-use data resulting from an existing good quality example of words compilation and as a point of *commonality* with previous research efforts for benchmarking purposes. In generating our own sentiment/opinion lexicon we have taken the following actions: (a) we have utilised, as a starting point, the *opinion-conveying-words* that are part of the opinion lexicon used by Hu and Liu in [30]. These words correspond *only* to four elements of part-of-speech (PoS) that have been proved to be capable of delivering opinions [24, 25, 33, 65]: nouns, verbs, adjectives and adverbs; (b) we have used SentiWordNet [19–21], which extends some of the functionality of WordNet [22, 39], as a source of *polarity or valence scores* for words that were originally in Hu and Liu's [30] list. As such, we have taken the words supplied by Hu and Liu, and looked them up in SentiWordNet. For those terms matching, the Positive, Negative and Objective scores available in SentiWordNet have been extracted and entries in our lexicon have been created combining the words from Hu and Liu's list and the semantic attributes present in SentiWordNet.

In terms of the characteristics of the polarity scores extracted from SentiWordNet, it is important to keep in mind that the polarity scores belong in the interval $[0, 1]$. Hence

$$0 \leq PositiveScore,\ NegativeScore,\ ObjectivityScore \leq 1$$

$$0 \leq (PositiveScore + NegativeScore) \leq 1$$

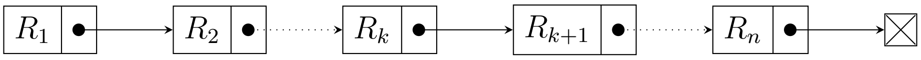
$$ObjectivityScore = 1 - (PositiveScore + NegativeScore)$$

As such, when the sum of $PositiveScore$ (PSC) and $NegativeScore$ (NSC) is equal to 1 for a given word $Word_k$, then the term $Word_k$ is fully opinionated, as opposed to the case when the addition of these two scores is zero, in which case the term $Word_k$ is fully Neutral or Objective. The *Objectivity Score* (COBJ) can be seen as a value of ambiguity/hesitancy, as it is the difference between 1 and the classification of a word as a negative/positive carrier of meaning. However, if PSC and NSC add, for example, to 0.9, then there is 0.1 points for the given word to occupy a semantic neutral space or hesitancy.

Not every word in Liu's opinion lexicon is present in SentiWordNet. Hence, for those absent words we have chosen to keep them in the new opinion lexicon, but they do not have polarity scores associated nor the proper PoS tag. They have been flagged in a special way so they can be recognised and enriched once the required information becomes available. Entries in the sentiment lexicon have the following structure:

$$R = (Word,\ SOL,\ PoS,\ PSC,\ NSC,\ COBJ,\ VDX,\ UPDC)$$

then we get the following graphical representation for elements in the sentiment lexicon (n = *length*(*lexicon*) = number of words in the lexicon):



A full description of the components of our Sentiment/Opinion Lexicon follows:

Word: word in the lexicon (entries).

SOL: semantic orientation label (pos/neg); inherited from Hu & Liu list [30].

PoS: part of speech (n=noun; v=verb; a=adjective; r=adverb; s=adjective satellite).

PSC: Positive Score as taken from SentiWordNet [20].

NSC: Negative Score as taken from SentiWordNet [20].

COBJ: Calculated Objectivity Score.

VDX: Versioning index for identifying/managing synonyms (future use).

UPDC: Update Counter to keep track of every time a given entry in the lexicon is updated.

Section 4.4.1 will explain the mechanics of how the sentiment lexicon is utilised.

Notice that our prototype has been built as a proof of concept tool, and not *yet* as a finished software product. Several of the programming constructs and data structures used correspond to native features of the programming language used for creating the prototype: Scheme [17, 60], a dialect of LISP [38]. As part of future work, we intend to port the code to a member of the family of the C programming language. Then, we will focus on algorithm efficiency by using the appropriate data structures (e.g. hash map instead of list data type) and effective programming techniques.

4.2. Component 2: semantic rules (SR)

In a classical SA approach with linguistic content semantic rules are utilised, as they assist in modelling the SA problem in a more rigorous fashion. In addition to the most common rules, a number of authors, among them [41, 59, 69], have pointed out the fact that having rules for negation handling and to deal with the use of specific PoS particles, like 'but, despite, unless, ...' could positively affect the final outcome of a classification exercise.

Thus, some rule strategy is needed to be put in place as the order of the different PoS play a role in the semantic of a sentence. Researchers have been, through time, improving the quality of these semantic rules so that they are more encompassing of the possible cases that must be managed. These research efforts are summarised by Xie et al. in [69], which includes a full presentation of their semantic rules approach. Up to certain extent, the semantic rules devised in the current hybrid proposed system are based on those presented by Xie et al, and as such a subset of the rules the aforementioned authors presented is utilised with the incorporation of new rules. We have followed the same naming convention for rules utilised by Xie et al. (R1, R2, ..., R13) with the addition of a super-index (Rk^{HSC}) to represent the rules actually utilised in the proposed method. Gaps in the sequential numbering utilised by Xie et al. represent rules that have not been implemented (R2, R4, R5, R8 and R9). The semantic rules utilised in the proposed method are displayed in Table 2 and Table 3.

Rule	**Semantic Rules**	**Example**
$R1^{HSC}$	Polarity (not var_k) = -Polarity (var_k)	'not *bad*.'
$R3^{HSC}$	Polarity (NP_1 VP_1) = *Compose* (NP_1, VP_1)	'*Crime* has decreased.'
$R6^{HSC}$	Polarity (ADJ to VP_1) = *Compose* (ADJ, VP_1)	'*Unlikely* to destroy the planet.'
$R7^{HSC}$	Polarity (VP_1 NP_1) = *Compose* (VP_1, NP_1)	'*Destroyed* terrorism.'
$R10^{HSC}$	Polarity (not as ADJ as NP) = -Polarity (ADJ)	'That wasn't as *bad* as the original.'
$R11^{HSC}$	If sentence contains "but", disregard all previous sentiment and only take the sentiment of the part after "but".	'And I've never liked that director, *but* I loved this movie.'
$R12^{HSC}$	If sentence contains "despite", only take the sentiment of the part before "despite".	'I love the movie, *despite* the fact that I hate that director.'
$R13^{HSC}$	If sentence contains "unless", and "unless" is followed by a negative clause, disregard the "unless" clause.	'Everyone likes the video *unless* he is a sociopath.'

Table 2: Semantic rules actually implemented in our Hybrid Approach (HSC)

Compose Functions Revised	**Algorithms**
Compose (arg1, arg2)	1. Return -Polarity(*arg2*) if *arg1* is negation.
	2. Return Polarity(*arg1*) if (Polarity(*arg1*) = Polarity(*arg2*).
	3. Otherwise, return the majority term polarity in *arg1* and *arg2*.

Table 3: Compose function implemented in HSC

Despite the apparent completeness of existing semantic rules by Xie et al., we have incorporated *two new rules* for managing particular PoS particles that were not included in the original set of rules provided in [69]: the particle **while** and the particle **however**. These new rules are given in Table 4.

Section 4.4.1 will explain the mechanics of how the semantic rules are put at work.

Rule	Semantic Rules	Example
$R14^{HSC}$ New	If sentence contains "while", disregard the sentence following the 'while' and take the sentiment only of the sentence that follows the one after the 'while'.	'*While* they did their best, the team played a horrible game.'
$R15^{HSC}$ New	If sentence contains "however", disregard the sentence preceding the 'however' and take the sentiment only of the sentence that follows the 'however'.	'The film counted with good actors. *However*, the plot was very poor.'

Table 4: New semantic rules extending those presented by Xie et al. in [69]

4.2.1. Negation effects

According to the well-known researcher Christopher Potts, Stanford University, Linguistics Department (`http://sentiment.christopherpotts.net/lingstruc.html`), "sentiment words behave very differently when under the semantic scope of negation". Dr. Potts notices that the so-called 'Weak' (mild) words such as good and bad behave like their opposites when negated (bad = not good, good = not bad), whilst 'Strong' (intense) words like *superb* and *terrible* have very general meanings under negation. According to Potts [53] "not superb is consistent with everything from horrible to just-shy-of-superb, and different lexical items for different senses. These observations suggest that it would be difficult to have a general *a priori* rule for how to handle negation. It does not just turn good to bad and bad to good. Its effects depend on the words being negated. An additional challenge for negation is that its expression is lexically diverse and its influences are far-reaching (syntactically speaking)". The method that Dr. Potts seems to favour for approximating the effects of negation is due to Das and Chen [16] and Pang, Lee, and Vaithyanathan [45]. When this method is incorporated at the tokenization level, the negation problem is relatively well managed. Let us look at an example:

Example 1. *I don't think I will enjoy it: it might be too spicy.*
As per the negation handling technique just mentioned, all words between the negation particle *don't* and the colon (:) would be tagged with the suffix '_NEG', clearly defining the scope of the negation. All words after the colon (:) would not be tagged at all.

Notice that even long-distance effects can be effectively managed. In our proposed approach, we have *chosen* to apply this *smart tokenization* strategy. There are good reasons for that. First of all, it saves us time as the scope of negation is defined early on, and if a polarity inversion is required, it can be done at tokenization time. Secondly, if a part of a sentence is identified as one that will not contribute to the final semantic orientation, then such part of the sentence can be discarded at this point, minimising the effort required at sentiment computing time.

4.3. Component 3: fuzzy sets approach to the SA problem

We have already established that we rely on an opinion lexicon that has been developed using a number of techniques and that started out with opinion-conveying words that were compiled by linguists and other scientists interested in the SA problem. In this sub-section we address the rest of components necessary to be able to classify sentences into *Positive* or *Negative* and, in addition, qualify the strength of the associated polarity. In order to do that we must address the following:

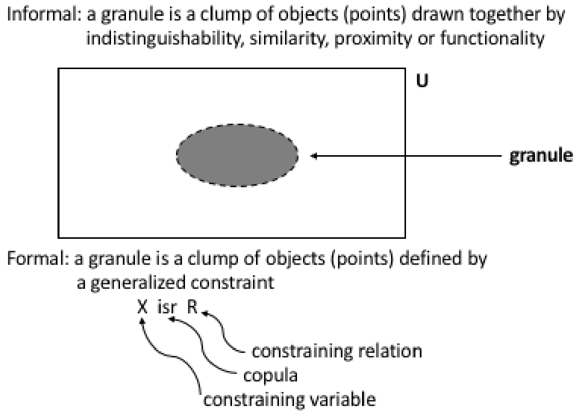

Informal: a granule is a clump of objects (points) drawn together by indistinguishability, similarity, proximity or functionality
U
granule
Formal: a granule is a clump of objects (points) defined by a generalized constraint
X isr R
constraining relation
copula
constraining variable

Figure 2: The Concept of a Granule as presented by Zadeh

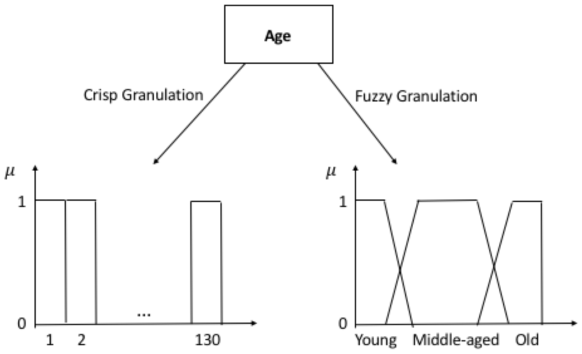


Figure 3: Crisp Granulation and Fuzzy Granulation as introduced by Zadeh

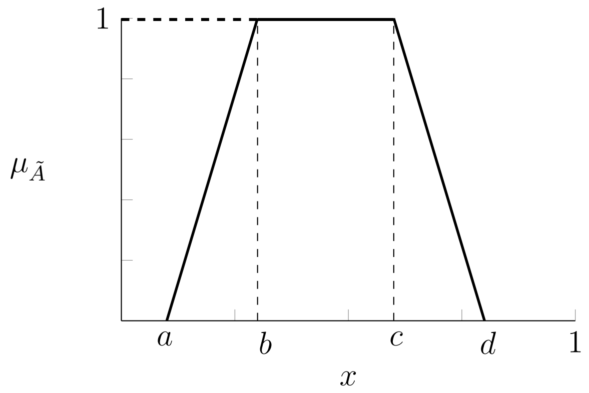
- Describe and construct the *fuzzy methodology* that will be utilised in this effort.
- Describe the fuzzy granulation, i.e. the linguistic discrimination, that will be implemented to represent the subjective classification of sentences into positive or negative.
- Provide the logic necessary, in combination with the lexicon and the fuzzy sets already mentioned, to address the classification problem at hand.
- Describe the mechanics behind the whole process as we incorporate the use of fuzzy sets components.

4.3.1. Basic concepts on perceptions and linguistic variables for polarity intensity

When we refer to Natural Languages (English in our case) it is clear that humans have developed the ability to classify objects, without the need to produce an actual measurement. When we say that someone or something is *slightly* large, *very* large, or *not very* large, we all understand the message. However, we have not measured or produced metrics for the object we are referring to. We continually use perceptions in the context of multiple events. According to Zadeh [73]: "reflecting the bounded ability of the human brain to resolve detail, perceptions are intrinsically imprecise. In more concrete terms, perceptions are f-granular, meaning that (1) the boundaries of perceived classes are unsharp and (2) the values of attributes are granulated, with a granule being a clump of values (points, objects) drawn together by indistinguishability, similarity, proximity, and function" (see Fig. 2 for a re-illustration of the graphic originally published by Zadeh in [74]). In [74], Zadeh continues, by saying that "a granule may be crisp or fuzzy, depending on whether its boundaries are or are not sharply defined. For example, age may be granulated crisply into years and granulated fuzzily into fuzzy intervals labeled very young, middle-aged, old and very old." Fig. 3 re-illustrates the graphical representation of the latter idea as originally presented by Zadeh in [73]. When it comes to the *Theory of Perceptions*, Zadeh's contribution is unique [73, 74]. Additionally, in 1973 Zadeh introduced the concept of linguistic variables: "a variable whose values are words instead of numbers" [72].

When deciding which *linguistic variables* to use in modelling our problem, we came to the realisation that the *intensity* or *degree of polarity* with which the grade of positivity or negativity of a sentence X could be understood corresponds to a *perception.* More specifically, the perception P_X that a given person Y has about how positive or negative a sentence X might be. A sentence could either be *Negative* or *Positive*, and then again 'Most

Positive' or 'Very Positive', or 'Most Negative' or 'Very Negative', and so on. Based on the definitions and concepts provided above by Zadeh, a fuzzy granulation of positive/negative sentiment using fuzzy intervals is considered appropriate. According to G.A. Miller [40], *7 plus or minus 2*, is the effective number of categories that a subject (individual or person) can maintain. In our case, we have chosen a conservative approach and have devised 5 labels (7 minus 2), symmetrically distributed in the domain $[0, 1]$. Additionally, our choice of trapezoidal function obeys to the fact that it generalises a triangular function and we have aimed for both: more generality and for more than one value at the top of every category. A trapezoidal membership function (MF), as shown in Fig. 4, is usually represented by the following 4-tuple (a, b, c, d).



$$\mu_{\tilde{A}}(x) = \begin{cases} 0 & \text{if} \quad x \leq a; \\ \frac{x-a}{b-a} & \text{if} \quad a \leq x \leq b; \\ 1 & \text{if} \quad b \leq x \leq c; \\ \frac{d-x}{d-c} & \text{if} \quad c \leq x \leq d; \\ 0 & \text{if} \quad d \leq x. \end{cases}$$

Figure 4: Trapezoidal membership function

Specifically, the following granules on the perception of the *positivity* or *negativity* of a given sentence X are suggested: $G = \{Poor;\ Slight;\ Moderate;\ Very;\ Most\}$, with the following 4-tuples:

- MF (Poor): (0, 0, 0.050, 0.150)
- MF (Slight): (0.050, 0.150, 0.250, 0.350)
- MF (Moderate): (0.250, 0.350, 0.650, 0.750)
- MF (Very): (0.650, 0.750, 0.850, 0.950)
- MF (Most): (0.850, 0.950, $1, 1$)

Thus, the *intensity* associated with the semantic positive/negative scores for words occupies a certain fuzzy interval as Fig. 5 illustrates. Section 4.4.2 will explain the mechanics of how the fuzzy sets described are utilised in determining the graduality of the intensity of polarity.

Calculating the level of intensity on the polarity of a sentence is advantageous in itself, as now it can be determined how strong or weak a given positive/negative sentiment might be in natural language. Hence, we are able to say that the sentiment towards a specific sentence is *moderately* positive/negative, *poorly* positive/negative, *most* positive/negative, etc. as per the linguistic labels we have already defined in section 4.3.1. Indeed, linguistic polarity intensity could be amenable to be further processed via the *computing with words* methodology introduced by Zadeh in [74], enabling computing with sentiments to be realised in practice.

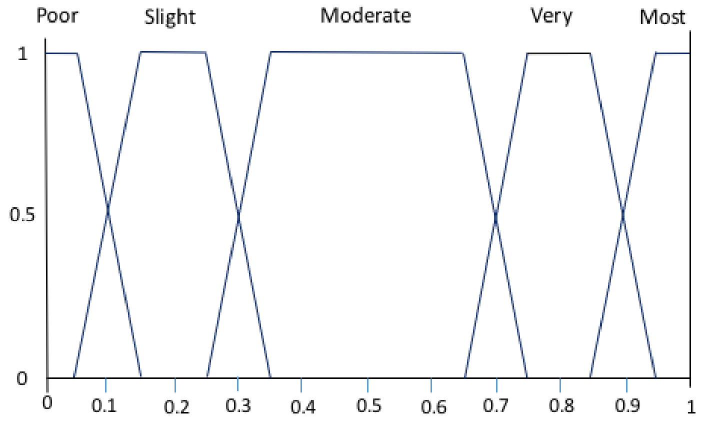


Figure 5: Linguistic variables, fuzzy granulation and trapezoidal membership functions

Natural Languages are a prime example of ambiguity, hidden meanings and multiple interpretations. In future, the proposed method could be taken to a next level, which should include the ability to incorporate approximate reasoning, via Fuzzy Logic, by bringing in the capability of computing with sentiments carried by words and sentences. Potential application around social media, such as product review sites, seems an obvious choice because the sentiment around a specific product based on the linguistic labels that the reviewers have given to the product (good, bad, acceptable, etc.) are possible to be computed. In this case, the entities manipulated to calculate the accumulated or aggregated sentiment would be **words** as opposed to **numbers**. Let us keep in mind that a sentiment in social media is typically expressed *in words and not in numbers*; at least for the regular user. In Fig. 6, a word is a label of a fuzzy set and those example labels are *poor, most and slight*. The computation of the aggregated sentiment is performed by directly manipulating the sentiment labels provided by each reviewer. In addition, in the presence of a proper fuzzy logic system, deductions can also be made out of facts expressed in words or determined via the SA approach presented here. In conjunction with social network analysis (SNA) [63, 66–68], it could be possible for a company, for marketing purposes for example, to identify the most influential nodes in a network that have very or most positive sentiment towards a particular product [47–49].

4.4. The hybrid approach and Its process

In this section we will describe how all the pieces fall into place in our proposed hybrid method in order to calculate both the sentiment polarity and the intensity of such polarity. Our approach consists of 2-steps, which will be described in the next two sub-sections.

4.4.1. Hybrid Standard Classification (HSC): calculating the polarity of sentiments in sentences

There are several tasks that must be executed in strict order in order to determine the polarity of sentences. Every intermediate step has an outcome that is consumed by the next step. Briefly, the tasks are as follows:

1. Tokenization, error cleansing, PoS tagging and smart parsing. The semantic rules as per section 4.2, are applied at tokenization/tagging time, and when applicable, sub-sentences may be discarded at that point (e.g. if the particle ‘but’ is present,

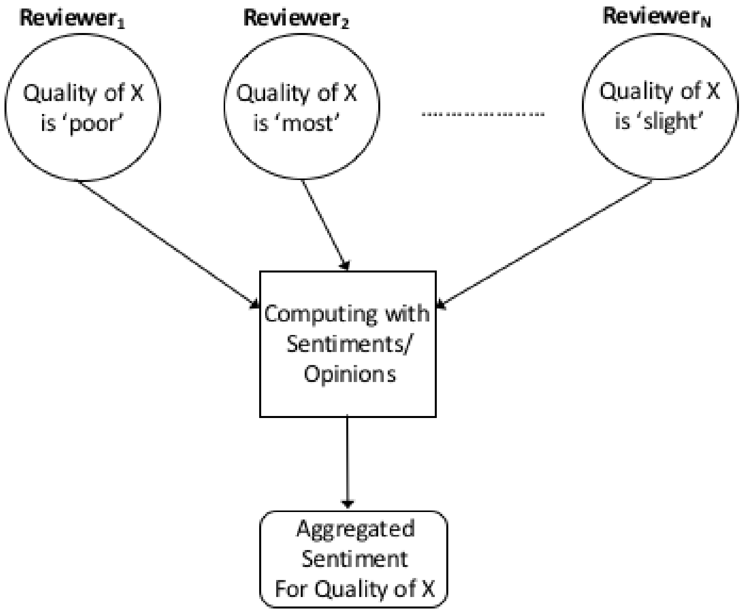


Figure 6: Computing with Sentiments - General Diagram

the sub-sentence before the particle 'but' will be discarded). In addition, if a sentence is made of two or more sub-sentences, the proper tagging is performed, so that at *interpretation time* the overall polarity is calculated as per the appropriate composition rule (Table 3). This step would imply changing the polarity of a given word-sentiment-carrying particle if such a particle is *negated*.

2. The resulting essential particles that convey sentiment/opinion (adjective, nouns, verbs and adverbs) are looked up in the sentiment lexicon bringing across the semantic properties (PoSs and polarity scores) of the term matched.

3. Words that are not in the Opinion Lexicon are tagged as such. Details on how these words are treated are provided in Section 4.5.

4. The semantic orientation (SOR) of each sentence is calculated following the process described in the first paragraph after this itemised list.

5. After processing a given sentence, those words without a pos/neg label are treated as an exception. This situation happens when the word in question was not in SentiWordNet or it was present, but there were no polarity scores available. More details are given in Section 4.5.

6. The overall sentiment of the sentence is produced, and if indicated as such by the semantic rules, a composition is performed for those sentences made of a collection of sub-sentences to derive its compounded semantic orientation (CSO) based on its sub-sentence SOR values. The actual process of computing a sentence SOR is addressed in the paragraph below.

Computing a sentence SOR. During the actual semantic orientation calculation, both the pos/neg label associated with the words in the lexicon and their respective polarity scores, are taken into consideration. The proposed system performs word counting of both orientations (neg/pos) for every sentence.

If *count*(positive words) $>$ *count*(negative words)
then [the sentence is classified as ‘positive’], hence SOR = ‘Positive’

If *count*(positive words) $<$ *count*(negative words)
then [the sentence is classified as ‘negative’], hence SOR = ‘Negative’

If *count*(positive words) = *count*(negative words)
then [There is a tie. Follow alternative process], hence SOR = Table 5 result.

Strata	**Task**
Stratus 1	The polarity scores or intensity of polarity (IP) are reviewed and the highest value (among negative and positive words) wins
Stratus 2	There is a hierarchy of importance around the PoS particles to which the words in a sentence belong. The aforementioned hierarchy, from most *influential* to least influential is: (i) adjectives, (ii) adverbs, (iii) verbs and (iv) nouns. If the previous step fails to produce a classification, the hierarchy just described is used and a higher priority is assigned to the IP values of adjectives, followed by adverbs, verbs and nouns
Stratus 3	If the two previous steps fail, we examine our dictionary and search for the participant words; we extract the frequencies with which each word has appeared in a sentence with a specific polarity (pos/neg); the polarity associated to the highest value wins

Table 5: Stratified Algorithm for Tie Breaks

Ties are resolved using a three-level stratified algorithm, as displayed in Table 5. The different strata shown are mutually exclusive, and every step is executed *only if* the previous step does not resolve the existing tie. As the positive/negative IP values in our lexicon range is $[0, 1]$, the semantic orientation calculation requires: (i) the Positive/Negative label in our lexicon (SOL) and (ii) the positive/negative IP values in our lexicon.

If a sentence S is made of n sub-sentences $(S_1, S_2, \ldots, S_n)$, then the CSO of the full paragraph/sentence is calculated by SOR sub-sentence counting.

1. **If** *count*(Positive SOR Sentences) $>$ *count*(Negative SOR Sentences)
   then $CSO_{(S_1,S_2,\ldots,S_n)}$ = Positive

2. **If** *count*(Positive SOR Sentences) $<$ *count*(Negative SOR Sentences)
   then $CSO_{(S_1,S_2,\ldots,S_n)}$ = Negative

3. **If** *count*(Positive SOR Sentences) = *count*(Negative SOR Sentences)
   then $CSO_{(S_1,S_2,\ldots,S_n)} = SOR$ of S_k; $IP(S_k) = \max\{IP(S_1), IP(S_2), \ldots, IP(S_n)\}$

Notice that the natural separators for sentences are punctuation marks (period, comma, exclamation sign, question mark, colon, semicolon, etc.), and naturally the sentences would be broken accordingly into sub-sentences at tagging/parsing time. For complex data inputs, like long paragraphs/sentences or even documents, the expectation is that there would be many sub-sentences participating in multiple compositions. For short paragraphs or snippets, like those exhibited in the Twitter datasets, we will have to compose semantic orientations for a low number of sub-sentences. By inspecting the data, there would be a composition of 3 to 5 sub-sentences at the most, with the majority of the cases being restricted to 2 or 3 of them. For Twitter data, it is not uncommon that the author of the tweet simply does not use punctuation marks at all, which would result in zero sub-sentences at tokenization time (just one longer than usual sentence).

4.4.2. Hybrid Advanced Classification (HAC): computing the intensity of polarity

This approach enhances the standard classification process by incorporating:

1. Determination of the intensity of polarity (IP) with which a given sentence leans towards being *positive* or *negative*.

   The IP of a sentence (X) is to be derived from the IP values of its associated list of sentiment-carrying words ($W_1, \ldots, W_n$). In other words, the partial IP values of words of a sentence X are to be fused appropriately to derive the global sentence IP value. Mathematically, this problem means that an appropriate mapping $f\colon [0,1]^n \to [0,1]$ is needed to be defined such that:

   $$IP(X) = f\big(IP(W_1), ..., IP(W_n)\big).$$

   Fusion operators can be roughly classified into the following categories: conjunctive, disjunctive and compensative: (i) Conjunctive operators that behave like a logical "and". In this case, the global IP value is high only when all the partial IP values are high but compensation is not possible as the presence of just one small partial IP value will result in a small global value no matter how big the rest of partial IP values are. A well known family of conjunctive operators is the t-norm family, and the minimum operator is the largest of all t-norms; (ii) Disjunctive operators behave like a logical "or", and can be seen as the dual operators of conjunctive operators. In this case, the global IP is low only when all the partial IP values are low. As with conjunctive operators, compensation is not possible as the presence of just one high partial IP value will result in a high global IP value no matter how low the rest of partial values are. The family of t-conorms belongs to this type of operators, and the maximum is the smallest of all t-conorms; (iii) Compensative operators are comprised between the minimum and the maximum, and therefore they are neither conjunctive nor disjunctive. In this kind of operators, a small partial IP value can be compensated by a high partial IP value. This type of operator is also known as an averaging operator with mean, weighted mean and ordered weighted averaging (OWA) operator being widely used in multicriteria decision making problems. A class of fusing operators that behaves like a conjunctive operator when all values are low, like a disjunctive operator the all values are high, and like a compensatory operator otherwise exists, is the family of uninorms operators [70].

   Definition 1. *A uninorm operator U is a mapping $U\colon [0,1]^2 \longrightarrow [0,1]$ having the following properties:*

   (a) Commutativity: $U(x,y) = U(y,x)$

   (b) Monotonicity: $U(x_1,y_1) \geq U(x_2,y_2)$ if $x_1 \geq x_2$ and $y_1 \geq y_2$

   (c) Associativity: $U(x,U(y,z)) = U(U(x,y),z)$

   (d) Identity element: $\exists\, e \in [0,1] :\ \forall\, x \in [0,1],\ U(x,e) = x$

   Uninorm operators share with t-norm and t-conorm operators the commutativity, associativity and monotonicity properties. Furthermore, the uninorm operator generalises both the t-norm operator and the t-conorm operators. In general, a uninorm operator has an identity element lying anywhere in the unit interval $[0,1]$; a t-norm operator has 1 as its identity element and therefore it is a uninorm operator with identity element 1; while a t-conorm operator has 0 as its identity element and therefore it is a uninorm operator with identity element 0. It is well known that a uninorm operator with identity element $e \in [0,1]$ behaves like (i) a t-norm operator when all partial IP values are below e; (ii) a t-conorm operator when all partial IP values are above e; (iii) a compensative operator in the presence of partial values below and above e. An interesting particular case of uninorm operators are the symmetric aggregative operators, i.e. uninorm operators that have a representation in terms of a single variable function. In particular, the representable uninorm operator with identity element $e = 0.5$ has been characterised as the most appropriate for modelling cardinal consistency of reciprocal preference relations [13].

Based on the above, a general approach in this step would be the implementation of a uninorm operator to derive the IP of a sentence X from the IP of its associated list of sentiment-carrying words. The experimental section reported in this paper made use of the minimum operator, which as mentioned above is a type of uninorm:

$$IP(X) = \min\{IP(W_1) \ \ldots \ IP(W_n)\}. \tag{1}$$

Once a sentence IP value is obtained, the linguistic labels (granules) $l \in G$ with highest $\mu_l(IP(X))$ is assigned to classify the positive/negative polarity. In the few cases when there exists two consecutive labels with equal $\mu_l(IP(X))$, we classify the polarity of the sentence with the label meaning higher as per the ordinal ordering implicitly expressed in the representation given in Fig. 5. For example, when $IP(X) = 0.3$, the polarity will be assigned the label *Moderate* rather than the label *Slight*.

2. Diagnosing when a given sentence could be considered rather *objective/neutral* as opposed to either *positive* or *negative*: not all sentences have been created equal, and even in the test dataset that have been carefully chosen, there are some sentences that one could argue that are rather neutral (not leaning towards negative or positive). With the Hybrid Advanced Classification (HAC) system we could consider those sentences classified as having an IP belonging in the *poor interval*, as prime candidates to have a Semantic Orientation that would be leaning more towards *Objective* than to *Subjective*.

4.5. Sentiment lexicon enrichment

This section addresses the classification issue that arises when dealing with sentences for which the data in the lexicon in not enough, as a SOR score cannot be produced. A response to such issues is presented in the form of an almost automated approach to enriching the sentiment lexicon by incorporating new opinion-carrying particles into the lexicon to minimise the number of cases when polarity classification is not possible.

Dealing with sentences when the data in the lexicon in not enough. One problem that we will encounter is that a given sentence being processed would include words that are not in the lexicon (none of them are in the lexicon). In such a case, our method cannot provide a *SOR* recommendation. If at least one word would have been included in the lexicon, the *SOR* would have been calculated using the information available, utilising the computation process given in the previous subsection. At this point, the only way forward is to incorporate the new words in the lexicon using some specific criteria as it is shown in the following paragraph. In the interim, no classification can be offered, but once the lexicon is updated, a polarity classification is very possible. Considering the frequency at which SentiWordNet is updated by the on-line feedback provided by regular users (`http://sentiwordnet.isti.cnr.it`), it is fair to say that the growth of the capabilities of our lexicon is guaranteed by the increase on the size of the corpora incorporated into SentiWordNet.

Enriching the sentiment/opinion lexicon. Methodologies based on the utilisation of a sentiment lexicon would eventually come across situations where words carrying opinions in a given sentence are *not* included in the lexicon. The only solution to this problem is to enrich the sentiment lexicon by either adding new words that were not originally part of the aforementioned lexicon or completing/modifying data attributes already in the lexicon. In this section we will focus on the former case and will explain how to incorporate new opinion-carrying particles into the lexicon.

In order to keep a strict control, we perform off-line the process of adding words to the lexicon using a semi-automated mechanism with some human intervention. The SentiWordNet database can be downloaded and utilised for off-line processing, which is the chosen mechanism. The process we have implemented is given below (every step in the list below uses the output of the previous step):

1. Process sentence using NLP techniques (tokenization, negation-handling, PoS tagging, parsing, etc.).

2. Compare sentiment-conveying words found in previous step against sentiment lexicon.

3. Obtain list of the words that were not found in the current/most-recent sentiment lexicon. These words are *candidates* to be added to the lexicon, but not all not-found words will be added to the lexicon; i.e. there are words that are not considered to be sentiment-carrying words, hence they should not be added to the lexicon.

4. Eliminate repeated words and check the PoS group the words belong to.

5. Compare off-line the list of words obtained in previous step against the available SentiWordNet's database and generate list of matches and no-matches.

6. Remove from the list generated in the previous step any particle not-found (there will be words that are not in SentiWordNet). The list of words generated in Step 5 represents candidate-words to be included in our Lexicon. However, those words that are not available in SentiWordNet cannot be added automatically to our lexicon, hence, they are rejected by being removed from the list of matches, and are placed instead in the list of exceptions that will require human intervention in order to make an educated decision.

7. Transform the list of matches from the previous step into a format that enables them to be potentially incorporated into our sentiment/opinion lexicon (the format of the lexicon of our proposed method).

8. Invoke the Lexicon Editor Program (written by our team) to provide a visual interface to an expert to analyse the candidate words already in opinion lexicon format. The human operator will decide the polarity label for each entry as the system prompts her with an input query. The expert (ideally, a linguist) will decide whether the candidate word should be (a) deleted, (b) classified as neutral polarity, or (c) classified as having negative or positive polarity.

9. Add the list of words previously obtained to the existing Sentiment Lexicon in order to generate an *updated* version of the lexicon.

Notice that all steps described above are automated by software that we have developed as part of our prototype, with the exception of step number 8. At a given point in time, some human interaction is usually required to make decisions about whether a given word should be part of the lexicon. A possible partial solution would be to simply add all words found in *SentiWordNet* as the latter is supposed to contain words labelled already as Positive, Negative or Objective. We have taken the latter approach in order to minimise user intervention. However, we still do a visual inspection *before* new words are added to the lexicon, in order to avoid the introduction of noise.

5. Experiments results

In this section we will look at the experimental results, starting with the outcome obtained when using the two mentioned SML methods, Naïve Bayes and Maximum Entropy. After this, we will show the results obtained from applying the proposed hybrid method. We will close the section with a comparison of the results. For the preparation and processing of the sentences using the SML classifiers mentioned above, we used extensively the material presented by Bird et al. [3] and Perkins [50].

5.1. Naïve Bayes classifier

In discussing the Naïve Bayes (NB) classifier, Pang et al. in [45] elaborate that one possible approach to text classification is to assign to a given document d the class $c^* = arg\,max_c\, P(c \mid d)$, with

$$P(c \mid d) = \frac{P(c)\, P(d \mid c)}{P(d)}.$$

In order to estimate $P(d \mid c)$, NB assumes the class features (f_i) are conditionally independent:

$$P_{NB}(c \mid d) := \frac{P(c)\ \left(\prod_{i=1}^{m} P(f_i \mid c)^{n_i(d)}\right)}{P(d)},$$

with $n_i(d)$ representing the number of possible classification classes (it would be 2 for a binary classifier).

In essence, NB is a probabilistic classifier that is based on the Bayes' theorem. Basically, in the presence of a sample input NB should be able to predict a probability distribution over a set of classes. In this case, word frequencies are the characteristics used to decide whether a paragraph belongs to one category or another. For that to happen, we would have to count with a dictionary (or corpus) previously labelled with the semantic orientation of words (i.e. 'fabulous' conveys a positive intention whilst 'terrible' would convey a bad one). However, despite its apparent simplicity, NB has proven to be very successful in many situations [45].

In the experiments, the NB classifier was trained using some of the recommendations presented by Perkins in [50]. The classifier uses the concept of 'bag of words' [64] to create 'feature vectors' exhibiting the main traits of each sentence. In this case, the NB classifier is a binary classifier, with a sentence being classified either as 'negative' or 'positive', with both categories being exclusive. The *movie_reviews* corpus available with NLTK 2.x was used to train the classifier. The training dataset consists of 1,500 instances (1,500 files containing full paragraphs) that have been pre-labelled as either *positive* or *negative.* There were 500 instances used to test the classifier. Once the classifier has been trained and tested, a *different* dataset of 'movie_reviews' sentences ($5,331$ sentences pre-labelled as Positive and $5,331$ pre-labelled as Negative), was used to evaluate all classifiers.

The NB classification algorithm returns a probability value that represents the sentence belonging with a specific label (negative or positive). Thus, the probability value has to be 0.5 or higher for a sentence to belong in a specific category (*positive* or *negative*, depending on which case is being tested). Table 6 shows the results obtained after running the NB classifier with the test datasets:

Metric	Twitter A dataset	Movie database dataset
Accuracy	0.6785	0.6717
Precision	0.6315	0.6274

Table 6: Naïve Bayes classifier performance indexes

5.2. Maximum Entropy classifier

The Maximum Entropy (ME) classification algorithm has been extensively used by the Machine Learning community to deal with text classification problems. One of the main properties of ME is that the algorithm does not make any assumptions about the relationships between features to derive the estimate of $P(c \mid d)$, which is expressed as follows:

$$P_{ME}(c \mid d) \;:=\; \frac{1}{Z(d)} \;\exp\left(\sum_i \lambda_{i,\,c}\, F_{i,\,c}(d,\; c)\right),$$

where $Z(d)$ is a normalisation function, and $F_{i,\,c}$ is the following *characteristic function* for feature f_i and class c:

$$F_{i,\,c}(d,\; c') \;:=\; \begin{cases} 1 & n_i(d) \;>\; 0 \text{ and } c' \;=\; c \\ 0 & \text{otherwise} \end{cases}$$

The $\lambda_{i,\,c}$'s are feature-weight parameters, and a large value for $\lambda_{i,\,c}$ would imply that "f_i is considered a strong indicator for class c. The parameter values are set so as to maximise the entropy of the induced distribution subject to the constraint that the expected values of the feature/class functions with respect to the model are equal to their expected value with respect to the training data" [45]. Additional information about ME can be found in the literature, for example, in [1, 4, 37].

Following Perkins's recommendations [50], the Generalized Iterative Scaling (GIS) learning method was used to train the ME classifier. As the NB classifier, the ME classifier returns a probability value of the sentence belonging with a specific label (negative or positive), and a probability value equal or higher than 0.5 is required for a sentence to belong in a specific category. Table 7 presents the results obtained when the trained classifier is applied to the test datasets:

Metric	Twitter A dataset	Movie database dataset
Accuracy	0.6759	0.6757
Precision	0.6293	0.6291

Table 7: Maximum Entropy classifier performance indexes

5.3. Proposed hybrid method (HSC/HAC)

The proposed hybrid method has been applied to the test datasets in two different sub-methods or incarnations that grow each time in complexity (HSC and HAC). The results of having applied the classification method HSC (polarity determination) to the test datasets, are followed by the results obtained by applying the HAC method to the test dataset (determining polarity intensity). The proposed hybrid method is first applied to both twitter datasets, Twitter A dataset and Twitter B dataset, and later on to the movie database dataset.

5.3.1. HSC results

Notice that we have a first and a second pass of the proposed HSC method. When running the experiments, we reset our lexicon to its initial state every time, for the sake of comparison against different data sets. The 2^{nd} pass of the method corresponds to the phase in which the sentiment lexicon and dictionary have learnt new words/terms. The latter mimics better real life scenarios, as every new run of our method should benefit from what it has already learnt.

Table 8 presents the HSC (1^{st} pass and 2^{nd} pass) results when applied to the data marked in this article as Twitter A dataset, while for performance confirmation purposes Table 9 presents the HSC 2^{nd} pass results when applied to the data marked in this article as Twitter B dataset. Table 10 presents only the HSC 2^{nd} pass results when applied to the test data marked as Movie Review dataset.

Metric	**HSC (1^{st} pass)**	**HSC (2^{nd} pass)**
Accuracy	0.8728	0.8802
Precision	0.8280	0.8424

Table 8: HSC classifier - Twitter A dataset performance indexes

Metric	**HSC (2^{nd} pass)**
Accuracy	0.8655
Precision	0.8406

Table 9: HSC classifier - Twitter B dataset performance indexes

Metric	**HSC (2^{nd} pass)**
Accuracy	0.7585
Precision	0.7278

Table 10: HSC classifier - Movie Review dataset performance indexes

At the beginning of this article we mentioned that we are focusing on Sentiment Analysis at the sentence level. Initial experimental results show that the closer the data utilised reflect the concept of a snippet (a short sentence usually found in one line or equivalent in social media systems like Twitter), the better our proposed system performs. We will discuss this further in section 5.4.

5.3.2. HAC results

In subsection 5.3.1 we have seen the results obtained by the proposed hybrid method in terms of estimating the polarity of three different datasets. With HAC, we incorporate the fuzzy sets approach already described and as a consequence we can incorporate a *finer granularity level* into the polarity classification process. Details of the results achieved with the Movie Review dataset are provided in Table 11 and Table 12.

False Negatives	929
No Semantic Orientation (NOSOR)	35
NOSOR (2^{nd} run)	0
True Positives	4,402
Poorly	577
Slight	1,106
Moderate	1,041
Very	1,365
Most	313
Total Number of Sentences	5,331

Table 11: HAC classifier increased granularity for **Positive Polarity** dataset

False Positives	1,646
No Semantic Orientation (NOSOR)	76
NOSOR (2^{nd} run)	0
True Negatives	3,685
Poorly	770
Slight	1,089
Moderate	789
Very	864
Most	173
Total Number of Sentences	5,331

Table 12: HAC classifier increased granularity for **Negative Polarity** dataset

Notice that in the 2^{nd} pass there are no cases of NOSORs. During the 1^{st} pass, the proposed system either learnt new terms (words) that were added to the lexicon, or was capable of finding polarity scores to terms already resident in the sentiment lexicon. With this added granularity to the polarity classification, we can inspect sentences classified in the lower end of the spectrum $[0, 1]$ as well, i.e. sentences labelled as 'poor', and revise them because in terms of classification they could be borderline with Neutral/Objective. Representative examples are sentences like '*The theatre was completed full.*' and "*The Sinner counted with great actors.*", which seem to express facts instead of opinions.

There were no annotations for polarity intensities in any of the utilised datasets. This was expected as the datasets were annotated only for polarity (negative or positive). We did annotate 10% of all sentences in the larger dataset though (movie review) in the positive-polarity dataset, which represents approximately 530 sentences distributed as follows: 100 each for the Poor, Slight, Very and Most labels, and 130 for the Moderate one. Table 13 presents the indicators for the 10% sample, which are considered very hopeful as efficiency in predicting polarity intensity accurately was above 80% in all cases.

5.4. Comparison of experimental results

In this subsection we will take a closer look at the experimental results. The first comparison table (Table 14) corresponds to results achieved when the different methods

Metric	Poor	Slight	Moderate	Very	Most
No. of Sentences	100	100	130	100	100
Estimated Correct (%)	81.00	89.00	93.08	91.00	87.00
Estimate Incorrect (%)	19.00	11.00	6.92	9.00	13.00

Table 13: Movie Review Positive Polarity dataset sample - HAC classifier performance

were applied to the dataset identified in this article as Twitter A dataset. As stated before, the 2^{nd} pass represents better real life scenarios, as our lexicon has already learnt new terms and their associated properties. The results obtained are very encouraging as the proposed hybrid method improves the results obtained by NB/ME by a significant magnitude. As a performance confirmation exercise for the proposed hybrid method, the data marked in this article as the Twitter B dataset was used. Similar results to those achieved for Twitter A dataset were obtained, with an Accuracy of 0.8655 and a Precision of 0.8406.

Metric	NB	ME	HSC (1^{st} pass)	HSC (2^{nd} pass)
Accuracy	0.6785	0.6759	0.8728	0.8802
Precision	0.6315	0.6293	0.8280	0.8424

Table 14: Twitter A dataset performance indexes comparison - NB/ME vs. HSC

Moving on to the results obtained using the Movie Review dataset (Table 15), we observe that the proposed HSC method performs better than NB/ME, although it is worth mentioning that the overall performance of HSC (for precision) is reduced by approximately 11.46% with respect to its performance on the Twitter datasets. The explanation that we offer for this behaviour is data related. The sentences in the Movie Review dataset are rather complex (a short paragraph or a long sentence made up of a few sub-sentences on average). The type of sentences available in both Twitter datasets are of a simpler nature and they are closer to the concept of a 'snippet'. Let us keep in mind as well that the focus of our research was directed toward presenting a sentiment analysis approach *at the sentence level*. In the following paragraphs, examples of the type of sentences found in the different datasets are provided to support this analysis.

Metric	NB	ME	HSC (1^{st} pass)	HSC (2^{nd} pass)
Accuracy	0.6717	0.6757	0.7559	0.7585
Precision	0.6274	0.6291	0.7263	0.7278

Table 15: Movie Review dataset performance indexes comparison - NB/ME vs. HSC

Movie Review dataset examples

Example 2. *"it was with great anticipation that i sat down to view braveheart last week as it premiered on american cable. the academy award winning film had been highly acclaimed. it also featured the music of one of my favorite film composers , james horner . what i was in for was a disappointing and overlong film which was anything but the best picture of 1995 …"*

Example 3. *"Vampire's is a rude , chauvinistic movie where women are portrayed as pawns of abuse , present only to pleasure men , feed vampires , readied to be bashed or beaten - till one's sensibilities is shocked by the low iq and mentality of this regressive movie . to make matters worse , the buffoons that go hunting vampires are all rednecks , and deserve to have their heads bitten off , if not , their bodies carved in half ."*

Twitter A dataset examples

Example 4. *"To hell with this economy. I hate aig and their non loan given asses."*

Example 5. *'US planning to resume the military tribunals at Guantanamo Bay ... only this time those DTS on trial will be AIG execs and Chrysler debt holders.'*

Twitter B dataset examples

Example 6. *"There are huge lines at the Apple store."*

Example 7. *"I had to wait for six friggin' hours in line at the Microsoft store. that's not cool man."*

5.4.1. Impact of different techniques in hybrid approach

Next we would like to show the improvements in the performance of the proposed system as some of the techniques mentioned in this article were introduced. Table 16 shows the precision of the proposed HSC method as specific enhancements were applied one after another, and as such provides a picture of the the impact that the different techniques generated as they were added to the proposed solution. Indeed, every step shown in the table inherits the benefits of having introduced a specific technique in the previous step. This results in the precision of the final process being close to 10% higher than that at the start.

Technique incorporated	**Precision**	**Acc. Impact (%)**
Using pre-existing semantic rules	76.77	
Adding effective PoS tagging	79.33	3.33
Adding smart negation handling	81.17	5.73
Adding new semantic rules (R14 & R15)	83.36	8.58
After 2^{nd} pass (once the lexicon has learnt new terms)	84.24	9.73

Table 16: Impact of different techniques in hybrid approach precision (Twitter A dataset)

5.4.2. Analysis of specific examples

Let us take a closer look at some examples that we believe are of interest. We will start by showing instances that exemplified the different intensities in polarity and their associated linguistic labels, as produced by the proposed HAC method, and then we will continue sharing examples of sentences that have proven to be too hard to classify for the proposed classifier.

Examples of polarity intensity graduality as per the five linguistic labels introduced.

Example 8. *Poor: "effective but too-tepid biopic."*

Example 9. *Slight: "if you sometimes like to go to the movies to have fun, wasabi is a good place to start."*

Example 10. *Moderate: "occasionally melodramatic , it's also extremely effective."*

Example 11. *Very: "the movie's ripe , enrapturing beauty will tempt those willing to probe its inscrutable mysteries ."*

Example 12. *Most: "one of the greatest family-oriented, fantasy-adventure movies ever."*

Examples of challenging sentences for the proposed hybrid classifier.

Example 13. *"spiderman rocks."*
In this case, the proposed classifier does not understand what the term 'rocks' means. As such, the sentence was wrongly classified as having a negative polarity.

Example 14. *"it extends the writings of jean genet and john rechy, the films of fassbinder, perhaps even the nocturnal works of goya."*
This sentence offers the names of great world-class film directors and actors, and claims that the director of the reviewed movie extends their work. However, the lack of context impacts on the ability of the proposed algorithm to classify this sentence properly.

Example 15. *"after watching the movie I found myself between a rock and a hard place."*
In this instance, the use of idioms creates problems for the proposed classifier.

5.5. Performance comparison against Machine Learning and state of the art

An accurate and strict comparison cannot be performed unless every method involved is evaluated against exactly the same dataset, and in the case of lexicon driven methods when the same lexicon is used. Therefore, the values shown below are rather *informative* of the performance achieved by each method in different experimental settings. A comparison against state of the art techniques that are not purely machine learning based was not part of this research, but it will be performed in the near future once we execute some of the recommendations offered in Section 6 such as the replacement of SentiWordNet with SenticNet [9].

Poria et al. [52] provides results for machine learning experiments that we have reused in the comparison shown below in Table 17. Notice that the proposed hybrid method does approximately 17% better than the machine learning techniques in general, and 21.50% better against NB/ME. For some time, Socher et al [57, 58] have been considered state of the art. At the sentence level the proposed hybrid method performs better than Socher et al. [57] and is close to the performance of Socher et al. [58]. Ensemble classification from Poria et al. [52] has achieved the best performance of all, with a precision of 86.21%.

Algorithm	**Precision (%)**
Naïve Bayes (from section 5.1)	62.74
Machine learning [52]	67.35
HSC/HAC - Movie Review dataset	72.78
Socher et al. [57]	80.00
HSC/HAC - Twitter dataset	84.24
Socher et al. [58]	85.40
Ensemble classification [52]	86.21

Table 17: Proposed hybrid method against state of the art

6. Conclusions and further work

In general, our proposed hybrid system works very well at the sentence level with a high level of *accuracy* (88.02%) and *precision* (84.24%) when the method is applied against twitter-like datasets. The fact that our hybrid system significantly improved the results obtained using Naïve Bayes (NB) and Maximum Entropy (ME), satisfies our initial hypothesis that a hybrid method using sentiment lexicons, NLP essential techniques and fuzzy sets, should be able to perform well. Another benefit of our proposed system is that we have managed to identify different strengths in the polarity degree of the input sentences with regard to the specific base-case (negative or positive). There is an interesting and intended effect of the introduction of the fuzzy sets component of our method. Those sentences classified in the 'poor' side of the polarity intensity spectrum are prime candidates to be considered rather neutral or objective sentences, instead of subjective (this functionality could be built into a subjectivity determination schema).

Our expectation is that the quality of the content of SentiWordNet, or more recent tools like SenticNet [9], should continue to improve with time. Those enhancements will contribute to the betterment of our proposed hybrid solution as it will reflect positively in the quality of our opinion lexicon. In theory, as time passes, both SentiWordNet and our proposed opinion lexicon should become better and more complete. The ability to incorporate new terms into the current opinion lexicon in an expedite way is another benefit provided by our proposed solution.

In essence, hybrid techniques can play an important role in the advancement of the Sentiment Analysis discipline by combining together a number of elements that tend to produce better results. Similar results, in terms of combining techniques effectively, have been reported by other researches [52].

By carefully analysing the sentences processed using the aforementioned classification methods, we are capable of extracting the main characteristics in those sentences that posed a problem for our classification method. If we group together the cases that our system have considered challenging to classify properly, we find the following traits:

- The use of jargon, argot, idiom and/or lingo is hard to deal with, and sometimes it misguides the system in classifying opinions properly.
- Imagery, metaphors, similes, sarcasm, humour and other language figures that rely on previous knowledge and/or context represent a big challenge for our system. For future research, a starting point would be the work of Justo et al. [32].
- Double negation offers difficulties that we must continue to study and improve.
- In the presence of very complex paragraphs the precision of our proposed hybrid method is negatively impacted

In terms of further work, we believe there are a number of avenues that should be pursued in the short-term:

- Create an automatic real-time interface, via API, with SentiWordNet or functionally equivalent tool (see next item) to search dynamically for polarity and PoS tagging updates for all terms of interest.
- Investigate the possibility of using SenticNet [9] as a source of a more mature and comprehensive set of semantic attributes to enrich our own lexicon -or replace it. The concept-level sentiment analysis approach introduced by Poria et al. [52], sentic patterns, and its dependency-based rules for concept-level sentiment analysis could provide a broader semantic coverage than the one we currently enjoy with SentiWordNet.
- Work on an algorithm to incorporate context in the tagging/parsing process and in the sentiment-determination modules in order to improve the ability of the system to deal with humour, metaphors, similes, sarcasm and irony (an initial approach could be the utilisation of context-sensitive grammars during the parsing process).

- Port the proof-of-concept prototype code from Scheme to C, C++, C# or Python, in order to increase efficiency and integration capabilities.

- Continue developing a *computing with sentiments* approach, using as a foundation the method presented in this article.

- Introduce the use of Induced OWA operators [71] as an aggregation mechanism in the calculation of the semantic orientation of sentences. This type of operator could be used to ensure that specific values among those elements considered would drive the aggregation according to a predefined operator. This aggregation could be performed at both levels, the words participating in a sentence and the sub-sentences making up a full sentence or paragraph. Work on this approach has already started and we expect to report results rather soon.

References

[1] Ethem Alpaydin. *Introduction to Machine Learning.* The MIT Press, 2nd edition, 2010.

[2] Orestes Appel, Francisco Chiclana, and Jenny Carter. Main Concepts, State of the Art and Future research Questions in Sentiment Analysis. *Acta Polytechnica Hungarica - Journal of Applied Sciences*, 12(3):87–108, 2015.

[3] Steven Bird, Edward Loper, and Ewan Klein. *Natural Language Processing with Python.* O'Reilly Media Inc, 2009.

[4] Christopher M. Bishop. *Pattern Recognition and Machine Learning.* Springer Science + Business Media, LLC, 1st edition, 2006.

[5] Felipe Bravo-Márquez, Marcelo Mendoza, and Barbara Poblete. Meta-level sentiment models for big social data analysis. *Knowledge-Based Systems*, 69:86–99, 2014.

[6] Erik Cambria and Amir Hussain. *Sentic computing: A common-sense-based framework for concept-level sentiment analysis.* Springer International Publishing Switzerland 2015.

[7] Erik Cambria. Affective computing and sentiment analysis. *IEEE Intelligent Systems*, 31(2):102–107, 2016.

[8] Erik Cambria, Björn W. Schuller, Bing Liu, Haixun Wang, and Catherine Havasi. Statistical approaches to concept-level sentiment analysis. *IEEE Intelligent Systems*, 28(3):6–9, 2013.

[9] Erik Cambria, Daniel Olsher, and Dheeraj Rajagopal. Senticnet 3: A common and common-sense knowledge base for cognition-driven sentiment analysis. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, 1515–1521, 2014.

[10] Erik Cambria, Dheeraj Rajagopal, Daniel Olsher, and Dipankar Das. Big social data analysis. In *Big Data Computing. CRC Press - A Chapman & Hall Book. Editor: Rajendra Akerkar*, 401–414, 2014.

[11] Erik Cambria, Haixun Wang, and Bebo White. Guest editorial: Big social data analysis. *Knowledge-Based Systems*, 69:1 – 2, 2014.

[12] Erik Cambria, Paolo Gastaldo, Federica Bisio, and Rodolfo Zunino. An ELM-based model for affective analogical reasoning. *Neurocomputing*, 149, Part A:443 – 455, 2015.

[13] F. Chiclana, E. Herrera-Viedma, S. Alonso, and F. Herrera. Cardinal consistency of reciprocal preference relations: A characterization of multiplicative transitivity. *IEEE Transactions on Fuzzy Systems*, 17(1):14–23, 2009.

[14] Heeryon Cho, Songkuk Kim, Jongseo Lee, and Jong-Seok Lee. Data-driven integration of multiple sentiment dictionaries for lexicon-based sentiment classification of product reviews. *Knowledge-Based Systems*, 71:61–71, 2014.

[15] Martine De Cock, Ulrich Bodenhofer, and Etienne E. Kerre. Modelling linguistic expressions using fuzzy relations. In *Proceedings of the 6th International Conference on Soft Computing*, 353–360, 2000.

[16] Sanjiv R. Das, Mike Y. Chen, To Vikas Agarwal, Chris Brooks, Yuk shee Chan, David Gibson, David Leinweber, Asis Martinez-Jerez, Priya Raghubir, Sridhar Rajagopalan, Ajit Ranade, Mark Rubinstein, and Peter Tufano. Yahoo! for Amazon: Sentiment extraction from small talk on the web. In *8th Asia Pacific Finance Association Annual Conference*, 2001.

[17] R. Kent Dybvig. *The Scheme Programming Language*. The MIT Press, 4th edition, 2009.

[18] Fabon Dzogang, Marie-Jeanne Lesot, Maria Rifqi, and Bernadette Bouchon-Meunier. Expressions of Graduality for Sentiments Analysis – A Survey. In *IEEE International Conference on Fuzzy Systems*, 1–7, 2010.

[19] Andrea Esuli and Fabrizio Sebastiani. Determining the semantic orientation of terms through gloss classification. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, 617–624, 2005.

[20] Andrea Esuli and Fabrizio Sebastiani. SentiWordNet – A Publicly Available Lexical Resource for Opinion Mining. *In Proceedings of the 5th Conference on Language Resources and Evaluation*, 417–422, 2006.

[21] Andrea Esuli and Fabrizio Sebastiani. SentiWordNet: a high-coverage lexical resource for opinion mining. Technical Report ISTI-PP-002/2007, Institute of Information Science and Technologies (ISTI) of the Italian National Research Council (CNR), 2006.

[22] Christiane Fellbaum. *WordNet: An Electronic Lexical Database*. A Bradford Book. Series: Language, Speech and Communication. Christiane Fellenbaum (Editor), 1st. edition, 1998.

[23] Mohammad Sadegh Hajmohammadi, Roliana Ibrahim, Ali Selamat, and Hamido Fujita. Combination of active learning and self-training for cross-lingual sentiment classification with density analysis of unlabelled samples. *Information Sciences*, 317:67–77, 2015.

[24] V. Hatzivassiloglou and K.R. McKeown. Predicting the semantic orientation of adjectives. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics (ACL) and the 8th Conference of the European Chapter of the ACL*, 174–181, 1997.

[25] Vasileios Hatzivassiloglou and Kathleen McKeown. Towards the automatic identification of adjectival scales: Clustering adjectives according to meaning. In Lenhart K. Schubert, editor, *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics (ACL)*, 172–182, 1993.

[26] Bas Heerschop, Frank Goossen, Alexander Hogenboom, Flavius Frasincar, Uzay Kaymak, and Franciska de Jong. Polarity analysis of texts using discourse structure. In *Proceedings of the 20th ACM Conference on Information and Knowledge Management*, 1061–1070, 2011.

[27] Alexander Hogenboom, Daniella Bal, Flavius Frasincar, Malissa Bal, Franciska de Jong, and Uzay Kaymak. Exploiting emoticons in sentiment analysis. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, 703–710, 2013.

[28] Alexander Hogenboom, Daniella Bal, Flavius Frasincar, Malissa Bal, Franciska de Jong, and Uzay Kaymak. Exploiting Emoticons in Polarity Classification of Text. *Journal of Web Engineering*, 14(1&2):22–40, 2015.

[29] Alexander Hogenboom, Flavius Frasincar, Franciska de Jong, and Uzay Kaymak. Using rhetorical structure in sentiment analysis. *Communications of the ACM*, 58(7):69–77, 2015.

[30] Minqing Hu and Bing Liu. Mining and summarizing customer reviews. *Proceedings – ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2004.

[31] Sheng Huang, Zhendong Niu, and Chongyang Shi. Automatic construction of domain-specific sentiment lexicon based on constrained label propagation. *Knowledge-Based Systems*, 56:191–200, 2014.

[32] Raquel Justo, Thomas Corcoran, Stephanie M. Lukin, Marilyn A. Walker, and M. Inés Torres. Extracting relevant knowledge for the detection of sarcasm and nastiness in the social web. *Knowledge-Based Systems*, 69:124–133, 2014.

[33] Jaap Kamps, Maarten Marx, Robert J. Mokken, and Maarten de Rijke. Using WordNet to measure semantic orientations of adjectives. In *Proceedings of the 4th International Conference on Language Resources and Evaluation, volume IV*, 1115–1118, 2004.

[34] Bing Liu. *Tutorial*: Sentiment Analysis Tutorial. *Twenty-Fifth Conference on Artificial Intelligence*, 2011.

[35] Bing Liu. *Sentiment Analysis and Opinion Mining.* Morgan and Claypool Publishers: Synthesis Lectures on Human Language Technologies, 1st edition, 2012.

[36] W.C Mann and S.A. Thompson. Rhetorical Structure Theory: Toward a functional theory of text organization. *Text*, 8(3):243–281, 1988.

[37] Stephen Marsland. *Machine Learning - An Algorithmic Perspective.* Chapman & Hall/CRC Machine Learning & Pattern Recognition Series CRC Press, 1st edition, 2009.

[38] John McCarthy. Recursive Functions of Symbolic Expressions and Their Computation by Machine, Part I. *Communications of the ACM*, 3(4):184–195, 1960.

[39] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. J. Miller. Introduction to WordNet: an on-line lexical database. *International Journal of Lexicography*, 3(4): 235–244, 1990.

[40] G.A. Miller. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, 63:81–97, 1956.

[41] Samaneh Nadali, Masrah Murad, and Rabiah Kadir. Sentiment classification of customer reviews based on fuzzy logic. *In International Symposium in Information Technology, Volume 2*:1037–1040, 2010.

[42] Bo Pang and Lillian Lee. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, 271–278, 2004.

[43] Bo Pang and Lillian Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, 115–124, 2005.

[44] Bo Pang and Lillian Lee. *Opinion mining and sentiment analysis.* NOW: the essence of knowledge, Foundations and Trends in Information Retrieval, Vol. 2, Nos. 1–2, pp. 1–135, 2008.

[45] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up? Sentiment Classification using Machine Learning Techniques. *Proceedings of the Association for Computational Linguistics Conference on Empirical Methods in Natural Language Processing*, 10:79–86, 2002.

[46] W. Pedrycz and AV Vasilakos. Linguistic models and linguistic modeling. *IEEE Transactions on Systems, Man and Cybernetics Part B-Cybernetics*, 29(6):745–757, 1999.

[47] Patrizia Pérez-Asurmendi and Francisco Chiclana. Linguistic majorities with difference in support. *Applied Soft Computing*, 18::196 – 208, 2004.

[48] Luis G. Pérez, Francisco Mata, and Francisco Chiclana. Social network decision making with linguistic trustworthiness-based induced OWA operators. *International Journal of Intelligent Systems*, 29(12):1117–1137, 2014.

[49] Luis G. Pérez, Francisco Mata, Francisco Chiclana, Gang Kou, and Enrique Herrera-Viedma. Modeling influence in group decision making. *Soft Computing*, 20(4):1653–1665, 2016.

[50] Jacob Perkins. *Python Text Processing with NLTK 2.0 Cookbook.* Packt Publishing, 2010.

[51] Rosalind W. Picard. *Affective* C*omputing.* The MIT Press, Cambridge, Massachusetts, 1st edition, 2000.

[52] Soujanya Poria, Erik Cambria, Grégoire Winterstein, and Guang-Bin Huang. Sentic patterns: Dependency-based rules for concept-level sentiment analysis. *Knowledge-Based Systems*, 69:45–63, 2014.

[53] Chris Potts. Sentiment Symposium Tutorial: Linguistic structure (part of the Sentiment Analysis Symposium held at San Francisco, november 8-9, 2011). Stanford Department of Linguistics, Stanford University. Accessed date: December 2014, November 2011.

[54] Kumar Ravi Ravi and Vadlamani Ravi. A survey on opinion mining and sentiment analysis: Tasks, approaches and applications. *Knowledge-Based Systems*, 89(C):14–46, 2015.

[55] Mohammad Sadegh and Roliana Ibrahim Zulaiha Ali Othman. Combining Lexicon-based and Learning-based Methods for Twitter Sentiment Analysis. *International Journal of Computers & Technology*, 2(3):171–178, 2012.

[56] Kim Schouten and Flavius Frasincar. Survey on aspect-level sentiment analysis. *IEEE Transactions on Knowledge and Data Engineering*, 28(3):813–830, 2016.

[57] R. Socher, A. Perelygin, J.Y. Wu, J. Chuang, C.D. Manning, A.Y. Ng, and C. Potts. Recursive deep models for semantic compositionality over a sentiment treebank. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 1631, 2013.

[58] Richard Socher, Brody Huval, Christopher D. Manning, and Andrew Y. Ng. Semantic Compositionality through Recursive Matrix-Vector Spaces. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 1201–1211, 2012.

[59] Pero Subasic and Alison Huettner. Affect Analysis of Text Using Fuzzy Semantic Typing. *IEEE Transactions on Fuzzy Systems*, 9(4):483–496, 2001.

[60] Gerald Jay Sussman and Guy Lewis Steele Jr. Scheme: An Interpreter for Extended Lambda Calculus. *MIT AI Lab Memo-349*, 1975.

[61] Sida Wang and Christopher D. Manning. Baselines and Bigrams: Simple, Good Sentiment and Topic Classification. *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, 2:90–94, 2012.

[62] XZ Wang and EE Kerre. Reasonable properties for the ordering of fuzzy quantities (II). *Fuzzy Sets and Systems*, 118(3):387–405, 2001.

[63] S. Wasserman and K. Faust. *Social Network Analysis: Methods and Applications.* Cambridge University Press, 2009.

[64] Dominic Widdows. *Geometry and Meaning.* CSLI Publications, CSLI Lecture Notes No. 172, 1st edition, 2004.

[65] Janyce Wiebe. Learning subjective adjectives from corpora. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, 735–740, 2000.

[66] Jian Wu and Francisco Chiclana. A social network analysis trust-consensus based approach to group decision-making problems with interval-valued fuzzy reciprocal preference relations. *Knowledge-Based Systems* 59: 97–107, 2014.

[67] Jian Wu and Francisco Chiclana. Trust based consensus model for social network in an incomplete linguistic information context. *Applied Soft Computing* 35: 827–839, 2015.

[68] Jian Wu, Ruoyun Xiong, and Francisco Chiclana. Uninorm trust propagation and aggregation methods for group decision making in social network with four tuples information. *Knowledge-Based Systems* 96: 29–39, 2016.

[69] Yusheng Xie, Zhengzhang Chen, Kunpeng Zhang, Yu Cheng, Daniel K. Honbo, Ankit Agrawal, and Alok N. Choudhary. MuSES: a multilingual sentiment elicitation system for Social Media Data. *IEEE Intelligent Systems*, 29(4):34–42, 2014.

[70] Ronald R. Yager and Alexander Rybalov. Uninorm aggregation operators. *Fuzzy Sets and Systems*, 80(1):111 – 120, 1996.

[71] R.R Yager and D.P. Filev. Induced ordered weighted averaging operators. *IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics*, 29(2):141–150, 1999.

[72] Lotfi A. Zadeh. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on Systems, Man and Cybernetics, Part B-Cybernetics*, SMC-3:28–44, 1973.

[73] Lotfi A. Zadeh. A New Direction in AI: Toward a Computational Theory of Perceptions. *AI Magazine*, 22(1):73–84, 2001.

[74] Lotfi A. Zadeh. From Computing with Numbers to Computing with Words - From Manipulation of Measurements to Manipulation of Perceptions. *International Journal of Applied Mathematics and Computer Science (AMCS)*, 12(3):307–324, 2002.