Hussain A & Cambria E (2018) Semi-supervised learning for big social data analysis, *Neurocomputing*, 275, pp. 1662-1673.

DOI: 10.1016/j.neucom.2017.10.010

Semi-Supervised Learning for Big Social Data Analysis

Amir Hussain[a], Erik Cambria[b,c]

[a]*Department of Computing Science and Mathematics, University of Stirling, UK*
[b]*School of Computer Science and Engineering, Nanyang Technological University, Singapore*
[c]*Corresponding author*

Abstract

In an era of social media and connectivity, web users are becoming increasingly enthusiastic about interacting, sharing, and working together through online collaborative media. More recently, this collective intelligence has spread to many different areas, with a growing impact on everyday life, such as in education, health, commerce and tourism, leading to an exponential growth in the size of the social Web. However, the distillation of knowledge from such unstructured Big data is, an extremely challenging task. Consequently, the semantic and multimodal contents of the Web in this present day are, whilst being well suited for human use, still barely accessible to machines. In this work, we explore the potential of a novel semi-supervised learning model based on the combined use of random projection scaling as part of a vector space model, and support vector machines to perform reasoning on a knowledge base. The latter is developed by merging a graph representation of commonsense with a linguistic resource for the lexical representation of affect. Comparative simulation results show a significant improvement in tasks such as emotion recognition and polarity detection, and pave the way for development of future semi-supervised learning approaches to big social data analytics.

1. Introduction

With the advent of social networks, web communities, blogs, Wikipedia, and other forms of online collaborative media, the way people express their opinions and sentiments has radically changed in recent years [1]. These new tools have facilitated the creation of original content, ideas, and opinions, connecting millions of people through the World Wide Web, in a financially and labour-effective manner. This has made a huge source of information and opinions easily available by the mere click of a mouse.

As a result, the distillation of knowledge from this huge amount of unstructured information comes into vital play for marketers looking to create and shape brand and product identities. The practical purpose this encapsulates has led to the emerging field of big social data analysis, which deals with information retrieval and knowledge discovery from natural language and social networks using graph mining and natural language processing (NLP) techniques to distill knowledge and opinions from the huge amount of information on the World Wide Web. Sentic computing [2] tackles these crucial issues by exploiting affective commonsense reasoning, modeled upon the intrinsically human capacity to interpret cognitive and affective information associated with natural language, so as to infer new knowledge and make decisions in connection with one's social and emotional values, censors, and ideals. In other words, we can say that commonsense computing techniques are applied to narrow the semantic gap between word-level natural language data and the concept-level opinions conveyed by these.

In the past, graph mining techniques and multi-dimensionality reduction techniques [3] were employed on a knowledge base obtained by merging ConceptNet [4], a directed graph representation of commonsense knowledge, with WordNet-Affect (WNA) [5], a linguistic resource for the lexical representation of affect. Our research fits within the sentic computing framework and aims to exploit machine learning for developing a cognitive model for emotion recognition in natural language text. Unlike purely syntactical techniques, concept-based approaches can

Email addresses: ahu@cs.stir.ac.uk (Amir Hussain), cambria@ntu.edu.sg (Erik Cambria)

even detect subtly expressed sentiments e.g. by analyzing concepts that do not explicitly convey any emotions, but are linked implicitly to others which do. On this note, the bag-of-concepts model represents semantics associated with natural language, much better than bag-of-words. Interestingly, in the bag-of-words model, a concept such as `cloud_computing` would be split into two separate words, and would hence disrupt the semantics of the input sentence (in which, for example, the word `cloud` could wrongly activate concepts related to `weather`).

Concept level analysis permits the inference of semantic and affective information associated with natural language opinions and, therefore, facilitates comparative fine-grained feature-based sentiment analysis [6]. Instead of collecting isolated opinions on a whole item (e.g., iPhone8), users prefer to compare different products according to specific features (e.g., iPhone8's vs GalaxyS8's touchscreen), or even sub-features (e.g., fragility of iPhone8's vs GalaxyS8's touchscreen). In this context, commonsense knowledge is essential for deconstructing natural language text into sentiments accurately; for instance, the concept `go_read_the_book` can be deemed positive if present in a book review, whereas in a movie review would indicate negative feedback. The inference of emotions and polarity from natural language concepts is, however, a formidable task as it requires advanced reasoning capabilities such as commonsense as well as analogical and affective reasoning.

In the proposed work, a novel semi-supervised learning model based on the merged use of multi-dimensional scaling (MDS) by means of random projections and biased support vector machine (bSVM) [7] is exploited for the task of emotion recognition. Semi-supervised classification should demonstrate an improvement over the classification rule, as both unlabeled and labeled data are utilised to empirically learn a classification function (compared to only labeled data). Interest in semi-supervised learning has grown in recent times, which can be attributed to the existence of application domains (e.g., text mining, natural language processing, image and video retrieval, and bioinformatics). The proposed scheme can benefit from biased regularization, which provides a viable approach to implementing an inductive bias in a kernel machine. This is of fundamental importance in learning theory given that it heavily influences the generalization ability of a learning system. From a mathematical perspective, inductive bias can be formalized as the set of assumptions which determine the choice of a particular class of functions for supporting the learning process. Therefore, it represents a powerful tool that embeds prior knowledge for the applicative problem at hand.

To this aim, semi-supervised learning is formalized as a supervised learning problem biased by an unsupervised reference solution. First, we introduce a novel, general biased-regularization scheme that integrates biased versions of two well-known kernel machines, specifically, support vector machines (SVMs) and regularized least squares (RLS). Subsequently, we propose a semi-supervised learning model, based on this biased-regularization scheme adopting a two-stage procedure. In the first stage, a reference solution is obtained using an unsupervised clustering of the complete dataset (including both unlabeled and labeled data). A primary impact of this is that the eventual semi-supervised classification framework can derive the reference function from any clustering algorithm, thus providing it with remarkable flexibility. In the following stage, clustering outcomes drive the learning process in a biased SVM (bSVM) or a biased RLS (bRLS) to acquire class information provided by the labels. The final outcome is that the overall learned function utilizes labeled and unlabeled data. The developed framework is applicable to linear and non-linear data distributions: the former works based on a cluster assumption applied to the data, whilst the latter operates based on a manifold hypothesis. Consequently, a semi-supervised learning process can only be valid, when unlabeled data can assume an intrinsic geometric structure, for example, a low-dimensional non-linear manifold in the ideal case. With respect to previous strategies, the results demonstrate significant enhancements and pave the way for future development of semi-supervised learning approaches echoing that of affective commonsense reasoning.

The rest of this paper is organized as follows: Section 2 introduces related work in the field of sentiment analysis research; Section 3 describes in detail the new semi-supervised learning architecture for affective commonsense reasoning; Section 4 illustrates results obtained by applying the new model to an affective benchmark and to an opinion mining dataset; finally, Section 5 offers some concluding remarks and recommendations for future work.

2. Related Work

In recent years, sentiment analysis [6] has become increasingly popular for processing social media data on online communities, blogs, Wikis, microblogging platforms, and other online collaborative media. Sentiment analysis is a branch of affective computing research that aims to classify text (but sometimes also audio and video [8]) into either positive or negative (but sometimes also neutral [9]). Sentiment analysis has raised growing interest both

within the scientific community, leading to many exciting open challenges, as well as in the business world, due to the remarkable benefits to be had from financial forecasting [10] and political forecasting [11], e-health [12] and e-tourism [13], community detection [14] and user profiling [15], and more.

While most works approach it as a simple categorization problem, sentiment analysis is actually a suitcase research problem [16] that requires tackling many NLP tasks, including aspect extraction [17], named entity recognition [18], word polarity disambiguation [19], temporal tagging [20], personality recognition [21], and sarcasm detection [22]. Most existing approaches to sentiment analysis rely on the extraction of a vector representing the most salient and important text features, which is later used for classification purposes [23]. Some of the most commonly used features are term frequency and presence. The latter is a binary-valued feature vector in which the entries merely indicate whether a term occurs (value 1) or not (value 0). Feature vectors can sometimes have term-based features with them. Position is one such example; considering that the position of tokens in text units can alter a token's effect on the text's sentiment. Presence of n-grams, usually bi-grams and tri-grams, can also be useful as features, as one can find methods which are dependent on distances between terms. Part-of-speech (POS) information (for example, nouns, verbs, adverbs, and adjectives) is commonly utilized for general textual analysis, in a basic form of word-sense disambiguation. There are some specific adjectives, which have been proven as useful indicators of sentiment, and as guides for feature selection in sentiment classification. Lastly, other studies carried out the detection of sentiments via selected phrases, selected through pre-specified POS patterns, the majority of which had either an adverb or an adjective. Numerous approaches exist which map given pieces of text to labels from predefined sets of categories, or real number representatives of a polarity degree. Nonetheless, these approaches and their performances are confined to an application's domain and relevant areas.

The transformation of sentiment analysis research can be evaluated by examining the token of analysis used, along with implicit associated information. In this way, current approaches can be sorted into four primary categories: keyword spotting, statistical methods, lexical affinity and concept based techniques.

Keyword spotting is very naïve and is the most popular approach, due to its accessibility and economical nature. Text can be grouped into affect categories depending on fairly unambiguous affect words like 'happy', 'bored, 'afraid', and 'sad' being present. For example, Elliott's Affective Reasoner [24], checks for 198 affect keywords (e.g., 'distressed', 'enraged') and affect intensity modifiers (e.g., 'extremely', 'mildly', and 'somewhat'). Other popular sources of affect words are Ortony's Affective Lexicon [25], which groups terms into affective categories, and Wiebe's linguistic annotation scheme [26].

Lexical affinity is slightly more sophisticated than keyword spotting as, rather than simply detecting obvious affect words, it assigns specific words a probabilistic ?affinity' for a particular emotion. For example, `accident` might be assigned a 75% probability of being indicative of a negative affect, as in the case of situations such as `car_accident` or `hurt_by_accident`. These probabilities are usually trained from linguistic corpora [27, 28, 29].

Statistical methods like that of latent semantic analysis (LSA) and SVM, have proven useful for the affective cataloging of texts. Researchers have applied these approaches on projects like Pang's movie review classifier [30], Goertzel's Webmind [31], and more [32, 33, 34, 17]. By feeding a machine learning algorithm a large training corpus of affectively annotated texts, it is possible for the systems to not only learn the affective valence of affect keywords, but to also take into account the valence of other arbitrary keywords (like lexical affinity), punctuation, and word co-occurrence frequencies. Statistical methods, however, tend to be semantically weak, which means that, with the exception of obvious affect keywords, other lexical or co-occurrence elements in a statistical model have little predictive value individually. Hence, statistical classifiers only have an acceptable accuracy when given a large enough text input. Therefore, while these methods may be able to classify text at a paragraph or page level, they are not effective for smaller text units such as sentences.

Concept-based approaches focus on a semantic analysis of text through the use of web ontologies [35] or semantic networks [36], which require grasping the conceptual and affective information associated with natural language opinions. By relying on large semantic knowledge bases, such approaches step away from the blind use of keywords and word co-occurrence count, instead relying on the implicit meaning/features associated with natural language concepts. Concept-based approaches differ from purely syntactical techniques, in that they can detect sentiments which are expressed in a subtle manner, e.g., through the analysis of concepts which are implicitly linked to other concepts that express emotions.

3. Semi-Supervised Reasoning

In the proposed framework, MDS is used to represent concepts in a multi-dimensional vector space and biased SVM (bSVM) is exploited to infer semantics and sentics (that is, the conceptual and affective information) associated with such concepts, according to an hourglass-shaped emotion categorization model [2] (Fig. 1). Under the aforementioned model, sentiments are organized around four independent dimensions (Pleasantness, Attention, Sensitivity, and Aptitude) whose different levels of activation make up the total emotional state of the mind. The bSVM model is adopted as a semi-supervised approach to tackle the classification task so as to overcome the lack of labeled commonsense data. In semi-supervised classification, in fact, both unlabeled and labeled data are exploited to learn a classification function empirically instead of learning a classification rule based only on labeled data. As a result, concepts for which affective information is missing can be employed in the classification phase. The main purpose of the bSVM-based framework developed in this study is to foretell the degree of affective valence each concept posses in a particular facet of the Hourglass model.

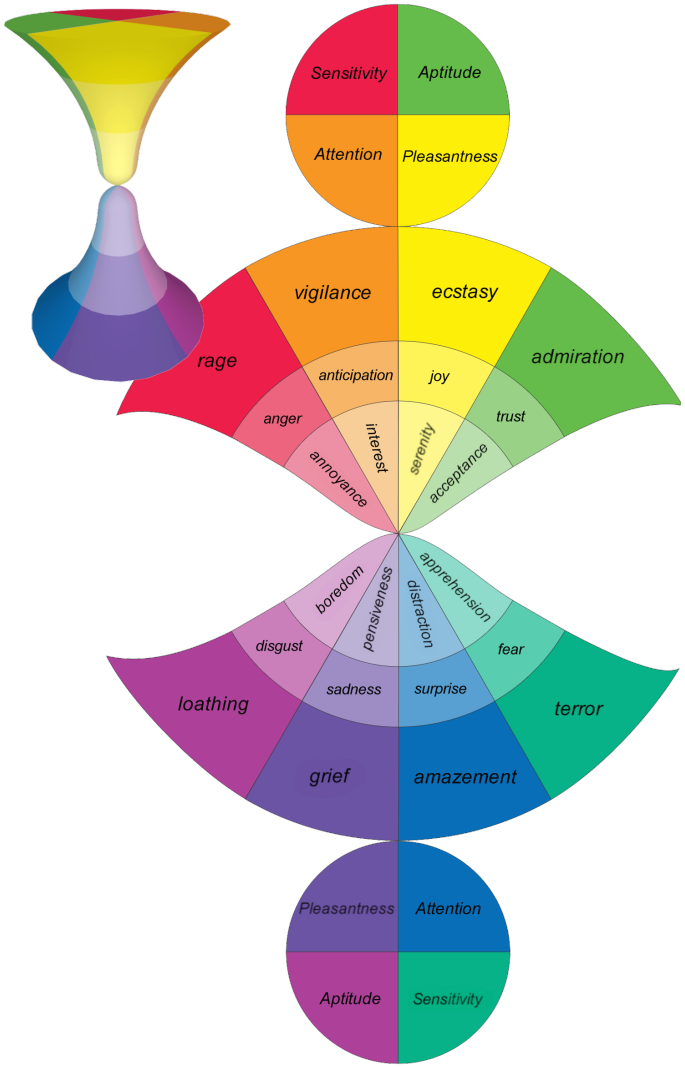

Figure 1: The Hourglass of Emotions

3.1. Affective Commonsense Knowledge Base

The core module of the framework hereby proposed is an affective commonsense knowledge base built upon ConceptNet, the graph representation of the Open Mind Common Sense (OMCS) corpus, which is structurally similar to WordNet [37], but whose scope of contents is general world knowledge, in the same vein as Cyc [38]. Instead of insisting on formalizing commonsense reasoning using mathematical logic [39], ConceptNet uses a new approach: it represents multi-word expressions in the form of a semantic network and makes it available for usage in NLP.

WordNet focuses on lexical categorization and word-similarity determination and Cyc focuses on formalized logical reasoning. Contrastingly, ConceptNet is characterized by contextual commonsense reasoning: this means that it is meant for the task of making practical context-based inferences over real-world texts. In ConceptNet, WordNet's notion of node in the semantic network is extended from purely lexical items (words and simple phrases with atomic meaning) to include higher-order compound concepts such as `satisfy_hunger` and `follow_recipe`, so as to represent knowledge of a greater range of concepts found in everyday life.

Moreover, in ConceptNet, WordNet's repertoire of semantic relations is extended from the triplet of synonym, *IsA* and *PartOf*, to a repertoire of twenty semantic relations including, for example, *EffectOf* (causality), *SubeventOf* (event hierarchy), *CapableOf* (agent's ability), *MotivationOf* (affect), *PropertyOf*, and *LocationOf*. ConceptNet's knowledge is also of a more informal and practical nature. For example, WordNet has formal taxonomic knowledge that a 'dog' is a 'canine', which is a 'carnivore', which is a 'placental mammal'; but it cannot make the logically oriented member-to-set association that a `dog` falls under the categories of `pet` or `family_member`. ConceptNet on the other hand describes something that is often true but not always. For instance, EffectOf(`fall_off_bicycle`, `get_hurt`); this shows it contains a lot of knowledge that is defeasible, which is something that cannot be left aside in commonsense reasoning. Most of the facts that interrelate ConceptNet's semantic network are dedicated to making rather generic connections between concepts.

ConceptNet is obtained via an automatic process, in which a set of extraction rules are applied to the semi-structured English sentences of the OMCS corpus, following which an additional set of 'relaxation' procedures are applied. This results in network gaps being filled in and smoothed over, thereby optimizing connectivity of semantic networks. ConceptNet is a good source of commonsense knowledge, but it alone is not enough for sentiment analysis despite detailing the semantic links between concepts, given that it frequently misses associations between concepts that express the same sentiment. To surmount this flaw, we employed the use of WNA – a semantic resource for the etymological embodiment of affective knowledge, which was built upon WordNet. By allocating a number of WordNet synsets to one or more affective labels (a-labels), WNA was developed. For instance, synsets marked with the a-label 'emotion' demarcated concepts representing emotional states. There are also other a-labels for concepts representing situations that provoke emotions, emotional responses as well as moods. Through a dual-stage process, WNA was born. The first stage pertained to the documentation of a base core of affective synsets, while the second saw the expansion of the core with associations outlined in WordNet. ConceptNet and WNA are then combined through the linear fusion of their respective matrix representations into a single matrix, in which the knowledge between the two databases is shared. In order for this combination process, the input data from both sources has to be transmuted so that it can be denoted in its entirety in the same matrix. Hence, the lemma forms of ConceptNet notions are allied with the lemma forms of words in WNA, and the most common associations in WMA are charted into ConceptNet's set of relations. For instance, Holonym is charted into *PartOf* and Hypernym is to *IsA*. Effectively, we transfigure ConceptNet into a matrix by divvying each assertion into two parts: a concept and a feature, wherein a feature is a contention without any quantified concept; for example 'is a type of fluid'.

Based on the dependability of assertions, the subsequent logs in the matrix are either positive or negative scores, which scale increases proportionally to their dependability. Like ConceptNet, WNA is also denoted as a matrix, in which rows are affective concepts and columns are associated qualities. In combining ConceptNet and WNA into a single matrix, we created a new affective semantic network, in which commonsense concepts are interrelated with a graded system of affective domain tags. We call this new framework AffectNet [2] (Fig. 2); through it, commonsense and emotional knowledge are not merely connected, but instead melded together. For example, concepts from daily life such as `meet_people` or `have_breakfast` are now associated with emotional domain notions such as. 'joy', 'anger', or 'surprise'. Such semantic associations can be of much benefit when tasks such as emotion recognition or polarity detection from natural language text are performed, as it is common for both sentiments and opinions to be conveyed implicitly through context and domain dependent concepts, instead of through specific affect words.

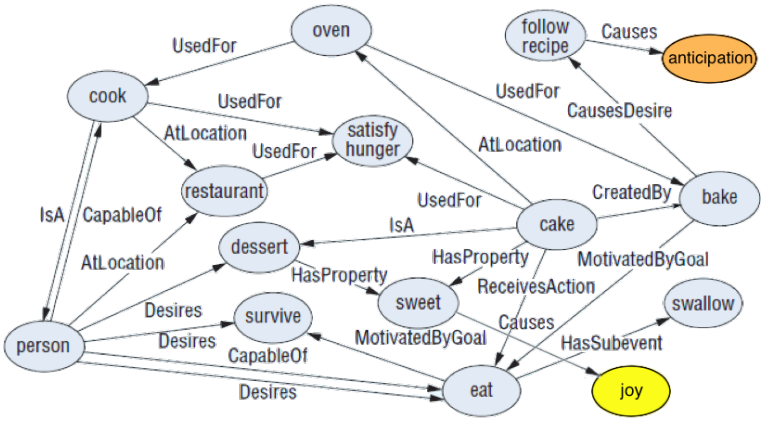

Figure 2: AffectNet

3.2. Affective Analogical Reasoning

Problems are solved best when we understand a solution for it. The tricky part is when we encounter problems we have never faced before, for which require intuition. Intuition can be explained as the process of forming analogies between a current problem and ones we have cracked previously to find a suitable solution. This process of thinking could well be the essence of human intelligence, as in daily life no two situations are identical, and we are continually having to apply analogical reasoning to solve problems and make decisions. The human mind continuously applies compression in all vital relations [40]. The compression principles aim to convert diffuse and distended conceptual structures to more focused versions, to become more congenial for human understanding.

In order to emulate such a process, MDS was previously applied on the matrix representation of AffectNet, a semantic network in which commonsense concepts were linked to semantic and affective features (Table 1). The result was AffectiveSpace. PCA is most widely used as a data-aware dimensionality reduction method [41] and is closely related to the low-rank approximation method, singular value decomposition (SVD), as both work on a transformed version of the data matrix [42]. Therefore, truncated singular value decomposition (TSVD) is applied to the concept-feature matrix for the purposes of expediently diminishing its dimensionality and netting key links.

The purpose of this is to ensure that the new joint model comprises solely of the key features that exemplify the general outlook. Employing TSVD on AffectNet, results in it defining other qualities that could be of parallel relevance to known affective concepts. If there is no known value for an item in a concept in the matrix, but the same item is inherent in other analogous concepts, then by parallel association, it is highly probable that the same item is inherent in the concept as well. Essentially, concepts and features that are aligned and possess high dot yields are ideal contenders for analogies.

Osgood et al. [43] produced a revolutionary piece on comprehending and visualizing affective knowledge connected to natural language text. In their work, MDS was utilized to construct visualizations of affective texts based on their parallel scores when contrasted with texts from other cultures. In a multi-dimensional space, words can be conceived of as points, and parallel scores then denote the distances between words. MDS rethinks these distances as points in a reduced dimensional space (most commonly two or three dimensioned). Similarly, AffectiveSpace's purpose is to visualize the semantic and affective likeness between dissimilar concepts by projecting them onto a multi-dimensional vector space.

However, different from that in Osgood's research, the basic foundation of AffectiveSpace is not merely a restricted set of parallel scores between affect words, but instead consists of millions of confidence scores linked to bits of commonsense knowledge that are in turn related to a structured system of affective domain labels. Instead of being defined by just a small number of human annotators and characterized as a word-word matrix, AffectiveSpace is solidly based on AffectNet and denoted as a concept-feature matrix. SVD seeks to decompose the AffectNet matrix $A \in \mathbb{R}^{n \times d}$ into three components,

$$A = USV^T, \tag{1}$$

where U and V are unitary matrices, and S is an rectangular diagonal matrix with nonnegative real numbers on the diagonal.

SVD has been proved to be optimal in preserving any unitarily invariant norm[1] $\| \cdot \|_M$ [42]:

$$\| A - A_k \|_M = \min_{\text{rank(B)=k}} \| \text{A} - \text{B} \|_\text{M}, \tag{2}$$

where A_k, i.e., AffectiveSpace, is formed by containing only the top k singular values in S. Hence, in AffectiveSpace, commonsense concepts and emotions are represented by vectors of k coordinates. These coordinates can be seen as describing concepts in terms of 'eigenmoods' which form the axes of AffectiveSpace, i.e., the basis e_0,...,e_{k-1} of the vector space. For example, the most significant eigenmood, e_0, represents concepts with positive affective valence. That is, the larger a concept's component in the e_0 direction is, the more affectively positive it is likely to be. Correspondingly, concepts with negative e_0 components are likely to have negative affective valence.

Hence, by exploiting the information sharing property of SVD, concepts with the same affective valence are likely to have similar features. This means that concepts which convey the same emotion are more likely to fall in close proximity to each other in AffectiveSpace. Concept similarity depends on, not their absolute positions in vector space, but the angle they make with the origin.

For example, concepts such as `beautiful day`, `birthday party`, and `make someone happy` are found very close in direction in the vector space, while concepts like `feel guilty`, `be laid off`, and `shed tear` are found in a completely different direction (nearly opposite with respect to the centre of the space).

Table 1: A snippet of the AffectNet matrix

AffectNet	*IsA-pet*	*KindOf-food*	*Arises-joy*	...
dog	0.981	0	0.789	...
cupcake	0	0.922	0.910	...
songbird	0.672	0	0.862	...
gift	0	0	0.899	...
sandwich	0	0.853	0.768	...
rotten fish	0	0.459	0	...
win lottery	0	0	0.991	...
bunny	0.611	0.892	0.594	...
police man	0	0	0	...
cat	0.913	0	0.699	...
rattlesnake	0.432	0.235	0	...
...	...	...	...	...

The difficulty with this sort of representation is a lack of model scalability: as the number of concepts and semantic features increases, the AffectNet matrix becomes so high-dimensional and sparse that it can no longer be computed by the SVD [44]. Although there has been substantial research seeking fast approximations of the SVD, the approximate methods are at best ≈ 5 times faster than the standard one [42], hence it is not viable for real-world big data applications.

There has been conjecture that neuronal learning has simple, yet powerful meta-algorithms underlying it [45], which should be fast, effective, scalable and biologically plausible, as well as having few-to-no assumptions [44]. Optimizing all the $\approx 10^{15}$ connections through the last few million years' evolution is very unlikely [44]. Alternatively, nature probably only optimizes the global connectivity (mainly white matter), but leaves the other details to randomness [44].

To handle the growing number of concepts and semantic features, we replace SVD with random projection (RP) [46], a data-oblivious method, to map the original high-dimensional data-set into a much lower-dimensional subspace by using a Gaussian N(0, 1) matrix, while preserving the pair-wise distances with high probability. This theoretically strong and empirically verified statement follows Johnson and Lindenstrauss's (JL) Lemma [44]. The JL

[1] A norm $\|\cdot\|_M$ is unitarily invariant if $\|UAV\|_M = \|A\|_M$ for all A and all unitary U, V.

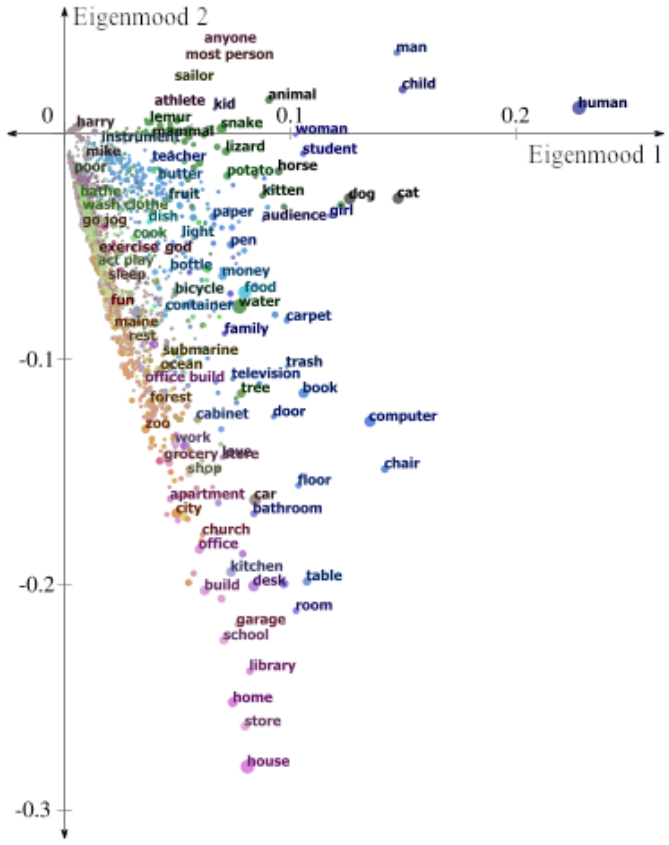

Figure 3: AffectiveSpace's first two eigenmoods

Lemma states that with high probability, for all pairs of points x, y ∈ X simultaneously,

$$\sqrt{\frac{m}{d}} \parallel x - y \parallel_2 (1-\varepsilon) \leq \parallel \Phi x - \Phi y \parallel_2 \leq \tag{3}$$

$$\leq \sqrt{\frac{m}{d}} \parallel x - y \parallel_2 (1+\varepsilon), \tag{4}$$

where X is a set of vectors in Euclidean space, d is the original dimension of this Euclidean space, m is the dimension of the space we wish to reduce the data points to, ε is a tolerance parameter measuring to what extent is the maximum allowed distortion rate of the metric space, and Φ is a random matrix.

Structured random projection for making matrix multiplication much faster was introduced in [47]. Achlioptas [48] proposed *sparse random projection* to replace the Gaussian matrix with i.i.d. entries in

$$\phi_{ji} = \sqrt{s} \begin{cases} 1 & \text{with prob. } \frac{1}{2s} \\ 0 & \text{with prob.} 1 - \frac{1}{s} \\ -1 & \text{with prob.} \frac{1}{2s} \end{cases}, \tag{5}$$

where one can achieve a ×3 speedup by setting $s = 3$, since only $\frac{1}{3}$ of the data need to be processed. However, since our input matrix is already too sparse, we avoid using sparse random projection.

When the number of features is a lot larger than the number of training samples ($d \gg n$), subsampled randomized Hadamard transform (SRHT) is preferred, as it behaves similarly to Gaussian random matrices but also accelerates

the process from $O(n\,d)$ to $O(n\,log\,d)$ time [49]. Following [50] [49], for $d = 2^p$ where p is any positive integer, a SRHT can be defined as:

$$\Phi = \sqrt{\frac{d}{m}}\text{RHD} \tag{6}$$

where

• m is the number we want to subsample from d features randomly.

• R is a random $m \times d$ matrix. The rows of R are m uniform samples (without replacement) from the standard basis of $\mathbb{R}^d$.

• $H \in \mathbb{R}^{d\times d}$ is a normalized Walsh-Hadamard matrix, which is defined recursively: $H_d = \begin{bmatrix} H_{d/2} & H_{d/2} \\ H_{d/2} & H_{d/2} \end{bmatrix}$ with $H_2 = \begin{bmatrix} +1 & +1 \\ +1 & -1 \end{bmatrix}$.

• D is a $d \times d$ diagonal matrix and the diagonal elements are i.i.d. Rademacher random variables.

Our subsequent analysis relies solely on distances and angles between pairs of vectors (i.e., the Euclidean geometry information), and is sufficient enough to set the projected space to be logarithmic in the size of the data [51] and apply SRHT. The result is a new vector space model, AffectiveSpace 2 (Fig. 4), which preserves the semantic and affective relatedness of commonsense concepts while being highly scalable [52]. The key to performing commonsense reasoning is to find a good trade-off for knowledge representation. Since no two situations in life are ever exactly the same, no representation should be too concrete or else it will fail apply to new situations. However, at the same time, no representation should be too abstract, or risk suppressing too many details.

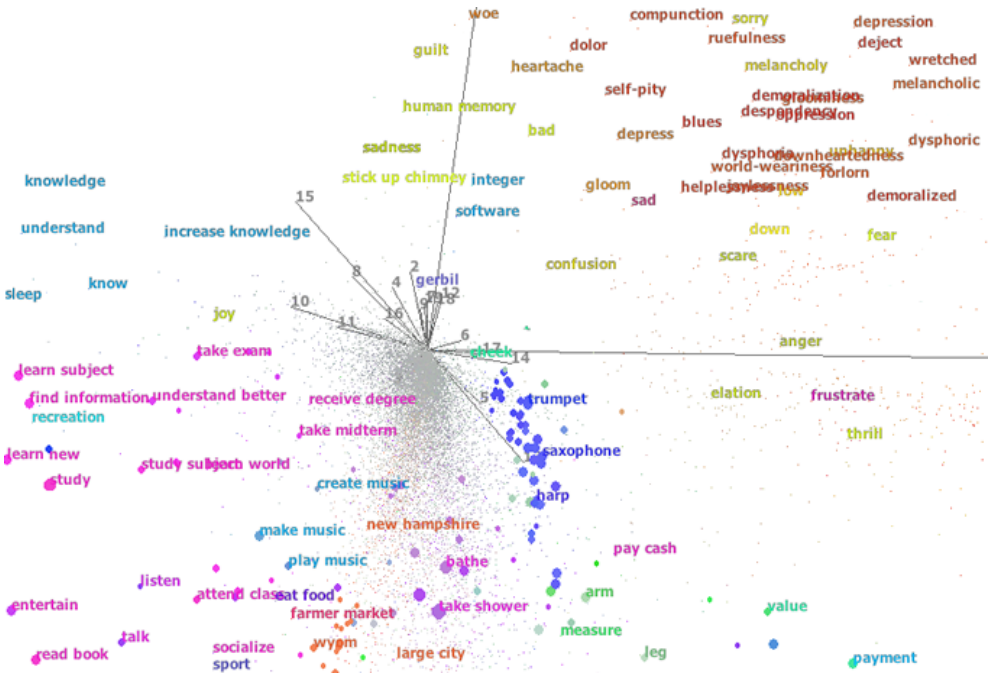

Figure 4: AffectiveSpace 2

Within AffectiveSpace 2, this knowledge representation trade-off can be seen in the choice of the vector space dimensionality. The number m indicates the dimension of AffectiveSpace and, in fact, is a measure of the trade-off between accuracy and efficiency in the representation of the affective commonsense knowledge base. The bigger m is, the more precisely AffectiveSpace represents AffectNet's knowledge; the smaller m is, on the other hand, the more efficiently AffectiveSpace represents affective commonsense knowledge both in terms of vector space generation and of dot product computation.

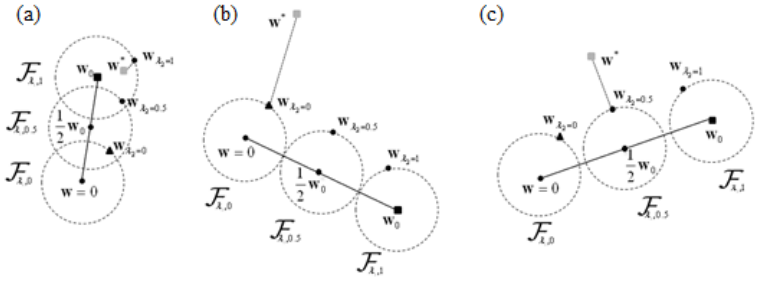

Figure 5: The role played by parameter λ_2 in the proposed problem setting. Three different situations are analyzed: (a): the reference $\mathbf{w}_0$ is closer to the true solution $\mathbf{w}^*$ than $\mathbf{w}_{\lambda_2=0}$; (b): the reference $\mathbf{w}_0$ is more distant from the true solution $\mathbf{w}^*$ than $\mathbf{w}_{\lambda_2=0}$; (c): the reference $\mathbf{w}_0$ is more distant from the true solution $\mathbf{w}^*$ than $\mathbf{w}_{\lambda_2=0}$, but biased regularization can be useful.

3.3. Semi-Supervised Learning Approach

Even though AffectiveSpace 2 is a powerful tool for discovering semantic and affective relatedness of natural language concepts, reasoning by analogy in such a multi-dimensional vector space is a difficult task as the distribution of concepts in the space is non-linear and only the affective valence of a relatively small set of concepts is known a priori. Hence, the bSVM and bRLS models are adopted as a semi-supervised approach in order to exploit both unlabeled and labeled commonsense data to learn a classification function empirically. They are both based on the biased regularization theory, realized as follows: a reference solution (e.g, a hyperplane) is used to bias the solution of a regularization-based learning machine.

3.3.1. Regularization-based Learning

Modern classification methods often rely on regularization theory. In a regularized functional, a positive parameter, λ , rules the tradeoff between the empirical risk, $R_{emp}[f]$, (loss function) of the decision functions f (i.e., regression or classification) and a regularizing term. The cost to be minimized can be expressed as:

$$R_{reg} = R_{emp}[f] + \lambda\Omega[f] \tag{7}$$

where the regularization operator, $\Omega[f]$, quantifies the complexity of the class of functions from which f is drawn. Usually f belongs to a Reproducing Kernel Hilbert Space (RKHS) $\mathbf{H}$. For a data set, $\mathbf{X}$, one computes a square matrix, $\mathbf{K}$, of elements, which is symmetric and positive definite. Every entry $K(\mathbf{s}, \mathbf{x})$ can be viewed as the inner product $< \phi(\mathbf{s}), \phi(\mathbf{x}) >$ where $\phi(.)$ is the (implicit, non linear) mapping function uniquely defined by $\mathbf{H}$. A kernel method implies the choice of an inner-product formulation; the simplest, linear kernel supports the inner vector product in the actual domain space: $< \phi(\mathbf{s}), \phi(\mathbf{x}) > \equiv < \mathbf{s}, \mathbf{x} >$.

When dealing with maximum-margin algorithms, $\Omega[f]$ is implemented by the term $\left\|f\right\|_H^2$, which supports a square norm in the feature space. The Representer Theorem proves that, when $\Omega[f] = \left\|f\right\|_H^2$, the solution of the regularized cost can be expressed as a finite summation over a set of labeled training patterns $\mathbf{X} = \{\mathbf{x}_i, y_i\}$, $i = 1, ..., l$, $y_i \in \{-1, +1\}$:

$$f(\mathbf{x}_j) = \mathbf{w} \cdot \mathbf{x}_j = \sum_{i=1}^{l} \beta_i K(\mathbf{x}_i, \mathbf{x}_j) \tag{8}$$

SVM and RLS are popular methods belonging to this family of regularizing algorithms; both provide excellent performance in pattern recognition problems. The two learning algorithms differ in their choice of loss function: the SVM model uses the 'hinge' loss function, whereas RLS operates on a square loss function.

The SVM training process requires one to solve the following optimization problem:

$$\min_{\{\tilde{\mathbf{w}}, b, \tilde{\epsilon}\}} C \sum_{i=1}^{l} \epsilon_i + \frac{1}{2} \left\|\mathbf{w}\right\|^2 \epsilon_i > 0 \tag{9}$$

subject to:

$$y_i(\mathbf{w}^T\mathbf{x}_i) \geq 1 - \epsilon_i \tag{10}$$

where ϵ_i is a penalty term to be added for each misclassified pattern, and C is a hyperparameter that plays the role of $1/\lambda$. The constant parameter b has been dropped because one can equivalently augment the space X with a feature of constant value +1. The problem can be efficiently solved in its dual form by using quadratic programming techniques. When using the dual formulation, one optimizes a set of Lagrange multipliers, α_i, and it can be shown that the series coefficients in (8) can be written as $\beta_i = \alpha_i y_i$.

When dealing with the RLS model, the problem to be optimized is:

$$\min_f \sum_{i=1}^{l} (y_i - f_i)^2 + \frac{\lambda}{2} \left\| f \right\|_H^2 \tag{11}$$

whose optimum in β is found by solving the following linear system:

$$(\mathbf{K} + \lambda \mathbf{I})\beta = \mathbf{y} \tag{12}$$

3.3.2. Maximal Discrepancy Bounds for Model Selection

One of the main obstacles in classification problems is tuning classifier regularization parameter(s). When tackling limited-sample problems, strategies such as k-fold cross validation may be difficult to apply due to the small size of both the training and the test sets. Thus, theoretical approaches which derive the analytical expressions of the generalization bounds, can give powerful options for attaining reliable model selection. These methods do not require data partitioning and are always based on the complexity on the hypothesis space, F. The bound value to the true generalization error, $R[f]$, is asserted with confidence at least $1 - \delta$, and is commonly written as the sum of several terms:

$$R[f] \leq R_{emp}[f] + \chi + \Psi \tag{13}$$

where $R_{emp}[f]$ is the error on the training set, χ measures the complexity of the space of classifying functions, and Ψ penalizes the finiteness of the training sample.

The Maximal-Discrepancy bound (MD) belongs to the class of theoretical approaches. In this research, the MD bound is exploited to confirm that the proposed semi-supervised learning scheme can ensure the shrinking of the generalization bound, thus providing effective tools for model selection in a semi-supervised setting.

3.3.3. A Biased Regularization

A general biased regularization model is realized as follows: a reference solution (e.g, a hyperplane) is used to bias the solution of a regularization-based learning machine.

In a linear domain one can define a generic convex loss function, $l(\mathbf{X}, \mathbf{Y}, \mathbf{w})$, and a biased regularizing term; the resulting cost function is:

$$l(\mathbf{X}, \mathbf{Y}, \mathbf{w}) + \lambda_1 \left\| \mathbf{w} - \lambda_2 \mathbf{w}_0 \right\|^2 \tag{14}$$

where $\mathbf{w}_0$ is a reference hyperplane, λ_1 is the classical regularization parameter that controls smoothness (e.g., $1/C$ in SVM), and λ_2 controls the adherence to the reference solution $\mathbf{w}_0$. Expression (14) is a convex functional and thus admits a global solution. From (14) one gets:

$$\begin{aligned} l(\mathbf{X}, \mathbf{Y}, \mathbf{w}) + \frac{\lambda_1}{2} \left\| \mathbf{w} - \lambda_2 \mathbf{w}_0 \right\|^2 = \\ l(\mathbf{X}, \mathbf{Y}, \mathbf{w}) + \frac{\lambda_1}{2} \left\| \mathbf{w} \right\|^2 - \\ \lambda_1 \lambda_2 \mathbf{w} \mathbf{w}_0 \end{aligned} \tag{15}$$

which actually involves two regularization parameters, λ_1 and λ_2; this problem setting differs from the one proposed for SVM, where only one regularization parameter was defined, obtaining $l(\mathbf{X}, \mathbf{Y}, \mathbf{w}) + \lambda_1 \left\|\mathbf{w} - \mathbf{w}_0\right\|^2$. The latter expression coincides in the special case $\lambda_2 = 1$.

Figure 5 explicates the role played by parameter λ_2 in three different cases. In all those figures, $\mathbf{w}$ is set as the origin, 0, of the space of hypothesis, whereas a black square denotes the reference hyperplane, $\mathbf{w}_0$, and a grey square indicates the 'true' optimal solution. For the sake of clarity, and without loss of generality, the examples assume that:

1. λ_1 is set to a fixed value (i.e., $\lambda_1 = 1$).
2. The distance $\left\|\mathbf{w} - \lambda_2 \mathbf{w}_0\right\|$ is constant for any λ_2.
3. $\mathbf{w}_{\lambda_2=0}$ (black triangle) is the best solution one can obtain from the unbiased learning (i.e., λ_2=0). Here, the best solution refers to the solution that is closest to $\mathbf{w}^*$ among all the possible $\mathbf{w}$ that lie at a distance $\left\|\mathbf{w} - \lambda_2 \mathbf{w}_0\right\|$ from $\mathbf{w}_0$ (the dashed circumference).

Fig. 5(a) refers to the situation in which the reference $\mathbf{w}_0$ is closer to the true solution $\mathbf{w}^*$ than $\mathbf{w}_{\lambda_2=0}$. The Figure shows that when λ_2 decreases from 1 to 0, the centre of the ideal circumference, which encloses the eventual solution $\mathbf{w}_{\lambda_2}$, drifts. When $\lambda_2 \to 0$, $\mathbf{w}_{\lambda_2}$ moves toward the origin $\mathbf{w} = 0$, which represents the condition of no reference exploited. Indeed, the draw highlights that, when $\mathbf{w}_0$ gives a reliable reference, one can take full advantage of biased regularization, as the best solution for $\lambda_2 = 1$, $\mathbf{w}_{\lambda_2=1}$, definitely improves over $\mathbf{w}_{\lambda_2=0}$.

Fig. 5(b) illustrates the opposite case: the reference $\mathbf{w}_0$ is more distant from the true solution $\mathbf{w}^*$ than $\mathbf{w}_{\lambda_2=0}$ (it is worth to note that the relative position of $\mathbf{w}^*$ and $\mathbf{w}_{\lambda_2=0}$ with respect to the origin $\mathbf{w} = 0$ remained unchanged when compared with Fig. 5(a)). In this situation, one would obtain the best outcome by setting $\lambda_2 = 0$, thus neutralizing the contribution of the biased regularization. Hence, $\mathbf{w}_0$ does not represent a helpful reference.

Finally, Fig. 5(c) illustrates another situation in which the reference $\mathbf{w}_0$ is more distant from the true solution, $\mathbf{w}^*$, than $\mathbf{w}_{\lambda_2=0}$, but biased regularization still remains useful, as by adjusting λ_2 (i.e., by modulating the contribution of the reference $\mathbf{w}_0$) one eventually obtains a solution $\mathbf{w}_{\lambda_2}$ that improves over $\mathbf{w}_{\lambda_2=0}$. As a result, one can take advantage of biased regularization even when the reference solution is not optimal.

The extension of (14) to non-linear models is obtained by considering a Reproducing Kernel Hilbert Space H. In that case one has a reference function f_0 and the functional (15) becomes:

$$l(\mathbf{X}, \mathbf{Y}, f) + \frac{\lambda_1}{2} \left\|f - \lambda_2 f_0\right\|_H^2 \tag{16}$$

where now the norm of the regularizer is taken in H. Eventually, one obtains the models for the biased SVM (bSVM) and the biased RLS (bRLS), respectively, by adopting the proper loss function $l(\mathbf{X}, \mathbf{Y}, \mathbf{w})$:

bSVM:

$$\sum_{i=1}^{l} (1 - y_i f(\mathbf{x}_i))_+ + \frac{\lambda_1}{2} \left\|f - \lambda_2 f_0\right\|_H^2 \tag{17}$$

bRLS:

$$\sum_{i=1}^{l} (y_i - f(\mathbf{x}_i))^2 + \frac{\lambda_1}{2} \left\|f - \lambda_2 f_0\right\|_H^2 \tag{18}$$

3.3.4. Biased SVM

The following theorem shows the formalization of the biased version of the support vector machine (bSVM) within this scheme with the inclusion of a regularizing bias.

Theorem1(bSVM): Given a reference hyperplane $\mathbf{w}_0$ (or a reference function f_0, if the domain is not linear), a regularization constant C, and a biasing constant λ_2, the dual form of the learning problem:

$$\begin{cases} \min_{\{\epsilon, \mathbf{w}\}} C \sum_{i=1}^{l} \epsilon_i + \frac{1}{2} \left\|\mathbf{w} - \lambda_2 \mathbf{w}_0\right\|^2 & \\ y_i(\mathbf{w} \cdot \mathbf{x}_i) \geq 1 - \epsilon_i & \forall i \\ \epsilon_i \geq 0 & \forall i \end{cases} \tag{19}$$

is written as

$$\begin{cases} \min_{\{\alpha\}} \frac{1}{2} \alpha^t \mathbf{Q} \alpha - \sum_{i=1}^{l} \alpha_i (1 - \lambda_2 y_i f_0(\mathbf{x}_i)) & \\ 0 \leq \alpha_i \leq C & \forall i \end{cases} \tag{20}$$

The model of the data is:

$$f(x) = \sum_{i=1}^{l} \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i) + \lambda_2 f_0(\mathbf{x}) \tag{21}$$

where α are the support vectors, ϵ the slack variables used to measure the grade of allowed misclassification and K the chosen kernel.

Unlike the conventional SVM formulation, the minimization problem (20) does not contain a linear constraint. This problem (20) can be optimized by an SMO version which uses only a single Lagrange multiplier at each iteration. In such a new procedure, the gradient integrates the new reference based-term and the regularization parameter. The gradient value for the i-th pattern is:

$$G_i = y_i \sum_{j=1}^{l} \alpha_j y_j K(\mathbf{x}_j, \mathbf{x}_i) - 1 + \lambda_2 y_i f_0(\mathbf{x}_i) \tag{22}$$

Then, as usual, the projected gradient PG is computed and the KKT optimality conditions are checked on this value. The algorithm runs till the KKT conditions are satisfied. In the following the pseudo-code of the algorithm is presented.

bSVMSolver:

1. Initialization: $\alpha = 0$, flag = 0
2. While (!flag):
    a. flag = 1
    b. for $i = 1, , l$
        i.

$$G = y_i \sum_{j=1}^{l} \alpha_j i y_j K(\mathbf{x}_j, \mathbf{x}_i) - 1 + \lambda_2 y_i f_0(\mathbf{x}_i) \tag{23}$$

        ii.

$$\begin{cases} min(G, 0) & \alpha_i = 0 \\ min(G, 0) & \alpha_i = C \\ G & 0 < \alpha_i < C \end{cases} \tag{24}$$

        iii. $if |PG| > \epsilon$
            1. $\alpha_i = min(max(\alpha_i - G/k_{ii}, 0), C)$
            2. $flag = 0$
        end if
    end for

end while

The pseudo-code listed above can be considerably accelerated by updating the gradient only when necessary, as well as by using shrinking and random permutations of indexes of patterns at each iteration.

3.3.5. Biased RLS

The following theorem formalizes the linear biased version of RLS.

Theorem2: Given a reference hyperplane $\mathbf{w}_0$, a regularization constant λ_1, and a biasing constant λ_2, the problem:

$$min_{\mathbf{w}} \left\| \mathbf{X}\mathbf{w} - \mathbf{y} \right\|^2 + \lambda_1 \left\| \mathbf{w} - \lambda_2 \mathbf{w}_0 \right\|^2 \tag{25}$$

has solution:

$$\mathbf{w} = (\mathbf{X}^t\mathbf{X} + \lambda_1 \mathbf{I})^{-1}(\mathbf{X}^t\mathbf{y} + \lambda_1\lambda_2\mathbf{w}_0) \tag{26}$$

The following theorem gives the dual form of biased RLS (bRLS):

Theorem3: Given a reference hyperplane $\mathbf{w}_0$ (or a reference function f_0), a regularization constant λ_1, and a biasing constant λ_2, the dual of the problem:

$$min_{\mathbf{w}} \left\| \mathbf{X}\mathbf{w} - \mathbf{y} \right\|^2 + \lambda_1 \left\| \mathbf{w} - \lambda_2 \mathbf{w}_0 \right\|^2 \tag{27}$$

is:

$$min_{\beta} \left\| \mathbf{K}\beta + \lambda_2 f_0(\mathbf{x}) - \mathbf{y} \right\|^2 + \lambda_1 \beta^t \mathbf{K}\beta \tag{28}$$

which has solution:

$$\beta = (\mathbf{K} + \lambda_1 \mathbf{I})^{-1} (\mathbf{y} - \lambda_2 f_0(\mathbf{X})) \tag{29}$$

The model of the data is:

$$f(x) = \sum_{i=1}^{l} \beta_i K(\mathbf{x}, \mathbf{x}_i) + \lambda_2 f_0(\mathbf{x}) \tag{30}$$

Corollary1: Given the RLS learning machine, the RKHS H and the representation,

$$f(x) = \sum_{i=1}^{l} \beta_i K(\mathbf{x}, \mathbf{x}_i) + \lambda_2 f_0(\mathbf{x}) \tag{31}$$

the solution of problem

$$min_f \left\| f - \mathbf{y} \right\|^2 + \lambda_1 \left\| f - \lambda_2 f_0 \right\|_H^2 \tag{32}$$

is:

$$(\mathbf{K} + \lambda_1 \mathbf{I})\beta = \mathbf{y} - \lambda_2 f_0(\mathbf{X}) \tag{33}$$

From a computational point of view, the choice between the primal or the dual solution depends on the characteristics of the available data. If samples lie in a low-dimensional space and can be separated by a linear classifier, the primal form is preferable because it scales linearly with the number of features. Conversely, when the number of patterns is lower than the number of features, the dual form should be used.

3.3.6. A Semi-Supervised Learning Scheme Based on Biased Regularization

Once the biased version of the SVM kernel machine has been defined, the following four-step procedure can be followed in order to formalize a semi-supervised framework for the classification task.

Let $\mathbf{X}$ be a dataset composed by l labeled patterns and u unlabeled patterns; let $\mathbf{X}_l$ denote the labeled subset, $\mathbf{y}_l$ denote the corresponding vector of labels, and $\mathbf{X}_u$ denote the unlabeled subset. Then the semi-supervised learning scheme can be formalized as follows:

1. Clustering: Use any clustering algorithm to perform an unsupervised partition of the dataset $\mathbf{X}$ (a bipartition in the simplest case).

2. Calibration: For every cluster, a majority voting scheme is adopted to set the cluster label; this is done by exploiting the labeled samples. Then, for each cluster, assign to each sample the cluster label. Let $\hat{y}$ denote this new set of labels.

3. Mapping: Given $\mathbf{X}$ and $\hat{y}$, train the selected learning machine and obtain the solution $\mathbf{w}_0$.

4. Biasing: Given $\mathbf{X}_l$ and the true labels $\mathbf{y}_l$, train the biased version of the learning machine (biased by $\mathbf{w}_0$). The solution $\mathbf{w}$ carries information derived from both the labeled data $\mathbf{X}_l$ and the unlabeled data $\mathbf{X}_u$.

The ultimate result is that the overall learned function exploits both labeled and unlabeled commonsense data. The semi-supervised learning scheme possesses some interesting features:

- Since the proposed method could be applied both to linear and non linear domains, the result is a completely generalizable learning scheme.

- The present learning scheme distinguishes between two principal actions: clustering and biasing. This means that the two tasks can be tackled independently. If one wants to adopt a particular solution for biasing or a new clustering algorithm is designed, then the two actions can be controlled and adjusted separately.
- If the learning machine is a single layer learning machine whose cost is convex then convexity is preserved and a global solution is granted.
- Every clustering method can be used to build the reference solution.

3.3.7. bSVM and bRLS for Semi-Supervised Learning: Computational Complexity

This semi-supervised learning scheme consists of 3 computationally intensive steps: clustering, mapping and biasing. Clustering tasks can be completed using various multiple clustering algorithms, which are then characterized by computational complexities. As follows, the complexity of clustering can be denoted generically as O_c. There also exists some solutions which allow the implementation of powerful clustering algorithms such as k-means or Special Clustering.

In the second step, mapping, the time complexity is entirely determined by the learning machine applied to all the $l + u$ available samples. For RLS this would mean a complexity of $O((l + u)^3)$, i.e., the solution of the system of linear equations (29). When adopting SVM as learning machine, one can exploit the SMO algorithm, which scales in between $O(l + u)$ and $O((l + u)^2)$.

The third step, biasing, consists of solving through either a linear system or using an SMO-like algorithm. In both cases, one also need to pre-compute the predictions of the reference model $f_0(\mathbf{x})$ for all the labeled patterns (with d-dimensional patterns the eventual cost is $O(ud)$). As a result, when bRLS is adopted as learning machine, the computational complexity is:

$$O_{bRLS} = O_c + O((l + u)^3) + O(ud) + O(l^3) \tag{34}$$

In O_{bRLS} the dominant terms are $O((l + u)^3)$ and possibly the complexity O_c associated to the clustering task. When instead bSVM is used, the computational complexity is:

$$O_{bSVM} = O_c + O((l + u)^2) + O(ud) + O(l^2) \tag{35}$$

where the dominant terms are $O((l + u)^2)$ and, again, the complexity O_c. In this case, one assumes that $O(ud) < O((l + u)^2)$; this is a reasonable hypothesis, except for those cases where the data lie in a highly dimensional space. Therefore, the complexity of the training procedure roughly scales with the same complexity of the original learning machine. SVM scales approximately as $O(l^2)$, and its semi-supervised version (bSVM) scales as $O((l + u)^2)$; a similar behavior characterizes RLS and bRLS.

Below are final considerations which can contribute to the discussion surrounding computational complexity:

- If it is known a priori that the data are almost linearly separable, then it is possible to build very efficient learning algorithms. For instance, one can couple fast linear k-means implementations with the linear version of SVM and bSVM. That set up leads to very efficient learning methods, in particular when data is highly sparse, such as in text mining problems.
- The proposed semi-supervised learning scheme can address large scale problems, as long as the clustering engines are scaling well. For certain domains, for example text mining, in which linearity and data sparsity are exploitable, adaptations of the learning algorithm can result in incredibly fast learning algorithms. These algorithms can hence process hundreds of thousands of patterns in a matter of seconds.
- New, unseen test patterns can be managed effectively as class assignment can exploit the closed form functions.

The proposed framework can provide attractive features when compared with other semi-supervised methods. First, the reference function can be worked out by exploiting any clustering algorithm. Second, biased regularization can support effectively model selection. This feature is crucial indeed, in particular when a model with few labeled data is addressed. Other approaches to semi-supervised learning do not provide this attribute. Furthermore, the present framework exploits a convex cost function.

The proposed framework also ensures satisfactory performance in terms of computational complexity. This is highlighted in Table 2, which reports – for each approach – the computational complexity. Computational complexity is formalized by using the number of labeled patterns, l, the number of unlabeled patterns, u, the dimensionality of the data, d, the number of iterations of the learning algorithm, k, and the complexity of the clustering algorithm, O_c. For the proposed biased learning machines, complexity has been formalized by assuming the use of efficient SMO-like routines.

Method	Complexity
bSVM	$O_c + O(l+u)^2$
bSVM linear	$O_c + O(d(l+u)k)$
bRLS	$O_c + O(l+u)^3$
LapRLS	$O(l+u)^3$
LapSVM	$O(l+u)^3$
LapSVM linear	$O(d^3)$
Cluster kernel	$O(max(l,u)^3)$
TSVM	$O(k(l+2u)^2)$
EM	-
Co-training	-
Semi Parametric Regularization	$O(l+u)^3$

Table 2: Comparison of Semi-supervised methods

It is evident from the table that the current semi-supervised learning scheme can achieve satisfactory performances in terms of computational complexity, whenever the clustering algorithm scales as (or better than) the adopted biased machine. In this regard, bSVM appears especially appealing as it scales quadratically, or even linearly if the underlying problem has particular characteristics. Indeed, one should take into account that the term O_c represents the added cost to be paid for a gain in flexibility.

3.3.8. Biased Regularization for Semi-Supervised Learning Supports Effective Model Selection

The proposed semi-supervised learning framework can extend the supervised classification scheme presented in [53], which demonstrated that when a cluster hypothesis holds, the use of clustering to set a reference solution leads to a significant reduction of the space of possible functions. Such a result is noteworthy in that it leads to tight generalization bounds, since the term χ in equation (13) measures the complexity of the space of classifying functions. Tight generalization bounds are in turn a necessary condition for supporting an effective model selection.

In [53], model selection of SVM is actually performed by exploiting an auxiliary machine, called VQSVM. In the VQSVM model the learning task is fully supervised because the clustering reference only derives from labeled data. Besides, an Ivanov biased regularization term is used and the regularization parameter λ_2 is implicitly set to a fixed value.

To extend the scheme [53] to semi-supervised learning, the present framework involves both labeled and unlabeled data in the clustering step. Indeed, the effectiveness of model selection is improved by adopting a formulation of biased regularization that fully exploits parameters λ_1 and λ_2.

Figure 6 considers four cases, and analyzes the relative positions of the origin $\mathbf{w} = 0$, the reference, $\mathbf{w}_0$, and the true solution, $\mathbf{w}^*$. Figure 2(a) exemplifies the case 'good reference / weak bias', whereas Fig. 2(b) illustrates the case 'good reference / strong bias'. In both situations, the reference solution $\mathbf{w}_0$ is not far from the true solution $\mathbf{w}^*$. Yet, the first case adopts a weak bias, which permits the biasing step to explore a relatively wide portion of space around the reference (the lighter circumference, which represents the space of functions). On the other hand, the second case adopts a strong bias, thereby exploring a smaller portion of space. Eventually, the area being explored via a strong bias does not include $\mathbf{w}^*$.

Figure 6(c) refers to the case 'bad reference / weak bias'. The reference $\mathbf{w}_0$ is quite distant from the true solution $\mathbf{w}^*$; by adopting a weak bias, though, one can still exploit biasing to reach $\mathbf{w}^*$. Finally, Figure 6(d) presents the case 'bad reference / strong bias'. In this situation, by adopting a strong bias one restricts the space to be explored to a small region around $\mathbf{w}_0$. As a result, the proposed solution will be very distant from the true solution $\mathbf{w}^*$.

On the whole, the four examples affirm that that by modulating the biasing mechanism through the parameters λ_1 and λ_2 one can take full advantage of the semi-supervised scheme and support the model selection procedure properly.

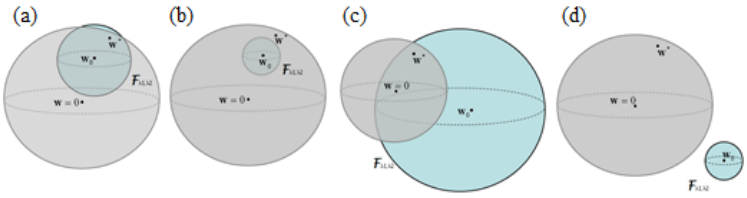

Figure 6: The effectiveness of model selection is improved by adopting a formulation of biased regularization which fully exploits parameters λ_1 and λ_2. Four cases are presented: (a) good reference / weak bias; (b) good reference / strong bias; (c) bad reference / weak bias; (d) bad reference / strong bias.

Eventually, the proposed framework involves a novel semi-supervised classification scheme, which supports a fully automated model selection and can be applied also when the size of the labeled dataset is small ($l < 50$).

4. Experimental Results

The proposed affective commonsense reasoning architecture based on bSVM and bRLS was tested on the publicly available AffectNet benchmark[2]. The AffectNet database provides four affective labels for 2,257 concepts. Each label corresponds to a level of activation in the Hourglass model. Concepts are described according to the m-dimensional vector space defined by AffectiveSpace 2. In the present experimental session, two different configurations of AffectiveSpace were compared: m=100 and m=50.

In order to robustly evaluate the performance of the proposed classification framework, a cross-validation procedure was applied. In particular, we considered ten different experimental runs. In each run, 200 concepts that were randomly extracted from the complete database provided the test set; the remaining concepts were evenly split into a training set and a validation set. In order to test the semi-supervised approach, 500 unlabeled patterns randomly extracted from a total of 16,431 unlabeled concepts were added to the training set. The reference function of the bSVM and bRLS frameworks can be derived from any clustering algorithm; we chose for this purpose the k-means algorithm.

In the present set up, the validation set was designed to support the model selection phase, i.e., the selection of the best parameterization for the classifier. A RBF kernel was adopted, thus the model selection actually involved two parameters: γ and C. In each run the classification accuracy was measured by using the prediction system obtained after the model selection phase. Only the patterns included in the test set were exploited to assess classification accuracy, i.e., the patterns that were not involved in the training phase or in the model selection phase.

Table 3 reports the average classification accuracy obtained by the bSVM and bRLS frameworks over the ten runs. The table provides the results of the two different experiments addressed in this research: the experiment involving the 100-dimensional AffectiveSpace 2 and the experiment involving the 50-dimensional AffectiveSpace 2. Classification accuracy is evaluated according to two different criteria. The first criterion, "Strict Accuracy", refers to the percentage of patterns for which the classification framework correctly predicted the activation level for every affective dimension. The second criterion, "Relaxed Accuracy", assumes that the prediction is correct even in the event the framework fails to properly assess one affective dimension out of four. The relaxed accuracy actually takes into account that a certain level of noise may hinder the AffectNet benchmark, as affective activation levels are assigned according to subjective tests.

The results showed in Table 3 provide a few interesting outcomes. Firstly, both the bSVM-based and the bRLS-based frameworks proved to be able to attain very promising performance in terms of classification accuracy. Indeed, the proposed approach significantly improves over standard classification techniques such as k-NN model, k-medoids, and artificial neural network (ANN). Secondly, both frameworks also attained reliable performance when the 50-dimensional AffectiveSpace 2 was adopted, which consistently reduced the complexity of the reasoning architecture. Such results confirm the ability to deal with complex problems. Moreover, bRLS slightly improves over bSVM.

[2] http://sentic.net/downloads

Method	Strict Accuracy	Relaxed Accuracy
bSVM (100 m)	63%	87%
bSVM (50 m)	62%	89%
bRLS (100 m)	63.5%	88.5%
bRLS (50 m)	64%	90.5%
ANN	46.9%	76.5%
k-medoids	43.2%	74.1%
k-NN	41.9%	72.3%
Random	14.3%	40.1%

Table 3: Emotion recognition accuracy

The proposed affective reasoning framework with bSVM was also evaluated on an opinion mining dataset derived from a corpus developed by Pang and Lee [32]. The corpus consists of 1000 positive and 1000 negative movie reviews from expert reviewers, collected from rottentomatoes.com. All text has been converted into lowercase and has been lemmatized, whilst HTML tags have also been removed. Pang and Lee initially labelled each review manually as either positive or negative, following which the Stanford NLP group annotated this dataset at sentence level [54, 55].

They extracted 11,855 sentences from the reviews and manually labeled them as positive and negative. We used the emotion-polarity formula provided by the Hourglass model to calculate a binary polarity value for each dataset sentence. Table 4 presents the comparison of the proposed system with the state-of-the-art accuracy.

System	Accuracy
Socher et al. 2012	80.00%
Socher et al. 2013	85.40%
Proposed Method	88.50%

Table 4: Comparison with the state of the art

5. Conclusion

We live in a world where millions express their views and opinions of commercial products on the web on a daily basis. This distillation of knowledge due to the massive amounts of unstructured information available, is of considerable importance for tasks such as social media marketing, product positioning, and financial market prediction.

While existing approaches are limited by the fact that they work at word-level need a lot of training, semi-supervised commonsense reasoning seems a good solution to the problem of big social data analysis. In this work, a novel learning model based on the combined use of random projections and support vector machines was exploited to perform reasoning on a knowledge base of affective commonsense. Results showed a significant improvement in both emotion recognition and polarity detection and, hence, paved the way for semi-supervised learning approaches to big social data analysis.

References

[1] S. Poria, E. Cambria, R. Bajpai, A. Hussain, A review of affective computing: From unimodal analysis to multimodal fusion, Information Fusion 37 (2017) 98–125.

[2] E. Cambria, A. Hussain, Sentic Computing: A Common-Sense-Based Framework for Concept-Level Sentiment Analysis, Springer, Cham, Switzerland, 2015.

[3] E. Cambria, D. Olsher, K. Kwok, Sentic activation: A two-level affective common sense reasoning framework, in: AAAI, Toronto, 2012, pp. 186–192.

[4] R. Speer, C. Havasi, ConceptNet 5: A large semantic network for relational knowledge, in: E. Hovy, M. Johnson, G. Hirst (Eds.), Theory and Applications of Natural Language Processing, Springer, 2012, Ch. 6.

[5] C. Strapparava, A. Valitutti, WordNet-Affect: An affective extension of WordNet, in: LREC, Lisbon, 2004, pp. 1083–1086.

[6] E. Cambria, D. Das, S. Bandyopadhyay, A. Feraco, A Practical Guide to Sentiment Analysis, Springer, Cham, Switzerland, 2017.

[7] F. Bisio, P. Gastaldo, R. Zunino, S. Decherchi, Semi-supervised machine learning approach for unknown malicious software detection, in: Innovations in Intelligent Systems and Applications (INISTA) Proceedings, 2014 IEEE International Symposium on, IEEE, 2014, pp. 52–59.

[8] S. Poria, E. Cambria, D. Hazarika, N. Mazumder, A. Zadeh, L.-P. Morency, Context-dependent sentiment analysis in user-generated videos, in: ACL, 2017, pp. 873–883.
[9] I. Chaturvedi, E. Ragusa, P. Gastaldo, R. Zunino, E. Cambria, Bayesian network based extreme learning machine for subjectivity detection, Journal of The Franklin Institute.
[10] F. Xing, E. Cambria, R. Welsch, Natural language based financial forecasting: A survey, Artificial Intelligence Review.
[11] M. Ebrahimi, A. Hossein, A. Sheth, Challenges of sentiment analysis for dynamic events, IEEE Intelligent Systems 32 (5).
[12] E. Cambria, A. Hussain, T. Durrani, C. Havasi, C. Eckl, J. Munro, Sentic computing for patient centered application, in: IEEE ICSP, Beijing, 2010, pp. 1279–1282.
[13] A. Valdivia, V. Luzon, F. Herrera, Sentiment analysis in tripadvisor, IEEE Intelligent Systems 32 (4) (2017) 2–7.
[14] S. Cavallari, V. Zheng, H. Cai, K. Chang, E. Cambria, Joint node and community embedding on graphs, in: CIKM, 2017.
[15] R. Mihalcea, A. Garimella, What men say, what women hear: Finding gender-specific meaning shades, IEEE Intelligent Systems 31 (4) (2016) 62–67.
[16] E. Cambria, S. Poria, A. Gelbukh, M. Thelwall, Sentiment analysis is a big suitcase, IEEE Intelligent Systems 32 (6).
[17] S. Poria, I. Chaturvedi, E. Cambria, F. Bisio, Sentic LDA: Improving on LDA with semantic similarity for aspect-based sentiment analysis, in: IJCNN, 2016, pp. 4465–4473.
[18] Y. Ma, E. Cambria, S. Gao, Label embedding for zero-shot fine-grained named entity typing, in: COLING, Osaka, 2016, pp. 171–180.
[19] Y. Xia, E. Cambria, A. Hussain, H. Zhao, Word polarity disambiguation using bayesian model and opinion-level features, Cognitive Computation 7 (3) (2015) 369–380.
[20] X. Zhong, A. Sun, E. Cambria, Time expression analysis and recognition using syntactic token types and general heuristic rules, in: ACL, 2017, pp. 420–429.
[21] N. Majumder, S. Poria, A. Gelbukh, E. Cambria, Deep learning-based document modeling for personality detection from text, IEEE Intelligent Systems 32 (2) (2017) 74–79.
[22] S. Poria, E. Cambria, D. Hazarika, P. Vij, A deeper look into sarcastic tweets using deep convolutional neural networks, in: COLING, 2016, pp. 1601–1612.
[23] L. Oneto, F. Bisio, E. Cambria, D. Anguita, Statistical learning theory and ELM for big social data analysis, IEEE Computational Intelligence Magazine 11 (3) (2016) 45–55.
[24] C. D. Elliott, The affective reasoner: A process model of emotions in a multi-agent system, Ph.D. thesis, Northwestern University, Evanston (1992).
[25] A. Ortony, G. Clore, A. Collins, The Cognitive Structure of Emotions, Cambridge University Press, Cambridge, 1988.
[26] J. Wiebe, T. Wilson, C. Cardie, Annotating expressions of opinions and emotions in language, Language Resources and Evaluation 39 (2) (2005) 165–210.
[27] T. Wilson, J. Wiebe, P. Hoffmann, Recognizing contextual polarity in phrase-level sentiment analysis, in: HLT/EMNLP, Vancouver, 2005, pp. 347–354.
[28] R. Stevenson, J. Mikels, T. James, Characterization of the affective norms for english words by discrete emotional categories, Behavior Research Methods 39 (2007) 1020–1024.
[29] S. Somasundaran, J. Wiebe, J. Ruppenhofer, Discourse level opinion interpretation, in: COLING, Manchester, 2008, pp. 801–808.
[30] B. Pang, L. Lee, S. Vaithyanathan, Thumbs up? Sentiment classification using machine learning techniques, in: EMNLP, Philadelphia, 2002, pp. 79–86.
[31] B. Goertzel, K. Silverman, C. Hartley, S. Bugaj, M. Ross, The Baby Webmind project, in: AISB, Birmingham, 2000.
[32] B. Pang, L. Lee, Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales, in: ACL, Ann Arbor, 2005, pp. 115–124.
[33] M. Hu, B. Liu, Mining and summarizing customer reviews, in: KDD, Seattle, 2004.
[34] L. Velikovich, S. Goldensohn, K. Hannan, R. McDonald, The viability of web-derived polarity lexicons, in: NAACL, Los Angeles, 2010, pp. 777–785.
[35] A. Gangemi, V. Presutti, D. Reforgiato, Frame-based detection of opinion holders and topics: a model and a tool, IEEE Computational Intelligence Magazine 9 (1) (2014) 20–30.
[36] E. Cambria, S. Poria, R. Bajpai, B. Schuller, SenticNet 4: A semantic resource for sentiment analysis based on conceptual primitives, in: COLING, 2016, pp. 2666–2677.
[37] C. Fellbaum, WordNet: An Electronic Lexical Database (Language, Speech, and Communication), The MIT Press, 1998.
[38] D. Lenat, R. Guha, Building Large Knowledge-Based Systems: Representation and Inference in the Cyc Project, Addison-Wesley, Boston, 1989.
[39] E. Mueller, Commonsense Reasoning, Morgan Kaufmann, 2006.
[40] G. Fauconnier, M. Turner, The Way We Think: Conceptual Blending and the Mind's Hidden Complexities, Basic Books, 2003.
[41] I. Jolliffe, Principal component analysis, Wiley Online Library, 2005.
[42] A. K. Menon, C. Elkan, Fast algorithms for approximating the singular value decomposition, ACM Transactions on Knowledge Discovery from Data (TKDD) 5 (2) (2011) 13.
[43] C. Osgood, W. May, M. Miron, Cross-Cultural Universals of Affective Meaning, Univ. of Illinois Press, 1975.
[44] D. Balduzzi, Randomized co-training: from cortical neurons to machine learning and back again, arXiv preprint arXiv:1310.6536.
[45] H. Lee, R. Grosse, R. Ranganath, A. Y. Ng, Unsupervised learning of hierarchical representations with convolutional deep belief networks, Communications of the ACM 54 (10) (2011) 95–103.
[46] E. Bingham, H. Mannila, Random projection in dimensionality reduction: applications to image and text data, in: ACM SIGKDD, 2001, pp. 245–250.
[47] T. Sarlos, Improved approximation algorithms for large matrices via random projections, in: FOCS, 2006, pp. 143–152.
[48] D. Achlioptas, Database-friendly random projections: Johnson-lindenstrauss with binary coins, Journal of computer and System Sciences 66 (4) (2003) 671–687.

[49] Y. Lu, P. Dhillon, D. P. Foster, L. Ungar, Faster ridge regression via the subsampled randomized hadamard transform, in: Advances in Neural Information Processing Systems, 2013, pp. 369–377.
[50] J. A. Tropp, Improved analysis of the subsampled randomized hadamard transform, Advances in Adaptive Data Analysis 3 (01n02) (2011) 115–126.
[51] N. Ailon, B. Chazelle, Faster dimension reduction, Communications of the ACM 53 (2) (2010) 97–104.
[52] E. Cambria, J. Fu, F. Bisio, S. Poria, AffectiveSpace 2: Enabling affective intuition for concept-level sentiment analysis, in: AAAI, Austin, 2015, pp. 508–514.
[53] T. Yu, S. Simoff, T. Jan, Vqsvm: A case study for incorporating prior domain knowledge into inductive machine learning, Neurocomputing 73 (13) (2010) 2614–2623.
[54] R. Socher, B. Huval, C. D. Manning, A. Y. Ng, Semantic compositionality through recursive matrix-vector spaces, in: EMNLP, 2012, pp. 1201–1211.
[55] R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng, C. Potts, Recursive deep models for semantic compositionality over a sentiment treebank, in: EMNLP, 2013, pp. 1642–1654.