Information Processing and Management 56 (2019) 554–564

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

Cognitive-inspired domain adaptation of sentiment lexicons

Frank Z. Xing[a], Filippo Pallucchini[b], Erik Cambria[*,1,a]

[a] School of Computer Science & Engineering, Nanyang Technological University, Singapore
[b] Dipartimento di Ingegneria Gestionale, Politecnico di Milano, Italy

ARTICLE INFO

Keywords:
Sentiment lexicon
Domain adaptation
Exploration-exploitation
Word polarity knowledge engineering

ABSTRACT

Sentiment lexicons are essential tools for polarity classification and opinion mining. In contrast to machine learning methods that only leverage text features or raw text for sentiment analysis, methods that use sentiment lexicons embrace higher interpretability. Although a number of domain-specific sentiment lexicons are made available, it is impractical to build an *ex ante* lexicon that fully reflects the characteristics of the language usage in endless domains. In this article, we propose a novel approach to simultaneously train a vanilla sentiment classifier and adapt word polarities to the target domain. Specifically, we sequentially track the wrongly predicted sentences and use them as the supervision instead of addressing the gold standard as a whole to emulate the life-long cognitive process of lexicon learning. An exploration-exploitation mechanism is designed to trade off between searching for new sentiment words and updating the polarity score of one word. Experimental results on several popular datasets show that our approach significantly improves the sentiment classification performance for a variety of domains by means of improving the quality of sentiment lexicons. Case-studies also illustrate how polarity scores of the same words are discovered for different domains.

1. Introduction

Domain adaptation (Blitzer, Dredze, & Pereira, 2007) has been identified as a key issue in sentiment analysis and its applications, such as customer opinion mining (Hu & Liu, 2004), human-computer interaction (Shi & Yu, 2018), and business intelligence (Xing, Cambria, & Welsch, 2018b). This is because that many topics we discuss are characterized by their own sub-languages, such as special terminologies and jargons. Being unfamiliar with these topics (or domains), e.g., food, finance, movie, sports etc. will lead to misunderstanding of the sentiment conveyed. For the same reason, we observe that the performances of sentiment analysis usually drop when using sentiment lexicons of the general domain or other irrelevant domains. Therefore, direct use of lexical resources is often suboptimal (Choi & Cardie, 2009). Domain adaptation is a necessary procedure to cast the general domain sentiment lexicon or sentiment classifier for practical use.

Although direct adaptation of the sentiment classifier is popular in recent studies, many critical tasks require their model to be equipped with a sentiment lexicon (Taboada, Brooke, Tofiloski, Voll, & Stede, 2011). These tasks include applications that prefer high interpretability, such as medicine, healthcare, and financial forecasting (Denecke & Deng, 2015; Xing, Cambria, Malandri, & Vercellis, 2018a). However, a frustrating fact is that the attempt to create an *ex ante* lexicon for every domain is Sisyphean: there are endless domains. For instance, the electronic product domain has sub-domains, e.g., camera and phone. As a result, supervision is still preferred to precisely define a language domain. Whereas word-level supervision is hardly accessible because word polarities are

* Corresponding author.
E-mail addresses: zxing001@e.ntu.edu.sg (F.Z. Xing), filippo@sentic.net (F. Pallucchini), cambria@ntu.edu.sg (E. Cambria).
[1] source code available at https://github.com/senticnet/cognitive-inspired-domain-adaptation.

https://doi.org/10.1016/j.ipm.2018.11.002
Received 26 July 2018; Received in revised form 30 October 2018; Accepted 6 November 2018
Available online 09 January 2019
0306-4573/

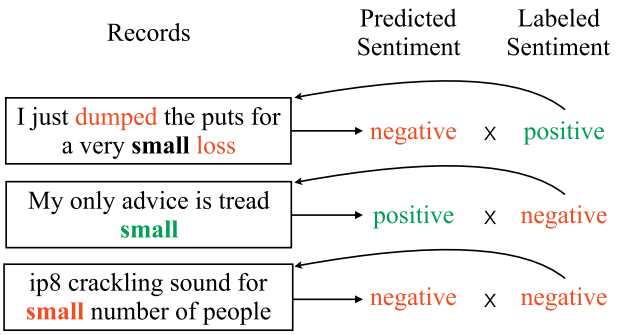

Fig. 1. Simplified illustration of the polarity score adaptation process of word *small* with regard to Apple's products.

considered as the latent information. And the task of deriving a domain-specific sentiment lexicon is trivial if we have direct access to the word-level supervision. On the contrary, language resources for sentiment analysis are usually from social media (Mohammad, Zhu, Kiritchenko, & Martin, 2015) or rating websites (Hung, 2017), where high-level supervisions are provided by users. Therefore, how to leverage high-level (e.g., expression-level, sentence-level, or even document-level) supervision becomes an intriguing question.

In this article, we propose a novel approach that explicitly presents a sentiment lexicon and expects all the polarity scores to change during the learning phase. Like what word embedding (Mikolov, Sutskever, Chen, Corrado, & Dean, 2013) is to the neural language model (Bengio, Ducharme, & Vincent, 2003), the adapted sentiment lexicon is a by-product, yet an important one as the algorithm gradually learns from the high-level supervision. We consider our approach as cognitive-inspired in a sense that the algorithm emulates several metacognition processes when a human agent is exposed to a new language domain: the agent has presumptions of word polarities according to his/her prior knowledge (Pinker, 2004); his/her lexical knowledge would *not* change before a conflict is detected; given the conflict, he/she would try to locate a word as the hypothetical cause of the misunderstanding and apply an alternative understanding, which is subject to further confirmation or disproval in future occasions when the word show up again.

We develop several heuristic rules, such as the exploration-exploitation trade-off and some cognitive constraints to model this process. Fig. 1 depicts the fundamental ideas of our approach. The first record is predicted as negative because *dump* and *loss* are negative words in the original lexicon. However, user labeled this record as positive. From the label our approach identifies neutral word *small* as the cause of this conflict and learns a positive polarity of word *small*. In fact, from the perspective of aspect-based or concept-level sentiment analysis, *small loss* is a positive phrase. However, this knowledge of word *small* is revised by the second record where *thread small* is a negative phrase. This negative polarity of word *small* can be used to predict the sentiment of the third record as negative, which is consistent with the supervision. Although the word that really carries a negative sentiment is *cracking*, all the polarity scores keep unchanged in this situation. In the long run, we could expect the discovery of true word sentiment as training samples continuously arrive. In the following sections, we conduct extensive experiments to demonstrate the effectiveness of our approach.

The remainder of the article is organized as follows. Section 2 reviews existing approaches to the problem of domain adaptation. Section 3 introduces several heuristic rules used in our approach and presents a domain adaptation algorithm. Experimental results are discussed in Section 4. Section 5 concludes the paper.

2. Related work

There are two clusters of existing methods in the research of domain adaptation. One is mainly based on transfer learning (Blitzer et al., 2007; Dong & de Melo, 2018; Socher et al., 2013; Wu & Huang, 2016; Wu, Huang, & Yan, 2017; Xia, Cambria, Hussain, & Zhao, 2015), the other is based on manipulation of sentiment lexicons (Araque, Guerini, Strapparava, & Iglesias, 2017; Choi & Cardie, 2009; Hamilton, Clark, Leskovec, & Jurafsky, 2016; Martineau & Finin, 2009; Tai & Kao, 2013; Turney, 2002). Transfer learning based methods try to learn a shared latent feature among different domains. For example, Blitzer et al. Blitzer et al. (2007) proposed to transfer domain knowledge by using pivot features; others also proposed to align domain-specific words (Xia et al., 2015), or to extract a global sentiment graph using multi-task learning (Wu & Huang, 2016). An important assumption of these methods is a universal distribution of sentiment features in different domains, which is questionable in practice especially when the transfer gap is large. Therefore, it could happen that adapted models perform even worse, which is known as *negative transfer* (Wu et al., 2017). Another disadvantage of these methods lays on the fact that the domain knowledge learned is stored as model parameters. These models could be powerful, however, not many human-interpretable parts can be analyzed from outside. The black-box metaphor is even apter when the model parameters do not hold a meaning of word representations. This transparency issue is not only limited to sentiment analysis, but also to a wider range of natural language processing tasks (Cambria, Poria, Gelbukh, & Thelwall, 2017).

Our approach falls into the second cluster, i.e., domain adaptation with the presence of sentiment lexicons. We categorize this cluster into three threads in terms of the type of supervision:

1. Seed-words supervised approaches assume a core set of words with consistent polarities across all the language domains, e.g., *love, hate, good, bad*. These polarities can be later propagated via a constructed graph. Besides the original words, out-of-vocabulary words can be added to sentiment lexicons (Wang, Zhang, & Liu, 2017) and polarity scores can be propagated to them on a constructed graph. The graph can either be a hand-crafted thesaurus (Vicente, Agerri, & Rigau, 2014) or based on corpus statistics, e.g., represented by a word similarity matrix in terms of context (Tai & Kao, 2013) or word embeddings (Tang et al., 2014). In particular, Tai and Kao Tai and Kao (2013) integrated three sources: WordNet (Fellbaum, 1998), conjunction rule (Hatzivassiloglou & McKeown, 1997), and the second-order co-occurrence pointwise mutual information (PMI) (Islam & Inkpen, 2006) to build the graph. Polarity scores are propagated using random walk (Hassan & Radev, 2010) and its variants (Hamilton et al., 2016; Velikovich, Blair-Goldensohn, Hannan, & McDonald, 2010). The bottleneck to these approaches is the lack of ground truth word correlation graph for every specific domain. A very large corpus is usually required to derive the good estimation of word relations.
2. Label supervised approaches try to leverage high-level supervision. The supervision can sometimes be hard to obtain and indirectly derived, especially in financial domain (Lee, Surdeanu, MacCartney, & Jurafsky, 2014; Xing, Cambria, & Welsch, 2018c). For instance, some researchers used stock price movement as implicit indicators for sentiment polarity of news articles (Ito, Izumi, Tsubouchi, & Yamashita, 2016; Moore, Rayson, & Young, 2016). Other ways to create surrogate supervision include training a classifier from ground truth lexicon, and fix the parameters to assist learning word polarities of the remaining unknown words (Bravo-Marquez, Frank, & Pfahringer, 2015). When labels are accessible, word polarities can be directly calculated using term frequency (Jimenez-Zafra, Martin-Valdivia, Molina-Gonzalez, & Lopez, 2016) or PMI. However, more complicated algorithms can be employed to tune polarity scores to achieve high-level consistency, e.g., linear programming (Choi & Cardie, 2009), stochastic gradient descent (Vo & Zhang, 2016), and adaptive query (Wu et al., 2017). Choi and Cardie (Choi & Cardie, 2009) formulated the lexicon adaptation task using integer linear programming to allow considering both word-to-word and word-to-expression relations. Other relations and features can be considered include n-grams, part-of-speech (POS), term frequency (Du, Tan, Cheng, & Yun, 2010; Moore et al., 2016), term frequency-inverse document frequency (TFIDF) and its variants, (positive) PMI (Hamilton et al., 2016; Martineau & Finin, 2009; Turney, 2002), etc. Araque et al. Araque et al. (2017) showed that a simple end-to-end multilayer perceptron (MLP) with polarity scores embedded in the neural network weights is trainable given multiple domains and their known "distance" as the supervision. This method is mathematically tractable, but seems not the way how we learn a language and requires data from many different domains. The polarities learned by the above-mentioned label supervised approaches usually do not guarantee consistency with each other. This issue is addressed by other studies (Schneider & Dragut, 2015).
3. Unsupervised approach is possible when deep syntactic analyses are applied to identify the clauses that carry polarity, and candidate general-domain words are selected to fit in the context (Kanayama & Nasukawa, 2006). Ofek et al. Ofek et al. (2016) further used an approach similar to that in Kanayama and Nasukawa (2006) for lexicon expansion and the statistical co-occurrence information to generate a direct acyclic graph to enrich a concept-level sentiment lexicon (Ofek et al., 2016). However, in the methods of Kanayama and Nasukawa (2006); Ofek et al. (2016), polarity scores for seed words or even the whole original sentiment lexicon are taken as the ground truth and not exposed to the learning phase. A very strong assumption here is the similarity between the target domain and the starting lexicon they use. Like discussed before, this is the situation when transfer learning can also achieve fairly good performance. In this sense, our approach has more flexibility when the starting lexicon performs poorly on the target domain.

The most similar to our idea are studies as in Melville, Gryc, and Lawrence (2009); Moore et al. (2016); Wu et al. (2017). Melville et al. Melville et al. (2009) observed that the sentiment polarity of terms learned by their Pooling multi-nominals model is very different from the background knowledge model, and this up-weighting/down-weighting of terms can be seen as a domain adaptation process. However, no discussion has been made on determining polarity scores in Melville et al. (2009). We extend their work by specifying an algorithm to do so from a cognitive perspective. In contrast to the active query strategy that selects the most informative instance to learn (Wu et al., 2017), our sequential learning approach is almost passive. However, our model performance is comparable to them without parameter tuning and has sentiment lexicon as a by-product. Instead of using price change directions as alternative supervision (Moore et al., 2016), we leverage user labels such as bullish and bearish. We believe that these sentiment labels reflect more direct features with regard to the natural language.

3. Our approach

In this section, we introduce notations used in the remaining parts of this paper and how we implement the exploration-exploitation strategy for word polarity adaptation. We denote the polarity score for word x by $\delta(x)$. A typical domain adaptation problem is to derive a mapping from the source domain lexicon to the target domain lexicon: $\mathcal{L} \mapsto \mathcal{L}'$. Furthermore, the sentiment lexicon as a starting point can be represented by $\mathcal{L}^D(\mathbf{x}: \delta(\mathbf{x}))$, where the vocabulary size of $\mathcal{L}$ is D. Suppose our training dataset consists of N records T_i[2], where $1 \le i \le N$ is the record index. The corresponding sentiment label for T_i is y_i. Hence, we can train a sentiment classification algorithm $f_{\mathbf{T}}(T, \mathcal{L})$ that outputs a prediction label y for input record T. Specific choice of $f(\cdot)$ does not affect the following discussions. The rest of this section explains the cognitive intuitions behind the proposed algorithm.

[2] Most of the records T_i contain only one sentence, though in practice this will depend on how the training data is labeled or partitioned.

3.1. Vectorization of sentiment features

Many features have been proposed for sentiment classification tasks besides sentiment lexicons and polarity scores, e.g., term frequencies, POS tags, negators, syntactic features, and more (Manek, Shenoy, Mohan, & Venugopal, 2017). However, the aim of our research is to adapt a sentiment lexicon rather than to train the best classifier. Therefore, we make use of solely sentiment information for training. We represent each record T with a D-dimensional vector, where the dimension indices of the vector indicate the unique location of the word in $\mathcal{L}$. That is, for $x = V(T)_i$, if x is in $\mathcal{L}$, we assign the polarity score of $x \in [-1, 1]$ to $V(T)_i$.

$$V(T)_i = \begin{cases} 0 & \text{if } x \notin \mathcal{L} \\ \delta(x) & \text{if } x \in \mathcal{L} \end{cases} \tag{1}$$

A binary label $y \in \{+,-\}$ is used to denote the classification result as either positive or negative.

3.2. Exploration-Exploitation

After training of the classification algorithm, for each record T_i, we can check the predicted sentiment label $\hat{y}_i = f_{\mathbf{T}}(T_i, \mathcal{L})$. If $\hat{y}_i \neq y_i$, our algorithm starts to check whether there exists a better representation of the record that can correct this error.

However, this error may be attributed to any word $V(T_i)_j$ in record $V(T_i)$. A search within all the combinations of word polarities would require an unrealistic amount of computing power. Therefore, we are facing a trade-off between exploiting the correct polarity score for a certain word and exploring other words to correct the predicted label. As several psycholinguistic theories have pointed out, the final comprehension is an integration of different levels of sentiment activation of words (Lemaire, Denhiere, Bellissens, & Jhean-Larose, 2006). We naturally explore activated words in descending order of the absolute value of their polarity scores.

For each word $x \in V(T_i) \cap \mathcal{L}$, the algorithm performs polarity score assignment according to the following rule:

$$\delta(x)' = \delta(x) + \epsilon \tag{2}$$

where $\epsilon \in [-1, 1]$ is a random float number generated from a uniform distribution. The algorithm decides whether to adapt the new polarity score $\delta(x)'$ before exploring another word. In this exploitation phase, we consider only a subset of the training dataset:

$$\mathbf{T}_x = \{T \in \mathbf{T} \mid x = V(T)_j, \forall j\} \tag{3}$$

where j is the index of words constituting the record.

Thus, classification performances with both polarity scores can be computed and compared. The original performance on this subset is:

$$P(x) = \frac{\sum_{k \in \mathbf{T}_x} \#\text{ of } \{\hat{y}_k = y_k\}}{\#\text{ of } \mathbf{T}_x} \tag{4}$$

If we substitute $\delta(x)$ with $\delta(x)'$ in $\mathcal{L}$, then with this new lexicon the predicted labels can be re-computed as $\hat{y}_k{}' = f_{\mathbf{T}}(T_k, \mathcal{L}')$. Using this $\hat{y}_k{}'$ in Eq. 4, we have the new performance P'. Thus, we only decide to register the new polarity score before exploring the next word if the performance improvement exceeds a certain threshold:

$$\Delta P(x) = P'(x) - P(x) \geq \theta \tag{5}$$

Otherwise, the algorithm continues to try with a new ϵ. We impose a maximum number of iterations on Eq. 2 to avoid endless exploitation.

In either exploration or exploitation phase, the algorithm will stop and process the next record T_{i+1} if the wrongly predicted label is corrected. Every time the sentiment lexicon is confirmed updated, we re-train the classification algorithm $f'(\mathbf{T}, \mathcal{L}')$. The next records will be predicted by this new algorithm.

3.3. Convergence constraints

The exploration-exploitation strategy does not guarantee the convergence of changing polarity scores. In early simulations, we observed this phenomenon: the updates of some word polarities exhibit no trend and swing from positive to negative. In theory, errors in updates can come from two sources: (1) a wrong word is identified for polarity exploitation and, (2) the condition described by Eq. 6 allows repeated updates of conjugated words, i.e., a minor improvement for one word causes major performance drop for another word.

However, human learning of word sentiments is more stable, because previous experience is stored in our memory. As sample size increases, the uncertainty of polarity scores will diminish. Then, we refine polarities in a narrower range. We emulate this mental process by the following convergence constraint, which is a refinement of Eq. 2:

$$\delta(x)'' = \delta(x)' + \zeta \tag{6}$$

where random variable ζ is drawn from a narrowing range $\zeta \in [-1/\text{count}(x),\ 1/\text{count}(x)]$, and $\text{count}(x) \in \mathbb{Z}$ records how many times the polarity score of x has been updated so far. This count includes not only times of exploitation, but also inter-instance exploration. The convergence constraint allows deeper and deeper investigation into the polarity score of word x.

3.4. Consistency constraints

Among all the different domains, words that switched their sentiment orientations are rarer and appear interesting to us. Previous research (Hamilton et al., 2016) showed that even from a diachronic perspective, there is only a small portion of words totally switched their polarity in a relatively long history. The reversal of word polarity will not appear very often compared to other types of sentiment shifting, especially in relevant domains, because language is a continuous system. Therefore, we decided to check for knowledge integration after the exploration-exploitation phase. The set of words to check is composed of words that switched their polarity and those not contributing much to the classification performance, that is:

$$\begin{aligned}\mathbf{x}_s = \{x \in \mathcal{L}' \mid \delta(x)'\delta(x) &< 0 \\ \cup P'(x) &< \beta\}\end{aligned} \tag{7}$$

where β is an empirically decided hyper-parameter that measures the desired performance level.

Then, for each word $x \in \mathbf{x}_s$, the algorithm performs another exploration-exploitation phase based on $\mathcal{L}'$.

3.5. Dealing with negators

Negation is a prevailing phenomenon and can appear at many levels of natural language (Choi & Cardie, 2009; Zhu, Guo, Mohammad, & Kiritchenko, 2014). However, to automatically identify the scope of negation is not an easy task (Fancellu, Lopez, Webber, & He, 2017). Generally, negators can be categorized into function words, e.g., *no, not, never, seldom*; and content words, e.g., *destroy, prevent*, etc. Since content words usually carry some polarity with themselves, we only deal with function-word negators in our vectorization of records. The simple rule we apply here is to reverse the output of $f(\cdot)$ when a single function-word negator is detected.

3.6. Lexicon expansion

Target domains are not only characterized by words with different polarity scores, but also neologisms, especially in web-based environments, e.g., new forms of microtext. Given expression-level or sentence-level supervision, such as in tweets, a record may not contain any word from the sentiment lexicon, that is

$$\mathbf{T}_e = \{T_i \mid x \notin \mathcal{L}\ ,\ \forall\, x \in T_i\} \tag{8}$$

In this case, we can confirm that some word in the record that carries a polarity is absent from the lexicon. The lexicon expansion algorithm first checks the POS tag of each word in records, and only adds *nouns, verbs*, or *adjectives* to the lexicon.

In order to determine the initial polarity score of newly added word x, we use a heuristic by considering $\mathbf{T}_x$. Assume $\mathbb{I}(pos)$ is the total number of positive records in the $\mathbf{T}_x$, and $\mathbb{I}(x, pos)$ denotes the frequency of word x appears in positive records of $\mathbf{T}_x$. The polarity score of word x can be calculated as a regularized difference of pointwise mutual information (PMI).

$$\begin{aligned}\delta(x) &= \tanh(\mathrm{PMI}(x, pos) - \mathrm{PMI}(x, neg)) \\ &= \tanh\left(\log_2 \frac{\mathbb{I}(x, pos)\cdot\mathbb{I}(neg)}{\mathbb{I}(x, neg)\cdot\mathbb{I}(pos)}\right)\end{aligned} \tag{9}$$

This heuristic is widely used in automatic induction of lexicon polarity scores (Mohammad, Kiritchenko, & Zhu, 2013; Vo & Zhang, 2016). We further implement regularization and smoothing technique to avoid division by zero for Eq. 9.

Later, polarity scores of newly added words will be updated using the same mechanism described above if they are activated again. In experiments, we observed this lexicon expansion for all lexicons during adaptation.

3.7. Boosting and the algorithm complexity

Since assignments of polarity scores allow randomness, the errors in the final lexicon would be different from time to time. Consequently, stochastic shifts of polarity scores can be eliminated by averaging polarity scores of the final lexicon from multiple experiments (loops). Meanwhile, deterministic polarity shifts are preserved and augmented. Finally, we present our supervised algorithm 1 termed “Cognitive-inspired Domain Adaptation with Higher-level Supervision (CDAHS)”. The algorithm ensembles the constraints introduced to the exploitation-exploration strategy and realizes the PMI based polarity scores for lexicon expansion. In particular, multiple conditions (line 9 to 12) together control which action to take in the current state.

The time complexity of algorithm 1 is non-trivial and depends on various factors. It is obvious that the complexity is proportional to the iteration times n of boosting. While the computational cost of training and performing the classifier $f(\cdot)$ can vary a great deal. Take the SVM implementation we used as an example, the worst training case involves with solving the inverse kernel matrix, which is $O(N^3)$. On average, we estimate the empirical training complexity as $O(N^2D)$ and discriminative complexity as $O(D)$ (Bordes, Ertekin, Weston, & Bottou, 2005; Shalev-Shwartz & Srebro, 2008), where N is the initial number of records for training and D is the lexicon size. Then, let t be the average assigning time under the consistency constraint. Apparently, t is both a function of hyper-parameter θ and an indicator of the quality of the original lexicon. The bad prior knowledge will severely slow down the adaptation process and a too high desired performance will kill the exploitation phase. The time complexity for one loop is

$O(N^2D + D(\frac{D}{2} + 2D{\cdot}t) + D)$. Therefore, the overall implementation complexity is $O(nN^2D + nD^2t)$.

4. Experiments

A common suspicion about the approach is that allowing some randomness, the adaptation of sentiment lexicon will not evolve toward a deterministic direction. Although intuitive and cognitive-inspired, the experimental results suggest that our approach works amazingly well on all the lexicons experimented and improves classifiers' performance.

4.1. Lexicons and datasets

In this section, we describe the original sentiment lexicons and the supervision from multiple domains used for domain adaptation. The four sentiment lexicons we experimented are: Opinion Lexicon, SentiWordNet, the Loughran & McDonald dictionary, and SenticNet. The six target domains we considered are: *Apparel, Electronics, Kitchen, Healthcare, Movie, and Finance*. Consequently, we obtained 24(4×6) combinations in total. We provide more details about the original lexicons as follows.

1. **Opinion Lexicon** (Hu & Liu, 2004) is a popular word list which contains around 6000 positive and negative terms. The lexicon also contains common social media misspelled words.
2. **SentiWordNet** (Baccianella, Esuli, & Sebastiani, 2010) is a lexicon that assigns continuous sentiment values to the nearly 117,000 synsets of the WordNet lexical database for English. A word may have multiple scores if it belongs to diverse synsets. To solve this issue, we average the sentiment values for all the POS tag entries under the same word.
3. **L&M** (Loughran & McDonald, 2011) is a widely recognized sentiment word list in finance domains. We adopt all of the 354 positive words and 2349 negative words commonly applied in financial documents.
4. **SenticNet** (Cambria, Poria, Hazarika, & Kwok, 2018) contains not only words, but also multi-word-concepts. The latest version has over 100,000 entries and each entry is associated with affective information, such as semantics, mood tags, and a polarity score.

We obtain supervision of *Apparel, Electronics, Kitchen, Healthcare* domains from the Multi-Domain Sentiment Dataset v2.0[3] (Blitzer et al., 2007); supervision of the *Movie* domain from sentence polarity dataset v1.0[4] (Pang & Lee, 2005); and supervision of the *Finance* domain from our own data stream collected from Stocktwits[5]. See Table 1 for the basic statistics of these labeled datasets.

4.2. Preprocessing

Our *Finance* dataset is very challenging because it is much noisier and many sentiments are expressed by prices and numbers, which requires commonsense knowledge to understand. Therefore, besides stop-word removal and lemmatization, which we do for all the datasets[6], we further remove URLs, non-ASCII characters, hashtags and substitute some microtexts and acronyms. Table 2 provides examples of the clean-up results.

4.3. Performance evaluation

We make use of the labeled data in Table 1 as supervision. The same number of positive and negative records (which is fewer) are used because dealing with unbalanced data is out-of-scope for our discussion. We train a linear SVM with squared-hinge loss function as the sentiment classification algorithm. That is to solve the following optimization problem (Fan, Chang, Hsieh, Wang, & Lin, 2008):

$$\min_{\mathbf{w},b} \ \frac{1}{2}\mathbf{w}^{\intercal}\mathbf{w} + \sum_{i=1}^{n} \ell[y_i(\mathbf{w}^{\intercal}V(T)_i + b) - 1] \tag{10}$$

where the loss function is:

$$\ell(t) = \begin{cases} (1-t)^2, & t < 1, \\ 0, & t \geq 1. \end{cases}$$

Other hyper-parameters are set as: the exploitation limit $\alpha = 10$, the desired performance level $\beta = 0.6$, the performance improvement threshold $\theta = 0.01$, the number of iterations for boosting $n = 5$. We do 3-fold cross-validation and report the average performance in terms of classification accuracy.

In particular, we compare the classification accuracies of our method with several strong benchmarks: (1) NBSVM (SVM with Naïve Bayes features) (Wang & Manning, 2012), which uses Naïve Bayes log-count ratios as features for the SVM. This modification

[3] http://www.cs.jhu.edu/~mdredze/datasets/sentiment/
[4] http://www.cs.cornell.edu/pabo/movie-review-data/
[5] https://stocktwits.com/
[6] Implemented with NLTK.

Table 1
Statistics for datasets.

Domain	Apparel	Electro.	Kitchen	Health.	Movie	Finance
positive	1000	1000	1,000	1,000	5,331	16,881
negative	1000	1000	1,000	1,000	5,331	4,866
unlabeled	7252	21,009	17,856	5,225	0	33,579

Table 2
Examples of record in *Finance* domain.

Record 1	Clean-up
Clean-up	Apple high price makes it a risky bet
Record 2	$AAPL needs to chew thru trendline rez & amp; build value in this area b4 resuming higher imho
Clean-up	Apple needs to chew through trendline reservation and build value in this area before resuming higher in my humble opinion
Record 3	Couldn't take any more of Bobbie's useless drivel
Clean-up	Couldn't take any more of Bobbie's useless drivel

trust Naïve Bayes classifier unless the SVM is very confident. In the context of sentiment lexicon rather than sentiment concepts, we adopt uni-gram features for the experiments; (2) TFIDF (Martineau & Finin, 2009), which does not present a sentiment lexicon just like NBSVM. In theory, the information presented by TFIDF is the upper bound for all the bag-of-words like models without any prior knowledge about word sentiment; (3) the widely recognized method in automatic induction of lexicon polarity scores (AIPS) from short internet texts (Mohammad et al., 2013). The results are provided in Table 3.

4.4. Results

In Table 3, classification accuracies of using the original sentiment lexicons are presented in the first row, and after domain adaptation in the second. In fact, AIPS is the main approach we should compare to because unlike NBSVM and TFIDF, AIPS induces a sentiment lexicon while the other two do not. We notice that the AIPS method is not robust across different domains. In *Finance* domain it works extremely well, probably because the dataset is consisting of short texts and there is abundant supervision. However, in other domains the performances are very poor. For example, in *Health* domain using the calculated polarity scores is worse than a random guess. In contrary, TFIDF can produce acceptable results, though this method does not provide any sentiment lexicon.

Our method can always outperform TFIDF with certain sentiment lexicon as a starting point. No negative learning is observed, i.e., the performance *always* improve after domain adaptation. This is the key difference between our CDAHS algorithm and other transfer learning based methods. The observation is not surprising since our algorithm forces strict constraints that the polarity shifts should (at least locally) correct the classification errors. The assumption that the target domain and incoming records share the same latent distribution is self-evident. While in the transfer learning case, the assumption that sentiment words share the same latent distribution across domains may not hold.

In general, SenticNet is of the highest quality before domain adaptation, in a sense that the average classification accuracy is 65.5%, followed by Opinion Lexicon (63.4%) and SentiWordNet (62.4%). L&M is already a domain-specific lexicon, therefore, it is not hard to understand that its performance is the lowest (58.0%) across different domains. Moreover, its performance in *Finance* domain is very ordinary. This shows how great the difference could be within the "financial domain group".

After domain adaptation, the gaps between different lexicons are narrowed. SenticNet is still slightly better (71.2%), followed by Opinion Lexicon (69.7%), SentiWordNet (67.4%), and L&M (65.8%). Our method improves L&M (7.8%) and Opinion Lexicon (6.3%)

Table 3
Sentiment classification accuracies for different domain and lexicon combinations.

	Apparel	Electronics	Kitchen	Healthcare	Movie	Finance	Average
NBSVM	**75.4%**	65.8%	67.1%	65.2%	77.9%	69.5%	70.2%
TFIDF	74.2%	66.0%	65.0%	65.0%	75.4%	68.1%	68.9%
AIPS	54.5%	53.3%	51.3%	49.0%	53.1%	**71.7%**	55.5%
Opinion Lexicon	66.2%	64.2%	62.5%	60.3%	69.4%	58.0%	63.4%
	72.8%	**69.2%**	**69.7%**	**66.5%**	74.7%	65.6%	69.8%
SentiWordNet	66.5%	63.2%	59.3%	59.2%	68.4%	57.7%	62.4%
	71.0%	65.9%	64.1%	64.2%	75.2%	63.6%	67.4%
L&M	63.2%	60.0%	61.5%	53.2%	58.0%	54.0%	58.0%
	70.5%	64.2%	68.3%	62.7%	69.1%	62.0%	65.8%
SenticNet	70.5%	64.8%	60.5%	63.2%	71.0%	62.7%	65.5%
	74.7%	**69.2%**	69.3%	65.7%	**77.9%**	**69.8%**	**71.2%**

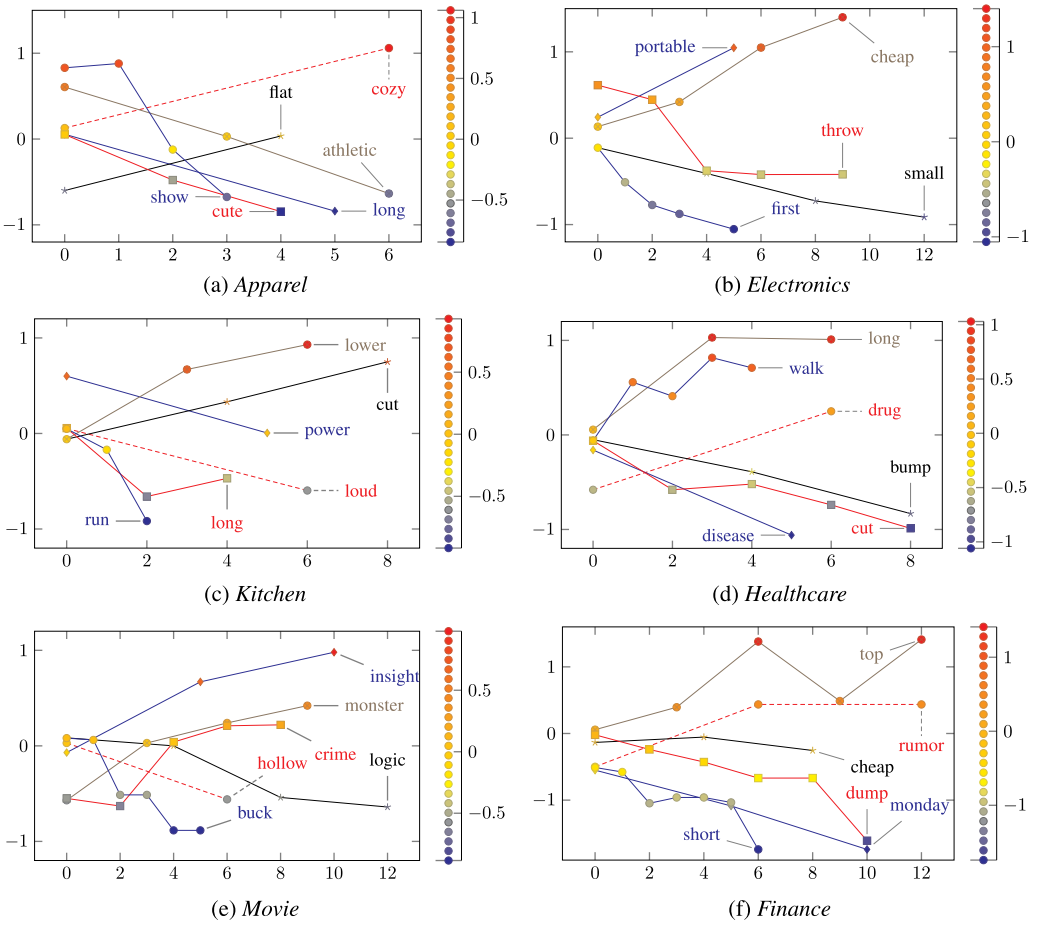

Fig. 2. Sentiment shifts of words in different domains (x-axis: the number of iterations experienced for polarity exploitation; y-axis: the polarity score of word sentiment).

more than SenticNet (5.7%) and SentiWordNet (5.0%). This is probably because the former two have relatively small vocabulary size (coverage issue of manually crafted lexicons). As a result, more words are added in the lexicon expansion phase with dataset-induced polarity scores. This may also imply that an optimal number of seed words exists: a balance between prior knowledge and the agility to adapt to a new domain.

Next, we provide case studies to check whether word polarities are effectively learned in each domain.

4.5. Qualitative analysis

Developing sentiment lexicons with machine learning based techniques usually introduces the problem of overfitting (Medhat, Hassan, & Korashy, 2014). Words that carry no sentiment are assigned polarity scores because of the coincidental statistical imbalance. These lexicons may perform extremely well on the training dataset, but have poor generalization ability. To examine whether the adaptation of word polarities in our method is consistent with commonsense knowledge, we look into some words that most intensively changed their polarity scores. These words are more important because other polarity shifts can be trimmed with various techniques. Fig. 2 provides such insight. Due to limited space, only results of SenticNet as the original sentiment lexicon are reported because of its exceptional performance both before and after adaptation.

Many examples are discussed by previous research: *war, dark* and *complex* are considered positive in description of movies (Melville et al., 2009); *easy* is usually used in positive reviews in *Electronics* domain, e.g., *easy to use*, however, it is negative in *Movie* domain (Wu & Huang, 2016); *unpredictable* is positive in *Movie* domain, e.g., *the plot of this movie is fun and unpredictable*, however, it is a negative word in *Kitchen* domain (Wu et al., 2017). Our observations support all these claims. Furthermore, we spot other examples generated by our domain adaptation process.

In our experiments, many word polarities shifted from almost neutral to opposite directions in different domains. For instance, *cheap* is neutral in general domain. However, it changed to positive in *Electronics* domain, because it is a desirable property for customers. In *Finance* domain, in contrary, people do not like *cheap stocks*, so the polarity became slightly negative. Similarly, *long* is positive in *Health* and *Finance* domain, e.g., *long life, long position.* However, it is negative in *Apparel* and *Kitchen* domain. *Crime* and

loop for n times
  train $f_{\mathbf{T}}(T, \mathcal{L})$;
  for $T_i \in \mathbf{T}$ **do**
    if $T_i \in \mathbf{T}_e$ && $x \in T_i$ **then**
      if $POS(x) = NN \,||\, VB \,||\, JJ$ **then**
        $\mathcal{L} \leftarrow [x; \delta(x)]$;
      end
    end
    if $f_{\mathbf{T}}(T_i, \mathcal{L}) \neq y_i$ **then**
      for $\mathbf{x}_{ij} \in T_i$ **do**
        while $\Delta P < \theta$ && $j < |\mathbf{x}_i|$ **do**
          if $count(\mathbf{x}_{ij}) < \alpha$ **then**
            $\delta(\mathbf{x}_{ij}) \leftarrow \delta(\mathbf{x}_{ij}) + \zeta$;
          else
            $j \leftarrow j + 1$;
          end
        end
        $\mathcal{L}' \leftarrow \delta(\mathbf{x}_{ij})$;
      end
    end
  end
  $\mathbf{x}_s \leftarrow$ comparing $\mathcal{L}$ and $\mathcal{L}'$;
  do line 9 to line 20 **for** $x \in \mathbf{x}_s \cap T_i$;
end
return $\bar{\mathcal{L}}' \leftarrow \sum(\mathcal{L}'/n)$;

Algorithm 1. CDAHS($\mathcal{L}$, **T**).

monster are usually regarded as negative words in general but not in the *Movie* domain; *power* does not refer to strength or capacity, but the source of energy in *Kitchen* domain.

Some less intuitive examples are associated with jargons and language usage. For instance, *monday* is neutral in the general domain. However, market crashes and liquidity problem are more likely to happen on Monday (Antweiler & Frank, 2004; Xing et al., 2018c), e.g., *Black Monday*.

The word *logic* changed to negative in the *Movie* domain because expressions like *I dislike the movie because it doesn't have any logic* are more natural and more likely than *I like the movie because it has logic*. Similarly, when people talk about rumor in *Finance* domain, they usually imply insider information that could make a profit, not bad news that could cause loss.

Another interesting point to notice is that in *Movie* and *Finance* domain, word polarity scores change more often. Since these two domains have quintuple the size of other domains in terms of the number of records, we believe it enables more accurate search for word polarities. When supervision is weak, the adjustment takes bigger steps and not suffices to correct small deviation from the underlying word polarities.

5. Conclusion and future work

In this article, we presented a novel approach to adapt an existing sentiment lexicon to the target domain by a sequence of labeled records. The sequential learning algorithm is almost passive, i.e., the order of learning samples can not be changed and the algorithm can not select samples to learn from. However, the performance is comparable to some active query strategy that selects the most informative instance to learn. Our method is robust, cognitive-inspired, and embraces high interpretability. We developed several heuristics to emulate the learning processes when we are exposed to a new language domain. In particular, our method has some desirable properties, i.e., no negative learning in terms of polarity scores and gives consistent gains on sentiment analysis tasks. Moreover, our approach presents a new sentiment lexicon for the target domain. Extensive experiments showed that our method improves sentiment classification accuracies by 6.2% on average. Qualitative analysis indicates that our method captures domain-specific word usage instead of other non-sentimental statistical features. In future work, we propose to extend the approach to concept-level sentiment adaptation with a hierarchy of primitives. This rich structure would help to achieve more accurate semantic compositionality. Another direction is to study the alignment of multiple sentiment lexicons and knowledge distillation from them. In the current form, the algorithm only starts from one existing lexicon and information such as the overlap and conflicts between lexicons are not utilized.

References

Antweiler, W., & Frank, M. Z. (2004). Is all that talk just noise? the information content of internet stock message boards. *Journal of Finance, 59*(3), 1259–1294.

Araque, O., Guerini, M., Strapparava, C., & Iglesias, C. A. (2017). *Neural domain adaptation of sentiment lexicons. Seventh international conference on affective computing and intelligent interaction workshops and demos (ACIIW)* 105–110.

Baccianella, S., Esuli, A., & Sebastiani, F. (2010). *Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. 7th language resources and evaluation conference (LREC)* 2200–2204.

Bengio, Y., Ducharme, R., & Vincent, P. (2003). A neural probabilistic language model. *Journal of Machine Learning Research, 3*, 1137–1155.

Blitzer, J., Dredze, M., & Pereira, F. (2007). *Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. ACL* 440–447.

Bordes, A., Ertekin, S., Weston, J., & Bottou, L. (2005). Fast kernel classifiers with online and active learning. *Journal of Machine Learning Research, 6*, 1579–1619.

Bravo-Marquez, F., Frank, E., & Pfahringer, B. (2015). *Positive, negative, or neutral: Learning an expanded opinion lexicon from emoticon-annotated tweets. IJCAI* 1229–1235.

Cambria, E., Poria, S., Gelbukh, A., & Thelwall, M. (2017). Sentiment analysis is a big suitcase. *IEEE Intelligent Systems, 32*(6), 74–80.

Cambria, E., Poria, S., Hazarika, D., & Kwok, K. (2018). *Senticnet 5: Discovering conceptual primitives for sentiment analysis by means of context embeddings. AAAI* 1795–1802.

Choi, Y., & Cardie, C. (2009). *Adapting a polarity lexicon using integer linear programming for domain-specific sentiment classification. EMNLP* 590–598.

Denecke, K., & Deng, Y. (2015). Sentiment analysis in medical settings: New opportunities and challenges. *Artificial Intelligence in Medicine, 64*(1), 17–27.

Dong, X., & de Melo, G. (2018). *A helping hand: Transfer learning for deep sentiment analysis. ACL* 2524–2534.

Du, W., Tan, S., Cheng, X., & Yun, X. (2010). *Adapting information bottleneck method for automatic construction of domain-oriented sentiment lexicon. WSDM* 111–120.

Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., & Lin, C.-J. (2008). Liblinear: A library for large linear classification. *Journal of Machine Learning Research, 9*, 1871–1874.

Fancellu, F., Lopez, A., Webber, B., & He, H. (2017). *Detecting negation scope is easy, except when it isn't. EACL* 58–63.

Fellbaum, C. (1998). *Wordnet: An electronic lexical database.* MIT Press.

Hamilton, W. L., Clark, K., Leskovec, J., & Jurafsky, D. (2016). *Inducing domain-specific sentiment lexicons from unlabeled corpora. EMNLP* 595–605.

Hassan, A., & Radev, D. (2010). *Identifying text polarity using random walks. ACL* 395–403.

Hatzivassiloglou, V., & McKeown, K. R. (1997). *Predicting the semantic orientation of adjectives. EACL* 174–181.

Hu, M., & Liu, B. (2004). *Mining and summarizing customer reviews. Proceedings of the tenth ACM SIGKDD international conference on knowledge discovery and data mining* 168–177.

Hung, C. (2017). Word of mouth quality classification based on contextual sentiment lexicons. *Information Processing & Management, 53*(4), 751–763.

Islam, A., & Inkpen, D. (2006). *Second order co-occurrence pmi for determining the semantic similarity of words. 5th language resources and evaluation conference (LREC)* 1033–1038.

Ito, T., Izumi, K., Tsubouchi, K., & Yamashita, T. (2016). *Polarity propagation of financial terms for market trend analyses using news articles. IEEE congress on evolutionary computation (CEC)* 3477–3482.

Jimenez-Zafra, S. M., Martin-Valdivia, M. T., Molina-Gonzalez, M. D., & Lopez, L. A. U. (2016). *Domain adaptation of polarity lexicon combining term frequency and bootstrapping. NAACL-HTL* 137–146.

Kanayama, H., & Nasukawa, T. (2006). *Fully automatic lexicon expansion for domain-oriented sentiment analysis. EMNLP* 355–363.

Lee, H., Surdeanu, M., MacCartney, B., & Jurafsky, D. (2014). *On the importance of text analysis for stock price prediction. 9th language resources and evaluation conference (LREC)* 1170–1175.

Lemaire, B., Denhiere, G., Bellissens, C., & Jhean-Larose, S. (2006). A computational model for simulating text comprehension. *Behavior Research Methods, 38*(4), 628–637.

Loughran, T., & McDonald, B. (2011). When is a liability not a liability? textual analysis, dictionaries, and 10-ks. *Journal of Finance, 66*, 67–97.

Manek, A. S., Shenoy, P. D., Mohan, M. C., & Venugopal, K. R. (2017). Aspect term extraction for sentiment analysis in large movie reviews using gini index feature selection method and svm classifier. *World Wide Web, 20*, 135–154.
Martineau, J., & Finin, T. (2009). *Delta TFIDF: An improved feature space for sentiment analysis. ICWSM* 258–261.
Medhat, W., Hassan, A., & Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal, 5*(4), 1093–1113.
Melville, P., Gryc, W., & Lawrence, R. D. (2009). *Sentiment analysis of blogs by combining lexical knowledge with text classification. KDD* 1275–1284.
Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). *Distributed representations of words and phrases and their compositionality. NIPS* 3111–3119.
Mohammad, S. M., Kiritchenko, S., & Zhu, X. (2013). *NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets. The seventh international workshop on semantic evaluation* 321–327.
Mohammad, S. M., Zhu, X., Kiritchenko, S., & Martin, J. (2015). Sentiment, emotion, purpose, and style in electoral tweets. *Information Processing & Management, 51*(4), 480–499.
Moore, A., Rayson, P., & Young, S. (2016). *Domain adaptation using stock market prices to refine sentiment dictionaries. Proceedings of ESA, LREC workshop* 63–66.
Ofek, N., Poria, S., Rokach, L., Cambria, E., Hussain, A., & Shabtai, A. (2016). Unsupervised commonsense knowledge enrichment for domain-specific sentiment analysis. *Cognitive Computation, 8*(3), 467–477.
Pang, B., & Lee, L. (2005). *Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. ACL* 115–124.
Pinker, S. (2004). Clarifying the logical problem of language acquisition. *Journal of Child Language, 31*(4), 949–953.
Schneider, A., & Dragut, E. (2015). *Towards debugging sentiment lexicons. ACL* 1024–1034.
Shalev-Shwartz, S., & Srebro, N. (2008). *SVM optimization: Inverse dependence on training set size. Proceedings of the twenty-fifth international conference on machine learning (ICML)* 928–935.
Shi, W., & Yu, Z. (2018). *Sentiment adaptive end-to-end dialog systems. Proceedings of the 56th annual meeting of the association for computational linguistics (ACL)* 1509–1519.
Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., & Potts, C. (2013). *Recursive deep models for semantic compositionality over a sentiment treebank. EMNLP* 1631–1642.
Taboada, M., Brooke, J., Tofiloski, M., Voll, K. D., & Stede, M. (2011). Lexicon-based methods for sentiment analysis. *Computational Linguistics, 37*(2), 267–307.
Tai, Y., & Kao, H. (2013). *Automatic domain-specific sentiment lexicon generation with label propagation. The 15th international conference on information integration and web-based applications & services (IIWAS)* 53–62.
Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T., & Qin, B. (2014). *Learning sentiment-specific word embedding for twitter sentiment classification. ACL* 1555–1565.
Turney, P. D. (2002). *Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. ACL* 417–424.
Velikovich, L., Blair-Goldensohn, S., Hannan, K., & McDonald, R. (2010). *The viability of web-derived polarity lexicons. HLT-NAACL* 777–785.
Vicente, I. S., Agerri, R., & Rigau, G. (2014). *Simple, robust and (almost) unsupervised generation of polarity lexicons for multiple languages. EACL* 88–97.
Vo, D.-T., & Zhang, Y. (2016). *Don't count, predict! an automatic approach to learning sentiment lexicons for short text. ACL* 219–224.
Wang, S., & Manning, C. D. (2012). *Baselines and bigrams: Simple, good sentiment and topic classification. ACL* 90–94.
Wang, Y., Zhang, Y., & Liu, B. (2017). *Sentiment lexicon expansion based on neural pu learning, double dictionary lookup, and polarity association. EMNLP* 553–563.
Wu, F., & Huang, Y. (2016). *Sentiment domain adaptation with multiple sources. ACL* 301–310.
Wu, F., Huang, Y., & Yan, J. (2017). *Active sentiment domain adaptation. ACL* 1701–1711.
Xia, Y., Cambria, E., Hussain, A., & Zhao, H. (2015). Word polarity disambiguation using bayesian model and opinion-level features. *Cognitive Computation, 7*(3), 369–380.
Xing, F. Z., Cambria, E., Malandri, L., & Vercellis, C. (Cambria, Malandri, Vercellis, 2018a). *Discovering bayesian market views for intelligent asset allocation. ECML-PKDD*.
Xing, F. Z., Cambria, E., & Welsch, R. E. (Cambria, Welsch, 2018b). Intelligent bayesian asset allocation via market sentiment views. *IEEE Computational Intelligence Magazine, 13*(4), 25–34.
Xing, F. Z., Cambria, E., & Welsch, R. E. (Cambria, Welsch, 2018c). Natural language based financial forecasting: A survey. *Artificial Intelligence Review, 50*(1), 49–73.
Zhu, X., Guo, H., Mohammad, S., & Kiritchenko, S. (2014). *An empirical study on the effect of negation words on sentiment. ACL* 304–313.