Creating Domain-Specific Sentiment Lexicons via Text Mining

Kevin Labille
Department of Computer Science and Computer Engineering
University of Arkansas
Fayetteville, Arkansas 72701
kclabill@uark.edu

Susan Gauch
Department of Computer Science and Computer Engineering
University of Arkansas
Fayetteville, Arkansas 72701
sgauch@uark.edu

Sultan Alfarhood
Department of Computer Science and Computer Engineering
University of Arkansas
Fayetteville, Arkansas 72701
salfarho@uark.edu

ABSTRACT

Sentiment analysis aims to identify and categorize customer's opinion and judgments using either traditional supervised learning techniques or unsupervised approaches. Traditionally, Sentiment Analysis is performed using machine learning techniques such as a naive Bayes classification or support vector machines (SVM), or could make use of a sentiment lexicon, that is, a list of words that are mapped to a sentiment score. Our work focuses on generating a domain-specific lexicon using probabilities and information theoretic techniques. By employing text mining, we overcome the poor performance of transferred supervised machine learning techniques and remove the need to adapt an existing lexicon while maintaining accuracy. We show that text mining techniques performs as well as traditional approaches and we demonstrate that domain specific lexicons perform better than general lexicons in a sentiment analysis task. We further review and compare the generated lexicons.

CCS CONCEPTS

• **Information systems** → **Sentiment analysis**;

KEYWORDS

Lexicons, sentiment analysis, data mining, text mining, opinion mining

ACM Reference format:
Kevin Labille, Susan Gauch, and Sultan Alfarhood. 2017. Creating Domain-Specific Sentiment Lexicons via Text Mining. In *Proceedings of Workshop on Issues of Sentiment Discovery and Opinion Mining, Halifax, Canada, August 2017 (WISDOM'17),* 8 pages.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

In recent years, it has become commonplace to leave feedback or a review upon an item that one has purchased online. Similarly, it is quite common to leave feedback about a restaurant or a hotel that one has visited and this feedback may reflect our satisfaction or dissatisfaction. These reviews are critical for both the community and the entity being reviewed since it helps the former to make decisions and it helps the latter to improve its service. User opinions have also been found to be useful in business intelligence, government intelligence, health care, tourism, and online services [35]. Because they are used to effect changes and decisions, it is very important to accurately summarize and evaluate people's opinions.

The rapid growth of online services and the increasing number of online reviews allow researchers to study how individuals express opinions and to mine the collections of opinions to identify trend and consensus. Sentiment analysis aims to extract user content regarding product features from their reviews and to identify their sentiment orientation, that is, if the reviews are positive or negative. Yet, it is not always an easy task to tell whether a statement is a fact or an opinion [21, 22]. Sentiment analysis approaches can be divided into two categories: corpus-based approaches and lexicon-based approaches. Corpus-based approaches consists of building classifiers from labelled instances and is often described as a machine-learning approach also known as supervised classification. The lexicon-based approach can be viewed as an unsupervised learning approach that uses a dictionary, or lexicon, that is a list of word associated with a sentiment orientation (positive/negative) and a sentiment strength. Sentiment lexicons play a key role in the sentiment analysis task. If the lexicon incorrectly assigns sentiment strength or orientation to words, the accuracy of the resulting sentiment analysis will be negatively impacted.

There are several approaches to word polarity annotations. *Discrete polarity annotation* labels words with a discrete value among *positive*, *negative*, or *neutral*. Such a polarity annotation is used in the MPQA Subjectivity lexicon [44]. *Continuous polarity annotation* assigns words a decimal value within a range (typically +1.0 to -1.0) that reflects the strength and the orientation of a word. Another polarity annotation is through *emotional sentiment*, in this case each word is assigned a discrete value among a list of predefined emotion such as {*joy, anger, sadness, disgust, surprise, fear etc...* }. The Word-Emotion Association Lexicon from Saif Mohammad [31, 32] is an example of this polarity annotation. One common polarity annotation is called fractional polarity annotation that is defined as a 3-tuple of positive numbers that sums up to 1, where each value corresponds to the positivity, negativity and neutrality respectively of the word. The popular SentiWordNet lexicon uses this type of polarity lexicon [2, 9] as is [12].

One of the advantages of using a lexicon approach is that the lexicon can be built from a large corpus and then used in other applications where there may not be enough information to do corpus-based approaches. Additionally, lexicons are widely used for cross-language sentiment classification of documents [4, 42] wherein the goal is to perform sentiment classification of document in multiple

WISDOM'17, August 2017, Halifax, Canada

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

languages [27]. Another useful application of lexicons is for cross-domain sentiment classification. This latter application has received less attention but still draws interest. Sentiment analysis is sensitive to the domain in which it is applied, that is because words could carry a different sentiment or meaning in different domains. Since a word's strength and sentiment is dependent upon the context in which it is used, it is unlikely that a word will have a single score across multiple domains. Most of the current approaches study the adaptation or sentiment transfer learning of a trained classifier (supervised techniques) or lexicon (unsupervised techniques) from one domain to another which involves having a general lexicon to start with, but very few works actually focus on techniques that build specific domain lexicons without requiring *a-priori* knowledge. Whilst Supervised sentiment classifier performs very well for the domain in which they were trained for, they usually perform very poorly when adapted or transfered to another domain [40].

Our work focuses on domain-specific lexicons. Specifically, we show how to build a domain-specific sentiment lexicon without *a-priori* knowledge, that is, without having to build a general lexicon. This removes the need to perform lexicon-adaptation and overcomes some performances issues that can arise when using a transferred supervised classifier. In our approach, domain-specific sentiment scores are calculated using probabilities and text mining techniques as introduced by [24]. In [27], the author suggests that finding domain-specific sentiment words is useful but insufficient in practice. Our intuition is contrary to that belief and we demonstrate that domain-specific lexicons are more accurate than generic lexicons when used for sentiment analysis. We compare the performance of two generic lexicons with several domain-specific lexicons that we build automatically and show that domain-specific lexicons perform better in the sentiment analysis task. In addition, we analyze and compare the structure and content of our various domain-specific lexicons.

The rest of the paper is organized as follow: In Section 2, we present various existing works on sentiment analysis and domain-specific lexicon generation. Section 3 describes how we generate domain-specific lexicons. In Section 4 we present our experimental evaluation and we discuss the results obtained. Finally, Section 5 summarizes our findings and highlights future work and improvements.

2 RELATED WORK

Sentiment analysis has become a field of great interest in recent years for computer scientists. We typically break sentiment analysis in two distinct tasks: opinion summarizing and opinion mining. The former aims to identify and extract product features from product reviews to summarize them [15] whereas the latter consists of analysing product reviews so as to determine the sentiment i.e., is the review reflecting a positive or negative sentiment [23, 26]. Sentiment analysis can be achieved using either supervised learning techniques or unsupervised learning techniques. Supervised learning techniques typically employ a naive Bayes classifier or use a Support Vector Machine (SVM) that is trained on a particular dataset. [10, 25, 28, 33, 34, 36, 46]. This approach generally performs well on the domain for which it is trained. Unsupervised learning technique are typically achieved by the means of a lexicon, that is, a list of words associated with a sentiment orientation and sentiment strength. The sentiment orientation of a review is calculated from the sentiment strength of each word found in that review [1, 8, 16, 20, 39, 41]

Sentiment Analysis can be performed at different levels, i.e., at the document level, sentence level, and aspect/feature level. Most of the cited work so far is done on the document level. Sentiment Analysis applied on the sentence level aims at evaluating the sentiment of a single sentence rather than the entire document. Yu and Hatzivassiloglou [45] used three unsupervised statistical techniques to identify the polarity of a sentence while Davidov et al [7] used supervised learning on text, hashtags and smileys to study the classification of tweets. Sentiment Analysis performed on the feature level aims to evaluate the sentiment of a particular feature from a review rather than evaluating the sentiment of the review. To that extent, Ding et al. employed a sentiment lexicon in their approach [8] whereas Wei and Gulla [43] modelled the problem as a hierarchical classification problem and utilized a Sentiment Ontology Tree.

Lexicon-based approaches are suitable at every level of Sentiment Analysis, it is therefore important to accurately capture the sentiment of each word in the lexicon. Sentiment lexicons can be generated (1) manually; (2) using a dictionary; or (3) using a corpus of documents. The second approach to generating sentiment lexicons uses a few seed words for which the sentiment orientation is already known. The list is then expanded by searching for the synonyms and antonyms of the seed words into a dictionary [18, 31, 37]. The corpus-based techniques have a similar approach but they use a domain corpus rather than a dictionary. Another corpus-based approach consists of adapting a general sentiment lexicon to a domain-specific one by using a domain corpus as well [5, 13, 19].

Sentiment classification is dependent upon the context in which it is used. It has been shown that a sentiment classifier used in one particular domain will perform poorly in another domain. Cross-domain sentiment analysis aims to study this problem. Cross-domain sentiment analysis is traditionally performed via domain-adaptation or transfer learning, i.e., adapting an existing classifier trained on a source domain to a target domain. Although adapting a supervised sentiment classifier often results in poor performances, [40] tackled this domain-transfer problem using an Adapted Naive Bayes classifier, a weighted transfer version of Naive Bayes Classifier. [11] used Probabilistic Latent Analysis in order to identify a common semantic space, i.e., common topics between the source domain and the target domain. [14] focused on automatically finding polarity-bearing topics from text. Their approach uses a joint sentiment-topic model along with a list of domain-independent sentiment word. Similarly, [6] focused on generating context-driven features (or clues). They proposed a boostrapping method that uses a small set of seed clues from different domain to generate new clues for a target domain. [3] use a different approach than the previously cited works. The authors are able to automatically generate a sentiment sensitive thesaurus from multiple source domains to a target a domain with no labeled data.

The works from [17] and [5] are the closest to ours. In the former, the authors are generating domain-specific sentiment lexicon through lexicon-adaptation. They use a bootstrapping method to generate a domain-specific sentiment lexicon from a generic lexicon.

In the latter paper, the authors use integer linear programming to adapt an existing lexicon into a new one. They consider the relations among words to derive the most accurate polarity of each lexical item.

Our approach differs from the aforementioned works by several ways. (1) Our method does not use lexicon-adaptation to generate a domain-specific lexicon. Rather, we use text mining to generate a lexicon without any *a-priori* knowledge on the domain for which the lexicon is built. (2) We use probabilities and information theoretic techniques to calculate the sentiment strength of each word.

3 ESTIMATING WORDS SCORE

We estimate the sentiment score for a word w by combining a probabilistic score $Score_{prob}(w)$ and an information theoretic score $Score_{it}(w)$.

3.1 Probabilistic Score

The probabilistic score $Score_{prob}(w)$ of a word w is computed using posterior probabilities and is defined as the difference of the probability of w of being positive, $p(pos|w)$, and its probability of being negative, $p(neg|w)$, as follows:

$$Score_{prob}(w) = p(pos|w) - p(neg|w) \tag{E1}$$

where:

$$p(pos|w) = \frac{p(pos) \times p(w|pos)}{p(w)}$$

$$p(neg|w) = \frac{p(neg) \times p(w|neg)}{p(w)}$$

and:

$$p(w|pos) = \frac{\frac{\sum_{r \in R_{pos}} n_{wr}}{\sum_{w'} \sum_{r \in R_{pos}} n_{w'r}} + 1}{k_{dic} + 1}$$

$$p(w|neg) = \frac{\frac{\sum_{r \in R_{neg}} n_{wr}}{\sum_{w'} \sum_{r \in R_{neg}} n_{w'r}} + 1}{k_{dic} + 1}$$

$$\sum_{r \in R_{pos}} n_{wr} = \gamma\, n_{w5^*} + n_{w4^*}$$

$$\sum_{r \in R_{neg}} n_{wr} = \gamma\, n_{w1^*} + n_{w2^*}$$

$p(pos)$ is the prior probability of the positive class, i.e., the proportion of words that belongs to the positive class; $p(neg)$ is the proportion of words that belongs to the negative class; and $p(w)$ is the total number of occurrences of w; $p(w|pos)$ is the probability to observe a word w given the positive class; and $p(w|neg)$ is the probability to observe w given the negative class. $\sum_{r \in R_{pos}} n_{wr}$ is the number of times word w appears in the positive class (i.e., the number of times it appears in each positive review r in corpus R); $\sum_{r \in R_{neg}} n_{wr}$ is the number of times w appears in the negative class; $\sum_{w'} \sum_{r \in R_{pos}} n_{w'r}$ is the number of occurrences of every word in the positive class; and $\sum_{w'} \sum_{r \in D_{neg}} n_{w'r}$ the number of occurrences of every word in the negative class. k_{dic} is the size of the dictionary. γ is a weight factor as described by [24].
This yields scores in the range from -1 to 1, with -1 indicating that the word is entirely negative, +1 that a word is entirely positive, and 0 indicating a neutral word.

3.2 Information Theoretic Score

The information theoretic score, $score_{it}(w)$, is based on the well-known information-theory based measure called TF-IDF (Term Frequency-Inverse Document Frequency) [38], that evaluates how important a word is in a document. Like in $E1$, the score is the difference between w's positive score and w's negative score times its inverse document frequency, and is defined as follows:

$$Score_{it}(w) = \Big(pos(w) - neg(w)\Big) \times IDF(w)$$
$$where: \tag{E2}$$
$$IDF(w) = \log \frac{N}{df_w}$$

and:

$$\begin{cases} pos(w) & = \gamma\ brtf(w_{5^*}) + brtf_c(w_{4^*}) \\ neg(w) & = \gamma\ brtf(w_{1^*}) + brtf_c(w_{2^*}) \end{cases}$$

$$brtf(w_c) = \frac{rtf_{wr}}{N_{neg}} \times N$$

$$brtf(w_c) = \frac{rtf_{wr}}{N_{pos}} \times N$$

$brtf$ stands for *balanced relative term frequency* and rtf_{wr} is the relative term frequency of word w as introduced by [24]; N_{neg} is the total number of negative review; N_{pos} is the total number of positive reviews; N is the total number of reviews. For example, $brtf(w_{1^*})$ $(c = 1^*)$ is the balanced relative term frequency of w in the 1-star reviews. γ is the same weight factor as in $E1$.

3.3 Hybrid Approach

Because the probabilistic score is related to the global frequency of each word, the resulting score reflects the word's sentiment at the corpus level. On the other hand, since the information theoretic score incorporates the relative term frequency of the word within documents and the distribution of the word across documents, it reflects the word's sentiment at the document level. We can benefit from both methods by averaging $Score_{prob}(w)$ and $Score_{it}(w)$ as follow:

$$Score(w) = \frac{Score_{prob}(w) + Score_{it}(w)}{2}$$

In previous work [24], this hybrid measure was shown to be more accurate than either the probabilistic or the information theoretic

approaches on their own. We compared several methods to merge both scores and report the results with averaging since that worked best. Additionally, we compared the performances using raw score, normalized score using feature scaling, and standard score (i.e., Z-score). Since raw scores performed best, we will report results using the raw score calculated as above.

3.4 Building a Domain-Specific Lexicon

In order to build a domain-specific lexicon, we begin with a collection of reviews for a specific domain. These are preprocessed to remove stop words and punctuation marks. Based on their number of occurrence, the resulting words are filtered to remove misspelled words. Then, each word is assigned a score based on the hybrid formula using γ equal to four.

We build our domain-specific lexicons by using Amazon product reviews [29, 30] for 15 different categories submitted from January 2013 through July 2014. Reviews are rated from 1 star to 5 stars. For our experiments, we consider reviews rated 1-star and 2-star to be negative whereas 4-star and 5-star reviews are considered positive. 3-star reviews are considered neutral and are therefore ignored throughout the experiments.

Table 1 shows the proportion of positive reviews and negative reviews for each of the 15 domains used.

Table 1: Proportion of positive and negative reviews in each domain

Domain	#Positive	#Negative
Automotive	1,083,639	186,272
Baby	699,255	133,260
Beauty	1,556,461	296,818
Books	18,489,343	2,095,422
CDs and Vinyl	3,191,727	292,452
Cellphones	2,341,166	754,761
Clothing and Shoes	4,424,033	750,290
Electronics	5,833,322	1,358,087
Health and care	2,276,578	464,115
Home and kitchen	3,248,403	660,429
Movies and TV	3,618,913	572,765
Office products	913,616	230,434
Sport and outdoor	2,573,342	417,153
Toys and games	1,750,036	308,794
Video games	970,030	230,353

Table 2 shows the dimension, i.e., the number of words they contain, as well as the proportion of positive and negative words, for each lexicon.

4 EXPERIMENT

We conducted a controlled experiment to compare domain-specific lexicons to two generic lexicons, measuring the accuracy when each was used to identify the sentiment on unlabeled reviews.

Table 2: Dimension of each lexicon

Domain	#words	%pos	%neg
Automotive	29,368	62%	38%
Baby	23,733	59%	41%
Beauty	34,449	61%	39%
Books	283,791	69%	31%
CDs and Vinyl	128,440	74%	26%
Cellphones	37,271	58%	42%
Clothing and Shoes	51,547	66%	34%
Electronics	80,092	62%	38%
Health and care	48,979	63%	37%
Home and kitchen	51,026	62%	38%
Movies and TV	119,726	70%	30%
Office products	29,445	32%	38%
Sport and outdoor	51,495	62%	38%
Toys and games	43,869	65%	35%
Video games	48,202	64%	36%

4.1 Experimental Setup

Each of the 15 datasets are randomly split into two subsets, using 80% for building the domain-specific lexicons (training) and 20% for testing the accuracy of the sentiment analysis using the domain-specific lexicons. We compare our results against two baseline generic lexicons and report our findings in Section 4.2. Our first baseline is built from the free lexical resource SentiWordNet [2]. SentiWordNet uses the fractional polarity annotation while our sentiment lexicons use continuous polarity annotation. To account for that, we use Petter Tonberg's sentiment value approximation[1] to turn SentiWordNet's fractional scores into continuous scores, ignoring part-of-speech. Our second baseline is the generic lexicon from [24] that was built using 8,903,505 Amazon product reviews combined from the 15 domains mentioned in Table 1.

We evaluate our domain-specific lexicons using sentiment analysis on each test dataset, and we compare them against both baselines. Our sentiment analysis method computes the score of a review by summing up each word score in the review from its domain-specific lexicon and by normalizing for length. If the resulting score is positive, then the review is deemed to be positive whereas if the resulting score is negative, the review is deemed to be negative.

4.2 Experimental Results

We evaluate our domain-specific lexicons versus the two baselines and report our results in Table 3. Table 3 shows the recall, precision, F1-Score, and accuracy averaged across all 15 domains. As shown

Table 3: Evaluation across all domain (average)

	Recall	Prec.	F1-Score	Acc.
sentiwordnet	0.85	0.90	0.87	80.01%
generic lex.	0.88	**0.95**	0.91	86.84%
domain spec.	**0.93**	0.94	**0.94**	**90.09%**

[1] http://sentiwordnet.isti.cnr.it/code/SentiWordNetDemoCode.java

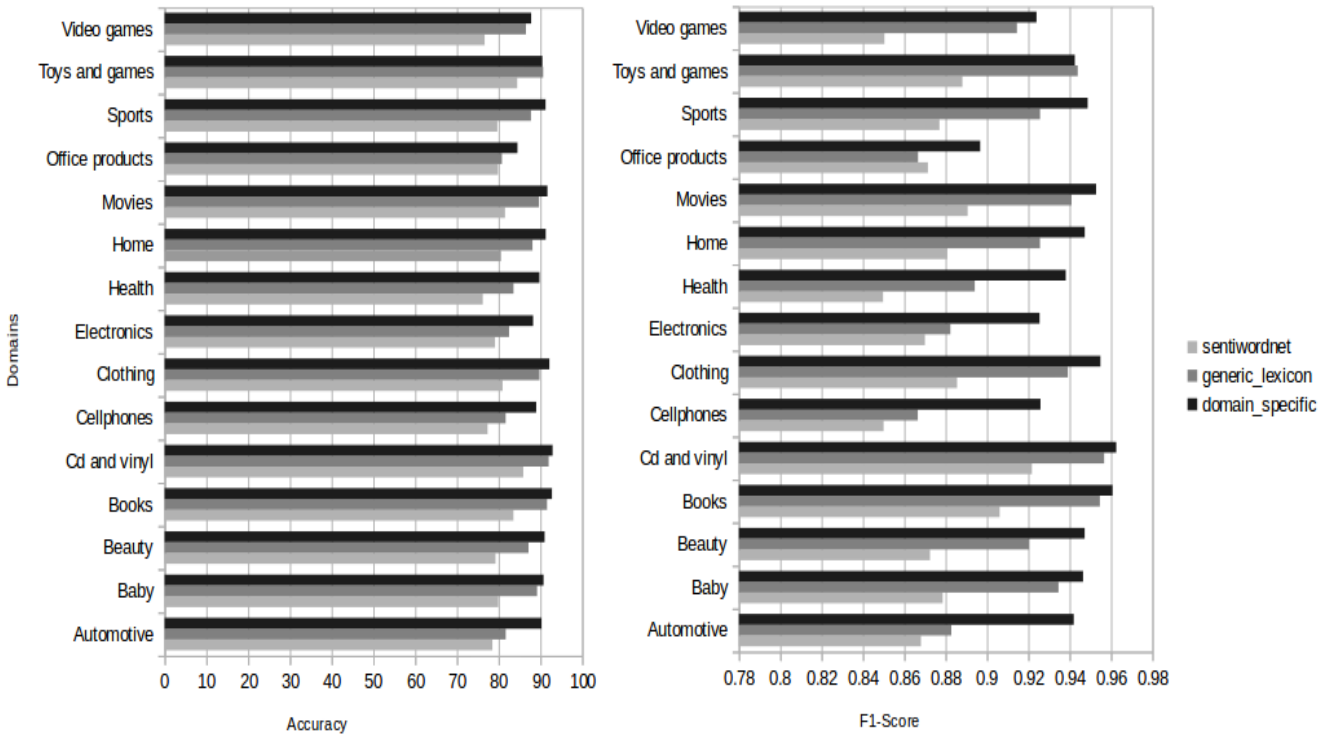

Figure 1: Accuracy and F1-Score of our approach on all domains.

in the table, domain-specific lexicons are more accurate than both generic lexicons. Our domain-specific lexicons achieve an accuracy of 90.09% on average, which is an improvement of 3.25% over the generic lexicon that achieves 86.84%, and it is an improvement of 10.08% over the SentiWordNet lexicon that achieves 80.23% accuracy. This validates our intuition that some words are associated with different sentiments and sentiment strengths depending on the domain in which they are used.

We additionally report the averaged recall, precision, and F1-Score and note that our domain-specific lexicons outperform the generic lexicons. Figure 1 depicts the F1-Score and accuracy with the generic lexicons and each domain-specific lexicon used on its respective domain. As shown in Figure 1, our domain-specific lexicons outperform both generic lexicons in all 15 domains. Our best domain-specific lexicon reaches 92.95% accuracy in the domain *Cds and vinyls* against a highest accuracy of 84.45% (in the domain *Toys and games*) for the SentiWordNet lexicon. Conversely, our lowest domain-specific accuracy is achieved in the field of *Office products* with an accuracy of 84.79% versus a lowest accuracy of 76.21% achieved by the SentiWordNet lexicon in *Health and care*.

These results suggest that estimating sentiment through text-mining techniques performs as good as traditional approaches and that domain-specific lexicons are more accurate than generic lexicons.

4.3 Discussion

We examine the content of each domain-specific lexicon to support our assumption that the sentiment of some words depends upon the context or the domain. While some words are "stable", i.e. their score barely fluctuates across domains, other words are perceived very differently from one domain to another. By computing the standard deviation σ of each word across all 15 domains, we are able to identify these words. Table 4 highlights a few stable words and a few variable words along with their minimum and maximum scores across all 15 domains.

As shown in the table, some words such as *misguidance, misconduct,* and *invalids* have a very low standard deviation and are consistently fully positive or fully negative, suggesting that they have the same meaning or perception regardless of the context in which they are used. Conversely, the sentiment of some words change considerably from one domain to another. For instance, *broke* is perceived as a very negative word in the *Cellphones* domain while it is positive in *Books*. This could be explained by the fact that *broke* has several meaning such as (1) the product is not working properly or (2) someone is in the state of having very little money.

Table 4: Sample of stable words and variable words

	Min score	Max score	Min Lex	Max Lex	Generic score
Stable words					
invalids	-0.001	-0.000	Movies	Books	0.039
misguidance	-0.053	-0.052	Books	Movies	-0.066
misconduct	-0.018	-0.015	Books	Cd and vinyls	-0.045
trafficker	0.017	0.017	Movies	Books	0.044
Variable words					
work	-0.499	0.100	Video games	Cd and vinyls	-0.195
violations	-0.455	0.007	Health and care	Sport and outdoor	-0.028
broke	-0.552	0.018	Cellphones	Books	-0.376
flammable	-0.36	0.02	Beauty	Video games	-0.194

To give a better feeling to the reader, we illustrate this with a negative review from the *Cellphones* domain which contains the word *broke* followed by a positive review from the *Books* domain which also contains the word *broke*: One-star review from the *Cellphones* domain:

> *This item broke right away plus it never click-in so it always comes off. See other reviews and pictures from other users and that's exactly what it will happen it other customers buy this product.*

Five-star review from the *Books* domain:

> *After reading this book, I was hungry for more Umrigar; she's an amazing storyteller! This novel was a compelling read. I've recommended it to friends and family who now concur that it's a great book. I loved the relationship between Bhima and Sera, and my heart broke for Bhima as she tried tirelessly to help her pregnant granddaughter Maya. An overall great read!*

Also, we should notice that the generic score of the variable words often falls in between the minimum and maximum domain-specific score, which reinforces our assumption that generic lexicons cannot capture the variation in sentiment for some words.

We also notice that few words are domain-specific, meaning that they can only be found in a certain domain. We report such words in Table 5. For example The word *nauseousness* was only used in

Table 5: Examples of domain-specific words

	Score	Domain
punitively	-0.0178	Books
chromate	-0.018	Automotive
destroyable	0.0223	Video games
nauseousness	0.013	Health and care
unpartitioned	-0.07	Electronics

the *Health and care* domain, which is unsurprising since it is likely to only occur when discussing illness or medications. Likewise, the word *unpartitioned* is only used when talking about hard disk drives and is therefore specific to the *Electronics* domain.

We further evaluate the performances of each domain-specific lexicon against each domain and report the accuracy, average accuracy i.e., the average accuracy of this lexicon across all domain, the rank (rank of the lexicon relative to its domain; e.g., if the rank is 2 it means that the lexicon is the second most accurate lexicon in its home domain), and average rank of each domain-specific lexicon in Table 5. From Table 6 we can see that most of the domain-specific

Table 6: Evaluation domain against domain

Lexicon	Rank	Avg rank	Acc %	Avg Acc %
Automotive	1	8	89.75	87.11
Baby	1	4	90.71	88.10
Beauty	1	10	91.01	86.36
Books	1	8	92.67	87.44
CD and vinyl	2	8	92.95	87.54
Cellphones	1	10	88.63	85.65
Clothing	1	2	92.18	89.30
Electronics	2	8	88.07	86.31
Health and care	1	3	89.46	88.15
Home	1	4	91.30	88.40
Movies	1	8	91.61	87.27
Office products	8	6	84.79	78.8
Sports and outdoors	1	4	91.07	88.65
Toys and games	12	16	89.4	80.89
Video games	2	10	87.83	84.98
sentiwordnet	NA	14	NA	80.01
generic lexicon	NA	8	NA	86.84

lexicons perform best in their own domain (rank = 1) except the *Toys and game* lexicon that ranked 12 in its own domain, meaning that most of the other lexicon are more accurate. The average rank of a lexicon is an indicator of how specific its vocabulary is. The higher the average rank, the less specific the vocabulary.

Indeed, we can notice that the *clothing* lexicon has an average rank of 2 and an average accuracy of 89.30%, that indicates that this lexicon is suitable to use in almost any domain. Conversely, the *video games* lexicon has an average rank of 10 but a relative rank of 2, that indicates that it performs very well in its own domain but poorly in other domains.

This could be explained by the fact that people tend to use fringe

words in the clothing domain (i.e. words such as good, bad, beautiful) whereas we tend to use specific words in the video games domain.

In table 6, the lexicon Toys-and-games has an accuracy in its home domain of 89.4% and an average accuracy of 80.89% across domain. The lexicon that performed best in the Toys-and-games domain is the Clothing lexicon with an accuracy of 91.59%, the second best is Baby with an accuracy of 91.49%. We believe this is due to the lexicon itself. The content of the Toys-and-games lexicon is much different from that of the other ones. Indeed, the average word weight in this lexicon is of 0.001647 while it is 0.05183 on average for the remaining 14 categories. The average positive score is 0.00144 in this lexicon versus 0.045 on average in the other 14 categories. Likewise, the average negative weight is -0.0020 versus -0.06 on average in the remaining 14 categories. The clustering of the word weights near 0.0 might negatively impact the classification using this lexicon, and therefore lower the accuracy. The lower variance of the word scores might be explained by the nature of the reviews in this particular domain.

Finally, both baselines have a high average rank across domains (14 for SentiWordNet and 8 for the generic lexicon versus 2 on average for the domain-specific lexicons within their appropriate domain) which support our hypothesis that generic lexicon are less suitable and accurate than domain-specific lexicons. The sentiwordnet lexicon performs poorly across specific domains with an average accuracy of 80.50% versus 86.84% for the text mining based generic lexicon and 86.93% for our domain-specific lexicons. This supports our hypothesis that domain-specific lexicons can outperform generic lexicons.

We explored the coverage of our domain-specific lexicons and compared it to our baselines. The *coverage* is the proportion of vocabulary covered by the lexicon. It is a good indicator of the specificity of the vocabulary within a domain. If a lexicon has a higher coverage but a lower accuracy than another lexicon, it could mean that it is missing some important (or strong) words from that domain. Table 7 sums up the coverage of each domain-lexicon as well as the baselines.

As Table 7 shows, our domain-specific lexicon achieves a high coverage across all domains. The sentiwordnet lexicon has the lowest average coverage (82.18%) and has an average coverage-rank of 17, meaning that this generic lexicon is not only missing many vocabulary words but the words that it covers are not highly indicative of sentiment. Conversely, the generic lexicon covers as many words as each domain-specific lexicon (average coverage-rank of 1 and average coverage of 96.54%). This is due to the fact that this lexicon was built from the same set of product reviews. However, as we showed earlier, the generic lexicon is not as accurate as the domain-specific lexicon, indicating that although the coverage is equivalent, the sentiment scores calculated ignoring domain boundaries are less accurate.

Table 7: Coverage of each lexicons

Lexicon	Domain Coverage	Avg coverage	Avg coverage rank
Automotive	96.57%	93.52%	13
Baby	96.31%	92.93%	14
Beauty	96.49%	93.89%	12
Books	96.74%	96.42%	2
CD and vinyl	95.51%	95.59%	8
Cellphones	96.27%	94.07%	11
Clothing	96.82%	95.09%	7
Electronics	96.64%	95.55%	5
Health and care	96.6%	95%	8
Home	97.03%	95.12%	7
Movies	96.37%	95.83%	5
Office products	96.61%	93.94%	12
Sports and outdoors	96.74%	95.23%	6
Toys and games	96.57%	95.05%	8
Video games	96.03%	94.75%	9
sentiwordnet	NA	82.18%	17
generic lexicon	NA	96.54%	1

5 CONCLUSIONS AND FUTURE WORK

In this paper, we introduce a method for generating a domain-specific lexicon based on a combination of probabilistic and information theoretic weights. Our work differs from the traditional approaches by creating the domain-specific lexicons without *a-priori* knowledge, that is, without having to perform lexicon-adaptation from a generic lexicon. It also overcomes some performances issues that can arise when using a transferred supervised classifier.

We assess the effectiveness of several domain-specific lexicons against two baseline generic lexicons by calculating their accuracy and F1-Score. Our domain-specific lexicons outperform both generic lexicons with an average accuracy of 90.09% in their appropriate domain versus 80.23% for the generic SentiWordNet lexicon and 86.84% for the generic information theoretic-based lexicon. Likewise, our domain-specific lexicons average an F1-Score of 0.94 against 0.87 and 0.91 for both generic lexicons.

We show that text mining techniques perform as well as traditional approaches for generating sentiment scores. Our experiment results indicate that domain-specific lexicons are more accurate than generic lexicons in the sentiment analysis task. Furthermore, our results show that domain-specific sentiment scores are more indicative of sentiment than generic sentiment scores.

Future work includes the investigation of sentiment rating prediction rather than sentiment analysis, i.e., predicting the rating of a review instead of identifying whether it is positive or negative. In addition, we will experiment the use of deep learning and word embedding for the sentiment lexicon creation.

REFERENCES

[1] Nawaf A Abdulla, Nizar A Ahmed, Mohammed A Shehab, Mahmoud Al-Ayyoub, Mohammed N Al-Kabi, and Saleh Al-rifai. 2014. Towards improving the lexicon-based approach for arabic sentiment analysis. *International Journal of Information Technology and Web Engineering (IJITWE)* 9, 3 (2014), 55–71.

[2] Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining.. In *LREC*, Vol. 10. 2200–2204.

[3] Danushka Bollegala, David Weir, and John Carroll. 2011. Using multiple sources to construct a sentiment sensitive thesaurus for cross-domain sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 132–141.

[4] Julian Brooke, Milan Tofiloski, and Maite Taboada. 2009. Cross-Linguistic Sentiment Analysis: From English to Spanish.. In *RANLP*. 50–54.

[5] Yejin Choi and Claire Cardie. 2009. Adapting a polarity lexicon using integer linear programming for domain-specific sentiment classification. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*. Association for Computational Linguistics, 590–598.

[6] Yoonjung Choi, Youngho Kim, and Sung-Hyon Myaeng. 2009. Domain-specific sentiment analysis using contextual feature generation. In *Proceedings of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*. ACM, 37–44.

[7] Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010. Enhanced sentiment learning using twitter hashtags and smileys. In *Proceedings of the 23rd international conference on computational linguistics: posters*. Association for Computational Linguistics, 241–249.

[8] Xiaowen Ding, Bing Liu, and Philip S Yu. 2008. A holistic lexicon-based approach to opinion mining. In *Proceedings of the 2008 international conference on web search and data mining*. ACM, 231–240.

[9] Andrea Esuli and Fabrizio Sebastiani. 2006. Sentiwordnet: A publicly available lexical resource for opinion mining. In *Proceedings of LREC*, Vol. 6. Citeseer, 417–422.

[10] Dehong Gao, Furu Wei, Wenjie Li, Xiaohua Liu, and Ming Zhou. 2015. Cross-lingual sentiment lexicon learning with bilingual word graph label propagation. *Computational Linguistics* (2015).

[11] Sheng Gao and Haizhou Li. 2011. A cross-domain adaptation method for sentiment classification using probabilistic latent analysis. In *Proceedings of the 20th ACM international conference on Information and knowledge management*. ACM, 1047–1052.

[12] Ahmed Hassan and Dragomir Radev. 2010. Identifying text polarity using random walks. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 395–403.

[13] Vasileios Hatzivassiloglou and Kathleen R McKeown. 1997. Predicting the semantic orientation of adjectives. In *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 174–181.

[14] Yulan He, Chenghua Lin, and Harith Alani. 2011. Automatically extracting polarity-bearing topics for cross-domain sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 123–131.

[15] Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 168–177.

[16] Minqing Hu and Bing Liu. 2004. Mining opinion features in customer reviews. In *AAAI*, Vol. 4. 755–760.

[17] Valentin Jijkoun, Maarten de Rijke, and Wouter Weerkamp. 2010. Generating focused topic-specific sentiment lexicons. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 585–594.

[18] Jaap Kamps, MJ Marx, Robert J Mokken, M de Rijke, and others. 2004. Using wordnet to measure semantic orientations of adjectives. (2004).

[19] Hiroshi Kanayama and Tetsuya Nasukawa. 2006. Fully automatic lexicon expansion for domain-oriented sentiment analysis. In *Proceedings of the 2006 conference on empirical methods in natural language processing*. Association for Computational Linguistics, 355–363.

[20] Aamera ZH Khan, Mohammad Atique, and VM Thakare. 2015. Combining lexicon-based and learning-based methods for Twitter sentiment analysis. *International Journal of Electronics, Communication and Soft Computing Science & Engineering (IJECSCSE)* (2015), 89.

[21] Soo-Min Kim and Eduard Hovy. 2004. Determining the sentiment of opinions. In *Proceedings of the 20th international conference on Computational Linguistics*. Association for Computational Linguistics, 1367.

[22] Soo-Min Kim and Eduard Hovy. 2006. Extracting opinions, opinion holders, and topics expressed in online news media text. In *Proceedings of the Workshop on Sentiment and Subjectivity in Text*. Association for Computational Linguistics, 1–8.

[23] Soo-Min Kim and Eduard Hovy. 2006. Identifying and analyzing judgment opinions. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*. Association for Computational Linguistics, 200–207.

[24] Kevin Labille, Sultan Alfarhood, and Susan Gauch. 2016. Estimating Sentiment via Probability and Information Theory. In *Proceedings of the 8th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management - Volume 1: KDIR,*. 121–129. DOI: https://doi.org/10.5220/0006072101210129

[25] Tao Li, Yi Zhang, and Vikas Sindhwani. 2009. A non-negative matrix tri-factorization approach to sentiment classification with lexical prior knowledge. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*. Association for Computational Linguistics, 244–252.

[26] Bing Liu. 2010. Sentiment Analysis and Subjectivity. *Handbook of natural language processing* 2 (2010), 627–666.

[27] Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies* 5, 1 (2012), 1–167.

[28] Feifan Liu, Dong Wang, Bin Li, and Yang Liu. 2010. Improving blog polarity classification via topic analysis and adaptive methods. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 309–312.

[29] Julian McAuley, Rahul Pandey, and Jure Leskovec. 2015. Inferring networks of substitutable and complementary products. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 785–794.

[30] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton van den Hengel. 2015. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 43–52.

[31] Saif Mohammad, Cody Dunne, and Bonnie Dorr. 2009. Generating high-coverage semantic orientation lexicons from overtly marked words and a thesaurus. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*. Association for Computational Linguistics, 599–608.

[32] Saif M Mohammad and Peter D Turney. 2013. Crowdsourcing a word–emotion association lexicon. *Computational Intelligence* 29, 3 (2013), 436–465.

[33] Vincent Ng, Sajib Dasgupta, and SM Arifin. 2006. Examining the role of linguistic knowledge sources in the automatic identification and classification of reviews. In *Proceedings of the COLING/ACL on Main conference poster sessions*. Association for Computational Linguistics, 611–618.

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

[35] Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval* 2, 1-2 (2008), 1–135.

[36] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*. Association for Computational Linguistics, 79–86.

[37] Wei Peng and Dae Hoon Park. 2004. Generate adjective sentiment dictionary for social media sentiment analysis using constrained nonnegative matrix factorization. *Urbana* 51 (2004), 61801.

[38] Gerard Salton and Michael J McGill. 1986. Introduction to modern information retrieval. (1986).

[39] Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. 2011. Lexicon-based methods for sentiment analysis. *Computational linguistics* 37, 2 (2011), 267–307.

[40] Songbo Tan, Xueqi Cheng, Yuefen Wang, and Hongbo Xu. 2009. Adapting naive bayes to domain adaptation for sentiment analysis. In *European Conference on Information Retrieval*. Springer, 337–349.

[41] Peter D Turney. 2002. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 417–424.

[42] Xiaojun Wan. 2008. Using bilingual knowledge and ensemble techniques for unsupervised Chinese sentiment analysis. In *Proceedings of the conference on empirical methods in natural language processing*. Association for Computational Linguistics, 553–561.

[43] Wei Wei and Jon Atle Gulla. 2010. Sentiment learning on product reviews via sentiment ontology tree. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 404–413.

[44] Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Language resources and evaluation* 39, 2-3 (2005), 165–210.

[45] Hong Yu and Vasileios Hatzivassiloglou. 2003. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*. Association for Computational Linguistics, 129–136.

[46] Shusen Zhou, Qingcai Chen, and Xiaolong Wang. 2010. Active deep networks for semi-supervised sentiment classification. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*. Association for Computational Linguistics, 1515–1523.