Combining Sentiment Lexica with a Multi-View Variational Autoencoder

Hoyle, Alexander Miserlis; Wolf-sonkin, Lawrence; Wallach, Hanna; Cotterell, Ryan; Augenstein, Isabelle

Published in:

DOI:
10.18653/v1/N19-1065

Publication date:
2019

Document version
Publisher's PDF, also known as Version of record

Document license:
CC BY

Citation for published version (APA):
Combining Sentiment Lexica with a Multi-View Variational Autoencoder

Alexander Hoyle\textsuperscript{3}  Lawrence Wolf-Sonkin\textsuperscript{\dagger}  Hanna Wallach\textsuperscript{5}
Ryan Cotterell\textsuperscript{\ddagger}  Isabelle Augenstein\textsuperscript{2}
\textsuperscript{3}University College London, London, UK
\textsuperscript{\dagger}Department of Computer Science, Johns Hopkins University, Baltimore, USA
\textsuperscript{\ddagger}Department of Computer Science, University of Cambridge, Cambridge, UK
\textsuperscript{2}Department of Computer Science, University of Copenhagen, Copenhagen, Denmark
alexander.hoyle.17@ucl.ac.uk, lawrencews@jhu.edu
hanna@dirichlet.net, rdc42@cam.ac.uk, augenstein@di.ku.dk

Abstract

When assigning quantitative labels to a dataset, different methodologies may rely on different scales. In particular, when assigning polarities to words in a sentiment lexicon, annotators may use binary, categorical, or continuous labels. Naturally, it is of interest to unify these labels from disparate scales to both achieve maximal coverage over words and to create a single, more robust sentiment lexicon while retaining scale coherence. We introduce a generative model of sentiment lexica to combine disparate scales into a common latent representation. We realize this model with a novel multi-view variational autoencoder (VAE), called SentiVAE. We evaluate our approach via a downstream text classification task involving nine English-Language sentiment analysis datasets; our representation outperforms six individual sentiment lexica, as well as a straightforward combination thereof.

1 Introduction

Sentiment lexica provide an easy way to automatically label texts with polarity values, and are also frequently transformed into features for supervised models, including neural networks (Palogiannidi et al., 2016; Ma et al., 2018). Indeed, given their utility, a veritable cottage industry has emerged focusing on the design of sentiment lexica. In practice, using any single lexicon, unless specifically and carefully designed for the particular domain of interest, has several downsides. For example, any lexicon will typically have low coverage compared to the language’s entire vocabulary, and may have misspecified labels for the domain. In many cases, it may therefore be desirable to combine multiple sentiment lexica into a single representation. Indeed, some research on unifying such lexica has emerged (Emerson and Declerck, 2014; Altrabsheh et al., 2017), borrowing ideas from crowdsourcing (Raykar et al., 2010; Hovy et al., 2013). However, this is a non-trivial task, because lexica can use binary, categorical, or continuous scales to quantify polarity—in addition to different interpretations for each—and thus cannot easily be combined. In Fig. 1, we show an example of the same word labeled using different lexica to illustrate the nature of the challenge.

To combine sentiment lexica with disparate scales, we introduce SentiVAE, a novel multi-view variant of the variational autoencoder (VAE) (Kingma and Welling, 2014). SentiVAE, visualized as a graphical model in Fig. 2, differs from the original VAE in two ways: (i) it uses a Dirichlet latent variable (rather than a Gaussian) for each word in the combined vocabulary, and (ii) it has multiple emission distributions—one for each lexicon. Because the latent variables are shared across the lex-
We use the following commonly used English-language sentiment lexica: SentiWordNet (Baccianella et al., 2010), MPQA (Wilson et al., 2005), SenticNet 5 (Cambria et al., 2014), Hu-Liu (Hu and Cianella et al., 2010), MPQA (Wilson et al., 2005), SentiWordNet (Baccianella et al., 2010), a Bayesian approach for aligning sentiment lexica, as well as the straightforward combination thereof. We then train a classifier to predict the overall sentiment of each text. For each dataset, we transform each text into an average polarity value using either our representation, one of the six commonly used sentiment lexica, or a straightforward combination thereof. We then train a classifier to predict the overall sentiment of each text from its average polarity value. We find that our representation outperforms the individual lexica, as well as the straightforward combination for some datasets. Our representation is particularly efficacious for datasets from domains that are not well-supported by standard sentiment lexica.1

The existing research that is most closely related to our work is SentiMerge (Emerson and Declerck, 2014), a Bayesian approach for aligning sentiment lexica with different continuous scales. SentiMerge consists of two steps: (i) aligning the lexica via rescaling, and (ii) combining the rescaled lexica using a Gaussian distribution. The authors performed token-level evaluation using a single sentiment analysis dataset where each token is labeled with a polarity value on a nine-point scale (where the midpoint is neutral), yielding a 10-dimensional label. MPQA, Hu-Liu, and GI all use binary scales. Lastly, each word in SenticNet is labeled with a real value in the interval $[-1, 1]$, where 0 is neutral.

### 2 Sentiment Lexica and Scales

We use the following commonly used English-language sentiment lexica: SentiWordNet (Baccianella et al., 2010), MPQA (Wilson et al., 2005), SenticNet 5 (Cambria et al., 2014), Hu-Liu (Hu and Liu, 2004), GI (Stone et al., 1962), and VADER (Hutto and Gilbert, 2014). Descriptive statistics for each lexicon are shown in Tab. 1. Each word in SentiWordNet is labeled with two real values, each in the interval $[0, 1]$, corresponding to the strength of positive and negative sentiment (e.g., the label (0 0) is neutral, while the label (1 0) is maximally positive). Each word in VADER is labeled by ten different human evaluators, with each evaluator providing a polarity value on a nine-point scale (where the midpoint is neutral), yielding a 10-dimensional label. MPQA, Hu-Liu, and GI all use binary scales. Lastly, each word in SenticNet is labeled with a real value in the interval $[-1, 1]$, where 0 is neutral.

### 3 SentiVAE

We first describe a figurative generative process for a single sentiment lexicon $d \in \mathcal{D}$, where $\mathcal{D}$ is a set of sentiment lexica. Imagine there is a true (latent) polarity value $z_w^d$ associated with each word $w$ in the lexicon’s vocabulary. When the lexicon’s creator labels that word according to their chosen scale (e.g., thumbs-up or thumbs-down, a real value in the interval $[0, 1]$), they deterministically transform this true value to their chosen scale via a function $f(\cdot; \theta_d)$.2 Sometimes, noise is introduced during this labeling process, corrupting the label as it leaves the ethereal realm and producing the (observed) polarity label $x_w^d$. They then add this potentially noisy label to the lexicon.

Given a lexicon of observed polarity labels, the latent polarity values can be inferred using a VAE. The original VAE posits a generative model of observed data $\mathcal{X}$ and latent variables $\mathcal{Z}$: $P(\mathcal{X}, \mathcal{Z}) = P(\mathcal{X} | \mathcal{Z}) P(\mathcal{Z})$. Inference of $\mathcal{Z}$ then proceeds by approximating the (intractable) posterior $P(\mathcal{Z} | \mathcal{X})$ with a Gaussian distribution, factorized over the individual latent variables. A parameterized encoder function compresses $\mathcal{X}$ into $\mathcal{Z}$, while a parameterized decoder function reconstructs $\mathcal{X}$ from $\mathcal{Z}$.

SentiVAE extends the original VAE model to combine multiple lexica with disparate scales, producing a common latent representation of the polarity value for each word in the combined vocabulary.

#### Generative process

Given a set of sentiment lexica $\mathcal{D}$ with a combined vocabulary $\mathcal{W}$, SentiVAE posits a common latent representation $z_w^\mathcal{W}$ of the polarity value for each word $w \in \mathcal{W}$, where $z_w^\mathcal{W}$ is a three-dimensional categorical distribution over

---

1Our representation and code are available at https://github.com/ahoho/SentiVAE.

2Parameterized by lexicon-specific weights $\theta_d$. 

---

<table>
<thead>
<tr>
<th>Lexicon</th>
<th>Source</th>
<th>$N$</th>
<th>Dom</th>
</tr>
</thead>
<tbody>
<tr>
<td>SentiWordNet</td>
<td>WordNet</td>
<td>14107</td>
<td>$[-1, 1]^2$</td>
</tr>
<tr>
<td>MPQA</td>
<td>Newswire</td>
<td>4397</td>
<td>$[0, 1]$</td>
</tr>
<tr>
<td>SenticNet</td>
<td>—</td>
<td>100000</td>
<td>$[-1, 1]$</td>
</tr>
<tr>
<td>Hu-Liu</td>
<td>Product reviews</td>
<td>6790</td>
<td>$[0, 1]$</td>
</tr>
<tr>
<td>GI</td>
<td>—</td>
<td>4206</td>
<td>$[0, 1]$</td>
</tr>
<tr>
<td>VADER</td>
<td>Social media</td>
<td>7489</td>
<td>$[0, \ldots, 8]^{10}$</td>
</tr>
</tbody>
</table>

Table 1: Descriptive statistics for the sentiment lexica. $N$: vocabulary size. Dom: Domain of polarity values.
the sentiments \textit{positive}, \textit{negative}, and \textit{neutral}.

The generative process starts by drawing each latent polarity value $z^w$ from a three-dimensional Dirichlet prior, parameterized by $\alpha^w = (1, 1, 1)$:

$$z^w \sim \text{Dir}(\alpha^w).$$

If the word is uncontroversial,\(^3\) we spur this prior somewhat using the number of lexica in which the word appears $c(w)$. Specifically, we add $c(w)$ to the parameter for the sentiment associated with that word in the lexica, e.g., $\alpha^\text{SUPERB} = (1 + c(\text{SUPERB}), 1, 1)$. This has the effect of regularizing the inferred latent polarity value toward the desired distribution over sentiments.

Having generated $z^w$, the process proceeds by “decoding” $z^w$ into each lexicon’s chosen scale. First, for each lexicon $d \in D$, $z^w$ is deterministically transformed via neural network $f(\cdot; \theta_d)$ with a single 32-dimensional hidden layer, parameterized by lexicon-specific weights $\theta_d$:

$$\rho^w_d = f(z^w; \theta_d).$$

The transformed value $\rho^w_d$ is then used to generate the (observed) polarity label $x^w_d$ for that lexicon:

$$x^w_d \sim P_d(x^w_d | \rho^w_d).$$

The dimensionality of $\rho^w_d$ and the emission distribution $P_d$ are lexicon-specific. For SentiWordNet, $P_d$ is a two-dimensional Gaussian with mean $\rho^w_d$ and a diagonal covariance matrix equal to $0.01I$; for VADER, $P_d$ consists of ten nine-dimensional categorical distributions, collectively parameterized by $\rho^w_d$; for MPQA, Hu-Liu, and GI, $P_d$ is a Bernoulli distribution, parameterized by $\rho^w_d$; and for SentiNet, $P_d$ is a univariate Gaussian with mean and variance each an element in a two-dimensional $\rho^w_d$.

\textbf{Inference.} Inference involves forming the posterior distribution over the latent polarity values $Z$ given the observed polarity labels $X$. Because computing the normalizing constant $P(X)$ is intractable, we instead approximate the posterior with a family of distributions $Q_\lambda(Z)$, indexed by variational parameters $\lambda$. Specifically, we use

$$Q_\lambda(Z) = \prod_{w \in W} Q_{\beta^w}(z^w) = \prod_{w \in W} \text{Dir}(\beta^w).$$

To construct $\beta^w$, we first define a neural network $g(\cdot; \phi_d)$, with a single 32-dimensional hidden layer, which “encodes” $x^w_d$ into a three-dimensional vector. The output of this neural network is then transformed via a softmax as follows:

$$\omega^w_d = \text{softmax}(g(x^w_d; \phi_d))$$

$$\beta^w = 1 + \sum_{d \in D} \omega^w_d.$$  

\(^3\)We say that a word is uncontroversial if there is strong agreement across the sentiment lexica in which it appears. Even without this spurring, the inferred latent representation typically separates into the three sentiment classes, but performance on our text classification task is somewhat diminished.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Source</th>
<th>$N$</th>
<th>Classes</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMDB</td>
<td>Movies</td>
<td>25000</td>
<td>2</td>
</tr>
<tr>
<td>Yelp</td>
<td>Product reviews</td>
<td>100000</td>
<td>5 / 3</td>
</tr>
<tr>
<td>SemEval</td>
<td>Twitter</td>
<td>7668</td>
<td>3</td>
</tr>
<tr>
<td>MultiDom</td>
<td>Product reviews</td>
<td>6500</td>
<td>2</td>
</tr>
<tr>
<td>ACL</td>
<td>Scientific reviews</td>
<td>248</td>
<td>5 / 3</td>
</tr>
<tr>
<td>ICLR</td>
<td>Scientific reviews</td>
<td>2166</td>
<td>10 / 3</td>
</tr>
</tbody>
</table>

Table 2: Descriptive statistics for the training portions of the sentiment analysis datasets. $N$: number of texts.
et al., 2013) using Adam (Kingma and Ba, 2015) in the Pyro framework (Bingham et al., 2018). The standard reparameterization trick used in the original VAE does not apply to models with Dirichlet-distributed latent variables, so we use the generalized reparameterization trick of Ruiz et al. (2016).

4 Experiments and Results

To evaluate our approach, we first use SentiVAE to combine the six lexica described in §2. For each word $w$ in the combined vocabulary, we obtain an estimate of $z^w$ by taking the mean of $Q_{\beta^w}(z^w) = \text{Dir}(\beta^w)$—i.e., by normalizing $\beta^w$. We compare this representation to using $\beta^w$ directly, because $\beta^w$ contains information about SentiVAE’s certainty about the word’s latent polarity value. We evaluate our common latent representation via a text classification task involving nine English-language sentiment analysis datasets: IMDB (Maas et al., 2011), Yelp (Zhang et al., 2015), SemEval 2017 Task 4 (SemEval, Rosenthal et al. (2017)), multi-domain sentiment analysis (MultiDom, Blitzer et al. (2007)), and PeerRead (Kang et al., 2018) with splits ACL 2017 and ICLR 2017 (Kang et al., 2018). Each dataset consists of multiple texts (e.g., tweets, articles), each labeled with an overall sentiment (e.g., positive). Descriptive statistics for each dataset are shown in Tab. 2. For the datasets with more than three sentiment labels, we consider two versions—the original and a version with only three (bucketed) sentiment labels.

For each dataset, we transform each text into an average polarity value using either our representation, one of the six lexica, or a straightforward combination thereof, where the polarity value for each word in the (combined) vocabulary is a 16-dimensional vector that consists of a concatenation of polarity values. (Unlike SentiVAE, this concatenation does not yield a single sentiment lexicon that retains scale coherence, while achieving maximal coverage over words.) Specifically, we replace each token with its corresponding polarity value, and then average the these values (Go et al., 2009; Özdemir and Bergler, 2015; Kiritchenko et al., 2014). We then use the training portion of the dataset to learn a logistic regression classifier to predict the overall sentiment of each text from its average polarity value. Finally, we use the testing portion to compute the accuracy of the classifier.

Results. The results in Tab. 3 show that our representation using $\beta^w$ outperforms the individual lexica for all but one dataset, and that our representation using the mean of $Q_{\beta^w}(z^w)$ outperforms them for six datasets. This is likely because SentiVAE has a richer representation of sentiment than any individual lexicon, and it has greater coverage over words (see Tab. 4). The results in Tab. 5 support the former reason: even when we limit the words in our representation to match those in an individual lexicon, our representation still outperforms the individual lexicon. Unsurprisingly, our representation especially outperforms lexica with unidimensional scales. We also find that our representation outperforms the straightforward combination for datasets from domains that are not well supported by the individual lexica (see Tabs. 1 and 2 for lexicon and dataset sources, respectively). By combining lexica from different domains, our representation captures a general notion of sentiment that is not tailored to any specific domain.

5 Conclusion

We introduced a generative model of sentiment lexica to combine disparate scales into a common
Table 4: Coverage over words (percentage) by lexicon for the training portions of four of the nine datasets.

<table>
<thead>
<tr>
<th></th>
<th>IMDB</th>
<th>SemEval</th>
<th>Multi</th>
<th>ICLR</th>
</tr>
</thead>
<tbody>
<tr>
<td>SentiVAE</td>
<td>70</td>
<td>64</td>
<td>81</td>
<td>71</td>
</tr>
<tr>
<td>SentiWordNet</td>
<td>15</td>
<td>14</td>
<td>24</td>
<td>16</td>
</tr>
<tr>
<td>MPQA</td>
<td>10</td>
<td>7</td>
<td>18</td>
<td>9</td>
</tr>
<tr>
<td>SenticNet</td>
<td>40</td>
<td>39</td>
<td>53</td>
<td>45</td>
</tr>
<tr>
<td>Hu-Liu</td>
<td>7</td>
<td>5</td>
<td>13</td>
<td>5</td>
</tr>
<tr>
<td>GI</td>
<td>8</td>
<td>7</td>
<td>15</td>
<td>6</td>
</tr>
<tr>
<td>VADER</td>
<td>7</td>
<td>6</td>
<td>13</td>
<td>5</td>
</tr>
</tbody>
</table>

Table 5: Classification accuracies for a 10% validation portion of two of the datasets. The first row, labeled SentiVAE, contains the classification accuracy for our representation using $\beta_w$. Subsequent (lexicon-specific) rows compare our representation (SV), restricted to the vocabulary of that lexicon, to the lexicon itself (Lex).

<table>
<thead>
<tr>
<th></th>
<th>IMDB 2C</th>
<th>SemEval 3C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SV Lex</td>
<td>SV Lex</td>
</tr>
<tr>
<td>SentiVAE</td>
<td>74.7</td>
<td>72.4</td>
</tr>
<tr>
<td>SentiWordNet</td>
<td>70.6</td>
<td>63.4</td>
</tr>
<tr>
<td>MPQA</td>
<td>73.5</td>
<td>66.6</td>
</tr>
<tr>
<td>SenticNet</td>
<td>74.4</td>
<td>60.9</td>
</tr>
<tr>
<td>Hu-Liu</td>
<td>73.6</td>
<td>68.4</td>
</tr>
<tr>
<td>GI</td>
<td>71.4</td>
<td>59.3</td>
</tr>
<tr>
<td>VADER</td>
<td>73.6</td>
<td>73.1</td>
</tr>
</tbody>
</table>

Acknowledgements

We would like to thank to Adam Forbes for the design of Fig. 1. We further acknowledge the support of the NVIDIA Corporation with the donation of the Titan Xp GPU used to conduct this research.

References


Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the Tenth...