Sentiment as an Ordinal Latent Variable

Niklas Stoehr[ʕ] **Ryan Cotterell**[ʕ] **Aaron Schein**[ə]
[ʕ]ETH Zürich [ə]The University of Chicago
niklas.stoehr@inf.ethz.ch ryan.cotterell@inf.ethz.ch schein@uchicago.edu

Abstract

Sentiment analysis has become a central tool in various disciplines outside of natural language processing. In particular in applied and domain-specific settings with strong requirements for interpretable methods, dictionary-based approaches are still a popular choice. However, existing dictionaries are often limited in coverage, static once annotation is completed and sentiment scales differ widely; some are discrete others continuous. We propose a Bayesian generative model that learns a composite sentiment dictionary as an interpolation between six existing dictionaries with different scales. We argue that sentiment is a latent concept with intrinsically ranking-based characteristics — the word "excellent" may be ranked more positive than "great" and "okay", but it is hard to express how much more exactly. This prompts us to enforce an ordinal scale of ordered discrete sentiment values in our dictionary. We achieve this through an ordering transformation in the priors of our model. We evaluate the model intrinsically by imputing missing values in existing dictionaries. Moreover, we conduct extrinsic evaluations through sentiment classification tasks. Finally, we present two extension: first, we present a method to augment dictionary-based approaches with word embeddings to construct sentiment scales along new semantic axes. Second, we demonstrate a Latent Dirichlet Allocation-inspired variant of our model that learns document topics that are ordered by sentiment.

https://github.com/niklasstoehr/ordinal-sentiment

1 Introduction

Sentiment analysis is being applied in various domains from political science (Young and Soroka, 2012; Gründl, 2020; Widmann and Wich, 2022) to economics (Stephany et al., 2022) and computational social science (West et al., 2014; Falck et al., 2020; Stoehr et al., 2021). In all of these applications, there is a strong demand for domain-specific and interpretable methods (Hofman et al., 2021; Widmann and Wich, 2022) making dictionary-based sentiment analysis still a popular choice (Young and Soroka, 2012; Hoyle et al., 2019; Gründl, 2020; Friedrichs et al., 2022).

Sentiment dictionaries describe a mapping between word types and some form of sentiment values. We consider the most general notion of sentiment value referring to the polarity score along a positive-negative axis, instead of fine-grained emotion dimensions (Plutchik, 1980) or stance (Mohammad, 2016). Sentiment values are measured on scales of different support (§2): some dictionaries assign binary "positive" and "negative" values (Hu and Liu, 2004; Wilson et al., 2005; Stone et al., 2007). These discrete values are often falsely interpreted as unordered, nominal categories. Other dictionaries have continuous scales that assign cardinal, floating point values (Hutto and Gilbert, 2014; Cambria et al., 2014).

In this work, we propose a method for merging sentiment dictionaries with different scales into a single, composite dictionary. Paying tribute to the subjective and ranking-based characteristics of sentiment, we design the dictionary to have an ordinal scale. Ordinal scales define discrete, ordered classes where interval sizes between classes are unequal and typically unknown (Stevens, 1946). For instance, the word "excellent" may be ranked more positive than "great" and "okay", but it is hard to express how much more positive. An example is the ordinal Likert scale (Likert, 1932) used to measure attitudes in psychometrics.

Our ordinal sentiment scale is derived from an ordinal latent variable within a probabilistic, generative model (§3). In particular, the latent variable's classes represent sentiment values. The classes are uniquely ordered which is achieved through an ordering transformation that is applied to the

Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pages 103–115
May 2-6, 2023

priors of our model (§3.2). Our model is tightly coupled with recent advancements in probabilistic programming (Bingham et al., 2018; Phan et al., 2019) and gradient-based inference (Homan and Gelman, 2014). These advancements alleviate the strict requirement of closed-formedness and conjugacy to perform posterior inference in complex Bayesian models with latent ordering motifs.

Our ordinal scale is learned as an unsupervised interpolation between 6 popular sentiment dictionaries. This has several benefits: on the other hand, we can impute missing sentiment values in existing dictionaries. We evaluate this capacity in a Bayesian data imputation task (§4.2). On the other hand, interpolating between different dictionaries causes our composite dictionary to have high coverage of word types from widely different sources. We evaluate our composite dictionary in 6 sentiment classification tasks from different domains (§4.3). Taking a Bayesian approach, we have access to uncertainty estimates for each sentiment value per word type. We find that uncertainty is larger for ambiguous and rare word types that are covered by only few dictionaries (§5).

In §6, we present two possible extension of our ordinal latent variable model. To further expand word type coverage, we incorporate sentiment values derived from bi-polar semantic axes within word embeddings (§6.1). To demonstrate the wide applicability of our ordinal modeling motif, we introduce a model variant that is closely related to Latent Dirichlet Allocation (LDA; Blei et al., 2003), but learns topics ordered by sentiment (§6.2). We publish our code together with our learned, high-coverage sentiment dictionary, annotated with posterior credible intervals.

2 Data: Sentiment Dictionaries

We consider 6 popular English-language sentiment dictionaries: SenticNet (SC) (Cambria et al., 2014), SentiWordNet (SW) (Baccianella et al., 2010), Vader (VA) (Hutto and Gilbert, 2014), General Inquirer (GI) (Stone et al., 2007) Hu-Liu (HL) (Hu and Liu, 2004) and MPQA (MP) (Wilson et al., 2005). The dictionaries vary in the number of included word types, the word source, application domain and the sentiment scale, see appendix Tab. 3. SC, SW and VA have continuous, bounded sentiment values, while GI, HL and MP have discrete, binary values as visualized in Fig. 1. We scale all continuous values to a $[0, 1]$ range. Some of the

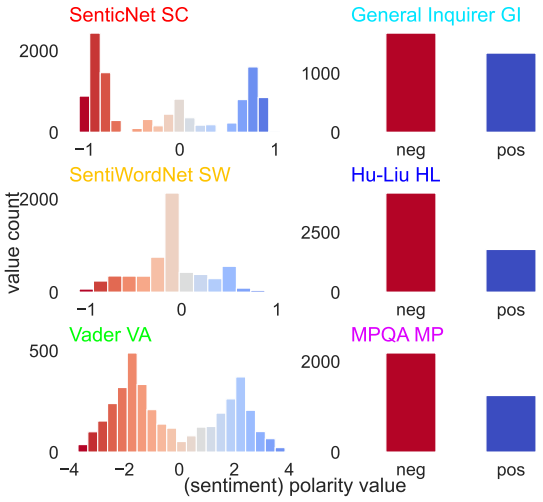

Figure 1: Sentiment value distributions of 6 sentiment dictionaries. Some dictionaries assign continuous float values to word types {SC, SW, VA}, others limit themselves to discrete, (binary) values {GI, HL, MP}.

dictionaries such as SW and VA feature multiple sentiment values per word type. We average those to consistently obtain one value per word type for all dictionaries, which allows for a fair comparison. We group the sentiment dictionaries in a single data table by word type. Since different dictionaries contain different word types, this results in many missing values. We filter the data table so that each word type is covered by at least 2 dictionaries. This leaves us with V =12,342 unique word types that serve as our dataset.

3 Model

Our goal is to learn a unifying sentiment dictionary as an interpolation between existing sentiment dictionaries. Each word type v is described by one or multiple sentiment values of a dictionary. Depending on the dictionary's scale, sentiment values can be continuous x_v^c or discrete x_v^d. The superscripts c and d represent continuous and discrete dictionaries respectively, i.e., $c \in \{\text{SC}, \text{SW}, \text{VA}\}$ and $d \in \{\text{GI}, \text{HL}, \text{MP}\}$. Considering all 6 sentiment dictionaries, we have a tuple of 6 sentiment values $\{x_v^{\text{SC}}, x_v^{\text{SW}}, x_v^{\text{VA}}, x_v^{\text{GI}}, x_v^{\text{HL}}, x_v^{\text{MP}}\}$ per word type. Due to our filtering in §2, at most 4 of those values can be missing (NaN).

3.1 Generative Story

For each word type v, we assume that its sentiment class z_v is sampled from a Categorical distribution over K classes, parameterized by $\boldsymbol{\pi}$, a K-dimensional vector of class probabilities. We

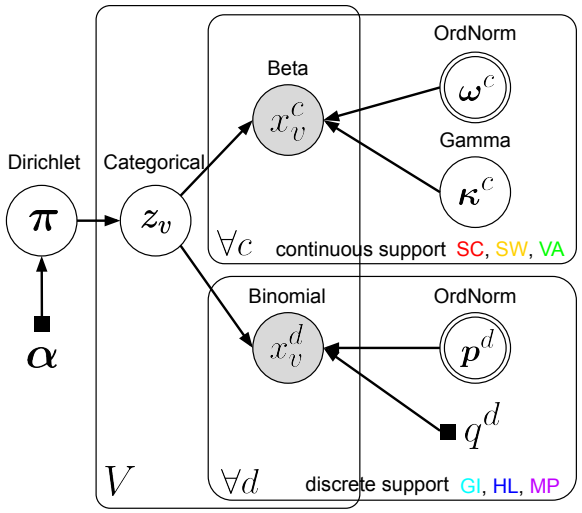

Figure 2: Model for interpolating sentiment dictionaries. Each word type v is described by observed sentiment values x_v^c and x_v^d from different sentiment dictionaries. The continuous, bounded dictionaries c are modeled by Beta and the discrete dictionaries d by Binomial distributions. Some priors are ordered, as indicated by double-border nodes (◎). This spurs the categorical latent z_v to be ordinal. Solid, black squares represent fixed hyperparameters.

further assume that $\boldsymbol{\pi}$ is drawn from a Dirichlet distribution. Conditioned on the sentiment class z_v, each observed continuous $x_v^c \in [0, 1]$ and discrete $x_v^d \in \{0, \ldots, q^d\}$ sentiment value per dictionary is independently sampled as depicted in Fig. 2. We assume that the values x_v^{SC}, x_v^{SW} and x_v^{VA} that come from dictionaries with continuous, bounded support are drawn from Beta distributions. The values from binary dictionaries, x_v^{GI}, x_v^{HL} and x_v^{MP}, are naturally Bernoulli random variables—however we represent them more generally as Binomial random variables, with number of trials equal to q^d (where $q^d = 1$ in our case), to accommodate dictionaries with arbitrary ordinal support:

$$\boldsymbol{\pi} \sim \text{Dirichlet}(\boldsymbol{\alpha}) \tag{1}$$

$$z_v \sim \text{Categorical}(\boldsymbol{\pi}) \tag{2}$$

$$x_v^c \mid z_v \sim \text{Beta}(\omega_{z_v}^c, \kappa_{z_v}^c) \tag{3}$$

$$x_v^d \mid z_v \sim \text{Binomial}(q^d, p_{z_v}^d) \tag{4}$$

We discuss the parameters $\omega_{z_v}^c, \kappa_{z_v}^c$ and $p_{z_v}^d$ that induce ordering on the latent variable z_v in the following section.

3.2 Ordinal Latent Variable

While the classes of a Categorical distribution are generically unordered, the structure of our model induces a natural ordering over the K classes that z_v can take. When $z_v = k$, the parameters ω_k^c, κ_k^c and p_k^d parameterize the Beta and the Binomial distributions from which word type v's sentiment scores are drawn. By imposing an ordering on those parameters (e.g., $\omega_k^c < \omega_{k+1}^c$), we induce ordering on z_v. In the following subsections, we introduce prior distributions over the vectors $\boldsymbol{\omega}^c$ and $\mathbf{p}^d$ that ensure they are ordered, such that higher classes correspond Beta and Binomial classes that are centered around higher sentiment values.

OrderedNormal Distribution. To induce ordering into the parameters $\boldsymbol{\omega}^c$ and $\mathbf{p}^d$ and thus the categories of z_v, we import the OrderedNormal distribution of Stoehr et al. (2022). The OrderedNormal is a distribution over a K-dimensional vector $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_K)$ whose elements are ordered, $\lambda_k < \lambda_{k+1}$. Specifically, for parameters $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_K)$ and $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_K)$, an OrderedNormal random variable $\boldsymbol{\lambda} \sim \text{OrderedNormal}(\boldsymbol{\mu}, \boldsymbol{\sigma})$ can be generated as:

$$\begin{aligned} s_k &\overset{\text{ind.}}{\sim} \text{Normal}(\mu_k, \sigma_k) \quad \text{for } k \text{ in } \{1, \ldots, K\} \\ (\lambda_1, \ldots, \lambda_K) &\leftarrow \text{Ord}(\{s_1, \ldots, s_K\}) \end{aligned} \tag{5}$$

where $\text{Ord}(\cdot)$ is a deterministic function that transforms the set of Normal variates $\{s_1, \ldots, s_K\}$, into a strictly increasing vector—specifically:

$$\lambda_k \leftarrow \begin{cases} s_1 & \text{if } k = 1 \\ s_1 + \sum_{i=2}^{k} \exp(s_i) & \text{if } k > 1 \end{cases} \tag{6}$$

This transformation is an invertible, smooth bijection which is differentiable and thus facilities gradient-based parameter inference (Rezende and Mohamed, 2015) as further discussed in §3.3.

Ordered Beta parameters. When $z_v = k$, the continuous sentiment score x_v^c is drawn from a $\text{Beta}(\omega_k^c, \kappa_k^c)$ distribution, where $\omega_k^c \in (0, 1)$ is the mode and $\kappa_k^c > 0$ is the concentration parameter. We impose ordering over the K-dimensional vector of mode parameters $\boldsymbol{\omega}^c = (\omega_1^c, \ldots, \omega_K^c)$ by positing the following prior:

$$S^{-1}(\boldsymbol{\omega}^c) \sim \text{OrderedNormal}(\boldsymbol{\mu}^c, \boldsymbol{\sigma}^c) \tag{7}$$

where $S^{-1}(\cdot)$ is the inverse sigmoid function. In other words, we first sample from an OrderedNormal, and then apply the element-wise sigmoid function to ensure that all elements of $\boldsymbol{\omega}^c$ are between 0 and 1. We do not impose any ordering on

the concentration parameters $(\kappa_1^c, \ldots, \kappa_K^c)$ and assume they are independent shifted Gamma random variables with shape γ_k^c and rate η_k^c:

$$(\kappa_k^c - 2) \overset{\text{ind.}}{\sim} \text{Gamma}(\gamma_k^c, \eta_k^c) \quad (8)$$

This formulation ensures that the concentration parameter is $\kappa_k^c \geq 2$ so that the Beta distribution is unimodal at ω_k^c.

Ordered Binomial parameters. Our model assumes observed discrete values x_v^d are sampled from a $\text{Binomial}(q^d, p_{z_v}^d)$ distribution where the number of trials q^d is based on the number of discrete sentiment classes in the dictionary d, and $p_{z_v}^d$ is the probability parameter. We impose ordering on the vector of probabilities $\mathbf{p}^d$ by positing the following prior:

$$S^{-1}(\mathbf{p}^d) \sim \text{OrderedNormal}(\boldsymbol{\mu}^d, \boldsymbol{\sigma}^d) \quad (9)$$

3.3 Posterior Inference

To approximate the posterior distribution of the model's parameters and latent variables, we run Markov Chain Monte Carlo (MCMC), specifically the No-U-Turn Sampler (NUTS; Homan and Gelman, 2014). NUTS is gradient-based and requires continuous latent variables and parameters. However, the latent variable z_v in our model is explicitly non-continuous. We implement our model using the probabilistic programming framework Pyro (Bingham et al., 2018; Phan et al., 2019) that offers an "enumeration" strategy, termed `parallel_enumeration`, to handle the discrete latent z_v during inference. This enumeration strategy effectively marginalizes z_v out numerically so that we can draw samples of the continuous parameters θ from $\theta^{(t)} \sim p(\theta \mid X)$, where X are all of the observed sentiment values. We can draw samples of the latent variables $z_v^{(t)} \sim p(z_v|\theta^{(t)}, x_v^c, x_v^d)$. To realize the ordering transformation presented in Eq. (6), we rely on Pyro's `OrderedTransform`.

Inferring Ordinal Sentiment Values. Ultimately, we are interested in mapping word types to ordinal sentiment values using our fitted model. As discussed, we approximate the posterior $p(z_v \mid X)$ using MCMC samples $\{z_v^{(t)}\}_{t=1}^T$, and then compute a point estimate either by taking the mean $\bar{z}_v = \frac{1}{T}\sum_{t=1}^T z_v^{(t)}$ or the mode $\dot{z}_v$. Considering the mode, we obtain an integer value $\dot{z}_v \in \{1, \ldots, K\}$ that may be interpreted as an ordinal sentiment value. In union, these values describe an ordinal scale that is part of a learned, composite sentiment dictionary that we term ORDSCALE.

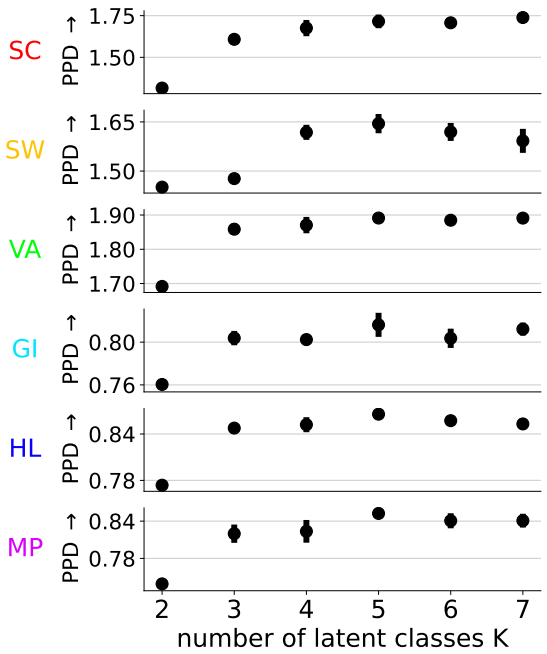

Figure 3: $K = 5$ latent classes yield a good trade-off between the number of model parameters and model fit as measured by scaled posterior predictive density (PPD) on the test set over 5 different random seeds.

4 Experiments

We evaluate our model intrinsically (§4.2) and our inferred ordinal sentiment dictionary, ORDSCALE, extrinsically (§4.3). Therefore, we first fit our model to existing sentiment dictionaries, identify the optimal number of latent classes and finally infer the ordinal scale. First, we split the $V = 12{,}342$ word types into a 70% training and 30% test set. Next, we run the NUTS sampler to perform posterior inference as introduced in §3.3. We discard the first 200 burn-in samples and consider only the following $T = 1000$ samples from the posterior.

4.1 Optimal Number of Latent Classes

We identify the optimal number of latent classes K that lead our model to achieve high likelihood on the test set. Therefore, we fit and evaluate our model on a range of class settings, e.g., $K = \{2, \ldots, 7\}$. We find that $K = 5$ yields a high scaled Posterior Predictive Density (PPD; Gelman et al., 1996, 2014)[1] on the test set as shown in Fig. 3. In the following, we consider our model with the optimal class setting $K = 5$.

[1]We explain all evaluation metrics in App. A.2.

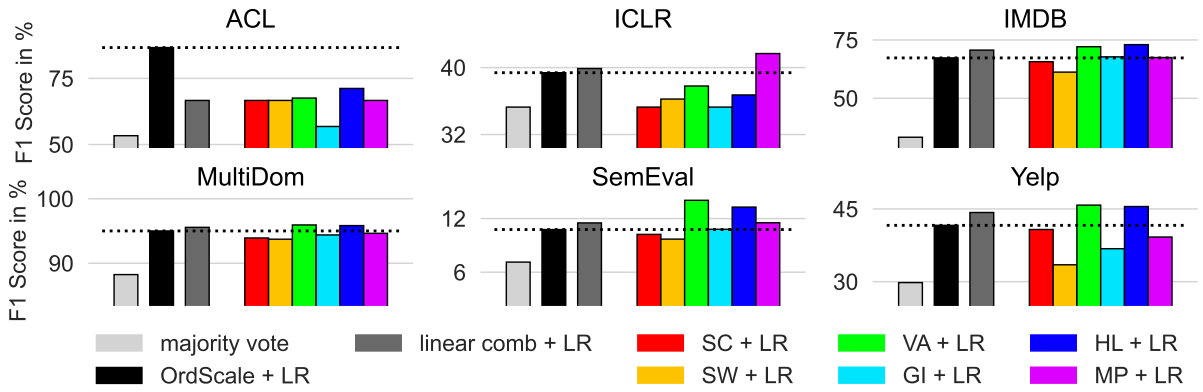

Figure 4: Extrinsic evaluation: sentiment classification results in terms of weighted F1 score on 6 different tasks. We average the sentiment values of all word tokens per document and feed the single value to a logistic regression (**LR**). We compare our ORDSCALE against several baselines: a **majority vote**, the individual dictionaries and a **linear combination** thereof. ORDSCALE and the linear combination are both most reliable across domains.

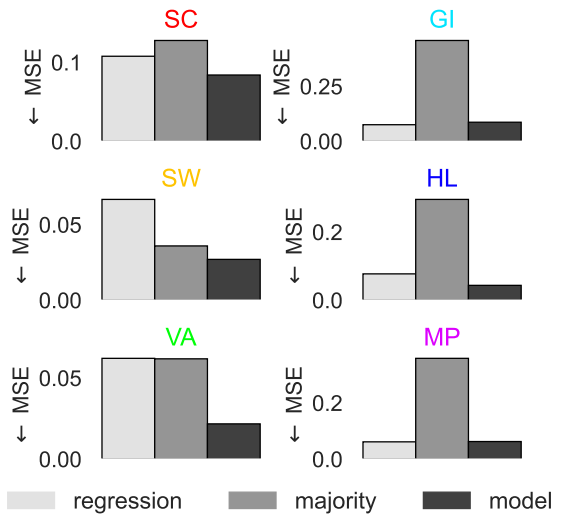

Figure 5: Intrinsic evaluation: we impute the missing sentiment value y_v^{missing} of one removed dictionary, e.g., SC, based on sentiment values y_v^{given} of other dictionaries, e.g., SW, VA, GI, HL, MP, in a test dataset. We consider two baselines: a **majority** vote fitted only to the later removed dictionary with $K = 5$; a linear **regression** trained on the direct mapping $y_v^{\text{given}} \rightarrow y_v^{\text{missing}}$. Our **model** condenses observations into a single latent.

4.2 Intrinsic Evaluation: Data Imputation

Experimental Setup. We perform an imputation task to evaluate model fit. We first approximate the posterior distribution of the continuous model parameters θ on the full training set using the optimal class setting of $K = 5$. On the testing set, we remove one sentiment dictionary entirely and refer to the corresponding, but now missing sentiment values as y_v^{missing}. For instance, we remove $y_v^{\text{missing}} = \{x_v^{\text{SC}}\}$, which leaves us with the five-way tuple $y_v^{\text{given}} = \{x_v^{\text{SW}}, x_v^{\text{VA}}, x_v^{\text{GI}}, x_v^{\text{HL}}, x_v^{\text{MP}}\}$. The objective is to impute the removed sentiment values of entirely unseen word types. To this end, we sample the discrete latent variable $z_v^{(t)}$ per word type v according to $z_v^{(t)} \sim p(z_v|\theta^{(t)}, y_v^{\text{given}})$. Then, we draw $\widehat{y}_v^{(t)} \sim p(y_v^{\text{missing}}|z_v^{(t)}, \theta^{(t)})$ and take the mean over samples $\frac{1}{T}\sum_{t=1}^{T}\widehat{y}_v^{(t)}$ to predict a single missing sentiment value.

Results. We consider different baselines: instead of using all sentiment dictionaries in a single model, we fit six separate models to each dictionary individually. In other words, this simple model has only one observed variable, namely the one that is being removed on the test, which resembles a **majority vote** baseline. Moreover, we train six linear **regression** models in a supervised task to predict y_v^{missing} from y_v^{given}. We report the results in terms of mean squared error (MSE) in Fig. 5. Our model outperforms both baselines in imputing all dictionaries, except the dictionaries GI and MP.

4.3 Extrinsic Evaluation: Classification

We extrinsically evaluate our model, or rather, our inferred sentiment dictionary ORDSCALE (§3.3) in a sentiment classification task. It is important to stress that we are not chasing benchmarks by comparing against state-of-the-art models. Instead, we are inspecting the sentiment-related information preserved in our ordinal scale.

Task Data. We consider 6 diverse sentiment classification tasks. These are PeerRead (Kang et al., 2018), specifically the splits ACL and ICLR, IMDB (Maas et al., 2011), MultiDom (Blitzer et al., 2007),

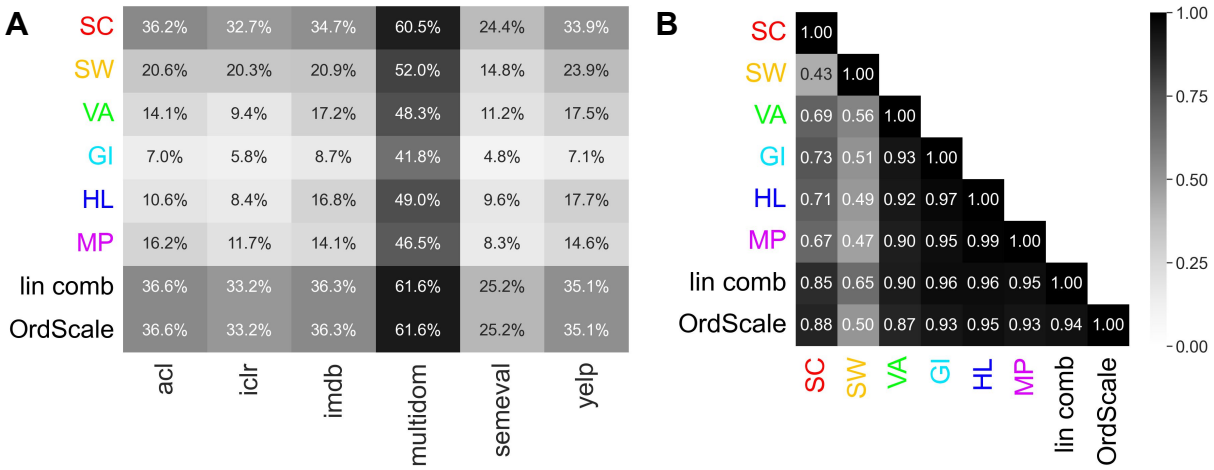

Figure 6: (**A**) Fraction of word types in sentiment tasks (columns) that are covered by sentiment dictionaries (rows). ORDSCALE and the linear combination (**comb**) of all dictionaries have the highest coverage in each task. (**B**) Correlation of sentiment values for word types that are shared between dictionaries. Overall, sentiment dictionaries are strongly correlated. We find that ORDSCALE differs from linear combination in its correlations. As can be seen in Fig. 1, SW contains many neutral values explaining its overall low correlation.

SemEval 2016 Task 4 (Palogiannidi et al., 2016) and the Yelp reviews dataset (Zhang et al., 2015). All tasks consists of full text (e.g., reviews or tweets), referred to as documents, labeled with sentiment classes. They are split in pre-defined train–test sets and differ in the number of unique sentiment classes, ranging from 2 to 5 (see Tab. 4).

Experimental Setup. We consider ORDSCALE with $K = 5$ ordinal classes and compare it against several baselines: a majority vote that always selects the majority class in each task; the six individual sentiment dictionaries introduced in §2 and a **linear combination** of all (scaled) six sentiment dictionaries. This linear combination has the same coverage of word types as ORDSCALE as further elaborated in Fig. 6). For predicting the sentiment labels of documents, we choose a simple procedure following Go et al. (2009); Kiritchenko et al. (2014); Ozdemir and Bergler (2015); Hoyle et al. (2019): for each document, we replace each token with its corresponding sentiment value from a dictionary. Then, we average all values per document and pass it to a logistic regression (**LR**) model that is fitted on the training set to predict document labels. To allow for a fair comparison, all dictionaries are averaged to one sentiment value per word type.

Results. Results expressed as weighted F1 Scores are presented in Fig. 4. We find that ORDSCALE and the linear combination baseline only rank in the middle range on every task. Yet, they never perform poorly and may be considered very reliable across different tasks and data domains. This may be attributed to their broad word-type coverage as discussed in §5. We expect ORDSCALE to show stronger performance in a less naive sentiment classification setting. In Fig. 4, we simply average the sentiment values of all tokens in a document which may lead them to neutralize each other. Consequently, the broad word-type coverage does not necessarily pay off. Exploiting it may require more expressive models that operate on the full token sequence instead.

	x_v^{SC}	x_v^{SW}	x_v^{VA}	x_v^{GI}	x_v^{HL}	x_v^{MP}	$\bar{z}_v$	$\dot{z}_v$
excellent	0.7	-	2.7	1	1	-	3.8	4
great	0.1	0.8	1.8	-	1	1	3.4	3
okay	0.1	0	-	-	-	-	2.0	2
bad	-0.3	-0.6	-2.5	0	0	0	1.0	1
horrible	-0.9	-	-2.5	0	0	0	0.1	0

Table 1: Sentiment scores for selected word types. $\bar{z}_v$ represents the mean and $\dot{z}_v$ the mode over samples per word type from our ordinal latent variable.

5 Discussion

Interpretability of Ordinal Sentiment Scale. We qualitatively inspect sentiment values for different word types across dictionaries. As shown in Tab. 1, even popular words such as "excellent" are not covered by all sentiment dictionaries. Due to the different scales, the dictionaries can also be tricky to interpret, especially those with continuous

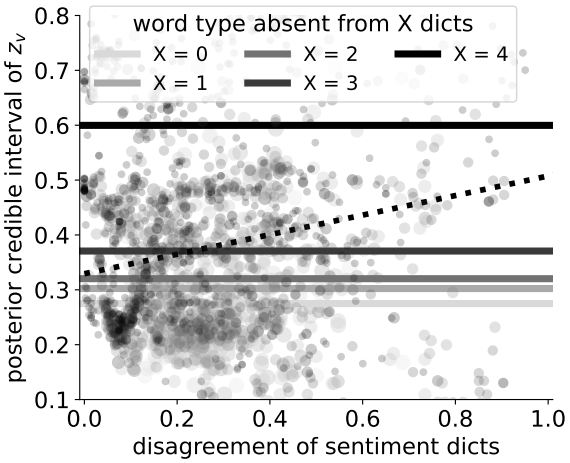

Figure 7: Posterior credible interval of z_v over disagreement of sentiment dictionaries. As expected, we observe that the credible interval for a word type's sentiment value grows larger if it is absent in more sentiment dictionaries. Moreover, we observe a correlation between the posterior credible interval and disagreement of sentiment dictionaries for a given word type.

support. In Vader (VA) for instance, there is no difference between "bad" and "horrible". Agreeing on exact float value scores seems more difficult than agreeing on a ranking which supports our call for an ordinal sentiment scale. The mode $\dot{z}_v$ of our latent variable represents 5 distinct ordinal levels that match the mental ordering of the words based on sentiment. It is ranking "excellent" as more positive than "great" and "okay".

Correlation Requirement. Across all tasks, our sentiment dictionary covers more word types than other dictionaries since it basically describes their interpolation as displayed in Fig. 6A. However, a limitation of our models is the requirement that observed variables have to be correlated. Considering Fig. 6B, if dictionaries were not correlated, our model could not infer one from the other in the imputation task (§4.2) nor learn a latent correlate. Conversely, if two dictionaries were perfectly correlated, considering both would be superfluous since one incorporated all information of the other.

Sentiment Uncertainty. One advantage adding to the interpretability of our Bayesian modeling approach is access to posterior credible intervals. Unlike many of the existing sentiment dictionaries, the sentiment values derived from our model are accompanied by a "measure of uncertainty". Fig. 7 shows that the posterior credible intervals are larger for word types that are missing in more sentiment dictionaries. Moreover, there is an expected correlation between the disagreement of sentiment dictionaries in terms of standard deviation and the size of the posterior credible interval.

pain	x_v^{emb}	$\dot{z}_v$	**humor**	x_v^{emb}	$\dot{z}_v$
painful	0.64	4	funny	0.66	4
unsettling	0.40	3	comic	0.33	3
stressful	0.25	2	normal	0.08	2
nontoxic	-0.21	1	tedious	-0.32	1
cured	-0.64	0	boring	-0.44	0

Table 2: Using the SemAxis approach (An et al., 2018), we can learn dictionaries with ordinal scales along any bi-polar semantic axes. We demonstrate this for the seed words "painful – cured" and "funny – boring".

Label Switching. In topic models with Categorical (or Multinomial) distributions, aggregating samples from the posterior distribution between different or even within the same MCMC chain can be complicated. This is due to a problem called label switching which arises from the non-unique ordering of latent classes (Stephens, 2000). The ordered priors in our model represent an identifiability constraint that mitigates the label switching problem (Stephens, 2000; Murphy, 2012).

6 Extensions and Applications

6.1 Word Embeddings

Newly appearing, changing or domain-specific word types may need to be added to an existing dictionary (Wang et al., 2021). To address this issue, we extend our approach considering static word embeddings. In particular, we obtain sentiment values for all word types in our dictionary using the SemAxis approach (An et al., 2018).

We consider 300-dimensional Glove embeddings (Pennington et al., 2014). First, we choose two pole word types such as v_+ ="good" and v_- ="bad" and obtain their vector representations $\mathbf{v}_+$ and $\mathbf{v}_-$.[2] Next, we compute the linear semantic axis between the poles according to $\mathbf{v}_+ - \mathbf{v}_-$. Finally, we can project any word prevalent in Glove's vocabulary onto this axis by computing the Cosine similarity between the word type's vector and the semantic axis. The similarity can be interpreted as a word's embedding-based polarity value x_v^{emb} on the respective semantic axis. We simply treat the

[2]We may also choose a set of word types, e.g., {good, positive} and {bad, negative} and consider their mean vector.

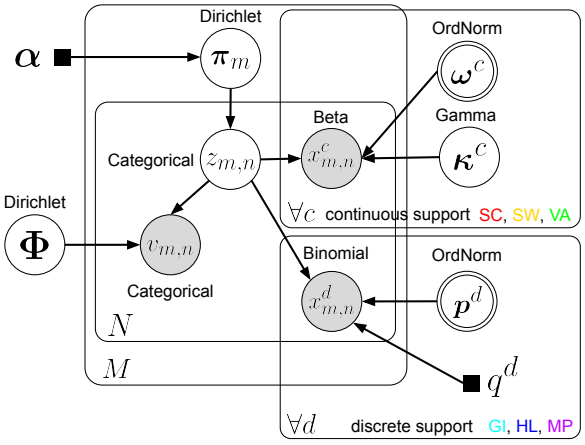

Figure 8: Document-level model with ordered topics. Following Latent Dirichlet Allocation (LDA), we can add another plate over M documents and include a topic-word type matrix $\mathbf{\Phi}$ where each row is sampled independently from a Dirichlet distribution. Different to LDA, we now have multiple observed variables: the word types $v_{m,n}$ and the sentiment values $x^c_{m,n}$, $x^d_{m,n}$ per word token n in each document m. Double-border nodes are ordered (◎).

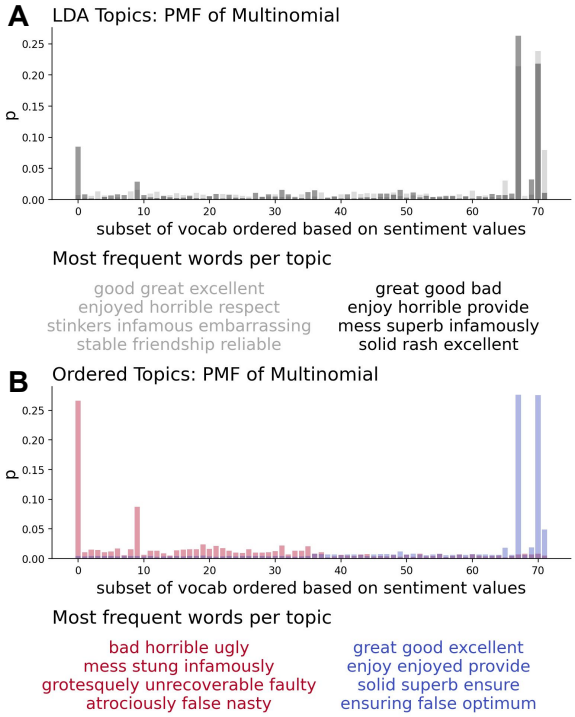

Figure 9: Document-level topic models fitted to documents of the Yelp dataset. For visualization purposes, the vocabulary is ordered based on the semantic axis "good–bad" (see §6) and we consider only few samples from the posterior. (**A**) The LDA model yields topics that are hard to interpret. (**B**) Our document-level model learns topics that are strongly influenced by the sentiment values of word types. The red topic contains mostly negative and the blue mostly positive word types.

word type–value pairs as its own dictionary with continuous support. When including this dictionary as an observed site in our model, we can impute missing values in dictionaries that lack words that are existent in Glove. Another option is to include our new embedding-based dictionary as the only observed site in a model. The model then learns an ordinal discretization of the semantic axis. In Tab. 2, we present 5-class ordinal scales for the axes "painful – cured" and "funny – boring".

6.2 Document-level Model

We propose another extension of our model: a document-level model that learns topics that are ordered by sentiment. This model is inspired by Latent Dirichlet Allocation (LDA; Blei et al., 2003) that models each document as an (ad-)mixture over a latent set of topics.

Generative Story. The generative story goes as follows: we have a corpus of M documents. A corpus-wide alpha concentration $\boldsymbol{\alpha}$ parameterizes a Dirichlet over K topics. Now, for each document m, a topic distribution $\boldsymbol{\pi}_m$ is sampled from the Dirichlet. Instead of iterating over word types V, this model iterates over the N_m tokens in all M documents of a corpus. Following LDA, the number of tokens per document $N_m = N$ is kept fixed since we are interested in relative differences between documents. For each token n, a topic $z_{m,n}$ is drawn from a Categorical parameterized by $\boldsymbol{\pi}_m$.

$$\boldsymbol{\pi}_m \sim \text{Dirichlet}(\boldsymbol{\alpha}) \tag{10}$$

$$z_{m,n} \sim \text{Categorical}(\boldsymbol{\pi}_m) \tag{11}$$

Conditioned on $z_{m,n}$, we sample multiple observed sites per word: the word type $v_{m,n}$ and the word type's associated sentiment values $x^c_{m,n} \in [0, 1]$ and $x^d_{m,n} \in \{0, 1\}$ as given by the dictionaries d and c. Sampling word types is identical to LDA: $z_{m,n}$ indexes into a $K \times |\mathcal{V}|$ topic-word type matrix $\mathbf{\Phi}$ where each row is sampled from a Dirichlet distribution. $|\mathcal{V}|$ is the size of the vocabulary. The selected row vector $\boldsymbol{\phi}_{z_{m,n}}$ parameterizes a Categorical distribution over words in the vocabulary associated with topic $z_{m,n} = k$. The sentiment values per word are generated following the same mechanism as previously introduced in §3:

$$v_{m,n} \mid z_{m,n} \sim \text{Categorical}(\boldsymbol{\phi}_{z_{m,n}}) \tag{12}$$

$$x^c_{m,n} \mid z_{m,n} \sim \text{Beta}(\omega^c_{z_{m,n}}, \kappa^c_{z_{m,n}}) \tag{13}$$

$$x^d_{m,n} \mid z_{m,n} \sim \text{Binomial}(q^d, p^d_{z_{m,n}}) \tag{14}$$

Applications. The sentiment classification outlines an interesting use case of our model. Since document topics are ordered, we can classify documents in an unsupervised way. Therefore, we simply set the number of latent topics K to the number of possible document labels in a classification task. Then, we predict labels based on the inferred topic $\dot{z}_m$ of a document. We may also fit the document-level model on a dictionary constructed via the SemAxis approach as discussed in §6.1. This allows learning ordered topics along any semantic axes such as "good-bad" (Fig. 9) or "funny-boring" (App. Fig. 10) without supervision.

7 Related Work

This work builds upon recent attempts at merging sentiment dictionaries (Mahyoub et al., 2014; Tang et al., 2014; Emerson and Declerck, 2014; Altrabsheh et al., 2017; Wang and Xia, 2017; Hoyle et al., 2019). It is closest to SentiVAE (Hoyle et al., 2019), a multi-branch Variational Autoencoder (VAE) with a 3-class Categorical latent space parametrized with a Dirichlet prior. Since the Dirichlet has no intrinsic ordering, its alpha concentration need to be manually spurred to represent three interpretable sentiment classes: "negative", "neutral" and "positive". In contrast to Hoyle et al. (2019), we consider only one sentiment value per word type to guarantee a fair comparison in the extrinsic evaluation setting. Our latent variable model is inspired by Stoehr et al. (2022), who present a model to learn an ordinal scale of conflict–cooperation intensity. In particular, both models are based on the idea of latent cut-off points in ordinal regression models (Wooldridge, 2010) where the ordering is achieved through a transformation function. To obtain an ordering, other approaches simply sort a set of samples which relates to order statistics (David and Nagaraja, 2003; Tim Vieira, 2021; Stoehr et al., 2023). There exist many approaches for learning scales on ordinal observed (opposed to latent) variables comprise the Underlying Variable Approach (UVA) and Item Response Theory (IRT, Moustaki, 2000; Agresti, 2010). Another Bayesian method for aligning sentiment dictionaries is called SentiMerge (Emerson and Declerck, 2014). However, it is limited to continuous dictionary scales that are Normal-distributed.

There exists a plethora of extensions of the Latent Dirichlet Allocation (LDA, Wallach, 2006; Mcauliffe and Blei, 2007; Chang and Blei, 2009; Blei, 2012; Dieng et al., 2020). Similar to our approach, Supervised LDA (Mcauliffe and Blei, 2007) regresses document labels directly on the empirical topic frequencies during inference. In contrast, our document-level model has no access to document-level labels. Dieng et al. (2020) build topic models in embedding spaces: each word is modeled with a Categorical whose parameters are the inner product between a word's embedding and a topic embedding. Stoehr et al. (2023) present an ordering constraint on the topic-word type matrix $\mathbf{\Phi}$ to learn ordered topics based on ordered vocabularies.

8 Conclusion

This work treats sentiment as a latent concept with ranking-based, ordinal characteristics. Other ordinal phenomena such as pain perception (Griffin et al., 2020), conflict intensity (Stoehr et al., 2022) or political ideology (Vafa et al., 2020; Russo et al., 2022) can similarly be measured on ordinal scales. Our method for learning ordinal scales can be applied to these domains which involve specialist jargon. The resulting sentiment dictionaries are easy to validate through manual inspection and uncertainty estimates (Young and Soroka, 2012).

Acknowledgements

We would like to thank and acknowledge Lucas Torroba Hennigen, Josef Valvoda, Robert West as well as the anonymous reviewers for helpful comments. A special thank you to Alexander Hoyle for sharing code and data for the extrinsic evaluation. Niklas Stoehr is supported by a scholarship from the Swiss Data Science Center (SDSC).

Limitations

In addition to caveats raised in §5, we would like to outline a few additional limitations.

Sensitivity of Priors. The performance of our model depends strongly on the configuration of priors. Their sensitivity is caused, in part, by the ordering transform in Eq. (6). In all experiments, we consistently choose the following parameter setting: $\mu_k^c = -1.0$, $\sigma_k^c = 1.0$, $\gamma_k^c = 1.0$, $\eta_k^c = 1.0$, $\mu_k^d = -1.0$ and $\sigma_k^d = 1.0$. In §6, we set $\mu_k^d = -5.0$, $\sigma_k^d = 10.0$, $\mu_k^c = -5.0$, $\sigma_k^c = 10.0$, $\gamma_k^c = 1.0$ and $\eta_k^c = 10.0$. Details on the inference procedure and implementation are given in §3.3.

Number of Parameters. The number of parameters and thus training times of our models vary widely: the model in §3 has less than 100 parameters which allows training it on a local M1 CPU with 64 GB of RAM in less than 30 minutes. The number of parameters of the document-level models depends on the vocabulary size $|\mathcal{V}|$ and the number of latent classes K. In particular, the $K \times |\mathcal{V}|$-shaped matrix $\mathbf{\Phi}$ represents a limiting factor. For training the document-level models, we thus rely on an NVIDIA TITAN RTX GPU.

Language Limitation. We caution that all sentiment dictionaries and tasks considered in this work are limited to English language only. Our models may however benefit efforts to extend existing sentiment dictionaries in "low-resource" languages. We provide dataset statistics in App. A.1.

Impact Statement

We do not foresee ethical concerns with the research presented in this paper. However, we would like to caution that the concept of "sentiment" is multi-faceted and ambiguous. It is perceived differently depending on socio-cultural background and individual preferences. Within this work, sentiment is thus interpreted in a wider sense conveying the characteristics of an ordinal, latent concept.

References

Alan Agresti. 2010. *Analysis of ordinal categorical data*. Wiley Series in Probability and Statistics. John Wiley & Sons, Inc., Hoboken, NJ, USA.

Nabeela Altrabsheh, Mazen El-Masri, and Hanady Mansour. 2017. Combining sentiment lexicons of Arabic terms. In *AMCIS*.

Jisun An, Haewoon Kwak, and Yong-Yeol Ahn. 2018. SemAxis: A lightweight framework to characterize domain-specific word semantics beyond sentiment. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, volume 1.

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*. European Language Resources Association (ELRA).

Eli Bingham, Jonathan P. Chen, Martin Jankowiak, Fritz Obermeyer, Neeraj Pradhan, Theofanis Karaletsos, Rohit Singh, Paul Szerlip, Paul Horsfall, and Noah D. Goodman. 2018. Pyro: Deep universal probabilistic programming. *Journal of Machine Learning Research*.

David M. Blei. 2012. Probabilistic topic models. *Communications of the ACM*, 55(4):77–84.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, Bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 440–447.

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

Jonathan Chang and David Blei. 2009. Relational topic models for document networks. In *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, pages 81–88.

H. A. David and H. N. Nagaraja. 2003. *Order statistics*, 3rd ed edition. John Wiley, Hoboken, N.J.

Adji B. Dieng, Francisco J. R. Ruiz, and David M. Blei. 2020. Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453.

Guy Emerson and Thierry Declerck. 2014. SentiMerge: Combining sentiment lexicons in a Bayesian framework. In *Proceedings of Workshop on Lexical and Grammatical Resources for Language Processing*, pages 30–38. Association for Computational Linguistics.

Fabian Falck, Julian Marstaller, Niklas Stoehr, Sören Maucher, Jeana Ren, Andreas Thalhammer, Achim Rettinger, and Rudi Studer. 2020. Measuring proximity between newspapers and political parties: The Sentiment Political Compass. *Policy & Internet*, 12(3):367–399.

Jörg Friedrichs, Niklas Stoehr, and Giuliano Formisano. 2022. Fear-anger contests: Governmental and populist politics of emotion. *Online Social Networks and Media*, 32:100240.

Andrew Gelman, Jessica Hwang, and Aki Vehtari. 2014. Understanding predictive information criteria for Bayesian models. *Statistics and Computing*, 24(6):997–1016.

Andrew Gelman, Xiao-Li Meng, and Hal Stern. 1996. Posterior predictive assessment of model fitness via realized discrepancies. *Statistica Sinica*, 6(4):733–760.

Alex Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. In *Stanford Online*.

Robert S. Griffin, Maria Antoniak, Phuong Dinh Mac, Vladimir Kramskiy, Seth Waldman, and David Mimno. 2020. Imagined examples of painful experiences provided by chronic low back pain patients and attributed a pain numerical rating score. *Frontiers in Neuroscience*, 13:1331.

Johann Gründl. 2020. Populist ideas on social media: A dictionary-based measurement of populist communication. *New Media & Society*.

Jake M. Hofman, Duncan J. Watts, Susan Athey, Filiz Garip, Thomas L. Griffiths, Jon Kleinberg, Helen Margetts, Sendhil Mullainathan, Matthew J. Salganik, Simine Vazire, Alessandro Vespignani, and Tal Yarkoni. 2021. Integrating explanation and prediction in computational social science. *Nature*, 595(7866):181–188.

Matthew D. Homan and Andrew Gelman. 2014. The No-U-Turn Sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research*, 15(1):1593–1623.

Alexander Miserlis Hoyle, Lawrence Wolf-Sonkin, Hanna Wallach, Ryan Cotterell, and Isabelle Augenstein. 2019. Combining sentiment lexica with a multi-view variational autoencoder. In *Proceedings of the 2019 Conference of the North*, pages 635–640, Minneapolis, Minnesota. Association for Computational Linguistics.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the 2004 ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '04*, page 168. ACM Press.

Clayton Hutto and Eric Gilbert. 2014. VADER: A parsimonious rule-based model for sentiment analysis of social media text. *Proceedings of the International AAAI Conference on Web and Social Media*, 8(1):216–225.

Dongyeop Kang, Waleed Ammar, Bhavana Dalvi, Madeleine van Zuylen, Sebastian Kohlmeier, Eduard Hovy, and Roy Schwartz. 2018. A dataset of peer reviews (PeerRead): collection, insights and NLP applications. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1647–1661.

Svetlana Kiritchenko, Xiaodan Zhu, and Mohammad Saif. 2014. Sentiment analysis of short informal texts. *Journal of Artificial Intelligence Research*, 50:723–762.

Rensis Likert. 1932. *A technique for the measurement of attitudes*, volume 18. The Science Press.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150.

Fawaz H.H. Mahyoub, Muazzam A. Siddiqui, and Mohamed Y. Dahab. 2014. Building an Arabic sentiment lexicon using semi-supervised learning. *Journal of King Saud University - Computer and Information Sciences*, 26(4):417–424.

Jon Mcauliffe and David Blei. 2007. Supervised topic models. In *Advances in Neural Information Processing Systems*, volume 20.

Saif M. Mohammad. 2016. Sentiment analysis – Detecting valence, emotions, and other affectual states from text. In *Emotion Measurement*, pages 201–237. Elsevier.

Irini Moustaki. 2000. A latent variable model for ordinal variables. *Applied Psychological Measurement*, 24(3):211–223.

Kevin P. Murphy. 2012. *Machine learning: A probabilistic perspective*. MIT Press, Cambridge, MA.

Canberk Ozdemir and Sabine Bergler. 2015. A comparative study of different sentiment lexica for sentiment analysis of tweets. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, pages 488–496.

Elisavet Palogiannidi, Athanasia Kolovou, Fenia Christopoulou, Filippos Kokkinos, Elias Iosif, Nikolaos Malandrakis, Haris Papageorgiou, Shrikanth Narayanan, and Alexandros Potamianos. 2016. Tweester at SemEval-2016 task 4: Sentiment analysis in Twitter using semantic-affective model adaptation. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 155–163. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543.

Du Phan, Neeraj Pradhan, and Martin Jankowiak. 2019. Composable effects for flexible and accelerated probabilistic programming in NumPyro. *arXiv*, 1912.11554.

Robert Plutchik. 1980. A general psychoevolutionary theory of emotion. In *Theories of Emotion*, pages 3–33. Elsevier.

Danilo Jimenez Rezende and Shakir Mohamed. 2015. Variational inference with normalizing flows. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning*, pages 1530–1538.

Giuseppe Russo, Christoph Gote, Laurence Brandenberger, Sophia Schlosser, and Frank Schweitzer. 2022. Disentangling active and passive cosponsorship in the U.S. congress. *arXiv*, 2205.09674.

Fabian Stephany, Leonie Neuhäuser, Niklas Stoehr, Philipp Darius, Ole Teutloff, and Fabian Braesemann. 2022. The CoRisk-index: A data-mining approach to identify industry-specific risk perceptions related to Covid-19. *Nature Humanities and Social Sciences Communications*, 9(1):41.

Matthew Stephens. 2000. Dealing with label switching in mixture models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 62(4):795–809.

Stanley S. Stevens. 1946. On the theory of scales of measurement. *Science*, 103(2684):677–680.

Niklas Stoehr, Lucas Torroba Hennigen, Samin Ahbab, Robert West, and Ryan Cotterell. 2021. Classifying dyads for militarized conflict analysis. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Niklas Stoehr, Lucas Torroba Hennigen, Josef Valvoda, Robert West, Ryan Cotterell, and Aaron Schein. 2022. An ordinal latent variable model of conflict intensity. In *arXiv*, volume 2210.03971.

Niklas Stoehr, Benjamin J. Radford, Ryan Cotterell, and Aaron Schein. 2023. The Ordered Matrix Dirichlet for modeling ordinal dynamics. In *arXiv*, volume 2212.04130.

Philip J. Stone, Robert F. Bales, J. Zvi Namenwirth, and Daniel M. Ogilvie. 2007. The general inquirer: A computer system for content analysis and retrieval based on the sentence as a unit of information. *Behavioral Science*, 7(4):484–498.

Duyu Tang, Furu Wei, Bing Qin, Ming Zhou, and Ting Liu. 2014. Building large-scale Twitter-specific sentiment lexicon: A representation learning approach. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 172–182, Dublin, Ireland.

Tim Vieira. 2021. On the distribution function of order statistics.

Keyon Vafa, Suresh Naidu, and David M. Blei. 2020. Text-based ideal points. *Proceedings of the 2020 Conference of the Association for Computational Linguistics*, pages 5345–5357.

Hanna M. Wallach. 2006. Topic modeling: Beyond bag-of-words. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 977–984.

Leyi Wang and Rui Xia. 2017. Sentiment lexicon construction with representation learning based on hierarchical sentiment supervision. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 502–510.

Shuai Wang, Guangyi Lv, Sahisnu Mazumder, and Bing Liu. 2021. Detecting domain polarity-changes of words in a sentiment lexicon. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3657–3668.

Robert West, Hristo S. Paskov, Jure Leskovec, and Christopher Potts. 2014. Exploiting social network structure for person-to-person sentiment analysis. *Transactions of the Association for Computational Linguistics*, 2.

Tobias Widmann and Maximilian Wich. 2022. Creating and comparing dictionary, word embedding, and Transformer-based models to measure discrete emotions in German political text. *Political Analysis*, pages 1–16.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 347–354.

Jeffrey M. Wooldridge. 2010. *Econometric analysis of cross section and panel data*, 2nd edition. MIT Press.

Lori Young and Stuart Soroka. 2012. Affective news: The automated coding of sentiment in political texts. *Political Communication*, 29(2):205–231.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, pages 649–657. MIT Press.

A Appendix

A.1 Dictionary and Task Statistics

We present dictionary statistics in Tab. 3 and task statistics Tab. 4, adopted from Hoyle et al. (2019).

dictionary	source	V	scale
SC SenticNet	-	100,000	cont., bound.
SW SentiWordNet	WordNet	14,107	cont., bound.
VA Vader	Social Media	7489	cont., bound.
GI General Inquirer	-	4206	disc., binary
HL Hu-Liu	Reviews	6790	disc., binary
MP MPQA	News	4397	disc., binary

Table 3: Descriptive statistics of 6 popular sentiment dictionaries. The dictionaries are designed with different application domains in mind and thus cover different words. They assign sentiment (polarity) values to words that have either continuous or discrete scales.

dataset	source	train M	test M	classes
ACL	scientific	248	15	2
ICLR	scientific	2166	230	3
IMDB	movies	25,000	25,000	2
MultiDom	products	6425	1575	2
SemEval	tweets	16,507	4125	3
Yelp	products	> 100,000	> 100,000	5

Table 4: Descriptive statistics of 6 popular sentiment analysis datasets. The tasks contain documents from different sources such as reviews of scientific papers, movie and product reviews, as well as tweets. The tasks also differ in the number of different sentiment classes.

A.2 Evaluation Metrics

We evaluate our model using a scaled variant of the posterior predictive density (PPD) (Gelman et al., 1996, 2014):

$$\text{PPD} = \exp\left(\frac{1}{V}\sum_{n=1}^{V}\log\left(\frac{1}{T}\sum_{t=1}^{T}p(y_v \mid x_v, \theta^{(t)})\right)\right)$$

PPD measures the exponentiated averaged predictive log-likelihood. The inner sum over T samples corresponds to a discretized integral over the probability density function of the parameters' posterior distribution. $\exp\frac{1}{V}\sum_{v=1}^{V}\log(\cdot)$ represents the geometric mean over V data points. By exponentiating, our metric ranges between 0 and ∞. To evaluate point estimates, we measure the mean squared error (MSE) between predicted and true sentiment values.

A.3 Inverse of OrderedNormal

The OrderedNormal distribution, defined in Eq. (6), is based on an ordering transformation. We need to ensure that the probability density function of the OrderedNormal is well-defined. To this end, the transformation needs to be a smooth bijection where $\forall k,\ \lambda_k > \lambda_{k-1}$, so the log is well-defined.

$$s_k \leftarrow \begin{cases} \lambda_1 & \text{if } k = 1 \\ \log(\lambda_k - \lambda_{k-1}) & \text{if } k > 1 \end{cases} \tag{15}$$

A.4 Document-level Model Details

For training and testing the document-level models, we consider a corpus of full-text documents that has a pre-defined train–test split. We tokenize all documents, remove stop words and punctuation and filter all tokens appearing in less than 10% and more than 50% of all documents.

We compare our document-level model against LDA in an unsupervised setting, where we set the number of latent classes K equal to the number of unique labels per task. This allows treating a document's inferred latent topic $\dot{z}_m$ directly as a predicted document label.

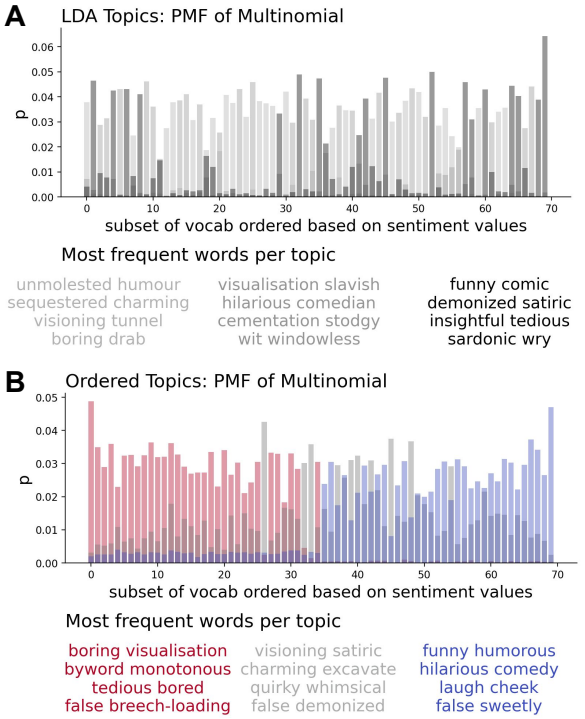

Figure 10: (**A**) LDA model fitted to documents of the Yelp dataset. (**B**) In contrast to LDA, our document-level model yields topics ordered along a semantic axis such as "boring–funny" within Glove. For visualization purposes, we consider only few posterior samples.