Semantic Annotation Aggregation with Conditional Crowdsourcing Models and Word Embeddings

Paul Felt
IBM Watson*
plfelt@us.ibm.com

Eric K. Ringger
Facebook*
eringger@fb.com

Kevin Seppi
Brigham Young University
kseppi@byu.edu

Abstract

In modern text annotation projects, crowdsourced annotations are often aggregated using item response models or by majority vote. Recently, item response models enhanced with generative data models have been shown to yield substantial benefits over those with conditional or no data models. However, suitable generative data models do not exist for many tasks, such as semantic labeling tasks. When no generative data model exists, we demonstrate that similar benefits may be derived by conditionally modeling documents that have been previously embedded in a semantic space using recent work in vector space models. We use this approach to show state-of-the-art results on a variety of semantic annotation aggregation tasks.

1 Introduction

Text annotation is a crucial part of natural language processing (NLP), enabling content analysis (Krippendorff, 2012) and providing training data for supervised and semi-supervised machine learning algorithms in NLP. Modern text annotation is often crowdsourced, meaning that the work is divided up and assigned to internet workers on micro-task marketplaces such as Amazon's Mechanical Turk[1] or CrowdFlower.[2] Although crowdsourced annotations tend to be error-prone, high quality labels may be obtained by aggregating multiple redundant low-quality annotations (Surowiecki, 2005). For many tasks, aggregated crowdsourced judgments have been shown to be more reliable than expert judgments (Snow et al., 2008; Cao et al., 2010; Jurgens, 2013).

Traditionally annotations were aggregated via majority vote. More sophisticated approaches jointly model annotator reliability and document labels. These models can down-weight the annotations of workers who often disagree with others and up-weight the annotations of workers who often agree with others. However, when annotation error is high or few annotations are available it can be difficult for these models to know which annotators to trust. Jin and Ghahramani (2002), Raykar et al. (2010), Liu et al. (2012), and Yan et al. (2014) show that crowdsourcing annotation models can be enhanced by conditioning the model on the document data (e.g., word content), improving the model by identifying annotators whose judgments tend to agree with word patterns found in the documents being annotated.

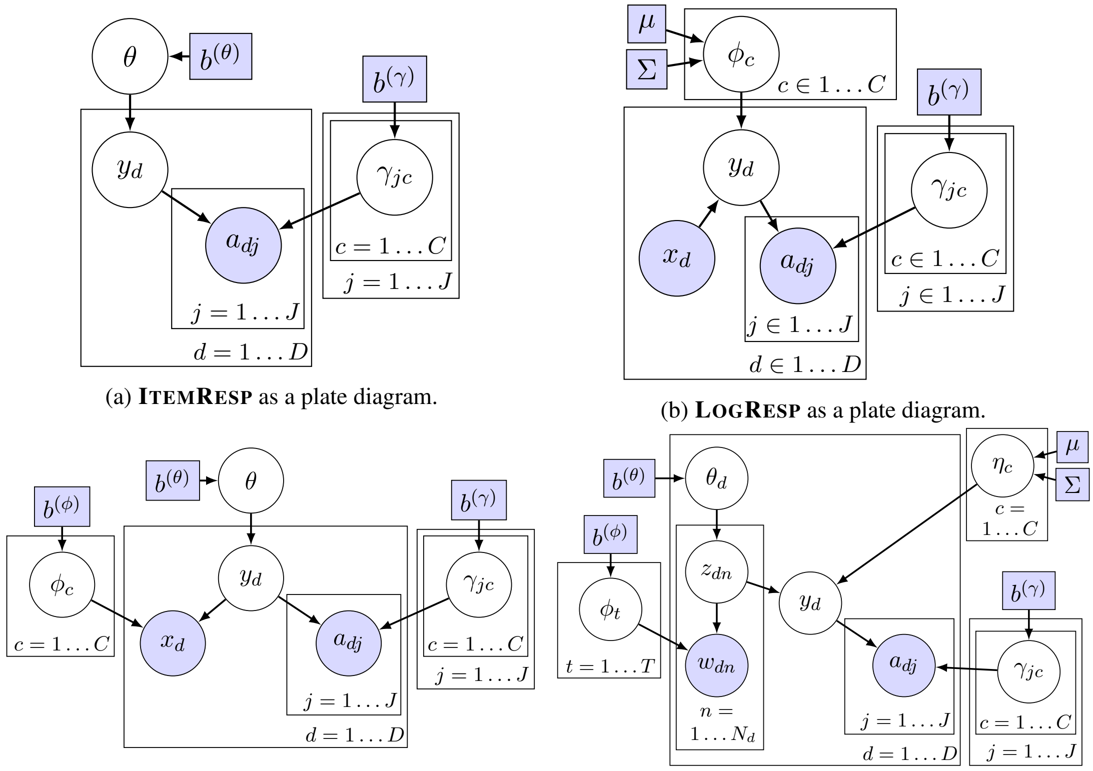
Recent work has shown that generative data models allow crowdsourcing models to converge to useful estimates even when few annotations are available, whereas by the time a conditional model has enough information (in the form of annotations) to be useful, the problem is often largely solved by majority vote (Felt et al., 2015b). However, although generative data modeling has been shown to be effective in categorizing text according to its topic, realistic generative data models are not always available. In this paper, we use advances in text representation to demonstrate that data-conditional annotation models can

* This work was completed while the first and second authors were at Brigham Young University.

[1] http://mturk.com

[2] http://crowdflower.com

Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 1787–1796, Osaka, Japan, December 11-17 2016.



(a) **ITEMRESP** as a plate diagram.

(b) **LOGRESP** as a plate diagram.

(c) **MOMRESP** as a plate diagram.

(d) **CSLDA** as a plate diagram.

Figure 1: Round nodes are variables with distributions. Rectangular nodes are values without distributions. Shaded nodes are observed. D, J, C, and T are the number of documents, annotators, classes, and topics, respectively. N_d is the number of words in document d.

achieve gains similar to those of data-generative annotation models, including for tasks where generative data models are currently unavailable, such as paired text similarity and compatibility.

In Section 2 we briefly review annotation models with generative and conditional data components and also discuss representing words and documents via embeddings in a semantic vector space. In Section 3 we show that data-conditional annotation models succeed on a variety of text datasets and classification tasks. In Section 4 we conduct error analysis on an anomalous dataset, and in Sections 5 and 6 we list additional related work and summarize our conclusions.

2 Background

Most crowdsourcing models extend the item-response model of Dawid and Skene (1979). The Bayesian version of this model, referred to here as ITEMRESP, is illustrated by Figure 1a and defines the joint distribution $p(y, a, \theta, \gamma)$, where a is the annotation and y is an unobserved document label. In the generative story for this model, a confusion matrix γ_j is drawn for each human annotator j. Each row γ_{jc} of the confusion matrix γ_j is drawn from $Dir(b^{(\gamma)}_{jc})$, and encodes a probability distribution over label classes that annotator j is apt to choose when presented with a document whose true label is c. A general prior over label classes θ is drawn from $Dirichlet(b^{(\theta)})$, then for each document d an unobserved document label y_d is drawn from categorical distribution $Cat(\theta)$. Finally, annotations are generated as annotator j corrupts the true label y_d according to the multinomial distribution $Mult(\gamma_{jy_d})$.

2.1 Data-aware annotation models

Notice that the ITEMRESP model entirely ignores document data x (e.g., words). ITEMRESP extensions model the data x and related feature parameters ϕ either conditionally $p(y, a, \gamma, \phi|x)$ or else generatively $p(y, a, x, \theta, \gamma, \phi)$.

Conditional crowdsourcing models make few assumptions about the data and can use the same general log-linear structure as maximum entropy classifiers, which have enjoyed success in a large number of classification problems. Figure 1b shows a Bayesian formulation of a conditional crowdsourcing model $p(y, a, \gamma, \phi|x)$. For each class k, ϕ_k is drawn from a multivariate Gaussian distribution $Gauss(0, \Sigma)$. Then for each document, y_d is drawn from a log-linear distribution $p(y_d|\phi, x) \propto e^{\phi_{y_d}^T x}$. For this reason we refer to this model as LOGRESP. LOGRESP is representative of a popular class of conditional crowdsourcing models (Jin and Ghahramani, 2002; Raykar et al., 2010; Liu et al., 2012; Yan et al., 2014). In previous work we found that LOGRESP often provides only incremental gains over majority vote (Felt et al., 2015b). This partly because its ϕ estimates, like other conditional models, tend to converge relatively slowly with $O(N)$ labeled examples (Ng and Jordan, 2001). By the time LOGRESP's ϕ estimates become useful, there are often enough annotations available that majority vote is sufficient.

Generative data-aware crowdsourcing models have complementary strengths. Although they make strong assumptions about the data that they model, their parameters can converge quickly with only $O(logn)$ labeled examples (Ng and Jordan, 2001). This means that when data does not violate a generative model's assumptions too badly, the generative model can offer dramatic improvements over majority vote, especially when few annotations are available. Figures 1c and 1d depict two such generative models. In Figure 1c, each document d draws its data x_d from a class-conditional multinomial word distribution $Mult(\phi_{y_d})$. We call this model MOMRESP because it models data as a mixture of multinomials. MOMRESP represents a common class of generative crowdsourcing models (Bragg et al., 2013; Lam and Stork, 2005; Simpson and Roberts, 2015). Figure 1d shows CSLDA, a more sophisticated generative crowdsourcing model based on supervised topic modeling (Felt et al., 2015a).

For inference in the ITEMRESP, LOGRESP and MOMRESP crowdsourcing models, we use existing variational inference (Felt et al., 2015b). Note that variational inference for ITEMRESP is easily derived as a special case of MOMRESP inference where terms involving the data are dropped. Inference for CSLDA is stochastic expectation maximization.

2.2 Word and Document Representations

Documents have historically been represented in NLP algorithms by large, sparse word count vectors $x_d = \sum_{n=1}^{|x_d|} \mathbb{1}(x_{dn})$ where $\mathbb{1}(x_{dn})$ is a one-hot vector having length equal to the size of the vocabulary. However, word-count document representations have a number of drawbacks. They define a space that is often so high-dimensional and sparse that inter-document distances and other vector computations have little meaning. In word-count representations, features that strongly relate to one another (e.g., the words "horse" and "equine") are represented as entirely orthogonal dimensions, exploding the number of parameters needed by downstream learning algorithms.

Recently, methods have been developed to represent words as locations in low-dimensional vector spaces where distance and direction encode semantic and syntactic meaning (Mikolov et al., 2013a; Pennington et al., 2014). These embedding vectors have been shown to improve a variety of language tasks including named entity recognition, phrase chunking (Turian et al., 2010), relation extraction (Nguyen and Grishman, 2014), and part of speech induction (Lin et al., 2015). The hypothesis investigated by the current work is that semantic, vector-based text representations can help conditional annotation aggregation models achieve some of the same early performance advantage seen in their generative counterparts, as well as help them operate on datasets that make semantic distinctions. This hypothesis is plausible *a priori* because using data embeddings is akin to using semi-supervision to enable faster learning. The reason for this is that data embeddings are traditionally induced in an unsupervised manner on extremely large corpora before being applied to a downstream supervised task. In addition, operating on dense, low-dimensional vector data reduces the number of model parameters which can also reduce the number of instances required to learn effectively.

Although it might be possible to extend the CSLDA model to generatively model the embedding as described by Das et al. (2015), but it is unclear how the inference approach used there (Gibbs sampling based on Cholesky decompositions) would be efficiently applied in the context of the CSLDA model, thus we leave this possibility to future work and focus on using embeddings discriminatively. We use the

Dataset	Size	Unique Annotators	Annotations per Instance	Classes	Average Doc size	Gold Labels	Timestamps
Sentiment	1,000	83	5	2	12.8	1,000	No
Weather	1,000	102	20	5	13.6	724	Yes
Compatibility	17,977	411	10	2	2 × 1	15,157	No
Paraphrase	4,000	119	5	2	2 × 11.2	838	Yes

Table 1: Dataset statistics. Evaluation metrics are calculated only over the subset of each dataset for which gold labels are available. The timestamps column indicates whether or not it is known exactly when each annotation was generated. When available, timestamps determine the order of annotation in reported learning curves.

word2vec algorithm introduced by Mikolov et al. (2013a) to convert words to vectors for the purposes of this paper, understanding that other text embedding methods may be swapped in for additional improvements as they are developed. Word2vec operates on individual words. When sentences or documents must be vectorized, we do so by averaging the vectors of each word in the sentence or document without any word filtering or selection. While we briefly experimented with gensim's *doc2vec* implementation of Le and Mikolov (2014), we noticed little benefit for the twitter data explored in this paper, possibly because of the short, focused nature of tweets.

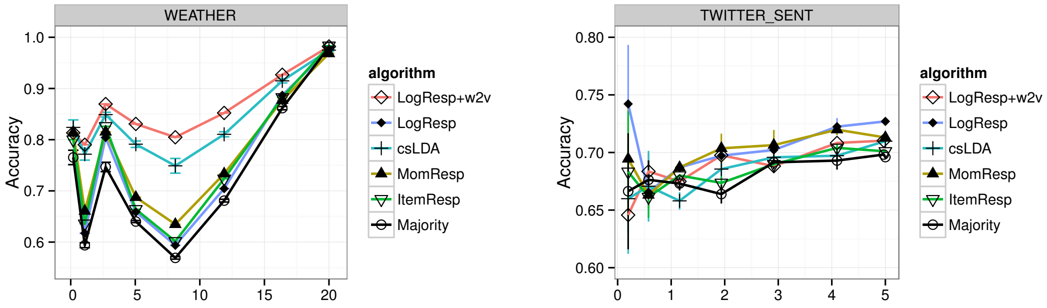
3 Experiments

In order to test the hypothesis that vector space document representations can improve conditional crowdsourcing model performance on semantic classification tasks, we plot and visually compare learning curves charting the accuracy of the labels inferred by various crowdsourcing models. All of the algorithms from Section 2 are trained on sparse one-hot vector representations of text; and an additional variant of LOGRESP is reported which is trained on low dimensional semantic vector representations (LOGRESP+w2v). Learning curves advance as annotations from multiple annotators are incrementally added to the set of annotations available to each model. When annotation timestamps are available, annotations are added in the empirical order in which they were created. When unavailable, annotations are added in randomized breadth-first order so that each document gets one annotation before any document receives a second. Accuracy is computed over the subset of gold labels having at least one annotation. This process illustrates model behavior both when few annotations per document are available (in the early stages of learning curves) and when many annotations per document are available (in the late stages of the learning curves).

For our word embedding model, we use the word2vec algorithm, implemented by the gensim document processing library (Řehůřek and Sojka, 2010) to train word embeddings on a June 2015 snapshot of the English Wikipedia pages and articles dump (approximately 2.1 billion words). The word2vec algorithm requires a number of parameters, which we report here for replicability. We train embeddings using a context window of 10 words, discarding words that occur fewer than 5 times. For training, we use hierarchical sampling with a skip-gram model and no negative sampling. Embeddings of size 300 are learned. All of these settings are rather standard for a large corpus like Wikipedia.

3.1 Datasets

In order to calculate the accuracy of inferred labels, we require datasets that have both crowdsourced annotations as well as gold standard labels for evaluation. We identify four suitable datasets, briefly describing both their annotation task as well as the way their gold standard labels are constructed. For all Twitter data, we use the Twitter text normalization dictionary of Han et al. (2012) to normalize tweets before embedding them. Note that for two of the datasets described below, **Compatibility** and **Weather**, the gold standard does not consist of the hand labels of an expert, but rather is constructed from the consensus vote of a reasonable number of crowd workers. A fundamental tenet of crowdsourcing is that



(a) The **Weather** Dataset

(b) The **Sentiment** Dataset

Figure 2: Inferred label accuracy (y axis) learning curves of various crowdsourcing models. The x axis is the number of annotations × 1,000.

inexpert workers are, in aggregate, trustworthy. The purpose of automatic aggregation models such as those decribed in Section 2 is to arrive at the same judgment with a few annotations that a simpler scheme like majority vote would have arrived at given many annotations. Therefore, a crowd-constructed gold standard is appropriate for evaluation of such models.

Paraphrase. During an exploratory annotation phase, Xu et al. (2014) paid Amazon Mechanical Turk workers to annotate 4,000 tweet pairs with binary judgments indicating whether or not the tweet pair communicates the same information.[3] For example, the tweets "Star Wars Return of the Jedi is on" and "My favorite Star Wars movie is on" communicate mostly the same information and are labeled as paraphrases of one another, while the tweet "and of course because I drink and like Star Wars I know nothing about football" communicates different information, and is labeled as not a paraphrase of the other two tweets. Each tweet pair received 5 binary annotations. Gold standard labels were constructed for a subset of 838 tweet pairs by experts who rated each pair on a scale from 0-5. Following the original authors, expert ratings of 0-2 are labeled *no paraphrase*, and 4-5 are labeled *paraphrase*. Ratings of 3 are ignored for evaluation purposes.

Compatibility. Kruszewski and Baroni (2015) paid CrowdFlower workers to rate word pairs according to their semantic compatibility, meaning that the two words can be used to refer to the same real-world entity.[3] For example, the words "artist" and "teacher" are compatible with one another, whereas "bread" and "rattlesnake" are not. Each word pair was rated by 10 different annotators on a 7-point scale. Following the original authors, the gold standard is constructed by labeling items with a mean rating less than 1.6 as incompatible, and those with a mean rating greater than 3.7 as compatible. Ratings between 1.7 and 3.7 are ignored for evaluation purposes.

Sentiment. Mozafari et al. (2014) paid Mechanical Turk workers to annotate tweets with binary sentiment labels: *Positive* and *Negative*, and manually created gold standard labels using trusted (non-crowdsourced) labelers.[4]

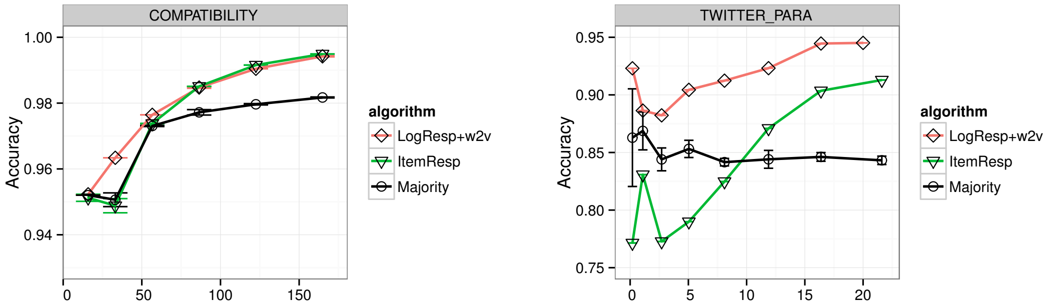
Weather. CrowdFlower has made a number of annotated datasets freely available.[5] In their "Weather sentiment" dataset, 20 annotators were paid to annotate weather-related tweets with sentiment labels: *Negative*, *Neutral*, *Positive*, *Unrelated to weather*, and *I can't tell*. A gold standard was constructed by running a separate evaluation task called "Weather sentiment evaluated" in which 10 additional annotators were paid to annotate the majority vote label from the previous task as correct or incorrect. We form a gold standard from those labels that are judged to be correct by at least 9/10 annotators.

Our focus in this paper is on challenging sentiment classification tasks which tend to have few classes. In preliminary experiments we observed that even on topical classification datasets such as 20 Newsgroups or the LDC-labeled Enron emails, LOGRESP is perceptibly improved by running on vector space features, although CSLDA remains dominant. Note that in the experiments described below, the **Weather** and **Paraphrase** annotations are applied in the order indicated in the dataset, however, the ac-

[3] Not publicly available at the time of writing.

[4] `http://web.eecs.umich.edu/~mozafari/datasets/crowdsourcing/index.html`

[5] `http://www.crowdflower.com/data-for-everyone`



(a) The **Compatibility** Dataset

(b) The **Paraphrase** Dataset

Figure 3: Inferred label accuracy (y axis) learning curves of vector space crowdsourcing models on tasks with paired-comparison data for which generative crowsdsourcing models are unsuitable. The x axis is the number of annotations × 1,000.

tual order of **Sentiment** and **Compatibility** annotations is not provided in the data set so they are applied in random order.

3.2 Comparison with generative methods

Two datasets, **Weather** and **Sentiment**, are traditional text classification tasks with instances consisting of one label per text document. We use these datasets to compare the performance of LogResp trained on vector space text features (LogResp+w2v) to the performance of alternatives using sparse word-count features, including LogResp as well as the generative models MomResp and csLDA. The majority vote and ItemResp algorithms serve as baselines. In Figure 2a we see that on the **Weather** dataset, LogResp with embeddings (LogResp+w2v) performs far better than traditional LogResp, and even outperforms the previous state-of-the-art for this dataset, csLDA. Although all algorithms eventually reach a high level of performance on the **Weather** dataset, we prefer algorithms like LogResp+w2v that reach high levels of accuracy using as few annotations as possible, potentially reducing annotation cost. The accuracy of all of the models is unstable until a reasonable number of annotations is obtained, 5,000 to 10,000 in this case. Models which make little or no use of the words themselves (especially Majority vote and ItemResp) are particularly susceptible to variability in the initial annotations. Data-sensitive models like csLDA and LogResp+w2v are far less susceptible to these swings.

In Figure 2b we see that no algorithm improves much over majority vote on the **Sentiment** dataset. Also, the baseline accuracy levels at the end of the curves are extremely low for a binary classification task with 5 annotations per instance, meaning that annotator accuracy is unusually low. We include this dataset as a reminder that the “no free lunch” theorem applies to crowdsourcing models the same as to any other class of models. In Section 4 we explore in more detail what makes the **Sentiment** dataset particularly difficult for crowdsourcing models.

3.3 When generative methods are unavailable

The datasets **Compatibility** and **Paraphrase** both involve data pairs being compared for semantic content (see Section 3.1 for examples of these tasks). **Compatibility** compares the semantic compatibility of word pairs while **Paraphrase** compares the semantic similarity of tweets pairs. Generative crowdsourcing models such as MomResp and csLDA do not natively accommodate such paired data since the data does not comport with these models’ generative stories. To make them do so would require restructuring the models and their inference procedures. On the other hand, it is straightforward to combine the semantic vector representations of two documents v_1 and v_2. We do so by forming a new feature vector

$$\begin{aligned} v_{new} = &\langle cos(v_1, v_2), \\ &L1(v_1 - v_2), L2(v_1 - v_2), \\ &PC_{50}(v_1), PC_{50}(v_2), \\ &PC_{50}(v_1) - PC_{50}(v_2)\rangle \end{aligned}$$

		Alternative Gold Standard			
		Neg	Pos	None	Hard
Gold	Neg	35	**6**	**5**	**1**
Standard	Pos	**9**	29	**11**	**3**

Table 2: Disagreement between the original gold standard (rows) and an alternative gold standard (columns) on 100 arbitrarily selected tweets. The alternative gold standard employs a more flexible label set. Neg=*Negative*, Pos=*Positive*, None=*No sentiment*, Hard=*Can't decide*. Bold values reflect various kinds of disagreement between the labelings.

where $cos(\cdot)$ is cosine distance, $L1(\cdot)$ and $L2(\cdot)$ are the first two p-norms, and $PC_n(v)$ is a vector consisting of the top-n components of v, found via PCA on the set of embedded documents.

Figure 3a shows that LogResp with semantic embeddings outperforms majority vote and ItemResp baselines on the word **Compatibility** dataset when there are fewer than 3 annotations per instance. Later in the learning curve, when annotations become sufficiently abundant (up to 20 per instance), the data appears to no longer be helpful. Fortunately, incorporating data information using LogResp with semantic embeddings appears to never actually hurt compared with using just ItemResp.

On the other hand, Figure 3b shows that on the **Paraphrase** dataset, LogResp with semantic embeddings dramatically outperforms the baselines along the entire learning curve. This is partly because, unlike the **Compatibility** task, **Paraphrase** accuracy is low enough to permit the improved vector data representation to benefit LogResp.

3.4 Summary of experiments

Overall, with the exception of the somewhat anomalous **Sentiment** dataset, which we examine in more detail in Section 4, running LogResp on semantically embedded data is always an improvement over LogResp running on traditional document representations. The gains in Figures 3a and 3b strongly confirm the hypothesis that semantic embeddings can allow crowdsourcing models to see some of the same efficiency gains for challenging semantic labeling tasks as previously observed using generative data-aware crowdsourcing models on more straight-forward topical labeling tasks. Not only that, but the fact that semantic embeddings lend themselves to sensible vector-space operations allows data-aware crowdsourcing models to be applied to complex tasks like labeling paired text similarity and compatibility, which was not previously possible.

4 Sentiment dataset error analysis

An analysis of the **Sentiment** dataset (results in Figure 2b) helps explain why algorithms behave so differently on it than on the other datasets. Kilgarriff (1998) identifies three sources of annotation noise: inherent data ambiguity, poor task definition, and annotator error. The crowdsourcing models used here account only for annotator error. However, the **Sentiment** dataset task definition dictates that each tweet be labeled with a binary sentiment label, forcing annotators to make arbitrary decisions when tweets encode little or ambiguous sentiment. For example, the tweet "EBTM.com is BACK?!" is genuinely ambiguous, and the tweet "@comeagainjen if you dont, neither do i" contains little explicit sentiment. Kilgarriff (1998) suggests that an important step towards addressing data ambiguity is ensuring that tasks are defined so that annotators have the ability to explicitly identify ambiguous instances.

To assess the impact of inherent data ambiguity and task definition on the **Sentiment** dataset, we arbitrarily chose 100 instances with gold labels and compared them with an alternative gold standard labeled according to a more flexible annotation scheme. The latter labels were generated by a pair of graduate students working in tandem. We added a *No sentiment* label to address problems with task definition and a *Can't decide* label to capture inherent data ambiguity. Table 2 shows the confusion matrix between the two gold standard sets. A large percentage (16%) of tweets were assigned to *No sentiment* in the alternative gold standard. This indicates that task definition affects this dataset strongly.

Although this analysis does not make this dataset any less interesting (indeed, the problems associated with modeling and correcting the effects of task misspecification are highly interesting), it does warn us that it is less representative than the others of annotation projects where effort is made up-front to iteratively refine an annotation specification before paying for large number of annotations.

5 Additional Related Work

A sizable body of research is currently underway to improve vector word representations. Although most commonly word embeddings are trained in an unsupervised manner, they may be tuned to maximize performance on a particular target task (Le and Mikolov, 2014). They may also be supervised by multiple tasks simultaneously (Collobert et al., 2011). Others fit one embedding per word sense rather than per lexical type, improving model fit (Neelakantan et al., 2014). Srikumar and Manning (2014) embed not only word types, but also label types, modeling the fact that some labels are more similar than others.

Another line of work explores ways of embedding larger spans of text. Although words tend to compose surprisingly well simply via linear combination, many phrases are more than the sum of their parts (e.g., collocations like "White House"). These can be dealt with by using heuristics to identify and combine token phrases (Mikolov et al., 2013b). Other approaches incorporate composition functions as first-class constituents of the objective function itself. Mitchell and Lapata (2010) motivate a general composition framework and compare a number of simple instantiations, including additive, multiplicative, and tensor product combination. Socher et al. (2012) assign vectors representing semantic content and matrices representing semantic transformations to every node in a parse tree. Fyshe et al. (2015) focus on learning phrasal representations whose dimensions are easily interpretable by humans, similar to successful models whose topics are easy for humans to recognize and name because they align with a topic distinction known *a priori* to the human.

In this work we focus on using instance data to improve probabilistic crowdsourcing models. Passonneau and Carpenter (2014) argue that probabilistic crowdsourcing models are generally more effective and reliable than traditional chance-adjusted agreement heuristics such as Krippendorff's alpha for assessing corpus quality (Krippendorff, 2012). Other previous work in crowdsourcing ignores the data being annotated, focusing instead on modeling other aspects of the annotation process, such as item difficulty and noise (Whitehill et al., 2009; Welinder et al., 2010). Hovy et al. (2013) model the non-linear nature of human reliability by adding binary variables to each annotator indicating whether they are a spammer or not. These extensions are orthogonal to the issue explored by this paper and could be incorporated into any of the models used here.

6 Conclusions and Future Work

Previous work indicates that generative crowdsourcing models enjoy significant learning advantages when aggregating topic-based document labels. Unfortunately, some text classification tasks make distinctions for which no good generative text models currently exist, such as labeling the similarity or compatibility of paired words and sentences. We have demonstrated that vector space text embeddings can be used to gain similar advantages using conditional models and for an even broader class of data. Using this approach, we have shown state-of-the-art annotation aggregation for several semantic annotation aggregation tasks. Future work includes experimenting with deep learning methods of jointly learning embeddings and hidden labels, rather than pipelining the two tasks.

Acknowledgments

This work was supported by the collaborative NSF Grant IIS-1409739 (BYU) and IIS-1409287 (UMD). Any opinions, findings, conclusions, or recommendations expressed here are those of the authors and do not necessarily reflect the view of the sponsor.

References

Jonathan Bragg, Mausam, and Daniel S. Weld. 2013. Crowdsourcing multi-label classification for taxonomy creation. In *Proc. Conference on Human Computation and Crowdsourcing (HCOMP)*.

Jing Cao, S Lynne Stokes, and Song Zhang. 2010. A Bayesian approach to ranking and rater evaluation an application to grant reviews. *Journal of Educational and Behavioral Statistics*, 35(2):194–214.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537.

Rajarshi Das, Manzil Zaheer, and Chris Dyer. 2015. Gaussian lda for topic models with word embeddings. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 795–804, Beijing, China, July. Association for Computational Linguistics.

Alexander P. Dawid and Allan M. Skene. 1979. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Applied Statistics*, pages 20–28.

Paul Felt, Eric K. Ringger, Jordan Boyd-Graber, and Kevin Seppi. 2015a. Making the most of crowdsourced document annotations: Confused supervised LDA. In *Proc. Conference on Computational Natural Language Learning (CoNLL)*.

Paul Felt, Eric K. Ringger, Kevin Seppi, and Robbie A. Haertel. 2015b. Early gains matter: A case for preferring generative over discriminative crowdsourcing models. In *Proc. Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Alona Fyshe, Leila Wehbe, Partha P. Talukdar, Brian Murphy, and Tom M. Mitchell. 2015. A compositional and interpretable semantic space. In *Proc. Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Bo Han, Paul Cook, and Timothy Baldwin. 2012. Automatically constructing a normalisation dictionary for microblogs. In *Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Dirk Hovy, Taylor Berg-Kirkpatrick, Ashish Vaswani, and Eduard H. Hovy. 2013. Learning whom to trust with MACE. In *Proc. Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Rong Jin and Zoubin Ghahramani. 2002. Learning with multiple labels. In *Proc. Advances in Neural Information Processing Systems (NIPS)*.

David Jurgens. 2013. Embracing ambiguity: A comparison of annotation methodologies for crowdsourcing word sense labels. In *Proc. Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Adam Kilgarriff. 1998. Gold standard datasets for evaluating word sense disambiguation programs. *Computer Speech & Language*, 12(4):453–472.

Klaus Krippendorff. 2012. *Content Analysis: An Introduction to its Methodology*. Sage.

Germán Kruszewski and Marco Baroni. 2015. So similar and yet incompatible: Toward the automated identification of semantically compatible words. In *Proc. Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Chuck P. Lam and David G. Stork. 2005. Toward optimal labeling strategy under multiple unreliable labelers. In *Proc. AAAI Spring Symposium: Knowledge Collection from Volunteer Contributors*.

Quoc V. Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proc. International Conference on Machine Learning (ICML)*.

Chu-Cheng Lin, Waleed Ammar, Chris Dyer, and Lori Levin. 2015. Unsupervised POS induction with word embeddings. In *Proc. Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Qiang Liu, Jian Peng, and Alex T. Ihler. 2012. Variational inference for crowdsourcing. In *Proc. Advances in Neural Information Processing Systems (NIPS)*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *Proc. Workshops at International Conference on Learning Representations (ICLR)*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems (NIPS)*.

Jeff Mitchell and Mirella Lapata. 2010. Composition in distributional models of semantics. *Cognitive Science*, 34(8):1388–1429.

Barzan Mozafari, Purna Sarkar, Michael Franklin, Michael Jordan, and Samuel Madden. 2014. Scaling up crowd-sourcing to very large datasets: a case for active learning. In *Proc. Very Large Databases (VLDB) Conference*.

Arvind Neelakantan, Jeevan Shankar, Alexandre Passos, and Andrew McCallum. 2014. Efficient non-parametric estimation of multiple embeddings per word in vector space. In *Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Andrew Y. Ng and Michael I. Jordan. 2001. On discriminative vs. generative classifiers: A comparison of logistic regression and Naive Bayes. In *Proc. Advances in Neural Information Processing Systems (NIPS)*.

Thien Huu Nguyen and Ralph Grishman. 2014. Employing word representations and regularization for domain adaptation of relation extraction. In *Proc. Annual Meeting of the Association for Computational Linguistics (ACL)*.

Rebecca J. Passonneau and Bob Carpenter. 2014. The benefits of a model of annotation. *Transactions of the Association for Computational Linguistics*, 2:311–326.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proc. Empiricial Methods in Natural Language Processing (EMNLP 2014)*.

Vikas C. Raykar, Shipeng Yu, Linda H. Zhao, Gerardo Hermosillo Valadez, Charles Florin, Luca Bogoni, and Linda Moy. 2010. Learning from crowds. *The Journal of Machine Learning Research*, 11:1297–1322.

Radim Řehůřek and Petr Sojka. 2010. Software framework for topic modelling with large corpora. In *Proc. LREC Workshop on New Challenges for NLP Frameworks*.

Edwin Simpson and Stephen Roberts. 2015. Bayesian methods for intelligent task assignment in crowdsourcing systems. In *Decision Making: Uncertainty, Imperfection, Deliberation and Scalability*, pages 1–32. Springer.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y. Ng. 2008. Cheap and fast—but is it good?: Evaluating non-expert annotations for natural language tasks. In *Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Richard Socher, Brody Huval, Christopher D. Manning, and Andrew Y. Ng. 2012. Semantic compositionality through recursive matrix-vector spaces. In *Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Vivek Srikumar and Christopher D. Manning. 2014. Learning distributed representations for structured output prediction. In *Advances in Neural Information Processing Systems 27*, pages 3266–3274.

James Surowiecki. 2005. *The Wisdom of Crowds*. Random House LLC.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In *Proc. Annual Meeting of the Association for Computational Linguistics (ACL)*.

Peter Welinder, Steve Branson, Pietro Perona, and Serge J. Belongie. 2010. The multidimensional wisdom of crowds. In *Proc. Advances in Neural Information Processing Systems (NIPS)*.

Jacob Whitehill, Ting-fan Wu, Jacob Bergsma, Javier R Movellan, and Paul L. Ruvolo. 2009. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *Proc. Advances in Neural Information Processing Systems (NIPS)*.

Wei Xu, Alan Ritter, Chris Callison-Burch, William B. Dolan, and Yangfeng Ji. 2014. Extracting lexically divergent paraphrases from Twitter. *Transactions of the Association for Computational Linguistics*, 2:435–448.

Yan Yan, Rómer Rosales, Glenn Fung, Ramanathan Subramanian, and Jennifer Dy. 2014. Learning from multiple annotators with varying expertise. *Machine Learning*, 95(3):291–327.