Learning Emphasis Selection for Written Text in Visual Media from Crowd-Sourced Label Distributions

Amirreza Shirani[†], Franck Dernoncourt[‡], Paul Asente[‡], Nedim Lipka[‡], Seokhwan Kim[§], Jose Echevarria[‡] and Thamar Solorio[†]

[†]University of Houston [‡]Adobe Research [§]Amazon Alexa AI

[†]{ashirani,tsolorio}@uh.edu

[‡]{dernonco,asente,lipka,echevarr}@adobe.com

[§]seokhwk@amazon.com

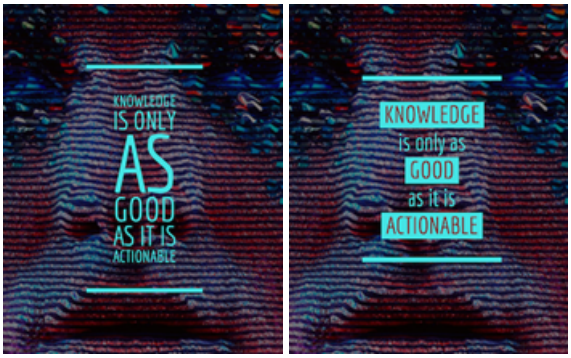
Abstract

In visual communication, text emphasis is used to increase the comprehension of written text and to convey the author's intent. We study the problem of emphasis selection, i.e. choosing candidates for emphasis in short written text, to enable automated design assistance in authoring. Without knowing the author's intent and only considering the input text, multiple emphasis selections are valid. We propose a model that employs end-to-end label distribution learning (LDL) on crowd-sourced data and predicts a selection distribution, capturing the inter-subjectivity (common-sense) in the audience as well as the ambiguity of the input. We compare the model with several baselines in which the problem is transformed to single-label learning by mapping label distributions to absolute labels via majority voting.

1 Introduction

Visual communication relies heavily on images and short texts. Whether it is flyers, posters, ads, social media posts or motivational messages, it is usually highly designed to grab a viewer's attention and convey a message in the most efficient way. For text, word emphasis is used to capture the intent better, removing the ambiguity that may exist in some plain texts. Word emphasis can clarify or even change the meaning of a sentence by drawing attention to some specific information. It can be done with colors, backgrounds, or fonts, or with styles like italic and boldface.

Some graphic design applications such as Adobe Spark[1] perform automatic text layout using templates that include images and text with different fonts and colors. However, their text layout algorithms are mainly driven by visual attributes like word length, rather than the semantics of the text or the user's intent, which can lead to unintended emphasis and the wrong message. Figure 1a shows an example that is aesthetically appealing but fails to effectively communicate its intent. Understanding the text would allow the system to propose a different layout that emphasizes words that contribute more to the communication of the intent, as shown in Figure 1b.



(a) (b)

Figure 1: Two different text layouts emphasizing different parts of the sentence.

We investigate models that aim to understand the most common interpretation of a short piece of text, so the right emphasis can be achieved automatically or interactively. The ultimate goal is to enable design assistance for the user during authoring. The main focus is on *short* text instances for social media, with a variety of examples from inspirational quotes to advertising slogans. We model emphasis using plain text with no additional context from the user or the rest of the design.

This task differs from related ones in that word emphasis patterns are person- and domain-specific, making different selections valid depending on the audience and the intent. For example, in Figure 1b, some users might prefer to just emphasize "knowledge" or "good." To tackle this, we model emphasis by learning label distributions (LDL) with a deep sequence labeling network and

[1]https://spark.adobe.com

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 1167–1172
Florence, Italy, July 28 - August 2, 2019.

the KL-Divergence loss function. LDL allows us to effectively capture the label ambiguity and inter-subjectivity within the annotators. Unlike single- or multi-label learning, LDL allows direct modeling of different importance of each label to the instance (Geng, 2016). The proposed model yields good performance despite the small amount of training data and can be used as a baseline for this task for future evaluations.

Our main contributions are: (1) Introducing a new NLP task: emphasis selection for short text instances as used in social media, learned from a new dataset. (2) Proposing a novel end-to-end sequence labeling architecture utilizing LDL to model the emphasis words in a given text. (3) Defining evaluation metrics and providing comparisons with several baselines to assess the model performance.

2 Related Work

A large amount of work in NLP addresses finding keywords or key-phrases in long texts from scientific articles, news, etc. (Augenstein et al., 2017; Zhang et al., 2016). Keyword detection mainly focuses on finding important nouns or noun phrases. In contrast, social media text is much shorter, and users tend to emphasize a subset of words with different roles to convey specific intent. Emphasis words are not necessarily the words with the highest or lowest frequency in the text. Often a high sentiment adjective can be emphasized, such as *Hot* in *Hot Summer*. Generally, word emphasis may express emotions, show contrast, capture a reader's interest or clarify a message.

In a different context, modeling word emphasis has been addressed in expressive prosody generation. Most studies detect emphasis words based on acoustic and prosodic features that exist in spoken data (Mishra et al., 2012; Chen and Pan, 2017). More recently, few works model emphasis from text to improve expressive prosody generation in modern Text-To-Speech (TTS) systems (Nakajima et al., 2014; Mass et al., 2018). For example, (Mass et al., 2018) trained a deep neural network model on audience-addressed speeches to predict word emphasis. The dataset consists of relatively long paragraphs which are labeled by four annotators based on words that clearly stand out in a recorded speech.

Many approaches have been proposed to deal with annotations coming from multiple annotators by essentially transforming the problem into single-label learning. Some rely on majority voting e.g. (Laws et al., 2011). More recent works (Yang et al., 2018; Rodrigues et al., 2014; Rodrigues and Pereira, 2018) use different strategies to learn individual annotator expertise or reliability, helping to infer the true labels from noisy and sparse annotations. All these approaches share one key aspect: only one label sequence is correct and should be considered as ground truth. This is contrary to the ambiguous nature of our task, where different interpretations are possible. Our solution is to utilize label distribution learning (Subsection 3.2). LDL methods have been used before to solve various visual recognition problems such as facial age prediction (Rondeau and Alvarez, 2018; Gao et al., 2017). We are the first to introduce LDL for sequence labeling.

3 Emphasis Selection

3.1 Task Definition

Given a sequence of words or tokens $C = \{x_1, ..., x_n\}$, we want to determine the subset S of words in C that are good candidates to emphasize, where $1 \leq |S| \leq n$.

3.2 Label Distribution Learning

We pose this task as a sequence labeling problem where the model assigns each token x from C a real number d_y^x to each possible label, representing the degree to which y describes x. Where $d_y^x \in [0, 1]$ and $\sum_y d_y^x = 1$. We use IO scheme $y \in \{I, O\}$, where "I" and "O" indicate emphasis and non-emphasis respectively. The final set of S_i can be generated by selecting tokens with different strategies (Subsection 5.3).

3.3 Dataset

We obtained 1,206 short text instances from Adobe Spark, which will be publicly available along with their annotations[2]. This collection contains a variety of subjects featured in flyers, posters, advertisement or motivational memes on social media. The dataset contains 7,550 tokens and the average number of tokens per instance is 6.16, ranging from 2 to 25 tokens. On average, each instance contains 2.38 emphases and the ratio of non-emphasis to emphasis tokens is 1.61.

[2] http://ritual.uh.edu/resources/emphasis-2019/

Words	A1	A2	A3	A4	A5	A6	A7	A8	A9	Freq. [I,O]
Enjoy	I	I	I	I	I	O	I	O	O	[6,3]
the	O	O	O	O	O	O	O	O	O	[0,9]
Last	O	O	O	O	I	I	O	O	O	[2,7]
Bit	O	O	O	O	I	I	O	O	I	[3,6]
of	O	O	O	O	O	O	O	O	O	[0,9]
Summer	I	I	I	O	I	O	I	I	O	[6,3]

Table 1: A short text example from our collected dataset along with its nine annotations.

We used Amazon Mechanical Turk and asked nine annotators to label each piece of text. To ensure high quality annotation, we included carefully-designed quality questions in 10 percent of the hits. We obtained a Fleiss' kappa agreement (Fleiss, 1971) of 63.59, which compared to similar tasks proves the subjectivity and multi-answer nature of our problem. We noticed higher annotation agreement in shorter length instances (2 to 5 words). Having many extremely short pieces of text in the dataset (∼60%) increased the annotation agreement.

We split up the data randomly into training (60%), development (10%) and test (30%) sets for further analysis. Table 1 shows an example of text annotated with the IO annotations. Ultimately, we compute the label distribution for each instance, which corresponds to the count per label normalized by the total number of annotations.

4 Model

We use an LSTM-based sequence labeling model to learn emphasis patterns. Figure 2 shows the overall architecture of the proposed model (DL-BiLSTM). Given a sequence of words, the model

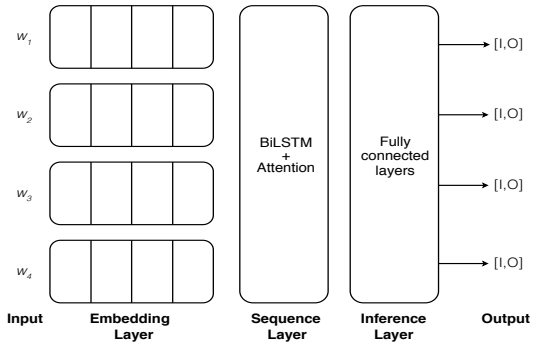


Figure 2: DL-BiLSTM Architecture

is to label each word with its appropriate label distribution. Words are represented with word embeddings for each input word sequence. We use two stacked bidirectional LSTM layers as an encoder to model word sequence information in both forward and backward directions. Having two BiLSTM layers helps to build a deeper feature extractor; having more than two does not help the performance as the model becomes too complicated. We investigate the impact of attention mechanisms to the model (Vinyals et al., 2015; Zhang et al., 2017), where attention weights a_i represent the relative contribution of a specific word to the text representation. We compute a_i at each output time i as follows:

$$a_i = \text{softmax}(v^T \tanh(W_h h_i + b_h)) \quad (1)$$

$$z_i = a_i \cdot h_i \quad (2)$$

where h_i is encoder hidden state and v and W_h are learnable parameters of the network. The output z_i is the element-wise dot product of a_i and h_i.

Subsequently, the inference layer assigns labels (probabilities) to each word using the hidden states of word sequence representations. This layer internally consists of two fully connected layers with size of 50. We use layer normalization (Ba et al., 2016) for improved results. [3]

KL-Divergence Loss During the training phase, the Kullback-Leibler Divergence (KL-DIV) (Kullback and Leibler, 1951) is used as the loss function. KL-DIV is a measure of how one probability distribution P is different from a second reference probability distribution Q:

$$\text{KL-DIV}(P||Q) = \sum_{x \in X} P(x) \log \frac{Q(x)}{P(x)}$$

5 Experimental Settings and Results

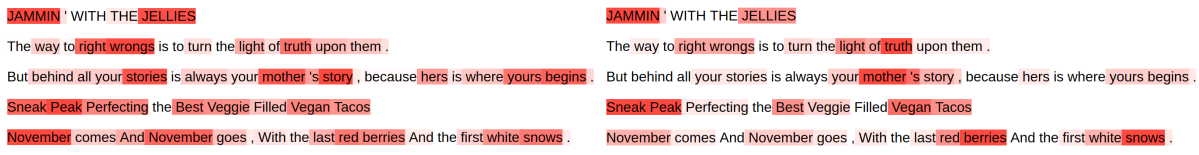
5.1 Training Details

We use two different word representations: pre-trained 100-dim GloVe embedding (Pennington et al., 2014), and 2048-dim ELMo embedding (Peters et al., 2018). We use BiLSTM layers with hidden size of 512 and 2048 when using GloVe and ELMo embeddings respectively. We use the Adam optimizer (Kingma and Ba, 2014) with the learning rate set to 0.001. In order to better train and to force the network finding different activation paths, we use two dropout layers with a rate of 0.5 in the sequence and inference layers. Fine-tuning is performed for 160 epochs, and the reported test result corresponds to the best accuracy obtained on the validation set.

[3]The implementation is available online: https://github.com/RiTUAL-UH/emphasis-2019

Model/Evals		Match$_m$				TopK				MAX
		m=1	m=2	m=3	m=4	k=1	k=2	k=3	k=4	
						F	F	F	F	ROC AUC
Label Distribution Learning Models										
M1	DL-BiLSTM+GloVe	54.8	69.4	77.2	81.6	47.5	68.2	78.1	83.6	0.874
M2	DL-BiLSTM+GloVe+Att	54.5	69.7	77.7	80.8	47.2	68.5	78.4	83.2	0.880
M3	DL-BiLSTM+ELMo	**57.4**	72.5	**79.2**	83.3	**49.7**	70.7	**79.4**	84.7	**0.887**
M4	DL-BiLSTM+ELMo+Att	56.2	**72.8**	77.9	**83.8**	48.7	**71.0**	78.5	**85.0**	0.883
Single Label Learning Models										
M5	SL-BiLSTM+GloVe	52.6	66.4	75.4	79.3	45.5	65.9	76.9	82.3	0.860
M6	SL-BiLSTM+GloVe+Att	52.3	66.1	77.2	78.5	45.3	65.6	78.1	81.7	0.862
M7	SL-BiLSTM+ELMO	53.7	68.7	76.9	80.5	46.5	67.7	77.9	83.0	0.865
M8	SL-BiLSTM+ELMo+Att	52.0	68.5	77.4	82.3	45.0	67.6	78.2	84.1	0.866
M9	CRF	44.0	65.3	73.0	79.2	38.1	65.0	75.3	82.2	0.818

Table 2: Experimental results of Label Distribution Learning and Single Label Learning models in three evaluation settings, Match$_m$, TopK, and MAX. F represents F1-score.



(a) Model's Output

(b) Ground Truth

Figure 3: Heatmap of emphases; highlighting words with model's output and ground truth probabilities.

5.2 Baselines

We compare our model against alternative setups in which the label distribution is mapped to binary labels using majority voting. We include the following single-label models:

SL-BiLSTM This model has a similar architecture compared to the DL-BiLSTM model but the input is a sequence of mapped labels and the negative log likelihood is used as the loss function in the training phase.

CRF This model is a Conditional Random Fields model (Lafferty et al., 2001) with handcrafted features including word identity, word suffix, word shape and word part-of-speech (POS) tag for the current and nearby words. The CRFsuite program (Okazaki, 2007) is used for this model.

5.3 Evaluation Settings

To assess the performance of the model, we propose three different evaluation settings:

Match$_m$ For each instance x in the test set D_{test}, we select a set $S_m^{(x)}$ of $m \in \{1 \dots 4\}$ words with the top m probabilities according to the ground truth. Analogously, we select a prediction set $\hat{S}_m^{(x)}$ for each m, based on the predicted probabilities. We define the metric Match$_m$ as follows:

$$\text{Match}_m := \frac{\sum_{x \in D_{test}} |S_m^{(x)} \cap \hat{S}_m^{(x)}|/(min(m, |x|))}{|D_{test}|}$$

TopK Similarly to Match$_m$, for each instance x, we select the top $k = \{1, 2, ..., 4\}$ words with the highest probabilities from both ground truth and prediction distributions. Then Precision, Recall and F1-score per each k can be computed accordingly.

MAX We map the ground truth and prediction distributions to absolute labels by selecting the class with the highest probability. Then we compute ROC_AUC. (e.g. a token with label probability of $[I = 0.75, O = 0.25]$ is mapped to "I").

5.4 Results

We run all models over 5 runs with different random seeds and report the scores of the best runs based on the dev set. Table 2 compares different models in terms of three evaluation settings. M1-M4 are four variants of the DL-BiLSTM model. Considering all evaluation settings, LDL models (M1-M4) either outperform SL-BiLSTM models

(M5-M8) or perform equally. Using ELMo instead of GloVe yields better results (M3 and M4). M3 and M4 with higher performance in all three metrics outperform the other models. Comparing the best results of both approaches, M3 and M4 with M7 and M8, we observe that both LDL results are statistically significant under paired t-test with 95% confidence interval. The improved performance of label distribution over single-label learning suggests that in LDL, the model exploits ordinal relationships among the classes during optimization, which results in better generalization.

Our model is more successful in predicting words with higher human annotation agreement. As we increase the confidence threshold and only consider words with higher ground-truth agreement, our model is able to achieve better results.

Figure 3 shows examples from the test set, with a heatmap showing the model's predicted score and ground truth probabilities.

6 SemEval-2020 Benchmarking

We are organizing a SemEval shared task on emphasis selection called "Task 10: Emphasis Selection for Written Text in Visual Media". In order to set out a comparable baseline for this shared task, in this section, we report results of our models according to the SemEval setting defined for the task. After the submission of this paper, we continued to improve the quality of the annotated data by cleaning the data and fixing the annotations of some noisy instances. The SemEval version of Spark dataset contains 1,200 instances with a different split: 70% training, 10% development and 20% test sets. We choose Match$_m$ as the evaluation metric for this shared task as it provides a comprehensive evaluation compared to MAX, as one can choose the value of m. Furthermore, compared to TopK, the Match$_m$ metric can better handle cases where multiple tokens have the same label distribution according to the annotators in the ground truth. Table 3 shows the results of all nine models under the SemEval setting, using the Match$_m$ evaluation metric. Similar to the results we showed in Table 2, M3 and M4 both perform competitively and outperform the other models.

7 Conclusion

We introduced a new task, emphasis selection in short text instances. Its goal is to develop models that suggest which part of the text to emphasize. To tackle the subjective nature of the task, we propose a sequence labeling architecture that optimizes the model to learn label distributions by capturing the inter-subjectivity within the audience. We provide comparisons to models trained with other objective functions where the ground truth probabilities are mapped to binary labels and show that LDL is more effective in selecting the emphasis. As future work, we plan to investigate emphasis selection on a larger and more diverse dataset. We also plan to investigate the role of word sentiment and emotion intensity as well as more advanced language models such as BERT (Devlin et al., 2018) in modeling emphasis.

Model/Eval		Match$_m$			
		m=1	m=2	m=3	m=4
Label Distribution Learning Models					
M1	DL-BiLSTM+GloVe	54.6	69.2	76.5	81.9
M2	DL-BiLSTM+GloVe+Att	57.5	69.7	76.7	80.7
M3	DL-BiLSTM+ELMo	**0.6**	71.7	**78.7**	84.1
M4	DL-BiLSTM+ELMo+Att	59.6	**72.7**	77.7	**84.6**
Single Label Learning Models					
M5	SL-BiLSTM+GloVe	51.7	66.7	75.0	81.1
M6	SL-BiLSTM+GloVe+Att	52.9	66.5	73.6	0.8
M7	SL-BiLSTM+ELMo	54.2	69.0	77.9	83.0
M8	SL-BiLSTM+ELMo+Att	54.2	70.7	78.5	82.8
M9	CRF	45.4	66.0	72.8	80.2

Table 3: Experimental results in SemEval setting

Acknowledgement

This research began during an internship at Adobe Research, and was sponsored in part by Adobe Research. We thank reviewers for their valuable suggestions. We also thank Amin Alipour and Niloofar Safi for comments that greatly improved the manuscript.

References

Isabelle Augenstein, Mrinal Das, Sebastian Riedel, Lakshmi Vikraman, and Andrew McCallum. 2017. Semeval 2017 task 10: Scienceie-extracting keyphrases and relations from scientific publications. *arXiv preprint arXiv:1704.02853*.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

Yanju Chen and Rong Pan. 2017. Automatic emphatic information extraction from aligned acoustic data and its application on sentence compression. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Bin-Bin Gao, Chao Xing, Chen-Wei Xie, Jianxin Wu, and Xin Geng. 2017. Deep label distribution learning with label ambiguity. *IEEE Transactions on Image Processing*, 26(6):2825–2838.

Xin Geng. 2016. Label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, 28(7):1734–1748.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Solomon Kullback and Richard A Leibler. 1951. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Florian Laws, Christian Scheible, and Hinrich Schütze. 2011. Active learning with amazon mechanical turk. In *Proceedings of the conference on empirical methods in natural language processing*, pages 1546–1556. Association for Computational Linguistics.

Yosi Mass, Slava Shechtman, Moran Mordechay, Ron Hoory, Oren Sar Shalom, Guy Lev, and David Konopnicki. 2018. Word emphasis prediction for expressive text to speech. pages 2868–2872.

Taniya Mishra, Vivek Rangarajan Sridhar, and Alistair Conkie. 2012. Word prominence detection using robust yet simple prosodic features. In *Thirteenth Annual Conference of the International Speech Communication Association*.

Hideharu Nakajima, Hideyuki Mizuno, and Sumitaka Sakauchi. 2014. Emphasized accent phrase prediction from text for advertisement text-to-speech synthesis. In *Proceedings of the 28th Pacific Asia Conference on Language, Information and Computing*.

Naoaki Okazaki. 2007. Crfsuite: a fast implementation of conditional random fields (crfs). http://www.chokkan.org/software/crfsuite/.

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 (EMNLP)*, pages 1532–1543.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Filipe Rodrigues, Francisco Pereira, and Bernardete Ribeiro. 2014. Sequence labeling with multiple annotators. *Machine learning*, 95(2):165–181.

Filipe Rodrigues and Francisco C Pereira. 2018. Deep learning from crowds. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Jared Rondeau and Marco Alvarez. 2018. Deep modeling of human age guesses for apparent age estimation. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 01–08. IEEE.

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015. Pointer networks. In *Advances in Neural Information Processing Systems*, pages 2692–2700.

Jie Yang, Thomas Drake, Andreas Damianou, and Yoelle Maarek. 2018. Leveraging crowdsourcing data for deep active learning an application: Learning intents in alexa. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*, pages 23–32. International World Wide Web Conferences Steering Committee.

Qi Zhang, Yang Wang, Yeyun Gong, and Xuanjing Huang. 2016. Keyphrase extraction using deep recurrent neural networks on twitter. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 836–845.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D Manning. 2017. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 35–45.