Predicting Helpful Posts in Open-Ended Discussion Forums: A Neural Architecture

Kishaloy Halder[1,2] **Min-Yen Kan**[1,2] **Kazunari Sugiyama**[1]

[1]School of Computing, National University of Singapore
[2]Institute for Application of Learning Science and Educational Technology, National University of Singapore
{kishaloy, kanmy, sugiyama}@comp.nus.edu.sg

Abstract

Users participate in online discussion forums to learn from others and share their knowledge with the community. They often start a thread with a question or by sharing their new findings on a certain topic. Unlike in Community Question Answering, where questions are mostly factoid based, we find that the threads in a forum are often open-ended (*e.g.*, asking for recommendations from others) without a definitive correct answer. We thus address the task of identifying helpful posts in a forum thread to help users comprehend long-running discussion threads, which often contain repetitive or irrelevant posts. We propose a recurrent neural network based architecture to model (i) the relevance of a post regarding the original post starting the thread, and (ii) the novelty it brings to the discussion, compared to the previous posts in the thread. Experimental results on five different types of online forum datasets show that our model significantly outperforms the state-of-the-art neural network models for text classification.

1 Introduction

Online discussion forums are widely used in many domains such as in generic web content[1], e-health[2], Massive Open Online Courses (MOOCs)[3], and e-commerce, among others. Users participate in these forums to gain knowledge from the collective wisdom of the community. Typically, users start a discussion thread by posting a question or asking others for opinions on a topic. Others then reply to threads relevant to their interests. Importantly, as these forums are indexed by search engines, they need to be discoverable by a wider audience — apart from just registered users — by enabling threads to be found in response to queries.

Due to the open nature of the forums and the various expertise level of users, the posts in the discussion threads vary in helpfulness. To address this, some websites provide actions for users to signal this, as in "Upvote" (`reddit`, `stackoverflow`) and "Highlight" (`coursera`). Such feedback is helpful for identifying important posts among the many. Such feedback rarely comes immediately following new post creation, affecting their visibility to the users (Singh et al., 2017). We can devise technology to proactively identify such helpful posts as they arrive, in a *helpfulness prediction task*, enabling users to efficiently assess relevance.

We observe that there is a key structural difference between online discussion forums and Community Question Answering (CQA) websites. Figure 1 shows the distribution of normalized helpful votes for the top-5 posts across a popular discussion forum (`reddit`), and a CQA website (`stackoverflow`[4]). In CQA, the vote distribution decays exponentially, indicating that usually there is a single correct answer with the largest number of votes (Omari et al., 2016). In contrast, votes for less helpful posts in discussion forums decay at a much lower rate, suggesting that discussion forum threads are more open-ended.

Table 1 shows a sample thread from `reddit` to understand the dynamics of online discussion. We observe the following two major differences compared to threads in CQA domain: (1) The first post (hereafter, *original post*) is not necessarily a question, but can be personal anecdotes or new findings on a certain topic, attracting more discussion. (2) Instead of searching for a single relevant answer as in CQA, discussion forum users find a post helpful

[1]https://www.reddit.com/
[2]https://www.healthboards.com/boards
[3]https://www.coursera.org
[4]https://www.kaggle.com/stackoverflow/stacksample/data

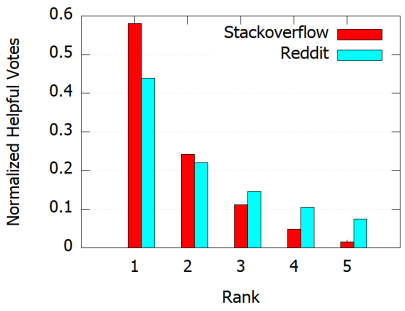


Figure 1: The helpful vote distribution for the top-5 posts across an online discussion forum (`reddit`), and the `stackoverflow` CQA website. The helpful votes decay at a slower rate for `reddit` compared to focused CQA.

when it introduces some *relevant* (with respect to the original post) and *novel* (*i.e.*, not presented in the earlier posts within the same thread) information. Motivated by these observations, we address helpfulness prediction by considering both the target post and its preceding posts.

We propose a novel neural architecture to predict the helpfulness of a post in a discussion thread. Our approach consists of two components: (1) modeling the *relevance* of a post and (2) determining the *novelty* with respect to the sequence of preceding posts. It combines the output from both components to predict the overall post helpfulness. As recurrent neural networks (RNNs) have shown good performance in sequence modeling tasks (Chung et al., 2014; Sutskever et al., 2014), we apply it to our architecture to model the (i) sequence of words in the post text, and the (ii) sequence of posts in a thread. Our model significantly outperforms other state-of-the-art models across experiments on five varied and large forum datasets. Our main contributions are:

- We reveal the key differences between posts in CQA and online discussion forums;
- We analyze the confounding factors behind the perceived helpfulness of posts in discussion forums. We observe that both *relevance* and *novelty* play important roles in determining the helpfulness of a post;
- We propose a novel neural network architecture to predict the helpfulness by using textual content of a target post as well as sequence of posts preceding it in the thread;
- We compare our model with current neural network classifiers and analyze the factors that influence our model's performance.

Order	Post Text	Helpful?
Original post	I was working yesterday..and my back was bent over and when I got up I felt like I strained my back but now my mind is linking it to my kidney..	Yes
1	I have this and my doc has told me it's muscular and physio might help..	Yes
2	Kidney pain is usually constant and doesn't change when you move, or get better when you change position, from how I understand it..you'll be fine :)	Yes
3	If it happens only when you move there is a big chance it's a muscle spasm, this happens after some physical activities.	No

Table 1: A sample discussion thread from `reddit`. Helpful votes are provided by the website users.

2 Related Work

To the best of our knowledge, predicting helpful posts in generic open-ended discussion forums has not been studied before. However, there is significant amounts of related work on similar directions; where researchers evaluate the *quality* (which may not correlate with perceived helpfulness by the community users) of posts in specific domains such as health (Oh et al., 2012; Oh and Worrall, 2013; Beloborodov et al., 2014) and online education (Chandrasekaran et al., 2015; Chandrasekaran and Kan, 2019; Jenders et al., 2016). External medical resources and thesauri such as UMLS[5] have been used to identify patterns of helpfulness in health (Asghar et al., 2014). In MOOC platforms, apart from the textual content of the forums, additional signals such as user reputation (*e.g.*, average homework scores, number of courses taken) have been used to estimate post quality (Jenders et al., 2016). However, these techniques are tightly coupled with the target domain, and may not be generalizable to new domains.

CQA Answer Quality: Past work has also addressed the evaluation of answer quality in CQA sites (Jeon et al., 2006; Hong and Davison, 2009; Shah and Pomerantz, 2010; Yao et al., 2015; Omari et al., 2016; Li et al., 2015). Typically posed as a classification problem, these use both textual and non-textual feature-based approaches. Since it is quite common for popular questions to attract many potential answers, answer ranking based on perceived quality is another line of approach (Surdeanu et al., 2008; Bian et al., 2008; Wang et al., 2009). Closer to our approach, Omari

[5] `https://www.nlm.nih.gov/research/umls/`

et al. (2016) proposed a novelty-based greedy ranking algorithm that depends on a pre-trained parser to identify different propositions, useful for predicting helpfulness. Li et al. (2015) propose a few features for answer quality detection from academic QA sites such as ResearchGate[6]. However this approach does not generalize well since the method uses many website-specific signals such as *reputation scores* for users and their institutions. Additionally, their approach relies on human annotations to identify a few key conversational characteristics in the answers, keeping it from being applied to use cases where scalability and automation are key.

In the CQA answer quality evaluation literature, quality is often measured through the human evaluators' annotations during experimentation (Shah and Pomerantz, 2010; Oh et al., 2012; Omari et al., 2016). However, we are interested in modeling the "helpfulness" for actual discussion forum users (in term of "Upvotes") and not annotators following guidelines to mark answer quality, which might present other forms of bias.

Modeling Novelty in IR, such as search result diversification (Carbonell and Goldstein, 1998; Soboroff and Harman, 2005; Ziegler et al., 2005; Clarke et al., 2008), also constitutes prior art. Carbonell and Goldstein (1998) proposed maximal marginal relevance (MMR) to diversify the set of documents returned for a search query. Similar approaches were also used later in Multi-Document Summarization (MDS) tasks (Nallapati et al., 2017). These approaches address the problem either as a ranking task (ordering search results) or as a subset selection problem (MDS), where all documents are simultaneously made available. In contrast, in our discussion thread scenario, we need to model the discussion posts' sequential nature to understand the context of a later post and, in turn, determine its helpfulness.

Neural Network Based Models have also recently outperformed existing classifiers in many text classification tasks. They have been widely adopted as they induce useful features on their own, given sufficient data. Although there are differences, the problem of answer selection is relevant: the goal is to rank the potential answers to a target question from multiple candidate answers in order of their similarity (Yu et al., 2014; Wang and Nyberg, 2015; Severyn and Moschitti, 2015). However in our case, all posts in a thread are similar to the original post to an extent. Helpful posts are thus more difficult to identify; computing similarity is not viable as a single source solution.

Inspired by all these previous works, we propose a neural architecture to predict the helpfulness of posts in open-ended discussion forums. To make it generic and easily adaptable to multiple domains, we study the problem from a linguistic viewpoint, where we consider only the textual contents of the discussion threads.

3 Methods

We propose a neural network architecture to model post helpfulness (*cf* Figure 2a). Our architecture is end-to-end trainable, adaptable to different domains. The model comprises two components to analyze a target post's thread *relevance* and *novelty* with respect to its past k posts.

3.1 Text Encoder

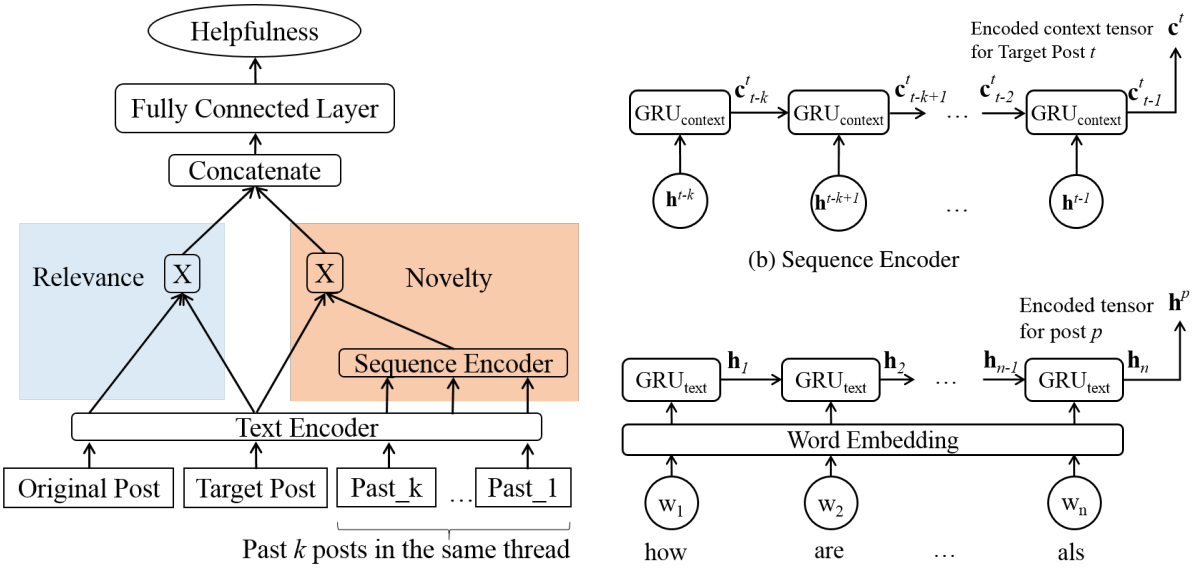
This component takes a post text p which consists of words $(w_1, w_2, \ldots, w_n)$ as input and encodes it to a tensor ($\mathbf{h}^p$) in two steps. We first use a word embedding initialized with GloVe[7] to transform all the words from the post text into finite d-dimensional vectors, *i.e.*, $w_i \mapsto \mathbb{R}^d$. Our experimental results show that the coverage of GloVe varies between $68 - 76\%$ on our datasets. To estimate the embeddings for the out-of-vocabulary words and reflect the domain dependence, we keep the embedding vectors trainable. In the second step, the sequence of words are provided to a gated recurrent unit (GRU) layer (Chung et al., 2014) to obtain a sequence of hidden vectors $(\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_n)$, where $\mathbf{h}_i \in \mathbb{R}^g$, and g is the output dimension of the GRU encoded tensor. The latent vector is defined as follows:

$$\mathbf{h}_i = \text{GRU}_{\text{text}}(\mathbf{h}_{i-1}, w_i).$$

The last vector in the sequence, $\mathbf{h}_n$, is considered as the encoded representation of a post text (*cf.* Figure 2c). For a post p, the GRU_{text} encoded representation is denoted as $\mathbf{h}^p$. We use a dropout layer after the GRU to prevent overfitting. In our model, note that there is only a single text encoder; all textual inputs — the target post, original post,

[6] https://www.researchgate.net/

[7] http://nlp.stanford.edu/data/glove.6B.zip



(a) Overall network

(b) Sequence Encoder

(c) Text Encoder

Figure 2: Our neural architecture and its components. (a) Overall network architecture. Shaded component on the left captures relevance with respect to the original post; ones on the right measure the novelty compared to the past k posts. (b) Unrolled Sequence Encoder ($\text{GRU}_{\text{context}}$). (c) Unrolled Text Encoder (GRU_{text}).

and each of the past posts in the thread — are encoded using a single text encoder, since as all of them are essentially text posts of similar nature.

Alternative Architectures. We also tried stacking additional GRUs in our experiments, but we did not observe accuracy improvements. We also tried to replace GRU with LSTM (Long-Short Term Memory) (Hochreiter and Schmidhuber, 1997), resulting in similar performance at the cost of much longer training time due to the larger number of parameters.

3.2 Modeling Post's Relevance

The left component of Figure 2a captures the relevance of a target post with respect to the original post. It takes as input two GRU encoded tensors: one for the target post $\mathbf{h}^t$, the other for the original post $\mathbf{h}^o$. It computes their similarity defined as:

$$\mathbf{r}_t = \mathbf{h}^t \otimes \mathbf{h}^o,$$

where $\otimes$ denotes the element-wise multiplication. We also experimented with element-wise difference and cosine similarity, but found that multiplication works best. Our relevance modeling component is inspired from the architecture for answer sentence selection model (Yu et al., 2014).

3.3 Modeling Post's Novelty

In Figure 2a, the right component models the target post's novelty compared to the past k posts from the same thread. It takes the encoded tensors for the target post $\mathbf{h}^t$ as input, as well as the past k posts $(\mathbf{h}^{t-k}, \mathbf{h}^{t-k+1}, ..., \mathbf{h}^{t-1})$.

We first encode the context of the discussion by modeling the sequence of the past k posts. In order to achieve this, we use another GRU (labeled as Sequence Encoder in Figure 2a) to transform the sequence of k post tensors to a single context tensor $\mathbf{c}^t$ of equal dimension g. Each timestep i of this is defined as follows:

$$\mathbf{c}^t_i = \text{GRU}_{\text{context}}(\mathbf{c}^t_{i-1}, \mathbf{h}^{t-i}).$$

Similar to GRU_{text}, $\mathbf{c}^t_{t-1}$, the last vector in the sequence, is considered as the context representation $\mathbf{c}^t$ (as shown in Figure 2b).

To determine the novelty of the target post, we compute its similarity $\mathbf{n}_t$ with the discussion thread context represented by its context tensor:

$$\mathbf{n}_t = \mathbf{h}^t \otimes \mathbf{c}^t.$$

Importantly, instead of considering all the previous posts in the thread, we limit the context to the past k posts for two reasons:

1. Users may not recall the entire context of discussion while reading a post appearing much later in a long-running thread.

2. Users often arrive at a discussion thread through search engine queries. Since long threads are paginated, a user may arrive on a page in the

middle of the discussion thread, thus also missing the previous context.

We find empirical evidence for these assumptions later in our experiments (see Section 5). In tuning our model, we observed that increasing the context length beyond a threshold does not yield improvements.

3.4 Final Helpfulness Prediction

We combine the relevance tensor ($\mathbf{r}_t$) and novelty tensor ($\mathbf{n}_t$) and feed through a fully connected layer to make the final post helpfulness prediction:

$$\mathbf{x}_t = \mathbf{r}_t \oplus \mathbf{n}_t,$$

$$p(y|\mathbf{x}_t) = sigmoid(\mathbf{W} \cdot \mathbf{x}_t + \mathbf{b}),$$

where $\oplus$ denotes concatenation; $\mathbf{x}_t$ is the concatenated tensor; y is the output label (0 or 1); $\mathbf{W}$ and $\mathbf{b}$ are the weight matrix and bias vector, respectively, learned for the fully connected layer. We use binary cross-entropy loss to train the model, optimizing with Adam (Kingma and Ba, 2014).

Alternative Architectures. We also investigated ensemble architectures. We fed the relevance and novelty tensors through two separate fully connected layers to obtain the binary predictions from both components concurrently, then merged the two predictions via a final fully connected layer for obtaining prediction. This approach fared worse compared to our concatenation-based model, possibly as our final concatenation model can exploit non-linear interactions between both components.

The actual post content is never presented to the fully connected layer so that it generalizes well. The final layer only gets to see the relevance, and novelty vectors, which we believe ameliorates the creation of overfitted (post-based or thread-based) features for the helpfulness prediction task.

4 Experiments

We first describe the datasets, evaluation metrics, and baseline models before our main results. We also conducted additional experiments to answer specific research questions about our model.

4.1 Datasets

We experiment with five real-world online discussion forums (Table 2) to validate model effectiveness. Typical of other research work, we also remove threads that have less than two posts.

1–2. **Reddit** is a popular platform for discussions on a wide-variety of topics on the web. We use a large number of discussion threads from a `reddit` data dump[8]. To diversify the datasets in terms of average thread length, we set different thresholds, and created two datasets: *Reddit_10+* ($\geq$ 10 posts) and *Reddit_3+* ($\geq$ 3 posts). Along with a chronologically ordered set of posts, `reddit` also has "Upvote" counts for every post.

Dataset	# Posts	# Threads	Avg # Posts / Thread	Avg # words / Post
1. Reddit_10+	200,006	9,744	20.52	29.45
2. Reddit_3+	200,016	28,763	6.95	30.58
3. Android Apps	11,643	2,077	5.60	56.53
4. Matrix	10,159	2,484	4.08	65.30
5. Travel	30,116	10,250	2.93	163.43

Table 2: Dataset statistics.

3–4. **Coursera** is a large MOOC platform, providing a discussion forum for the course participants. We select two courses with the largest number of posts: "Matrix-001" and "Android Apps 101-001" from a MOOC dataset (Chandrasekaran et al., 2015). Course participants can "vote" for a post if they find it helpful. We refer to these datasets as *Matrix* and *Android Apps*, hereafter.

5. **Travel Stack Exchange** is one of many QA websites in the Stack Exchange community. We use a data dump[9] of the website and refer to it as *Travel* dataset. In Travel Stack Exchange, a user can "Upvote" a post if she deems it helpful. Although not strictly a discussion forum, the threads in this forum appear to be less objective (by our vote distribution analysis, similar to Figure 1), compared to other CQA sites like `stackoverflow`.

4.2 Post Annotation and Evaluation Metrics

We use the user-provided feedback in form of "mark as helpful", "like", "upvote" actions as a proxy of the actual helpfulness of a post. Vote counts vary widely across posts and threads, (*i.e.*, 0 to 3,100 for the `reddit` dataset), making it infeasible to formulate the task as a regression problem. Following by prior published research (Cheng et al., 2014; Lo et al., 2017), we model it as a binary classification problem, and use the 80^{th} percentile expected value of helpful vote count across all the posts as the boundary between the two classes. We assume that a post is

[8] https://files.pushshift.io/reddit/comments/

[9] https://archive.org/download/stackexchange/travel.stackexchange.com.7z

helpful if it has received more helpful votes than the 80^{th} percentile, and *not helpful* otherwise.

Since our goal is to predict the helpful posts and the class distribution is inherently skewed from our definition, we evaluate the model performance in terms of prediction accuracy for only the positive, helpful class. We evaluate using standard precision, recall, and F_1 score across all datasets.

4.3 Baselines

Code for our model is publicly available[10] to aid the reproduction of our results. We experiment with the following state-of-the-art neural text classification methods:

1. BiLSTM (Sun et al., 2017): a stack of two layers of Bidirectional LSTM encoders on post text.

2. Stacked LSTM (Liu et al., 2016): a stack of two layers of LSTM encoders on the post text.

3. LSTM with Attention (Rocktäschel et al., 2016): an LSTM layer with hierarchical attention.

4. Answer Sentence Selection (Yu et al., 2014): a CNN model pioneered in a TREC QA[11] task.

5. Our Model (Relevance based): only the relevance component of our model.

6. Our Model (Novelty based): only the novelty component of our model.

We do not include traditional feature-based models as part of our reported baseline portfolio, as in our study, neural models have outperformed them as well, which is corroborated in recent studies (Kim, 2014). Additionally, such approaches are fragile, as we experiment with datasets from multiple domains with various discussion styles, and extracting hand crafted features for each is non-trivial and labour intensive. As a preliminary experiment, we tried with a traditional bag-of-words based model. However, we do not include it in the baseline portfolio given its poor performance on our datasets.

4.4 Training

We used the Keras[12] library with TensorFlow as the backend for model implementation. We split the dataset 80:10:10 for train, validation, and test, respectively, and perform 5-fold cross validation. We tuned the hyper-parameters via grid search on the validation set for all the models.

The rest of the parameters used follow standard values from the recent literature. We set word embedding dimension (d) to 100, vocabulary size to 100K, hidden dimension of GRU (g) to 128, batch size to 512, the dimension of the final fully connected layer to 128, and use 70% dropout. For the CNN-based Answer Sentence Selection baseline, we tuned the number and size of filters (128 and 3, respectively). The maximum length of post text was set according to average post length (in the training split) for each dataset.

4.5 Results

Table 3 shows the comparison of model performance over the five datasets. We observe that our full model consistently outperforms others in terms of F_1 across all datasets. Our novelty-based model gives the second best score in all datasets except for *Android Apps*. Comparing our novelty-based model against answer selection model, we observe that the helpfulness of a post depends on both its relevance to the original post and the novelty with respect to earlier posts in the same thread. The evaluation scores obtained by the state-of-the-art neural text classification models strongly support this observation. They consistently make less accurate prediction compared to the relevance- and/or novelty-based models. Among them, BiLSTM or LSTM with Attention model achieves the best performance, dependent on the dataset. We discuss the confounding factor affecting performance in Section 5.

We also observe that the prediction is more accurate when there is sufficient context to learn the dynamics of the discussion forums. In *Reddit_10+* and *Reddit_3+*, where both datasets average about 20 and 7 posts per thread respectively, we obtain an F_1 score of 0.40 to 0.51. In the other datasets, where the average thread length is much shorter ($\sim$ 3 to 5), we obtain relatively low F_1 scores of 0.34 to 0.38. Our model is more accurate in `reddit` datasets where threads are longer on average, indicative of more open-ended discussion centered on the original post.

4.6 Case Study

We now highlight a few corner cases successfully handled by our model.

Table 4 shows three target posts along with the original posts and their previous posts from dif-

[10] https://github.com/WING-NUS/post-helpfulness

[11] http://trec.nist.gov/data/qa.html

[12] https://keras.io

Model	1. Reddit_10+			2. Reddit_3+			3. Android Apps			4. Matrix			5. Travel		
	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
BiLSTM (Sun et al., 2017)	0.23	0.23	0.23	0.23	0.22	0.22	**0.36**	0.32	0.34	0.29	0.35	0.32	0.28	**0.31**	0.29
Stacked LSTM (Liu et al., 2016)	0.24	0.21	0.22	0.23	0.20	0.21	0.34	0.29	0.31	0.32	0.29	0.31	0.23	0.26	0.25
LSTM w/ Attention (Rocktäschel et al., 2016)	0.24	0.21	0.23	0.24	0.21	0.22	0.34	0.27	0.30	0.30	0.36	0.33	0.25	0.26	0.25
Answer Selection (Yu et al., 2014)	0.28	0.27	0.27	0.31	0.32	0.32	0.28	0.21	0.24	0.33	0.34	0.33	0.30	**0.31**	0.31
Our Model (Relevance-based)	0.30	0.30	0.30	0.32	0.34	0.33	0.31	0.35	0.33	0.38	0.31	0.34	0.35	0.30	0.32
Our Model (Novelty-based)	**0.53**	0.38	0.44	**0.42**	0.27	0.33	0.33	0.24	0.28	**0.43**	0.27	0.33	**0.47**	0.27	**0.34**
Our Model (Full)	0.48	**0.53**	**0.51**	0.41	**0.39**	**0.40**	0.35	**0.40**	**0.38**	0.37	**0.37**	**0.37**	0.37	**0.31**	**0.34**

Table 3: (P)recision, (R)ecall and F_1 comparison of model performances across our five datasets representing three domains. Our model outperforms other state-of-the-art neural text classifiers consistently. Ablation study with Answer Selection, Relevance-based, and Novelty-based model shows that modelling both relevance, and novelty is important.

Original Post	**My fiancée and I are looking for a good Caribbean cruise in October and were wondering which islands are best to see and which Cruise line to take?..**	**I've had bouts of heart burn & this time its sticking around for a while. I ate something really spicy on Tuesday night & its Thursday & Im having heart burn on & off... Please help**	**In a few weeks' time, I will be visiting the US for 14 days. Coming from the EU, roaming is very expensive, so I am considering getting a temporary SIM card..**
Past Posts	Friends I am staying with are travelling with Royal Carribean on a cruise in October. They are starting from Miami..	You're probably fine. People get heartburn from time to time.. Eat bland food for a few days and that inflammation should subside..	There are many options you can have as far as mobile phone data prepaid plans are concerned. Since you need coverage along the route..
	The Princess Cruise line has a Caribbean cruise in the fall. It may start in November rather than October but could be suitable for your needs..	Heartburn can last a few days and its not always spicy food that triggers it. I assume youre concerned it might be a heart attack. If that was it you would know it by now.	You may want to check your existing phone plan. For example, quite a few providers in the UK offer free or cheap roaming with data included..
	There are plenty of options for the Caribbean in October regradless of it being in hurricane season..	Heartburn doesn't JUST occur from spicy food. If you're having it over multiple days, it could simply be other food. Fatty foods in particular cause it.	If your main goal is price, MetroPCS has no-contract 30 month plans which have unlimited calling US numbers, unlimited SMS, and unlimited data in the US..
Target Post	If you like to dress up and eat high-end food, the cruise line you want is not the one that caters to honeymooners on a tight budget or to families with small kids. If you like things to be..	Stay calm. Drink lots of water. Do you have an antacid you could take? Try to avoid spicy, acidic, caffeine, alcohol for a while..	`willmyphonework.net` is good for checking a phone's compatibility with the various networks. Suggestion before departure, print-out a list of the carriers your phone will work with hard copy is the way to go here..
Helpful?	Yes	No	No

Table 4: Illustration of different corner cases for helpfulness prediction. The target post needs to be both relevant to the original post, and novel compared to the previous posts in the thread in order to be helpful.

ferent datasets. In the first case, we observe that the target post introduces some relevant and novel information into the thread, and thus our model predicts it as helpful.

In the second example, we find that the target post is quite similar to some of the previous posts. Since it introduces less novelty in the discussion, our model predicts the target post as unhelpful, although relevant to the discussion topic. In the third example, the target post seems to be novel compared to the previous posts but it deviates from discussion topic in the original post. Hence, our model does not predict it as helpful.

These observations indicate that our model treats each of the two qualities of a target post, *i.e.*, relevance with the original post, and novelty compared to the previous discussion individually as necessary but not sufficient conditions. A target post needs both relevance and novelty so that our model predicts it as helpful.

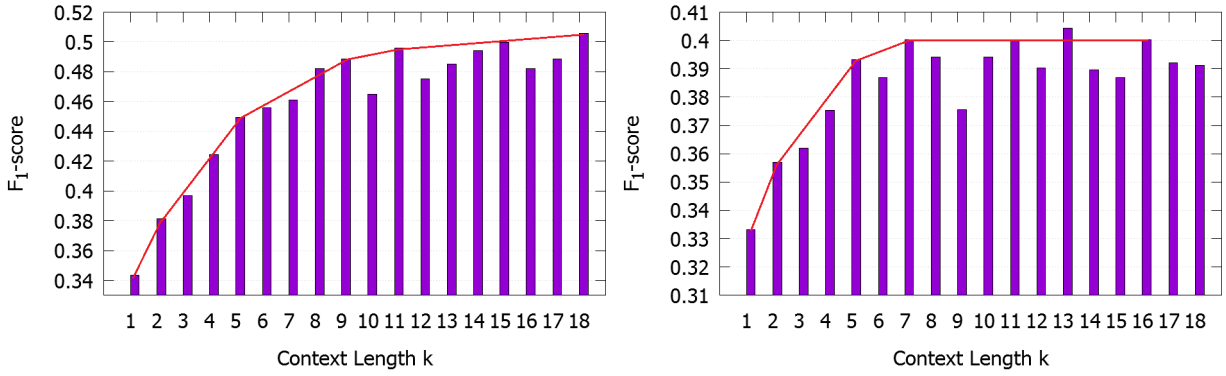
5 Discussion

We now answer the following research questions (RQ) to further analyze prediction of helpful posts:

RQ1: How does the past context length influence model performance? The number of posts across threads varies widely, making it difficult to estimate the optimal value for past context length (k in Section 3.3). To understand the effect of k on model performance, we vary k ranging from 1 to 18 and report F_1 for the *Reddit_10+*, and *Reddit_3+* datasets in Figure 3. Interestingly, we observe that, the performance stops improving after a certain number of posts in both cases: k=11 and k=7 for *Reddit_10+*, and *Reddit_3+*, respectively.

Setting too low a k limits the number of past posts the model gets to see, underfitting the data. Large k gives modest performance gains but incurs significant increase in training cost. As discussed in Section 3.3, the entire context might be redundant to determine target posts' helpfulness in long threads.
We believe the context length analysis would be necessary to achieve optimal model performance while exploring other domains.

RQ2: Does the order of contextual posts matter? To investigate whether the order of the past posts matter in determining the helpfulness of a target post, instead of modeling the past posts by $GRU_{context}$ layer, we just use the average of the



(a) Reddit_10+ (b) Reddit_3+

Figure 3: Model performance while varying context length k for *Reddit_10+*, and *Reddit_3+* datasets. F_1 stabilizes after a certain context length in both cases. Trend line in red.

Context Modeling	Reddit_10+	Reddit_3+	Andriod Apps	Matrix	Travel
Average	0.40	0.35	0.36	0.36	0.33
$GRU_{context}$	**0.53**	**0.40**	**0.38**	**0.37**	**0.34**

Table 5: F_1 obtained by the model variation that uses the average tensor of the past post tensors as the context tensor, compared to our $GRU_{context}$ based model.

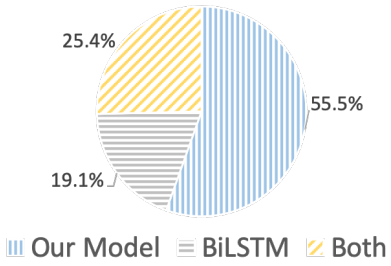


Figure 4: Correct prediction share of helpful posts for Reddit_all. Yellow: both models; blue: only our model; grey: only BiLSTM.

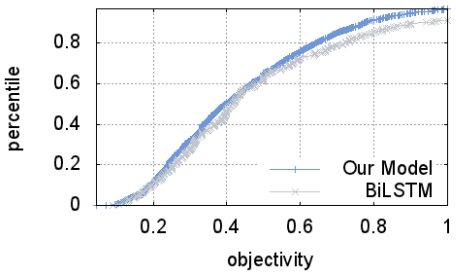


Figure 5: Thread objectivity score CDF. The blue curve shows threads where our model is correct and BiLSTM is not; vice versa for the grey.

past post tensors to get the context tensor. Table 5 shows the F_1 achieved by this variation compared to our model.

We observe that the model performance significantly degrades when the order of the past posts are ignored and represented by an average. Crucially, we find that the datasets with longer threads suffer more compared to the ones with shorter threads. This observation indicates that the sequential nature of discussion is integral to model construction.

RQ3: What factors influence performance among the text classification models and our model? Table 3 shows that BiLSTM achieved better scores compared to the other neural text classification models. To better understand the modeling differences between the BiLSTM and our models, we focus on the cases where one model is correct but not the other (as illustrated for *Reddit_10+* in Figure 4). While both models can predict the correct class in 25.4% cases (in yellow), in the other cases (blue and grey), they differ.

We study the objectivity of the posts where such differences were observed. Without loss of generality, we define a metric called thread *objectivity* spread, in terms of the vote shares for the top-5 posts:

$$objectivity = \frac{max(vote(x)) - min(vote(x))}{\sum vote(x)},$$

where $x \in \{\text{top-5 posts}\}$ in the thread and $vote(x)$ gives the helpfulness score of post x. $objectivity$ is unit bound $[0, 1]$. While a high objectivity score indicates skewed helpfulness distribution in a thread, a low score indicates that there are multiple helpful answers in a thread; in other words, the thread is less objective in nature.

We analyze the cumulative distribution func-

tions (CDFs) of objectivity spread scores for all threads belonging to the grey or blue wedge of Figure 4 (*cf.* Figure 5). We observe that the CDF for our model (blue) gives lower objectivity scores with 80^{th} percentile score of 0.64 for our model and 0.72 for BiLSTM, respectively. This indicates that our model performs better when the thread is more open-ended in nature.

6 Conclusion

We studied the problem of predicting helpfulness of posts in open-ended discussion forums. We found key differences in discussion forums compared to traditional CQA platforms: we observe that forum threads are often non-factoid and subjective in nature with many helpful answers. We hypothesize that post helpfulness crucially relies on two factors: (i) its relevance to the discussion thread and (ii) the novelty of the information introduced. We propose a generic and novel neural architecture using GRU encoders to embody this intuition. Our model outperforms state-of-the-art neural text classification baselines over a diverse set of forums representing three distinct domains. Through deeper analysis, we demonstrate that our model is able to encode the sequential nature of contextual posts, and capture the open-ended nature of discussion threads, thus achieving superior performance over other neural approaches.

We plan to apply our work towards building a notification system for incoming helpful posts. In the current work, we addressed the information need aspect present in the discussion forums in general. However, helpfulness might be conflated with other reasons such as humour, sentiment in certain domains. We would like to investigate those aspects in the future.

Acknowledgments

We acknowledge the support of NVIDIA Corporation for their donation of the Titan X GPU that facilitated this research. This research is supported by the Singapore National Research Foundation under its International Research Centre.

References

Muhammad Z Asghar, Aurangzeb Khan, Fazal M Kundi, Maria Qasim, Furqan Khan, Rahman Ullah, and Irfan U Nawaz. 2014. Medical Opinion Lexicon: an Incremental Model for Mining Health Reviews. *International Journal of Academic Research*, 6(1):295–302.

Alexander Beloborodov, Pavel Braslavski, and Marina Driker. 2014. Towards Automatic Evaluation of Health-Related CQA Data. In *Proc. of CLEF*, pages 7–18. Springer.

Jiang Bian, Yandong Liu, Eugene Agichtein, and Hongyuan Zha. 2008. Finding the Right Facts in the Crowd: Factoid Question Answering over Social Media. In *Proc. of WWW*, pages 467–476.

Jaime Carbonell and Jade Goldstein. 1998. The Use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries. In *Proc. of SIGIR*, pages 335–336.

Muthu Kumar Chandrasekaran and Min-Yen Kan. 2019. When to reply? context sensitive models to predict instructor interventions in mooc forums. *arXiv preprint arXiv:1905.10851*.

Muthu Kumar Chandrasekaran, Min-Yen Kan, Bernard C. Y. Tan, and Kiruthika Ragupathi. 2015. Learning Instructor Intervention from MOOC Forums: Early Results and Issues. In *Proc. of EDM*, pages 218–225.

Justin Cheng, Lada Adamic, P. Alex Dow, Jon Michael Kleinberg, and Jure Leskovec. 2014. Can Cascades be Predicted? In *Proc. of WWW*, pages 925–936.

Junyoung Chung, Çaglar Gülçehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. In *Proc. of NIPS 2014 Deep Learning and Representation Learning Workshop*.

Charles L.A. Clarke, Maheedhar Kolla, Gordon V. Cormack, Olga Vechtomova, Azin Ashkan, Stefan Büttcher, and Ian MacKinnon. 2008. Novelty and Diversity in Information Retrieval Evaluation. In *Proc. of SIGIR*, pages 659–666.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Liangjie Hong and Brian D Davison. 2009. A Classification-based Approach to Question Answering in Discussion Boards. In *Proc. of SIGIR*, pages 171–178.

Maximilian Jenders, Ralf Krestel, and Felix Naumann. 2016. Which Answer is Best?: Predicting Accepted Answers in MOOC Forums. In *Proc. of Q4APS*, pages 679–684.

Jiwoon Jeon, W Bruce Croft, Joon Ho Lee, and Soyeon Park. 2006. A Framework to Predict the Quality of Answers with Non-Textual Features. In *Proc. of SIGIR*, pages 228–235.

Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. In *Proc. of EMNLP*, pages 1746–1751.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. In *Proc. of ICLR*.

Lei Li, Daqing He, Wei Jeng, Spencer Goodwin, and Chengzhi Zhang. 2015. Answer Quality Characteristics and Prediction on an Academic QA Site: A Case Study on ResearchGate. In *Proc. of WWW*, pages 1453–1458. ACM.

Pengfei Liu, Xipeng Qiu, Yaqian Zhou, Jifan Chen, and Xuanjing Huang. 2016. Modelling Interaction of Sentence Pair with Coupled-LSTMs. In *Proc. of EMNLP*, pages 1703–1712.

Caroline Lo, Justin Cheng, and Jure Leskovec. 2017. Understanding Online Collection Growth Over Time: A Case Study of Pinterest. In *Proc. of WWW 2017 Industry Track*, pages 545–554.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. SummaRuNNer: A Recurrent Neural Network Based Sequence Model for Extractive Summarization of Documents. In *Proc. of AAAI*, pages 3075–3081.

Sanghee Oh and Adam Worrall. 2013. Health Answer Quality Evaluation by Librarians, Nurses, and Users in Social Q&A. *Library & Information Science Research*, 35(4):288–298.

Sanghee Oh, Yong Jeong Yi, and Adam Worrall. 2012. Quality of Health Answers in Social Q&A. *Proc. of AIST*, 49(1):1–6.

Adi Omari, David Carmel, Oleg Rokhlenko, and Idan Szpektor. 2016. Novelty based Ranking of Human Answers for Community Questions. In *Proc. of SIGIR*, pages 215–224.

Tim Rocktäschel, Edward Grefenstette, Karl Moritz Hermann, Tomáš Kočiskỳ, and Phil Blunsom. 2016. Reasoning about Entailment with Neural Attention. In *Proc. of ICLR*.

Aliaksei Severyn and Alessandro Moschitti. 2015. Learning to Rank Short Text Pairs with Convolutional Deep Neural Networks. In *Proc. of SIGIR*, pages 373–382.

Chirag Shah and Jefferey Pomerantz. 2010. Evaluating and Predicting Answer Quality in Community QA. In *Proc. of SIGIR*, pages 411–418.

Jyoti Prakash Singh, Seda Irani, Nripendra P Rana, Yogesh K Dwivedi, Sunil Saumya, and Pradeep Kumar Roy. 2017. Predicting the "Helpfulness" of Online Consumer Reviews. *Journal of Business Research*, 70:346–355.

Ian Soboroff and Donna Harman. 2005. Novelty Detection: The TREC Experience. In *Proc. of HLT-EMNLP*, pages 105–112.

Chengjie Sun, Yang Liu, Change Jia, Bingquan Liu, and Lei Lin. 2017. Recognizing Text Entailment via Bidirectional LSTM Model with Inner-Attention. In *Proc. of ICIC*, pages 448–457.

Mihai Surdeanu, Massimiliano Ciaramita, and Hugo Zaragoza. 2008. Learning to Rank Answers on Large Online QA Collections. In *Proc. of ACL*, pages 719–727.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to Sequence Learning with Neural Networks. In *Proc. of NIPS*, pages 3104–3112.

Di Wang and Eric Nyberg. 2015. A Long Short-Term Memory Model for Answer Sentence Selection in Question Answering. In *Proc. of ACL and IJCNLP*, pages 707–712.

Xin-Jing Wang, Xudong Tu, Dan Feng, and Lei Zhang. 2009. Ranking Community Answers by Modeling Question-Answer Relationships via Analogical Reasoning. In *Proc. of SIGIR*, pages 179–186.

Yuan Yao, Hanghang Tong, Tao Xie, Leman Akoglu, Feng Xu, and Jian Lu. 2015. Detecting High-Quality Posts in Community Question Answering Sites. *Information Sciences*, 302(1):70–82.

Lei Yu, Karl Moritz Hermann, Phil Blunsom, and Stephen Pulman. 2014. Deep Learning for Answer Sentence Selection. In *Proc. of NIPS 2014 Deep Learning and Representation Learning Workshop*.

Cai-Nicolas Ziegler, Sean M. McNee, Joseph A. Konstan, and Georg Lausen. 2005. Improving Recommendation Lists Through Topic Diversification. In *Proc. of WWW*, pages 22–32.