Decision-Focused Summarization

Chao-Chun Hsu
University of Chicago
chaochunh@uchicago.edu

Chenhao Tan
University of Chicago
chenhao@uchicago.edu

Abstract

Relevance in summarization is typically defined based on textual information alone, *without* incorporating insights about a particular decision. As a result, to support risk analysis of pancreatic cancer, summaries of medical notes may include irrelevant information such as a knee injury. We propose a novel problem, decision-focused summarization, where the goal is to summarize relevant information *for a decision*. We leverage a predictive model that makes the decision based on the full text to provide valuable insights on how a decision can be inferred from text. To build a summary, we then select *representative* sentences that lead to similar model decisions as using the full text while accounting for textual non-redundancy. To evaluate our method (DecSum), we build a testbed where the task is to summarize the first ten reviews of a restaurant in support of predicting its future rating on Yelp. DecSum substantially outperforms text-only summarization methods and model-based explanation methods in decision faithfulness and representativeness. We further demonstrate that DecSum is the only method that enables humans to outperform random chance in predicting which restaurant will be better rated in the future.

1 Introduction

Human decision making often requires making sense of a large amount of information. For instance, doctors go through a myriad of medical notes to determine the risk of pancreatic cancer, and investors need to decide whether a stock price will increase based on hundreds of analyst reports. In these cases, summarization can potentially support human decision making by identifying the most relevant information for these decisions (Demner-Fushman et al., 2009; Workman et al., 2012).

Ideally, decision-focused summarization should incorporate insights about how decisions can be inferred from text. However, typical summarization methods in NLP define relevance based on the textual information exclusively. An example desideratum is textual non-redundancy (Carbonell and Goldstein, 1998), which encourages the summaries to cover diverse information in the input documents. Fully optimizing this text-only criterion can be counter-productive for decision making: information about a knee injury does not really help understand the risk of pancreatic cancer, and the disclaimers in financial analysts may not be the most relevant for investment decisions.

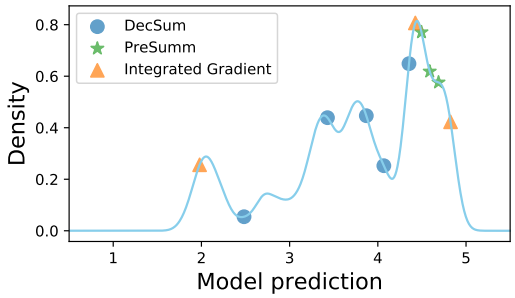


Figure 1: Illustration of the selected sentences by different methods on the distribution of model predictions on all individual sentences. Our method (DecSum) covers the full distribution, while PreSumm, a text-only summarization method, concentrates on the right side, and integrated gradients, a model-explanation method, misses the middle part.

In this work, we investigate the potential of leveraging a supervised decision model for *extractive* decision-focused summarization. A predictive model that learns to make a decision given the full text can encode valuable insights about how the decision can be inferred from text. Given that Kleinberg et al. (2015) shows that many policy problems depend on predictive inference, incorporating model-based insights into summarization can be widely applicable to many decisions in high-stake scenarios such as finance and healthcare.

We propose novel desiderata for decision-focused summarization in addition to textual non-redundancy and formalize them based on model

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 117–132
November 7–11, 2021.

behavior. First, *decision faithfulness* suggests that the selected sentences should lead to the same decision as using the full text based on the model. This desideratum is analogous to sufficiency in evaluating the interpretability of attribution methods (DeYoung et al., 2019), as attribution methods should ideally identify sentences that would "explain" the model's decision with all sentences. This observation also highlights the connection between explanation and decision-focused summarization.

In addition to faithfulness, *decision representativeness* resembles textual non-redundancy in the decision space. Fig. 1 illustrates the decision distribution of all individual sentences in the input documents, i.e., model predictions given each sentence, and sentences chosen by different methods. Ideally, the selected sentences should be representative of this overall decision distribution. Our method is designed to optimize this desideratum, whereas text-only summarization methods and model-based explanation methods do not aim to select sentences that represent the whole distribution.

To evaluate our proposed method, we formulate a future rating prediction task on Yelp, inspired by investment decisions. The task is to predict a restaurant's future rating given the first ten reviews. Automatic metrics demonstrate that our method (`DecSum`) outperforms text-only summarization methods and model-based explanation methods in decision faithfulness and decision representativeness. `DecSum` also improves textual non-redundancy over the baselines, although at the cost of grammaticality and coherence. Human evaluation further shows that `DecSum` is the *only* method that enables humans to statistically outperform random chance in predicting which restaurant will be rated better in the future.

To summarize, our main contributions are:

- We propose a novel summarization task that emphasizes supporting decision making.
- We propose *decision faithfulness* and *decision representativeness* as important desiderata for this task in addition to textual non-redundancy, based on the behavior of a supervised model.
- Using Yelp future rating prediction as a testbed, we show that the proposed approach outperforms text-only summarization methods and model-based explanation methods.
- We show that the proposed approach effectively supports human decision making in a very challenging classification task.

2 Method

In this section, we formalize decision-focused summarization and three desiderata. We then provide a greedy algorithm to optimize the three desiderata.

2.1 Problem Formulation

Decision-focused summarization is conditioned on a decision of interest, e.g., whether a stock price will increase. We refer to this decision as y. It is challenging for humans to make decisions based on the full input text, X, which can be hundreds of analyst reports. The task is thus to identify the most relevant information from the input for a particular decision as a summary in support of human decision making. We formulate the extractive version of decision-focused summarization as follows.

Definition 1 (Decision-focused summarization)**.** Given an input text $X = \{x_s\}_{s=1}^{s=S}$, where S is the number of sentences, select a subset of sentences $\tilde{X} \subset X$ to support making the decision y.

Unlike typical summarization where we only have access to textual information, decision-focused summarization requires knowledge of how the decision can be inferred from the text. Our problem setup thus has a training set analogous to supervised learning, $D_{\text{train}} = \{(X_i, y_i)\}$, which can provide insights on the relation between the text and the decision.

Yelp future rating prediction task. Inspired by investment decisions given analyst reports, we consider a future rating prediction task in the context of Yelp as a testbed. This allows us to have access to both a dataset[1] and participants who may be able to perform this task. Specifically, for each restaurant in Yelp, we define X as the text of the first k reviews and y is the average rating of the first t reviews where $t > k$ so that the task is to forecast future ratings. We use $k = 10$ and $t = 50$ in this work. Our problem is then to select sentences from a restaurant's first 10 reviews in support of predicting its future rating after 50 reviews.

2.2 DecSum

The key intuition of our approach (`DecSum`) is to develop a model that makes the decision given the text ($f : X \rightarrow y$) and then build summaries that can both support this model in making accurate decisions and account for properties in text-only summarization. This model can be seen as a virtual

[1] https://www.yelp.com/dataset.

decision maker and hopefully encodes valuable information of how the decision can be inferred from the text. We obtain f from D_{train} using standard supervised models.

As discussed in §1, decision-focused summaries should satisfy decision faithfulness, decision representativeness, and textual non-redundancy. Next, we formally define these desiderata as objective (loss) functions that can be minimized to extract decision-focused summaries.

Decision faithfulness. The first desideratum is that the selected sentences should lead to similar decisions as the full text: $f(\tilde{X}) \simeq f(X)$. A natural loss function is the absolute difference between $f(\tilde{X})$ and $f(X)$, and here we use its logarithm:

$$\mathcal{L}_{\text{F}}(\tilde{X}, X, f) = \log |f(\tilde{X}) - f(X)|.$$

This desideratum resonates with faithfulness in interpretability (Jacovi and Goldberg, 2020). However, our focus is not on whether the model *actually* uses these sentences in its prediction, but on the behavioral outcome of the sentences, i.e., whether they supports model/human decision making by identifying relevant information for the decision.

Decision representativeness. Sentences in the full input X can lead to very different decisions on their own. Thus, in addition to decision faithfulness, model decisions of selected sentences should be representative of the decision distribution of sentences in the full input (Fig. 1). In other words, the decision distribution of the summary $\hat{Y}_{\tilde{X}} = \{f(x) \mid x \in \tilde{X}\}$ should be close to the decision distribution of all sentences in the full text $\hat{Y}_X = \{f(x) \mid x \in X\}$. To measure the distance between $\hat{Y}_{\tilde{X}}$ and $\hat{Y}_X$, we use the Wasserstein Distance (Ramdas et al., 2017):

$$W(\hat{Y}_{\tilde{X}}, \hat{Y}_X) = \inf_{\gamma \in \Gamma(\hat{Y}_{\tilde{X}}, \hat{Y}_X)} \int_{\mathbb{R}\times\mathbb{R}} ||f - f'|| d\gamma(f, f'),$$

where $\Gamma(\hat{Y}_{\tilde{X}}, \hat{Y}_X)$ denotes the collection of all measures on $\mathbb{R} \times \mathbb{R}$ with marginals $\hat{Y}_{\tilde{X}}$ and $\hat{Y}_X$ on the first and second factors respectively. Our second loss function is then the logarithm of the Wasserstein distance between the decision distribution of the summary and that of the full text:

$$\mathcal{L}_{\text{R}}(\tilde{X}, X, f) = \log(W(\hat{Y}_{\tilde{X}}, \hat{Y}_X)).$$

Textual non-redundancy. Our third desired property is inspired by prior work on diversity in textual summarization: the selected sentences should capture diverse contents and provide an overview of the textual information in the input text (Lin and Bilmes, 2011; Dasgupta et al., 2013; Carbonell and Goldstein, 1998). To operationalize this intuition, we adopt a loss function to encourage sentences in the summary to be dissimilar to each other. We operationalize similarity using the cosine similarity based on SentBERT sentence representation $s(x)$ (Reimers and Gurevych, 2019):

$$\mathcal{L}_{\text{D}}(\tilde{X}) = \sum_{x \in \tilde{X}} \max_{x' \in \tilde{X} - \{x\}} \text{cossim}(s(x), s(x')).$$

To summarize, our objective function consists of the above three parts:

$$\mathcal{L}(\tilde{X}, X, f) = \alpha \mathcal{L}_{\text{F}}(\tilde{X}, X, f) + \beta \mathcal{L}_{\text{R}}(\tilde{X}, X, f) + \gamma \mathcal{L}_{\text{D}}(\tilde{X}),$$

where α, β, γ control the tradeoff between the three desiderata. Note that decision faithfulness ($\mathcal{L}_{\text{F}}$) and decision representativeness ($\mathcal{L}_{\text{R}}$) both rely on f, while textual non-redundancy ($\mathcal{L}_{\text{D}}$) depends on the textual information alone. We use $\log$ in $\mathcal{L}_{\text{F}}$ and $\mathcal{L}_{\text{R}}$ because they are unbounded.

Algorithm implementation. Inspired by traditional summarization methods (Carbonell and Goldstein, 1998; Mihalcea and Tarau, 2004), we develop an iterative algorithm that greedily selects a sentence that minimizes our loss function. A key advantage of this approach is that it exposes the design space and presents a white box for researchers.

Algorithm 1 shows the full algorithm. To select K sentences from input X, in each step $k = \{1, ..., K\}$, we iteratively choose a sentence among the remaining sentences, $\hat{x} \in X - \tilde{X}_{k-1}$, that achieves the lowest loss $\mathcal{L}(\tilde{X}_{k-1} \cup \{\hat{x}\}, X, f)$ where $\tilde{X}_{k-1}$ is the current summary with $k-1$ sentences. When $\beta > 0$, we only use $\mathcal{L}_R$ at the first step to encourage the algorithm to explore the full distribution rather than stalling at the sentence that is most faithful to $f(X)$. In practice, we use beam search with beam size of 4 to improve our greedy algorithm. Our code and data are available at https://github.com/ChicagoHAI/decsum.

3 Experiment Setup

Our approach is contingent on a machine learning model that can make decisions based on the input text. In this section, we discuss our dataset split and choice of this ML model, baselines summarization approaches, and evaluation strategies.

Algorithm 1: `DecSum`

Input: X, f, K
Output: $\tilde{X}$
$\tilde{X} \leftarrow \emptyset,\ k \leftarrow 1$;
while $k \leq K$ **do**
  if $\beta > 0$ *and* $k = 1$ **then**
    $\hat{x} \leftarrow \text{argmin}_{\hat{x} \in X} \mathcal{L}_{\text{R}}(\{\hat{x}\}, X, f)$
  else
    $\hat{x} \leftarrow \text{argmin}_{\hat{x} \in X} \mathcal{L}(\tilde{X} \cup \{\hat{x}\}, X, f)$
  end
  $\tilde{X} \leftarrow \tilde{X} \cup \{\hat{x}\}$;
  $X \leftarrow X - \{\hat{x}\}$;
  $k \leftarrow k + 1$
end

3.1 Regression Model and Baselines

We split the Yelp dataset (18,112 restaurants) into training/validation/test sets with 64%/16%/20% ratio. Since the text of 10 reviews has 1,621 tokens on average, we use Longformer (Beltagy et al., 2020) to fine-tune a regression model. See details of hyperparameter tuning in the appendix.

In addition to Longformer, we also considered logistic regression and deep averaging networks (Iyyer et al., 2015) for this problem. However, we find that only Longformer leads to an appropriate distribution of the predicted score ($f(x)$) at the sentence level (see the appendix), suggesting that Longformer may better generalize to shorter inputs. We refer to this model as the regression model or f to differentiate from summarization methods.

We consider two types of baselines: text-only summarization and model-based explanation.

Text-only summarization baselines. We compare `DecSum` with both extractive and abstractive summarization methods.

- **PreSumm** is an extractive summarization method with hierarchical encoders (Liu and Lapata, 2019). We use distilbert-base-uncased[2] built on the CNN/DM dataset (Hermann et al., 2015), as DistillBERT is competitive with BERT.
- **BART** is a seq2seq model trained with a denoising objective (Lewis et al., 2020). We use bart-large-cnn model fine-tuned on CNN/DM.
- **Random** simply selects random sentences from the input reviews. This method can extract somewhat representative sentences, and we hypothesize that it may be competitive against PreSumm and BART in this task.

[2] https://transformersum.readthedocs.io/en/latest/extractive/models-results.html.

Model-based explanations. PreSumm and BART do not depend on our regression model, we thus consider attribution methods based on the same model that `DecSum` uses as the second type of baselines. These attribution methods used in Jain et al. (2020a) are supposed to extract sentences that explain the model decision.

- **Integrated Gradients (IG)** is a gradient-based method (Sundararajan et al., 2017). Following Jain et al. (2020a), we sum up the importance score of input tokens for each sentence and select top K sentences as the results of IG.
- **Attention** may also be used to interpret transformers. We use the mean attention weights of all 12 heads for the `[CLS]` token at the last layer in Longformer as importance scores for each token, following Jain et al. (2020b). Similar to IG, we rank sentences based on the summed importance scores over tokens in a sentence.

`DecSum`, PreSumm, IG, Attention, and Random can all generate a ranking/order for sentences and allow us to control the summary length.

3.2 Evaluation Metrics and Setup

Our evaluation consists of both automatic metrics and human evaluations. All the evaluations are based on the test set, similar to supervised learning.

Automatic metrics. We design evaluation metrics based on our three desiderata.

- **Faithfulness to the original model prediction.** We rely on the regression model trained based on the full text of the first 10 reviews to measure faithfulness. Specifically, we measure the mean squared error between the predicted score based on the summary with the predicted score of the full text, $(f(\tilde{X}) - f(X))^2$.
- **Representativeness compared to the decision distribution of all sentences.** We measure the Wasserstein distance between the distribution of model predictions of the summary $\hat{Y}_{\tilde{X}}$ and that of all sentences in the first 10 reviews $\hat{Y}_X$.
- **Text-only summary evaluation metrics.** We use SUM-QE (Xenouleas et al., 2019), BERT-based automatic summarization evaluation, to evaluate five aspects, i.e., grammaticality, non-redundancy, referential clarity, focus, and structure & coherence. Note that coherence of decision-focused summaries may differ from that of typical summaries, as they are supposed to provide diverse and even conflicting opinions.

In addition, we also use MSE with the restaurant rating after 50 reviews to measure the quality of the summaries in the forecasting task, $(f(\tilde{X}) - y)^2$.

Human evaluation. While an obvious idea is asking humans to forecast a restaurant's future rating, this regression task is too challenging for humans. It is not humans' strength to tell the difference between 4.1 and 4.2 in average restaurant ratings. Therefore, inspired by prior work on pairwise tasks (Tan et al., 2016, 2014; Zhang et al., 2018), we develop a simplified pairwise classification task: given a pair of restaurants with the same average rating of first 10 reviews, we ask participants to guess which will be rated better after 50 reviews. We ensure that these two restaurants are located in the same city and their rating difference is at least one star after 50 reviews. 1,028 restaurant pairs from the test set satisfy these criteria, and we randomly select 200 pairs for our human evaluation and limit the number of pairs per city to 25.

We use Mechanical Turk to conduct our human evaluation. A crowdworker is shown task instructions, an example pair, 10 pairs of restaurants (main task), and an exit survey. Fig. 2 illustrates the experiment interface of the main task. We only allow participants who have 99% or more HITs acceptance rate, have 50 or more finished HITs, and are located in the US. We also require turkers to spend at least 20 seconds for each pair (the hourly salary is ∼$10). Participants enjoyed our tasks and reported their heuristics in decision making. See appendix for more details of our experiments. We collect three human guesses for each pair and consider four summarization methods. In addition to random[3] and `DecSum`, we choose one text-only summarization method (PreSumm) and one model-based explanation method (IG) according to automatic metrics (see §4.1).

To make sure that the summaries of different methods are comparable to each other, we control for token length in summaries. Recall that the summarization length of BART model is not easily controllable. Thus, we constrain token length of summaries to the average of BART summaries. Specifically, we sequentially select sentences until their length exceeds 50 tokens in the other methods. For `DecSum`, we set $K = 15$ in beam search and then truncate the same way as other methods. As a result, the summaries from all methods are comparable in length (see the appendix).

[3]We considered using the full text of 10 reviews as a baseline. However, participants in pilot studies found the information too overwhelming. Summaries consisting of random sentences provide a more comparable baseline as `DecSum`.

4 Results

In this section, we compare the quality of summaries from our proposed decision-focused summarization with other existing approaches, both through automatic evaluation metrics and human evaluation. Automatic metrics show that `DecSum` provides better decision faithfulness, decision representativeness, textual non-redundancy than other baselines, but sacrifices other text-only qualities such as coherence and grammaticality. Human evaluation shows that `DecSum` also leads to better human decision making.

4.1 Automatic Evaluation

We next evaluate three desired properties in §3.2.

4.1.1 Decision Faithfulness

We measure faithfulness by comparing the prediction derived from the summary with the prediction derived from the 10 reviews (*MSE with full*). Table 1 shows that `DecSum` with all components on, "(1, 1, 1)", achieves much better faithfulness than any of other baselines, close to 0. All the text-only summarization methods have an *MSE with full* of about 0.34, more than 100 times as much as that of `DecSum`. Model-based explanation methods, surprisingly, lead to even poorer faithfulness than text-only methods (IG: ∼0.44; attention: ∼0.54).

Effect of different components. Our first component, decision faithfulness, is critical for achieving low *MSE with full* (all the underlined numbers are below 0.05). Furthermore, textual non-redundancy improves *MSE with full* over optimizing decision faithfulness alone, suggesting that text-only desiderata can in fact support decision making, at least for the AI decision maker.

Using only textual non-redundancy (0, 0, 1), a deep version of Maximum Marginal Relevance (Carbonell and Goldstein, 1998), is not better than other text-only summarization methods, i.e., BART and PreSumm. Interestingly, decision representativeness alone (0, 1, 0) leads to better faithfulness than any other baselines, although not as good as directly optimizing *MSE with full*. Henceforth, we use `DecSum` to refer to the system with all components on (1, 1, 1) unless otherwise specified.

Prediction performance. We also present the MSE with the ground truth rating after 50 reviews.

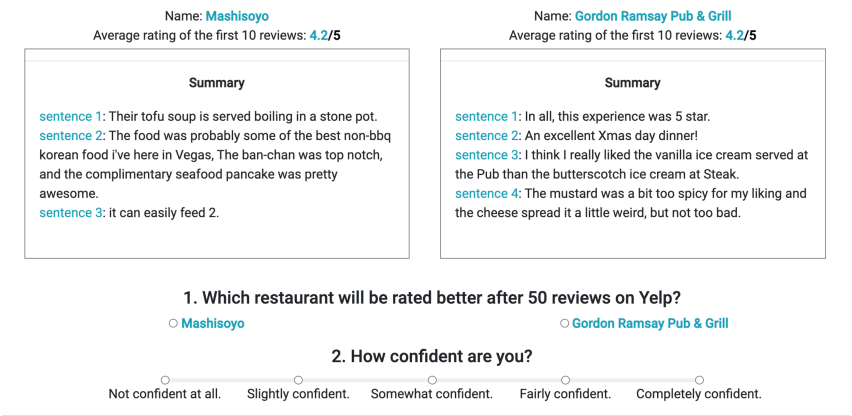


Figure 2: Screenshot of the experiment interface for human evaluation. Participants are asked to predict which restaurant will be rated higher after 50 reviews based on the summaries of the first 10 reviews where these two restaurants have the same average rating in the first 10 reviews.

Method	MSE with **Full** (faithfulness) ↓	MSE ↓
Full (oracle)	0	0.135
Text-only summarization methods		
Random	0.356	0.475
BART	0.368	0.502
PreSumm	0.339	0.478
Model-based explanation methods		
IG	0.436	0.565
Attention	0.539	0.715
`DecSum` w/ (α decision faithfulness, β decision representativeness, γ textual non-redundancy)		
(1, 1, 1)	0.0005	0.136
(1, 1, 0)	0.0378	0.164
(1, 0, 1)	**0.0002**	0.135
(0, 1, 1)	0.162	0.283
(1, 0, 0)	0.0264	0.155
(0, 1, 0)	0.175	0.287
(0, 0, 1)	0.504	0.565

Table 1: MSE of model predictions based on summaries of different methods. **Full** denotes using all reviews without summarization.

As expected, using the full text of all ten reviews achieves the best MSE compared to summarization methods. The prediction performance of summaries is aligned with *MSE with full*. `DecSum` leads to the best performance compared to baseline models. Text-only summarization (PreSumm and BART) provides similar performance as random, and outperforms explanation methods (IG and attention), which again highlights that explanation methods do not lead to good summaries even for model decision making.

4.1.2 Decision Representativeness

We start by measuring the Wasserstein distance between model predictions of the selected sentences

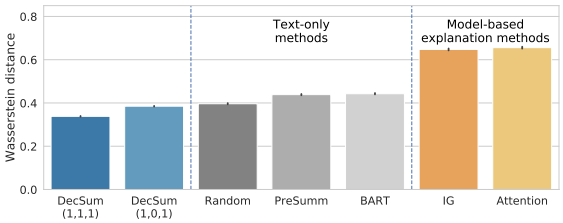
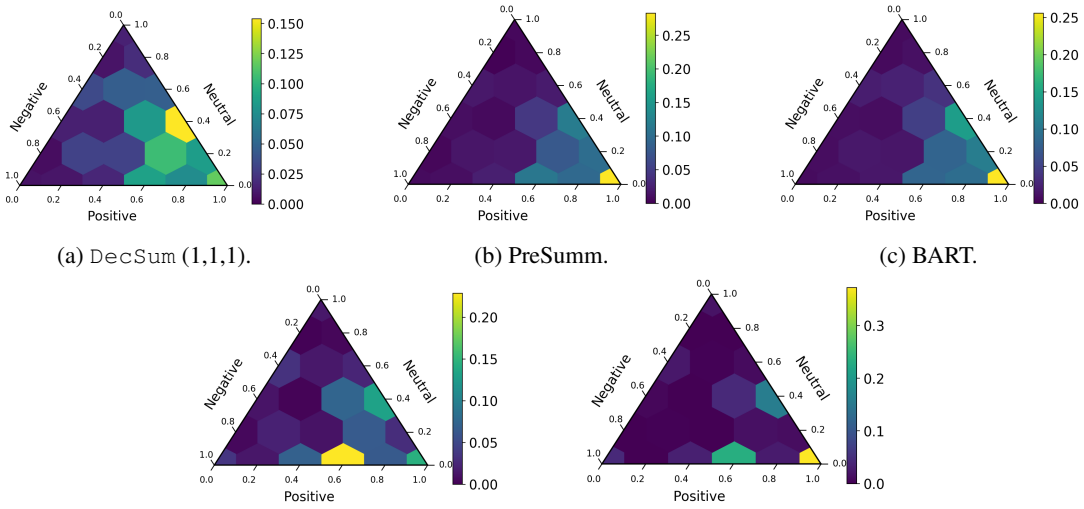


Figure 3: Wasserstien distance between model predictions of summary sentences and all sentences of the first ten reviews. *Lower* values indicate better representativeness. Error bars represent standard errors. `DecSum` $(1, 1, 1)$ is significantly better than other approaches, including `DecSum` $(1, 0, 1)$, with p-value ≤ 0.0001 with paired t-tests.

with those of all the sentences. Fig. 3 shows that `DecSum` is significantly better than random, text-only summarization, and model-based explanation. In other words, `DecSum` can select sentences that are more representative of the decision distribution derived from individual sentences in the first ten reviews. We also compare $(1, 1, 1)$ with $(1, 0, 1)$ to examine the effect of the decision representativeness component. While optimizing decision faithfulness naturally encourages selecting sentences that overall reflect the final decision, the second component further improves the representativeness.

To further examine the effectiveness of our approach, we study the sentiment distribution using an independent classifier other than our own model. We use a pretrained BERT model fine-tuned on sentiment analysis for product reivews[4] to determine the sentiment of sentences. Specifically, the 5-class

[4] https://huggingface.co/nlptown/bert-base-multilingual-uncased-sentiment.



(a) DecSum (1,1,1). (b) PreSumm. (c) BART.

(d) Integrated gradient. (e) Attention.

Figure 4: Sentence-level sentiment distribution of summaries. DecSum can select a wider range of sentences w.r.t. sentiment diversity.

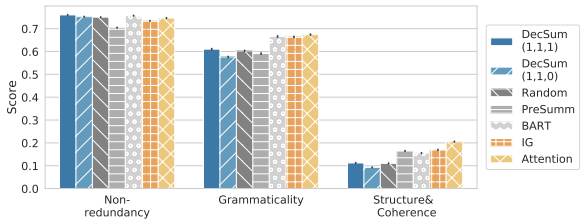


Figure 5: Summary quality evaluation using SUM-QE (Xenouleas et al., 2019). DecSum achieves strong textual non-redundancy, but leads to lower grammaticality and coherence.

sentiment classification model outputs a class with the highest probability, and we define sentences with class 1 and 2 as negative, 3 as neutral, and 4 and 5 as positive. Ideally, a representative summary should cover diverse sentiments. Fig. 4 shows that DecSum can select a more diverse set of sentences with regard to sentiment diversity compared to other methods. PreSumm and BART tend to select positive sentences over negative sentences which results in a less representative summary and can potentially mislead human decision making. In comparison, model-based methods (i.e., IG and attention) tend to avoid neutral sentences.

4.1.3 Text-only Summary Evaluation

Finally, we evaluate textual non-redundancy and other text-only properties commonly used in standard text-only summarization (Fig. 5). Overall, we find that DecSum achieves strong textual non-redundancy (0.760 vs. 0.757 with BART, $p = 0.046$ with paired t-tests; comparisons with other baselines are all statistically significant with $p < 0.001$). In comparison, PreSumm achieves the worst non-redundancy among the baselines. Explanation methods (IG and attention) also provide worse non-redundancy than DecSum, as they do not explicitly optimize textual non-redundancy.

Meanwhile, DecSum leads to inferior performance based on other text-only evaluation metrics such as grammaticality and coherence. Textual non-redundancy improves the grammaticality and coherence compared to (1, 1, 0). Surprisingly, although attention does not take coherence into account, it leads to better coherence than text-only summarization methods. We hypothesize that this is related to the fact that attention tends to select sentences that are more concentrated in sentiment distribution.

4.2 Human Evaluation

As the regression task is simplified to a binary classification task in human evaluations (§3.2), we first obtain model accuracy on the simplified task (Table 3). DecSum is the best summarization method with an accuracy of 76.1%, comparable to using the full text. Among our baselines, only PreSumm achieves above 60% in the simplified task. We choose four methods for our human evaluation based on this result: random as a control condition, PreSumm as the better text-only summarization method, IG as our model-based explanation method, and DecSum.

Fig. 6a shows human performance in this sim-

Method	Restaurant 1: IHOP	Restaurant 2: Tasty Kabob (rated better after 50 reviews.)	#correct
PreSumm	$\tilde{x}_1$: I had a pancake combo with New York cheese cake pancakes and they were delicious ! ! !. $\tilde{x}_2$: This place was great $\tilde{x}_3$: I got to eat breakfast and watch the football game !. $\tilde{x}_4$: Finally a local IHOP , great service and always delicious breakfast. $\tilde{x}_5$: Nice clean place.	$\tilde{x}_1$: Also they have the best Persian Ice Cream which is only one flavor .... $\tilde{x}_2$: what is the flavor?? $\tilde{x}_3$: (its a secret , you will have to go there and find out !). $\tilde{x}_4$: Tasty Kabob is a must see on any Hookah bar tour. $\tilde{x}_5$: Tasty Kabob , while among the best Persian restaurants in Arizona , falls short of Famous Kabob in Sacramento and many Los Angeles joints.	1/3
DecSum	$\tilde{x}_1$: Love this place and they got big screen TV'S always playing football, great idea. $\tilde{x}_2$: My soup came out cold, our server forgot our drinks, and they just microwaved it to warm it up and it literally over cooked everything in the soup. $\tilde{x}_3$: I had a pancake combo with New York cheese cake pancakes and they were delicious!!!	$\tilde{x}_1$: Regardless, both versions were moist and very appealing. $\tilde{x}_2$: If you thought you didn't like Persian food, this place will definitely make you think again. $\tilde{x}_3$: It was a generous portion for two, but I found myself munching on it just to pass the time until our lunches came, not because it was exceptionally well done.	3/3

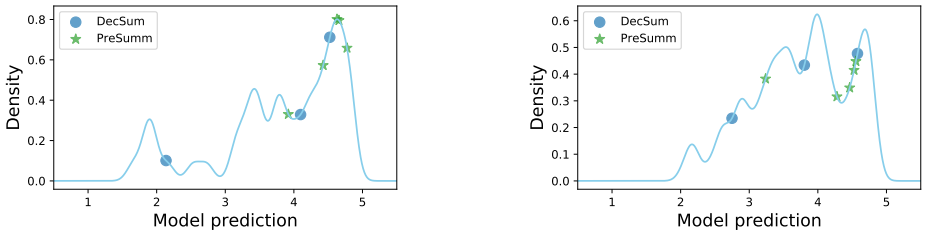
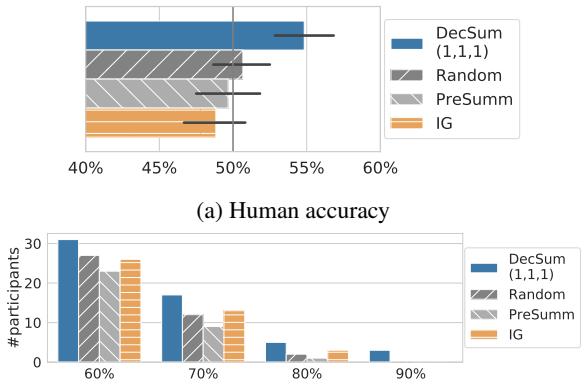


Table 2: Example summaries from PreSumm and DecSum for two restaurants in Tempe, AZ. *#correct* is the result from human evaluation. Dots on the plots represent selected sentences on the distribution of model predictions. DecSum is able to capture sentence $\tilde{x}_2$ with a low predicted rating from reviews of IHOP to help participants distinguish future ratings between two restaurants.

Method	Accuracy (%)	Experiment subset accuracy (%)
Full (oracle)	76.0	85.5
Text-only summarization methods		
Random	55.6	58.0
BART	57.3	60.5
PreSumm	64.4	66.0
Model-based explanation methods		
IG	57.3	59.0
Attention	52.8	52.5
DecSum	76.1	85.5

Table 3: Model performance on the simplified binary classification task as described in §3.2. We sample 200 restaurant pairs for human evaluation.



(a) Human accuracy

(b) #participants with over 60% accuracy

Figure 6: Fig. 6a shows that DecSum is the only method that enables humans to statistically outperform random chance. Fig. 6b further shows that DecSum leads to more individuals with high performance.

plified classification task. This task turns out to be very challenging for humans and the best human performance is only 54.7%, much lower than model accuracy in Table 3. This best human accuracy is achieved with DecSum and is statistically different from 50% (p =0.017), while other baselines are all about chance (50.7%, 49.7%, and 48.8 for Random, PreSumm, and IG respectively; Random indeed slightly exceeds PreSumm and IG as it selects somewhat representative sentences). DecSum also leads to more individuals with great performance: three participants obtained 90% accuracy with DecSum, but none with baseline methods did. 31 participants reached 60%, 4 more than the second best (27 with random).

Although text-only qualities show that summaries from DecSum are less grammatical and coherent, the effect on human perception of usefulness is limited. For instance, while 16 participants with IG strongly agree that summaries were useful in helping decide future ratings compared to 12 with DecSum, 15 with DecSum strongly agree that summaries were useful in helping assess confidence compared to 10 with IG.

Finally, Table 2 shows summaries of the same restaurant pair from DecSum and PreSumm, and the distribution plots present the corresponding se-

lected sentences on the distribution of model predictions from all sentences. Summaries from DecSum can better present the overall distribution and allow participants to evaluate these two restaurants. For example, DecSum includes a negative sentence ($\tilde{x}_2$) from IHOP reviews to help users determine that IHOP is not better rated. In contrast, PreSumm only selects positive sentences and fails to form a decision-representative summary.

5 Related Work

We review additional related work in three areas: query/aspect-based summarization, forecasting with NLP, and evaluation of summarization.

Our problem formulation is closely related to query-focused summarization (Daumé III and Marcu, 2006; Wang et al., 2014; Schilder and Kondadadi, 2008; Damova and Koychev, 2010). In fact, Wang and Cardie (2012) also uses the term "decision" and provides summaries for each decision made in a meeting. Note that relevance in query-focused summarization is still based on textual information, whereas we incorporate potential insights about a decision from a supervised model into the summarization framework. For example, query-focused summarization for pancreatic cancer may summarize all sentences that mention pancreas, but a supervised model may learn that smoking relates to pancreatic cancer and our approach then includes smoking history in the summary.

Similar to our work, aspect-based summarization uses a predictive model to provide summaries for food, service, decor for reviews (Titov and McDonald, 2008). Another related direction is identifying helpful sentences in product reviews (Gamzu et al., 2021). It is useful to highlight our motivation in support decision making in *challenging* tasks towards effective human-AI collaboration (Green and Chen, 2019; Lai et al., 2020; Lai and Tan, 2019). Unlike tasks such as textual entailment where models aim to emulate human intelligence, forecasting future outcomes, such as stock markets (Xing et al., 2018) and message popularity (Tan et al., 2014), is challenging both for humans and for machines. Humans and machines may offer complementary insights in these tasks. We chose restaurant rating prediction as an example about which laypeople may have valid intuitions. We thus also propose novel desiderata, decision faithfulness and decision representativeness.

Evaluation of summarization is very challenging, partly because the goal of summarization is usually vague (Nenkova and Passonneau, 2004; Fabbri et al., 2021). Popular metrics such as ROUGE require reference summaries (Lin, 2004), but it is unclear that humans can provide useful summaries for decision making in challenging tasks given their limited performance and the scale of inputs. Our formulation adopts a task-driven evaluation, i.e., human performance on the decision task which the summaries are supposed to support. This resembles application-based evaluation of explanations in interpretability (Doshi-Velez and Kim, 2017).

6 Conclusion

We propose a novel task, decision-focused summarization, and demonstrate that DecSum outperforms text-only summarization methods and model-explanation methods in both automatic metrics and human evaluation. There are many exciting future directions in advancing decision-focused summarization to support human decision making. In particular, our human evaluation demonstrates a substantial gap between human performance and model performance. One possible approach is to leverage visualizations similar to Fig. 1 to enable interactive summarization so that users can see the decision variance and explore the textual information beyond a static set of sentences. As humans are final decision makers in a wide variety of high-stake scenarios, ranging from healthcare to justice systems, it is critical to investigate human-centered approaches to support human decision making.

Ethics considerations. Our work promotes intelligent models that can be used to support human decision making. We advocate the perspective of augmented intelligence: the goal of our system is to best support humans as final decision makers instead of maximizing model performance. However, in decisions with fairness concerns (e.g., bailing decisions), important future directions include examining fairness-related metrics for the summaries and human-AI interaction.

Acknowledgement. We thank anonymous reviewers for their valuable feedbacks. We thank Rebecca Willett, Kevin Gimpel, and members of the Chicago Human+AI Lab for their insightful suggestions. This work is supported in part by research awards from Amazon, IBM, Salesforce, and NSF IIS-2125116, 2126602.

References

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 335–336.

Mariana Damova and Ivan Koychev. 2010. Query-based summarization: A survey.

Anirban Dasgupta, Ravi Kumar, and Sujith Ravi. 2013. Summarization through submodularity and dispersion. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1014–1022.

Hal Daumé III and Daniel Marcu. 2006. Bayesian query-focused summarization. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 305–312, Sydney, Australia. Association for Computational Linguistics.

Dina Demner-Fushman, Wendy W Chapman, and Clement J McDonald. 2009. What can natural language processing do for clinical decision support? *Journal of biomedical informatics*, 42(5):760–772.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C Wallace. 2019. Eraser: A benchmark to evaluate rationalized nlp models.

Finale Doshi-Velez and Been Kim. 2017. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

Alexander R Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2021. Summeval: Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics*, 9:391–409.

Iftah Gamzu, Hila Gonen, Gilad Kutiel, Ran Levy, and Eugene Agichtein. 2021. Identifying helpful sentences in product reviews. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 678–691, Online. Association for Computational Linguistics.

Ben Green and Yiling Chen. 2019. The principles and limits of algorithm-in-the-loop decision making. *Proceedings of the ACM on Human-Computer Interaction*, 3(CSCW):1–24.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in neural information processing systems*, pages 1693–1701.

Mohit Iyyer, Varun Manjunatha, Jordan Boyd-Graber, and Hal Daumé III. 2015. Deep unordered composition rivals syntactic methods for text classification. In *Association for Computational Linguistics*.

Alon Jacovi and Yoav Goldberg. 2020. Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4198–4205, Online. Association for Computational Linguistics.

Sarthak Jain, Sarah Wiegreffe, Yuval Pinter, and Byron C. Wallace. 2020a. Learning to faithfully rationalize by construction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4459–4473, Online. Association for Computational Linguistics.

Sarthak Jain, Sarah Wiegreffe, Yuval Pinter, and Byron C Wallace. 2020b. Learning to faithfully rationalize by construction. *arXiv preprint arXiv:2005.00115*.

Jon Kleinberg, Jens Ludwig, Sendhil Mullainathan, and Ziad Obermeyer. 2015. Prediction policy problems. *American Economic Review*, 105(5):491–95.

Vivian Lai, Han Liu, and Chenhao Tan. 2020. "why is 'chicago' deceptive?" towards building model-driven tutorials for humans. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–13.

Vivian Lai and Chenhao Tan. 2019. On human predictions with explanations and predictions of machine learning models: A case study on deception detection. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 29–38.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Hui Lin and Jeff Bilmes. 2011. A class of submodular functions for document summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 510–520.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411, Barcelona, Spain. Association for Computational Linguistics.

Ani Nenkova and Rebecca J Passonneau. 2004. Evaluating content selection in summarization: The pyramid method. In *Proceedings of the human language technology conference of the north american chapter of the association for computational linguistics: Hlt-naacl 2004*, pages 145–152.

Aaditya Ramdas, Nicolás Trillos, and Marco Cuturi. 2017. On wasserstein two-sample testing and related families of nonparametric tests. *Entropy*, 19(2):47.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Frank Schilder and Ravikumar Kondadadi. 2008. Fastsum: Fast and accurate query-based multi-document summarization. In *Proceedings of ACL-08: HLT, short papers*, pages 205–208.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pages 3319–3328. PMLR.

Chenhao Tan, Lillian Lee, and Bo Pang. 2014. The effect of wording on message propagation: Topic- and author-controlled natural experiments on twitter. In *Proceedings of ACL*.

Chenhao Tan, Vlad Niculae, Cristian Danescu-Niculescu-Mizil, and Lillian Lee. 2016. Winning arguments: Interaction dynamics and persuasion strategies in good-faith online discussions. In *Proceedings of WWW*.

Ivan Titov and Ryan McDonald. 2008. A joint model of text and aspect ratings for sentiment summarization. In *proceedings of ACL-08: HLT*, pages 308–316.

Lu Wang and Claire Cardie. 2012. Focused meeting summarization via unsupervised relation extraction. In *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 304–313, Seoul, South Korea. Association for Computational Linguistics.

Lu Wang, Hema Raghavan, Claire Cardie, and Vittorio Castelli. 2014. Query-focused opinion summarization for user-generated content. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1660–1669, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

T Elizabeth Workman, Marcelo Fiszman, and John F Hurdle. 2012. Text summarization as a decision support aid. *BMC medical informatics and decision making*, 12(1):1–12.

Stratos Xenouleas, Prodromos Malakasiotis, Marianna Apidianaki, and Ion Androutsopoulos. 2019. SUM-QE: a BERT-based Summary Quality Estimation Model. In *EMNLP-IJCNLP 2019*, Hong Kong.

Frank Z Xing, Erik Cambria, and Roy E Welsch. 2018. Natural language based financial forecasting: a survey. *Artificial Intelligence Review*, 50(1):49–73.

Justine Zhang, Jonathan Chang, Cristian Danescu-Niculescu-Mizil, Lucas Dixon, Yiqing Hua, Dario Taraborelli, and Nithum Thain. 2018. Conversations gone awry: Detecting early signs of conversational failure. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1350–1361, Melbourne, Australia. Association for Computational Linguistics.

A Model Training Details and Comparisons with DAN and LR

We fine-tune Longformer with 102M parameters on Nvidia RTX Titan GPU with half precision using Huggingface transformers package (Wolf et al., 2020). We use AdamW (Loshchilov and Hutter, 2017) optimizer with learning rate 5e-5 and linear warm-up of 500 steps. We train Longormer for 3 epochs where the batch size is 4 and the maximum input token length is 3,000. We search for hyperparameters for epochs $\{3, 4, 5\}$ and max sequence lengths $\{2000, 3000\}$ and choose the model checkpoint with lowest MSE on the validation set. The model training time of Longformer is about an hour per epoch. The beam search algorithm takes 3 days to find 15 sentences[5] for processing the whole test set (3,623 restaurants) if the setting includes the faithfulness component. Without faithfulness component, `DecSum` takes less than an hour on the test set.

[5]We do not need that many sentences for the main paper, but we did that to understand the effect of summary length.

Besides Longformer, we have tried logistic regression (LR) and Deep Averaging Networks (DAN) as our regression model. However, as shown in Fig. 7, only Longformer can provide appropriate prediction distributions of individual sentences. We group restaurants into four groups where their average ratings of first 10 reviews are in $[1.5, 2.5)$, $[2.5, 3.5)$, $[3.5, 4.5)$, and $[4.5, 5]$ as group 2, 3, 4, and 5 respectively. Then, we use a regression model trained with full 10 reviews $f : X \rightarrow y$ to predict ratings of individual sentences from different restaurants in the group. Finally, we use Gaussian kernel density function to obtain the score distribution and plot sentence score distributions of different groups in the same figure. Note that we do not show restaurants with ratings in the range of $[0, 1.5)$ because there are only a very small number of restaurants in this range. We can see that the distributions from LR and DAN are close to normal distributions with different means for each group. More importantly, LR and DAN are not robust to distribution shift of input length, where the models are trained with full 10 reviews and are tested on individual sentences. LR can make predictions beyond 5 stars and DAN even makes predictions above 15. In comparison, Longformer is able to distinguish positive, neural, and negative sentences and the distributions of different groups also reflect the sentiment distributions of each group.

B The Effect of Summary Length

To generate DecSum summaries in this paper, we use beam search to find 15 sentences for each restaurant and then truncate these sentences at the one that exceeds 50 tokens as summaries in our evaluation section. Fig. 8a shows the average token length of different methods after controlling for the length. They are all comparable to each other.

Next, we investigate the effect of summary length in model prediction. Note that we do hard truncation without considering the sentence boundaries in this section, so the results are not directly comparable to Table 1 and Table 3 in the main paper. As show in Fig. 8, BART summaries do not improve along with the increase of token length because its average token length is only 60 where other extractive summarization approaches can select as many sentences as the full text in ten reviews. It's worth noting that random baseline becomes better than other baseline except IG after 100 tokens.

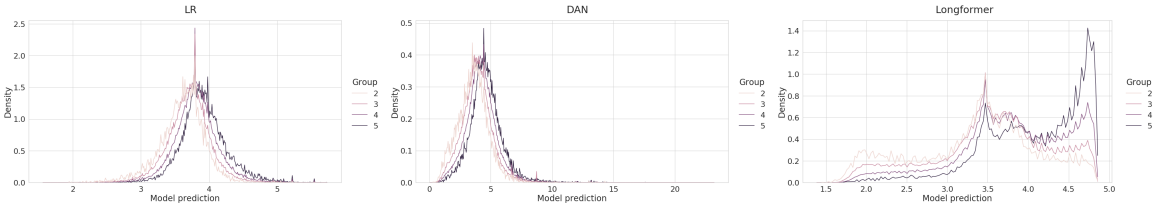
Method	MSE with **Full** (faithfulness) ↓	MSE ↓
Full (oracle)	0	0.135
DecSum (selected order) w/ (α decision faithfulness, β decision representativeness, γ textual non-redundancy)		
(1, 1, 1)	0.028	0.157
(1, 1, 0)	0.076	0.200
(1, 0, 1)	**0.024**	0.154
(0, 1, 1)	0.174	0.288
(1, 0, 0)	0.069	0.188
(0, 1, 0)	0.180	0.290
(0, 0, 1)	0.537	0.588

Table 4: MSE of model predictions based on summaries of DecSum where the sentences are concatenated with the **selected order** which is different from DecSum algorithm.

The reason can be that random selection is more representative of the original reviews compared to PreSumm and attention methods. We also present model accuracy of the simplified task on various token lengths. Fig. 8c shows DecSum still outperforms baselines substantially. PreSumm is the second best model but is surpassed by IG after 120 tokens.

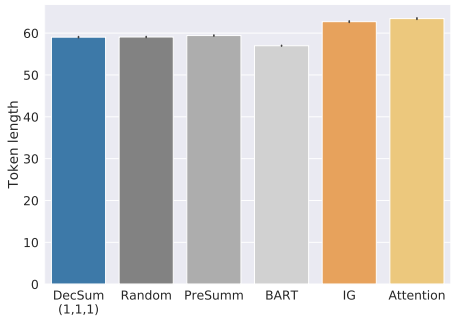
C The Effect of Sentence Order

While computing the score of decision faithfulness component in DecSum algorithm, we concatenate the selected sentences in the original order of the first ten reviews. We find that the LongFormer supervised model is sensitive to the sentence order of summary. For example, for three selected sentences x_1, x_4, x_8 from the first ten reviews $X = \{x_1, x_2, ..., x_i, ..., x_N\}$ where i is sentence index of concatenated first ten reviews, summary constructed from the selected order of DecSum, e.g., x_8, x_1, x_4, yields different results from summary with the original order x_1, x_4, x_8. As shown in Table 4 and Table 5, summaries built from selected order, which is different from DecSum algorithm, weaken the performance of DecSum on the decision faithfulness objective, and diminish the predictive power of the supervised model on simplified binary classification task. Thus, building a supervised model which is robust to different sentence orders in the summary can be a future direction to pursue.

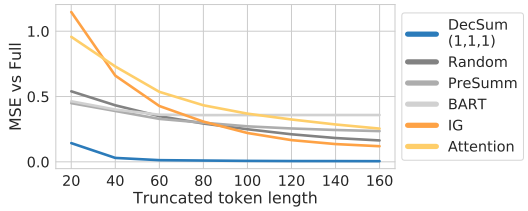


(a) Logistic Regression. (b) DAN. (c) Longformer.

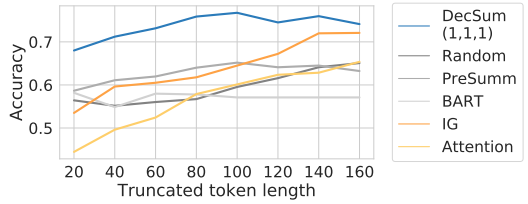
Figure 7: Model prediction distributions of each rating group from logistic regression (LR), deep averaging networks (DAN), and Longformer. Only Longformer model can properly distinguish sentences located at different score range. LR and DAN are not robust to input length shift where models are trained with input of full 10 reviews but are tested with sentences.



(a) Token lengths of summarization approaches for human evaluation. The summary lengths are comparable after length truncation.



(b) Faithfulness with predictions from all reviews using various token lengths.



(c) Model accuracy on the simplified classification task with different input token lengths.

Figure 8: The effect of summary length.

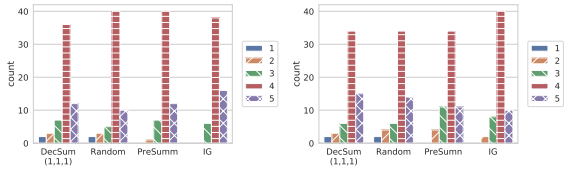
Method	Accuracy (%)	Experiment subset accuracy (%)
Full (oracle)	76.0	85.5
`DecSum`		
(original order)	76.1	85.5
(selected order)	73.8	75.0

Table 5: Comparison between `DecSum` with different sentence order methods on the simplified binary classification task.

D Human Evaluation Details and Additional Results

To choose 200 restaurant pairs for human evaluation, we randomly select from eligible restaurant pairs and limit restaurants per city to 25. We make sure a restaurant does not appear twice in a HIT with 10 restaurant pairs. In the end, 320 restaurants are used in human study, including 2 restaurants for the example pair. In human evaluation, we disallow duplicate participants in our HITs by checking the worker id. We rejected 5 assignments for submitting a confirmation code but not actually doing the experiment. The human study takes about 10 minutes for crowdworkers on average.

In the exit survey, many participants found our experiment interesting and the experience was smooth. They also shared the heuristics used while doing the HITs. For example, *"For the most part, I considered the tone of the reviews. If one review had a more positive tone than the other, I figured that one would get better reviews in the future"* and *"I only used the summaries. I decided based on what I thought seemed like it was an ongoing issue. I didn't read too much into them if it seemed like it was a one-off issue."* Some people may rely on information beyond reviews: *"I focused mostly on the summaries. However, when summaries weren't enough I also focused on the locations and names."* As for the experiment experience, one participant indicated, *"I really enjoyed this survey, and it was unique/different in many aspects, and one of my favorite things to do is read reviews so it was actually fun for me."*. Another crowdworker said, *"The experiment was easy to follow and enjoyable because it was not like any others."* Also, *"I basi-*



(a) Usefulness in deciding rating.

(b) Usefulness in assessing confidence.

Figure 9: Self-reported usefulness.

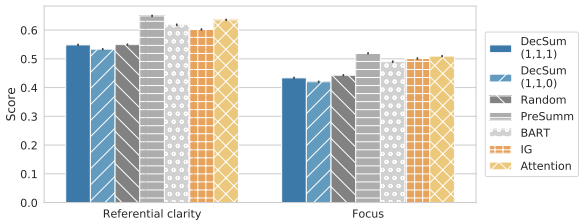


Figure 10: SUM-QE evluation on referential clarity and focus.

cally felt like I was guessing considering I got the practice question wrong but I did give my earnest best answers. Interesting and engaging, thank you." However, a small fraction of participants found that the 20-second timer is too long and preferred a timer of 10 or 15 seconds.

Participants provided self-reported usefulness rating as shown in Fig. 9. In general, these self evaluations are not correlated to the actual performance on the simplified task.

E Summary Quality

Fig. 10 shows two additional summary quality evaluations with SUM-QE, referential clarity and focus. As `DecSum` encourages textual non-redundancy, `DecSum` is worse than text-only summarizations and model-based explanations on these two metrics.

F More Example Summaries from Experiment Subset

In Table 6 and Table 7, we present more example restaurant pairs along with model prediction distributions.

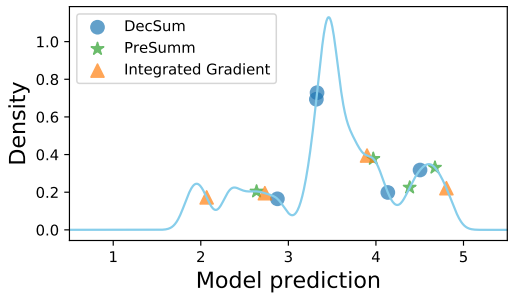
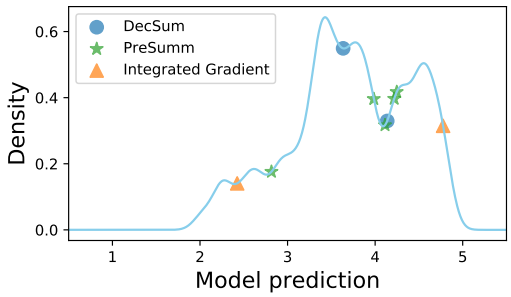
method	Jimmy John's	Bueno Burger (rated better after 50 reviews.)
Random	$\tilde{x}_1$: I was impressed - not only was it fresh, but the bread was delicious. $\tilde{x}_2$: All of the kids acted completely un interested in making sure we had what we needed. $\tilde{x}_3$: My decision: a turkey unwich. $\tilde{x}_4$: The only thing that I was disappointed with $\tilde{x}_5$: Now I crave sandwiches from there.	$\tilde{x}_1$: They cook the patties to pink in the middle, so if you like $\tilde{x}_2$: Not to mention the flavor from the mesquite grill which is unlike anything else out there. $\tilde{x}_3$: Went here with a couple of other people. $\tilde{x}_4$: Perhaps they didn't finish constructing the place? $\tilde{x}_5$: My friend who ordered the hot dog said the bun was hard, and he felt it was stale vs. being toasted.
PreSumm	$\tilde{x}_1$: I highly recommend JJ 's on Hayden ! ! ! !. $\tilde{x}_2$: I noticed that a new Jimmy John 's had opened on Hayden in McCormick Ranch. $\tilde{x}_3$: We used to order sandwiches a couple times a week at work in Seattle. $\tilde{x}_4$: When I moved to Scottsdale about a year ago , I would make the drive to North Scottsdale (8 - 10 miles) just for a delicious sandwich.	$\tilde{x}_1$: I split the an Arizona style and a Glendale style burger. $\tilde{x}_2$: This restaurant just opened 10/21/11. $\tilde{x}_3$: Gary also brought around a " Arizona " style burger that had been mis - ordered. $\tilde{x}_4$: The Arizona burger was definitely the better of the two. $\tilde{x}_5$: Gary hooked my wife up with their super - fire - burner - hot sauce , and while I ca n't do those types of things , my wife said it has great flavor and was indeed very hot.
IG	$\tilde{x}_1$: Great food, amazing customer service, and a great atmosphere. $\tilde{x}_2$: Terrific bread, great tasting sandwich! Music was too loud to hold a conversation and the staff seem disinterested. $\tilde{x}_3$: Food was great but employees were disgusting. $\tilde{x}_4$: Was told they couldn't and the reason is "it's against company policy".	$\tilde{x}_1$: With so many burger joints out there offering a lot of the same, Bueno Burger offers fresh local ingredients and a unique menu which allows you to customize your burger experience. $\tilde{x}_2$: The tables are rickety, the lighting is weird, and particle board design has not sufficiently replaced the skeleton of Boston Market.
DecSum	$\tilde{x}_1$: Thank you, Jimmy John's! :) $\tilde{x}_2$: It took everything inside of me not to walk back in and put them in their place. $\tilde{x}_3$: Thank you Jimmy John's, for adding a little brightness to my day. $\tilde{x}_4$: Freaky fast! $\tilde{x}_5$: The kids pointed and laughed behind his back while mocking him as he walked away.	$\tilde{x}_1$: The chimi and burger were full sized, however I'm used to a bit more fries (and/or rings), but personally i'm trying to avoid the "super size" mentality, so it's fine by me. $\tilde{x}_2$: Showing up at a new restaurant on opening day is a real treat because usually it's about the only time you'll see owners and managers.
		

Table 6: Restaurant pair at Scottsdale, AZ.

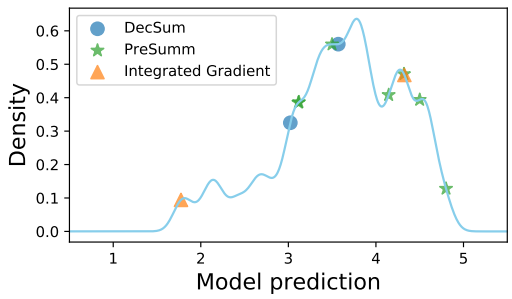
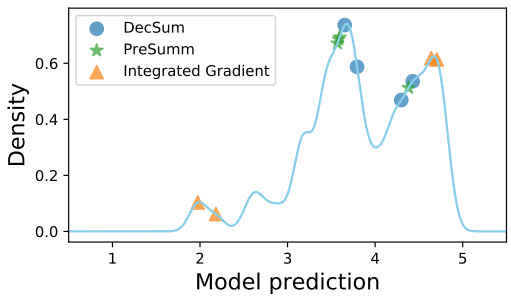
method	Village Pub & Poker (rated better after 50 reviews.)	Cafe Pan
Random	$\tilde{x}_1$: but I sure was glad to have ordered my burger $\tilde{x}_2$: he told me" the girls". $\tilde{x}_3$: the prime rib was OK... $\tilde{x}_4$: They started going to the Village Pub several months ago and have been telling me how great the food is and how reasonable the prices were, so I was looking forward to it.	$\tilde{x}_1$: They are in the same location as Cocolini but have changed the name. $\tilde{x}_2$: They are in the same location as Cocolini but have changed the name. $\tilde{x}_3$: and it was wonderful! $\tilde{x}_4$: Not too bad in the new Vegas. $\tilde{x}_5$: Arrangement is a bit slapped together for a $12 dessert crepe.
PreSumm	$\tilde{x}_1$: great food good prices $\tilde{x}_2$: 1.00 ellis island beer $\tilde{x}_3$: 6.99 steak salad bake potatoe as good as its gets $\tilde{x}_4$: i felt guilty when i paid my bill $\tilde{x}_5$: i thought they made a mistake $\tilde{x}_6$: This is our favorite place. $\tilde{x}_7$: This is a local chain w/ locations all over the valley.	$\tilde{x}_1$: The ham and cheese croissant sandwich was a great on - the - go breakfast. $\tilde{x}_2$: Located in the food court off the casino floor of the Venetian. $\tilde{x}_3$: Not too bad in the new Vegas. $\tilde{x}_4$: Located in the food court off the casino floor of the Venetian.
IG	$\tilde{x}_1$: They started going to the Village Pub several months ago and have been telling me how great the food is and how reasonable the prices were, so I was looking forward to it. $\tilde{x}_2$: This was by far the worst service I have received in a long time.	$\tilde{x}_1$: Had a very bad experience here. $\tilde{x}_2$: It was our first time eating at this place and we definetly wouldn't recommend it to anybody else. $\tilde{x}_3$: I went back later for gelato, and that was incredible, as well. $\tilde{x}_4$: Possibly the best espresso I've had outside of Europe.
DecSum	$\tilde{x}_1$: Oh, how I wish that this place was able to take advantage of its Desert Shores location and offer outside dining on the lake, but it's angled location makes that impossible. $\tilde{x}_2$: The food was okay and the prices were reasonable, but unless my parents are treating I'm not going back.	$\tilde{x}_1$: It is near other food places, almost like a food court and plenty of seating available. $\tilde{x}_2$: Will update this review next time after I try them. $\tilde{x}_3$: but they're known for their crepes, gelato and what I usually get is the waffles. $\tilde{x}_4$: There are MANY restaurants and coffee shops to eat at...
		

Table 7: Restaurant pair at Las Vegas, NV.