Key point Analysis: Improvement with better Candidate Extraction

Akhil Ayyanki, Chaitanya Rajesh, Praneeth Reddy, Veda Sree Bojanapally *
{aayyanki, cbanala, schirasani, vbojanapally}@umass.edu

1 Introduction

With the ever-increasing demand for e-commerce and usage of digital applications for different needs, users' dependence on reviews and ratings for their decisions has surged up. However, given that there are thousands of reviews online, it is often challenging to read through all of them.

Sometimes these reviews can be repetitive and they might miss reviews that are specific on particular features. It is generally an inefficient task to go through all of them. Hence, users read only a part of them, or they decide only based on numerical ratings. In any case, a significant part of the information is being ignored, which might cause a biased decision. Therefore, a quantitative and textual summary of these reviews would help make exhaustive use of the information in hand.

In a practical real world scenario, summarization task of product reviews can show a significant influence on businesses. It provides them with useful knowledge of user's opinions in a compact form that can be exploited to improve their products. The prime motivation to study this problem is that the techniques learnt in solving this problem can be generalized and extended to other domains like summarizing news articles etc. With the internet filled with lengthy articles, this kind of task comes in handy in extracting useful information, thereby saving user's time.

There has been significant improvements in the field of NLP recently with the transformer architecture and transfer learning. Transformer models combined with self-supervised pre-training such as BERT, GPT-2, RoBERTa have shown to be a powerful framework for producing general language learning, achieving state-of-the-art performance when fine-tuned on a wide array of language tasks. Taking advantage of transfer learning, we explore summarization of reviews using keypoints that are generated by using a model that measures quality and rank them based on their match with a whole set of reviews for a business.

In this Project, we explore transfer learning on a wide range of tasks with different kinds of data for each task. We use Bert-based model Roberta-large for all these tasks in our baseline and a sequence to sequence model based on Pegasus to automatically generate Keypoint Candidates from reviews and compare how it performs with keypoints that are created from a argument quality model

2 What you proposed vs. what you accomplished

- Generation of Keypoints
  - Rule based + High quality Candidates
  - Model based
  - ~~Using multiple sentences to generate keypoints~~
    - Achieved with model based, in which multiple sentences are used by the model to give high quality keypoint
- Matching model to match Keypoints with reviews
- Quality model to find quality of Keypoints and reviews
- Sentiment Model application on Keypoints
- Collective Keypoint Mining
- Error Analysis

3 Related work

Early work in the field of summarization focused to extract and quantify the sentiment toward the main aspects of the reviewed entity (Snyder and

Barzilay (2007),Titov and McDonald (2008)). Such aspect-based summaries are highly informative, but it is hard for a user to understand why an aspect received a particular rating, lacking explanations and justifications for the assigned score.

There has been another line of work such as multi-document summarization, that aims to create a textual summary from input reviews. (Ganesan et al. (2010)) uses a graph-based approach to summarize highly redundant opinions. In a similar way, a recent research(Alshomary et al. (2021)) looked at another graph-based extractive summarization model for generating key points. It utilizes a PageRank variant to rank sentences in the input arguments by quality and predicts the top-ranked sentences to be key points. In addition experimentation(Alshomary et al. (2020)) they performed aspect identification on arguments, followed by aspect clustering to ensure diversity. where as (Bražinskas et al. (2020)) adapts an unsupervised framework for Summarization as copycat-review generation. These approaches are able to provide more detail but lacking a quantitative view of the data. Also, the salience of each element in the summary is not indicated, which doesn't allow to evaluate their relative significance. To capture opposing viewpoints, a summary should ideally indicate the proportion of favorable vs unfavorable reviews of each viewpoints.

In order to address the limitations of above approaches, recently a novel extractive summarization framework called Key Point Analysis (KPA) has been proposed by (Bar-Haim et al. (2020a)) in the context of argument summarization. Their approach aims to match each argument to as short list of key points, defined as high-level arguments. These were manually composed by an expert and matched automatically. To further improvement this, (Bar-Haim et al. (2020b)) tries to semi automate the creation of Key points(KP). The set of key points is selected out of a set of candidates - short input sentences with high argumentative quality so that together they achieve high coverage while aiming to avoid redundancy. The resulting summary provides both textual and quantitative views of the data. The high argumentative quality of the candidates is inferred by training a model with IBM-ArgQ-Rank-30kArgs dataset of (Gretz et al. (2020a)). Using this argument quality score of each comment is computed and only high quality candidates are included

To further improve the **candidate selection** or **generation**, (Bar-Haim et al. (2021)) proposed some improvements that they thought might help in this approach. As a part of it, they defined some requirements for KP in review summarization such as validity, Sentiment, Informativeness, should talk of only single aspect. Further building on that approach and the original KPA we propose an automatic KP generation approach that could help to achieve the same result based on Zhang et al. (2019) which is more effective in modeling the dependencies present in the long sequences encountered in summarization.

4 Datasets

For our work, we are using Yelp open dataset[1] for the task of review summarization. In addition to this, we use argumentation datasets to fine tune our matching and quality models.

4.1 Yelp Open Dataset

Yelp dataset is a subset of Yelp's businesses, reviews and user data put together for academic or research purposes. The most recent dataset contains about 8.5million reviews of about 150k businesses.

We used this dataset for MLM pre-training objective on the review training data as well as Key Point Analysis. The dataset contains 5 json files namely, yelp_academic_dataset_business.json, yelp_academic_dataset_checkin.json, yelp_academic_dataset_review.json, yelp_academic_dataset_tip.json, yelp_academic_dataset_user.json containing various information about each business. The files for our concern are yelp_academic_dataset_review.json and yelp_academic_dataset_business.json. In yelp_academic_dataset_review.json, we have the 8million review texts of various lengths, their corresponding business ids, star ratings and other attributes. yelp_academic_dataset_business.json mainly contains the mapping between the various business_ids to the categories that fall under that specific id. Our first task is to choose which category we would like to focus and then taking a

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

	Split	Arguments
Train	25%	1,345,639
Val	25%	1,379,989
Test	50%	2,698,799

Table 1: Yelp dataset split

Split	Arguments
Train	20974
Dev	3208
Test	6315

Table 2: ArgQ-Rank-30k dataset split

subset of the original dataset as there are about 8 million reviews and this would be computationally infeasible to process with resources in hand.

4.1.1 Preprocessing

For the initial task of further pre-training the ROBERTA-large model, we use a subset of 1.5 million reviews. These reviews are taken from all categories as our aim here was to further pre-train so that the model learns review structure. For this task, each review text is further split into single sentences and ROBERTA-large model is further pre-trained with Masked Language Modelling(MLM) objective.

Apart from this, we create a dataset with train/val/test split as displayed in Table 4 from which we sample data for various other tasks. We create this dataset from reviews across two domains RESTAURANTS(5.5 million reviews) and HOTELS(385k reviews) which has the largest number of reviews. When creating this spilt, we remove businesses which has less than 50 reviews and reviews which has sentences greater than 15

4.2 IBM ArgKP dataset

ArgKP[2] is a dataset created by IBM for argument to key-point mapping task. It contains around 24k examples with (argument,key-point,label) where label $\in$ {0,1} denoting a match (1) / mismatch (0) for 24 topics. Out of the 24,093 data tuples 4,998 are positive. It has 6515 distinct arguments and 243 unique key-points. For each of the (argument, key-point) pair, the topic and stance are also indicated. The purpose of this dataset is to finetune the matching model to generate matching score between reviews and keypoints.

From the dataset, only alphabetical characters are preserved. Emoticons , symbols and pictographs have been filtered out. This is achieved by matching their unicode character ranges suing regular expressions.

After a shuffle of the data, a split of 60:20:20 has been chosen for the train, validation and test datasets respectively.

4.3 IBM-ArgQ-Rank-30kArgs dataset

It is a large scale benchmark dataset[3] for ranking Argument Quality created by IBM(Gretz et al. (2020b). As the name suggests, there are around 30k arguments for 71 topics, with corresponding quality scores given in the dataset. In this dataset, there is also a column called 'set' which indicates the split in which the particular argument is present.

Similar to ArgKP dataset, only alphabetical characters are preserved. Emoticons , symbols and pictographs have been filtered out from the arguments.

For this dataset, as mentioned, the 'set' column indicates whether a particular argument is in train, dev or test split. The details of each split are described in Table - 2

4.4 Yelp Sentiment Dataset

We create this dataset by leveraging the abundance of available star ratings for reviews in the yelp review dataset. We extract reviews from train set that only have at most 3 sentences and 64 tokens. The reviews are divided into a training set, comprising 234206 reviews and 26023 in the test set.The labels for classified into three classes positive(4-5 star ratings), negative(1-2 star ratings) and neutral(3 star ratings).

5 Baselines

As a part of our baseline, we generate keypoints based on a set of rules and their quality, and then rank them based on their matching scores with the reviews.

The step by step procedure is outlined below and shown in figure 1.

[2]Downloaded from https://research.ibm.com/haifa/dept/vst/debating_data.shtml

[3]Downloaded from https://research.ibm.com/haifa/dept/vst/debating_data.shtml

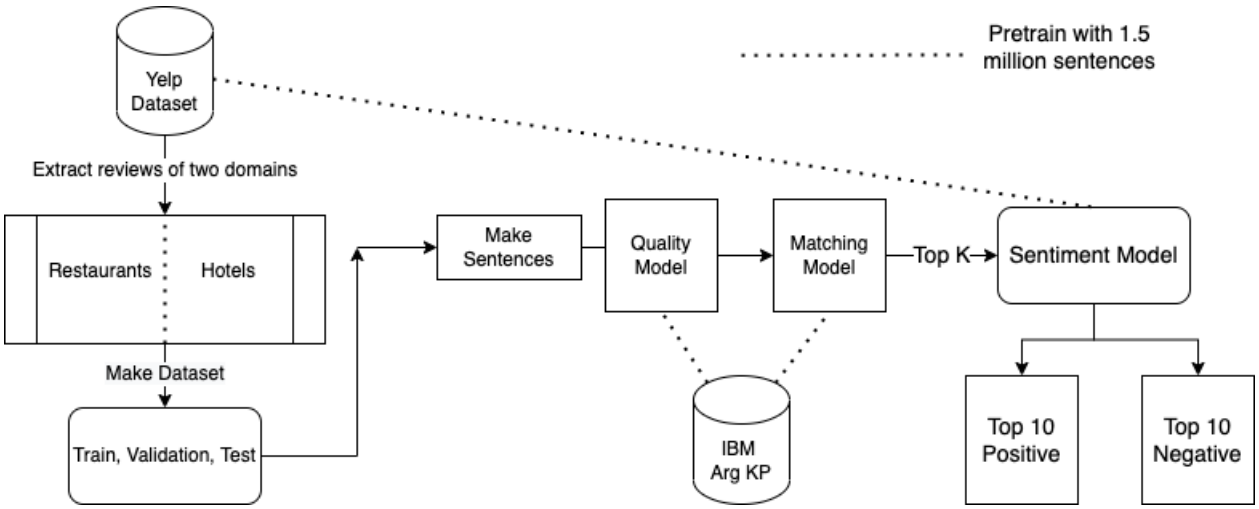

Figure 1: Rule based Method for extracting top-K Key Points

5.1 KeyPoint Generation

Key Point Anlaysis(KPA) was first developed for extracting summary of large argument data (Bar-Haim et al. (2020a)). In this, the arguments are matched to a set of key points. The number of matching arguments for each Key Point is included in the summary. These Key Points can be either given as input or automatically extracted from the data. One of the ways to automatically extract the Key Points is described by Bar-Haim et al. (2020b).

For the Key Point extraction, we follow the below steps:

1. Firstly, we process all the reviews from the train set and then filter the sentences of reviews with 3-6 tokens and high quality

2. We further filter these sentences using Parts of Speech(POS) tagging to remove sentences that start with pronouns.

3. Then each review is matched to its best matching Key Point, the one which has the highest matching score

4. Each of these sentences are ranked according to their number of matches

5. Then the top-k of these sentences are selected and outputted as the candidate Key Points

We have used Spacy library (Honnibal and Montani (2017)) for the POS tagging to remove the sentences which start with pronouns. Since we only focus on the first token of the sentence, we didn't have to do any more processing to adjust to the different tokenization that Spacy uses.

The function of Matching model is to assign a matching score given a (Review, Key Point) pair. And, the quality model assigns quality score for a Key Point or a review given. The training and further details about Matching Model and the Quality Model is described in sections 5.3 and 5.4 respectively.

5.2 Pre-trained model

The model architecture we chose for all our language models is RoBERTa. RoBERTa has some important improvements over BERT such as dynamic masking, Byte pair encoding on raw bytes instead of unicode characters. RoBERTa has shown increased performance compared to BERT on a number of tasks. Hence, we chose RoBERTa large as our model architecture.

Our work has three classifier models - A quality model A matching model and a sentiment model. Each of these models and their training has been described in greater detail in respective subsections. All the three classifier models we use as the part of baseline which are matching, quality and sentiment models uses the same pretrained roberta-large model. In order to generalize these models for reviews domain, we perform Masked Language(ML) Pretraining(Devlin et al. (2019))by masking out some of the words and trying to predicting them.
We perform the ML pretraining using 1.5 million sentences,with a length filter of 20-150 characters, extracted from the reviews of the train set constructed from restaurant and hotels domains. We

have used the following hyperparameters for our training: learning rate 1e-5; 2 epochs as proposed in (Bar-Haim et al. (2021)).
We utilized Google Colab pro for the training process and it took about little more than a day for the training to complete. But it wasn't a single run due to the problem of time limits set by colab pro. To avoid the problem, we have done it in batches by saving the checkpoint at each run and initializing it for the next run.

5.3 Matching Model

The Matching Model is obtained by fine-tuning the pre-trained Roberta-large model on IBM ArgKP datset(described in 4.2). Given a (Review, Key Point) pair, this model predicts a score that denotes the extent to which a review matches a key point. This is achieved by modelling the pre-trained Roberta-large with a few additional layers for classification. Over the output of the pre-trained model, we apply 3 liner neural network layers and a final sigmoid output neuron denoting the probability of a match. This sigmoid output was chosen as the Arg-KP dataset has score ('label' column) as a binary variable and we intend to model the probability that there is a match. Since our final output closely relates to a binary classification we chose binary cross entorpy loss as the objective for training.

We split the ArgKp dataset into 60:20:20 for all experiments on the matching model for the train,validation,test datasets respectively. We experimented with the following hyperparameters for finetuning (the best performing values and tried ranges are indicated). This was achieved by measuring the loss on our validation set. The least loss on the validation set was 0.29.

- learning rate - 1e-5 [1e-5,2e-5]
- batch size - 32 [32,16]
- epochs - 3 - Used reference from (Devlin et al. (2019))
- accumulation steps - 2. this hyperparameter divides a batch into mini-batches and accumulates gradient before applying any updates to parameters.
- dropout 0.2 on layers of the transformer

5.4 Quality Model

The Quality Model is obtained by fine-tuning the pre-trained Roberta-large model on IBM ArgQ-Rank-30kArgs datset(described in 4.3). Given a review or a key point, this model gives a score that denotes the quality and relevancy of the input to the topic. The input is (comment,topic) pair where the quality of a comment(review) is assessed based on the its relevance to a topic(for example restaurant hygiene/quality of food.) The dataset is a reduced version of human annotation results denoting two metrics MACE-P and WA . MACE-P can be seen as a quality measure while WA denotes the annotator agreement. By this analogy,we train our model using the MACE-P scores from the dataset. The architecture for fine-tuning is similar to that described in the matching model - 3 linear layers with a sigmoid output trained using a binary cross entropy objective. The Arg30K dataset has a 'set' column to denote if the data point is being considered for training/validation or testing. We reuse the same dataset splits for our experiments. The chosen hyperparameters and ranges are similar to the model in the matching section. Best loss of 0.32 was obtained on the validation set.

- learning rate - 2e-5 [1e-5,2e-5]
- batch size - 32 [32,16]
- epochs - 3
- accumulation steps - 2
- dropout 0.2 on layers of the transformer

5.5 Matching reviews with Key Points by leveraging Sentiment

Incorporating Sentiment into KP matching with reviews offers advantage in terms of understanding positive KPs and negative KPs summaries separately. In addition to that we may reduce matching errors by trying to match only sentences and KPs with similar sentiment. In order to do this we need to train a sentiment model, which we were able to do so by taking advantage of abundance of star ratings available for reviews in yelp dataset.

For the dataset creation related to this task, we extract reviews from train set that has atmost 3 sentences and 64 tokens and label reviews with 1-2 rating as negative, 3 as neutral and 4-5 star

Label	Precision Score
Negative	0.80350877
Neutral	0.96
Positive	0.89705882

Table 3: Precision Score of Sentiment Model for each class

rating are labelled as positive. The sentiment classifier was trained by finetuning the pre-trained model(Section 5.2) on the above training data, for 3 epoch.

Finally we incorporate sentiment into KPA by extracting positive KPs and matching them with positive sentences and likewise for negative KPs match them with negative sentences.

6 Your approach

6.1 Automatic Key point Generation using Abstractive Summarization

The problem with the using the above extractive approach for key point generationis that it cannot summarize the reviews with novel words. Summarization in general aims to generate very concise text and, the generated text should be indicative of the entire input and include as much information as possible. So, we resorted to a different approach - "Abstractive Summarization" rather than using the extractive summarization as proposed in the original paper.

Experiments are conducted on three summarization models - T5, BART, and Pegasus to examine how different models pretrained on different objectives perform in this task. To outline, BART (Lewis et al. (2020)) is a GPT-like self-supervised auto-encoder trained on CNN/Daily Mail that uses a noise-added source text as input and trains to reconstruct the original text by predicting the true replacement of corrupted tokens. T5(Raffel et al. (2019)) is an encoder-decoder architecture trained on the C4 dataset with fill-in-the-blank-style denoising objectives (where the model is trained to recover missing words in the input). Pegasus (Zhang et al. (2019)) with a encoder-decoder architecture, on the other hand, is pretrained with the following objective: important sentences are canned/masked from an input document and are generated together as one output sequence from the remaining sentences, similar to an extractive summary.

6.1.1 Reviewing Models for Summarization

Some sample summaries generated for each of the pretrained model are shown in table - 4(Restaurant Reviews) and 5 (Hotel & Travel reviews). It is evident from the table that T5 outputs repetitive sentences for some reviews and can sometimes be nonsensical. The generation from BART is good though because it uses beam search to generate sentences. However, BART generates sequences with very high sequence length. This is not desirable because key points are meant to be short and informative. If the output sequence length of BART is limited, the outputs become worse. Pegasus, on the other hand strikes a good balance between these two - it does not generate repetitive sentences as frequently as T5 does and the output sequence length is not too long.

However, the output from pretrained pegasus cannot be used as candidates because the generated sentences go against the basic rules of a key point i.e - it should not start with pronouns (key points should not be specific to a person) and each key point must present only one idea (sentences that describes about multiple attributes is not considered a key point). These additional constraints while generating summaries can only be learnt by the model if it sees many such examples of key points. Hence, the model is finetuned on IBM's ArgKP dataset.

6.1.2 Fine tuning Pegasus

The pegasus model is finetuned on the IBM ArgKP dataset (described in 4.3). The model is run for 3 epochs with batch size of 8 (limited by colab resources). The learning rate used is $1e^{-}5$ with a learning rate scheduler with warm up every 500 steps. The input-output to Pegasus model while training is the (argument,key point) pair with decoder trying to generate key points. Used HuggingFace's Seq2SeqTrainer for training the model with a cross entropy loss used by default.

The following are the hyper parameters used for the training procedure :

- learning rate - 1e-5
- batch size - 8
- epochs - 3
- weight decay - 0.01,
- dropout 0.1 on layers of the transformer

Sample Generated Summaries		
Model	Example 1	Example 2
bart-large-cnn	I have tasted the coconut cake (amazing!), german chocolate cake (delicious!)	Legal seafood restaurant at Boston Logan airport. The clam chowder is a must and is the best I've ever tasted hands down. The lobster roll is also a great choice. The service
T5-base	The food was mediocre. The food was mediocre. The food	I had the crab cake sandwich. It was nice. It comes with chips. I
pegasus-xsum	I have tasted the coconut cake (amazing!), german chocolate cake (delicious!)	The food was delicious and the service was top notch.

Table 4: Sample generated summaries of Restaurant reviews with different models

Sample Generated Summaries		
Model	Example 1	Example 2
bart-large-cnn	A fun thing to do while in Portland. It's close enough to the city on beautiful grounds and the tour was so interesting. Edi did a fantastic job of providing the history of	i was excited to learn that BrewDog was going to open up their brewery in Canal Winchester
T5-base	a lot of money. 15 rooms. a full room.	I'm a Hertz Gold Member so rarely stop at the ocunte
pegasus-xsum	"I've never been to a place like BrewDog before.	One of the best things to do in Portland is to take a tour of the Declaration of Independence house

Table 5: Sample generated summaries of Hotel/Travel reviews with different models

6.1.3 Training a Sentiment Classifier

A sentiment classifier is obtained by finetuning a pretrained Roberta-base Model on a subset of 1.5 million reviews from Yelp Dataset (described in 4.1.1). This sentiment classifier is used to group positive and negative sentences and can be used to select high quality sentences as candidates for key points. As a preprocessing step, the reviews are assigned with labels {0,1,2} each corresponding to the sentiment of the sentence. A review is labelled 0 if its rating is either 1 or 2. Similarly, label 2 corresponds to reviews with ratings - 4 and 5. The reviews with rating = 3 correspond to neutral label.

The Roberta-base is added with a classification head with a full linear layer of hidden size 768. The output of the classification head is 3 logits, one for each of positive, neutral and negative. A cross entropy loss is calculated and is back propogated using Adam optimizer.

The model is trained for a total of 10 epochs with a batch size of 8. The Adam optimizer is used with epsilon value = 1e-8. The learning rate is taken as 1e-6. The model achieved a validation accuracy of 0.88 with the precision scores reported in Table 3.

6.1.4 Procedure to generate top K Key Points

We used the following method for selecting top 10 positive and negative Key Points by leveraging all the trained models like Matching, Quality and Pegasus-ArgKP. The flow chart for the whole procedure is outlined in fig.2.

1. Divide the Yelp Dataset seperately into Restaurant and Hotel/Travel reviews.

2. For each of the reviews dataset, divide it further into 2*N (N=80) chunks based on the polarity (review rating/sentiment) s.t reviews with same polarity are grouped together.

3. For each chunk summarize the whole set of reviews using finetuned pegasus to generate N summaries for each of the positive and negative sentiment.

4. These generated summaries of the reviews are checked for quality using a Quality model. Any point that has a quality score less than a threshold are removed to make sure only high quality sentences are considered as candidates.

5. Map each review to its best matching KP using a threshold for matching score.

6. Rank the candidates according to the number of their matches and get top K candidates for each of the positive and negative reviews set.

   The final Key points obtained after following the above procedure are selected and clculated their percentage of reviews they match

with are shown in table. Key Points for Hotel/Travel reviews - Table10. Key Points for Restaurant reviews - Table9.

6.2 Models implemented by us

- Roberta-large Sentiment model (This model code is under Sentiment Classifier folder)
- Roberta-large Quality model (This model code is under Quality Model folder)
- Roberta-large Matching model (This model code is under Matching Model folder)
- Rule-based Key Point Generation(Codes are under Rule-based folder)
- Finetuned pegasus on ArgKP model(The code related to this is under Summarization Models).

7 Experiments

7.1 Rule-based KP generation experiments

We have followed the procedure mentioned earlier and found the top 120 Keypoints(60 positive and 60 negative) across each domains and using these we match them with 100 reviews and a threshold of 0.95 is used in the matching model to establish a match. We got mean matches per review of 6.35 for HOTELS and 8.58 for RESTAURANTS. Restaurants have more reviews compared to Hotels, this is reason we observed higher mean matches per review for RESTAURANTS.

Table 3 and 5 summarize the percentage of reviews matched for the top-10 positive and negative Key Points, for Restaurants and Hotels respectively. Table 6 displays sample matches of sentences for a positive and a negative Key Point.

7.2 Automatic KP generation experiments

To see how different generation models (not fine-tuned) perform on the summarization task, the base models T5, BART and Pegasus are used to generate sample summaries, detailed in section 6.1.

We followed the procedure mentioned in section 6.1.4 to generate top 160 candidates (80 positive and 80 negative) for each of the domain. We feed these to Quality model and Matching model as described to find the matching scores and hence the top-10 negative and positive key points.

Table 9 and 10 summarizes the percentage of reviews matched for the top-10 positive and negative Key Points, for Restaurants and Hotels respectively.

Figures 3,4 give the Key Point coverages in percentage for the top-10 Positive and Negative Key Points for Restaurants For Rule-based and Pegasus models.

8 Error analysis

8.1 Reviews not matching any keypoints

Some of the reviews which are too specific to a business failed to be matched to any key point after threshold is applied to the match score. One of the examples of such reviews is this -

> ”I called WVM on the recommendation of a couple of friends who had used them in the past and thought they did a nice job. I’m a fan now, too.and Cody showed up right on time for my move this past weekend. They were friendly and energetic, working quickly but carefully to get all my things moved out of the old place and into the new one in less than 2.5 hours. All of my (heavy) furniture arrived in perfect condition, and they took extra care not to scratch the wood floors in the process.hope not to move again anytime soon, but next time I do, I’ll be calling WVM.”

Since this review is too specific to the use case, and describes about the services offered by the travel agency ’WVM’, the matching score for this review with the key points was below the threshold and hence wasn’t matched to any of the key points.

Similarly, the below review is unmatched to any Key Point after threshold is applied

> ’Not enough service-13 minutes sometimes for another train to comewords: Braintree/Ashmont split=Delays, very rude staff.e.:THE DOORS! THE DOORS WILL BE CLOSING! WATCH THE DOORS! screeched at obnoxious volume, whereas service delays are made at a level no one can hearbeen called names by bus drivers in the good old days for being slow with putting in money and/or scanning my pass (and I always have them out ahead of time)gladly pay more if they addressed their problems, but it all goes to debt reductionI do get reliably each day where I want to go and can read books while doing so’

8.2 Key points generation

Baseline model :

1. The generated keypoints had some overlap for example the key points - ”Service was efficient and friendly” and ”The service was good ”. Both of them are essentially the same key point.

2. Few Key points are not informative and are mere expressions - For example - ”Loved the experience”,”Each time enjoyed it”

Pegasus fine-tuned model:

Key point : Excellent food and great service	Key point : Food is not great but good
The food is always great here. The service from both the manager as well as the staff is super. Only draw back of this restaurant is it's super loud. If you can, snag a patio table!	Nothing special but good enough. I like another one much better in Dorchester, but hardly get to that area, so Le's is my goto place for Pho"
I have been here twice and have had really good food and service both times. I got a burger with caramelized onions	I was expecting the tomatillo salsa to be amazing, considering the name of the restaurant was called tomatillos. Instead, it just tasted like they added waaaaaaaaay too much lime juice to me.
It's a pleasure doing business with them. Very good service and hospitable staff.	All 3 of us ordered 2 different rolls and it took like 30 mins to come out. Sushi was alright but I would say to much for the quality

Table 6: Sample matches of sentences to key points

Positive Key Points	% Reviews	Negative keypoints	% Reviews
Excellent food and great Service	14.62%	Food is not great but good	18.4%
Great atmosphere, service and food!	14.38%	Nothing special and definitely not great	15.5%
Service was efficient and friendly	12.98%	slow service and disappointing food	15.13%
Good food, affordable, will go again	11.54%	Amazing food but awful service	14.89%
Good service and fresh food!	11.46%	Terrible service & food was delivered cold	14.53%
The staff was nice and attentive	11.17%	Horrible Service! Bland food!	13.31%
Highly recommend- very friendly staff	10.75%	Again, not outstanding, just good	12.5%
The service was good	9.54%	Food was terrible and overpriced	9.3%
Super fast and cheap	8.92%	Great atmosphere but terrible food	9%
The staff is hospitable and friendly	8.83%	No flavor to anything	6%

Table 7: Summary of reviews produced by Rule-based KP Generation for Restaurants.

Positive Key Points	% Reviews	Negative keypoints	% Reviews
Excellent food and great service	22.7%	It was dirty and very worn	18.3%
Excellent and fast service	18.6%	Nothing special and definitely not great	16.6%
The staff is hospitable and friendly	18.2%	Not worth the money	14.9%
They're clean and updated	18.1%	Food is not great but good	14.4%
We would definitely stay there again	9.32%	Would never recommend this place	12.4%
I definitely recommend this hotel!	16.7%	The elevator was broken	10.9%
The service was good	14.8%	There was no microwave or fridge	10.7%
All new appliances and very clean	12.9%	No hot water in my room	10.7%
Each time enjoyed it	12.7%	Parking garage is terrible	10.6%
Loved the experience	12.6%	Shitty furniture and shitty tv	10.4%

Table 8: Summary of reviews produced by Rule-based KP Generation for Hotels Travel

Positive Key Points	% Reviews	Negative keypoints	% Reviews
We were quite impressed. Food was served hot and delicious	18.6%	The service was slow as a snail	16.5%
This is one of my favourite restaurants .	16.3%	This is one of the worst restaurants I have ever been to.	16.35%
It has a fun atmosphere, great for big groups	14.5%	This was the worst meal I've ever had	14.74%
A good place to stop for quick food	13.74%	Smelly environment, very bad ventilation system	12.7%
Lots of seating! Fast, friendly service!	12.85%	The atmosphere is so loud and anxious!	11.67%
Check it out!	10.76%	I've eaten at this location and others at the past and this is the worst experience I've had so far.	10.5%
Has amazing food, and is affordable, will definitely go again	10.43%	the food was just bland	9.73%
The dishes have deep and spicy flavors'	10.12%	Food is nothing special but definitely pricey	8.5%
I've never had a lobster roll quite like this before.	9.4%	The menu though is quite limited now.	7.5%
Staff is attentive. Nice service.	8.5%	Food is bland.	6.9%

Table 9: Summary of key points produced by Pegasus model for restaurants.

Positive Key Points	% Reviews	Negative keypoints	% Reviews
Great food. Great atmosphere.	14.8%	Expensive, Not Worth it !	15.7%
Hotel is nice and kept clean	12.4%	It's disgusting and way over priced	14.76%
Crazy cheap for good food	12.2%	Disappointed, the room was very small.	11.2%
The room was clean and spacious	11.1%	Very poor service and you have to haul	10.5%
We had a great time and were able to explore the city	10.7%	Customer service is terrible	9.2%
each visit has been a great experience	9.5%	Mansion it self is a bit pricey for what you get.	8.95%
This was one of the best service I have ever seen.	8.4%	Dissappointed in sales staff	8.7%
Absolutely fabulous experience	8.1%	the room stinks	8.1%
Very nice decor. Elegant bedrooms with very comfortable beds	7.85%	Totally inexcusable, the staff doesn't care at all	7.95%
I have stayed at this hotel many times and have never had a bad experience	7.2%	Was full of mold, cracks and rust in my room	7.79%

Table 10: Summary of key points produced by Pegasus model for Hotel/Travel reviews.

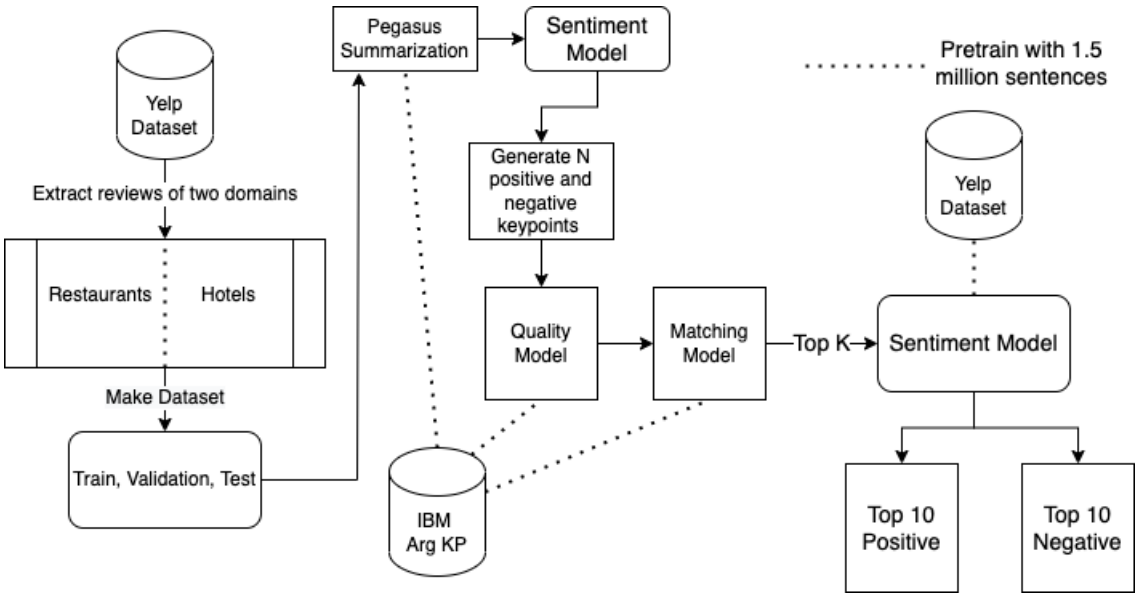

Figure 2: Key Point Generation Flow Chart using Pegasus

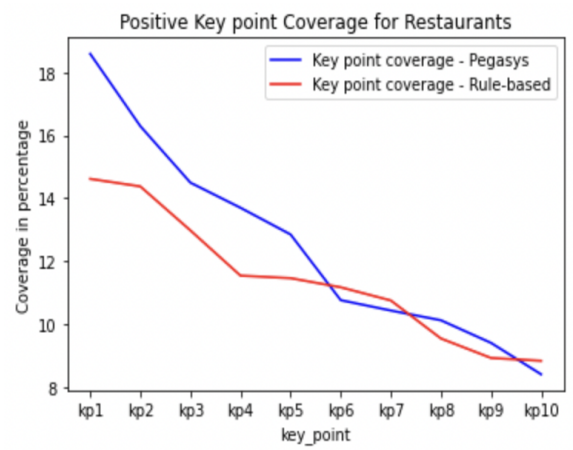

Figure 3: Coverage percentage of positive key points for Restaurants

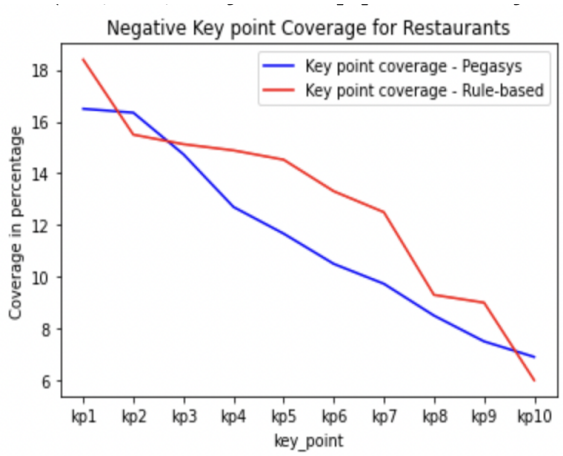

Figure 4: Coverage percentage of negative key points for Restaurants

1. Generated key points become too specific and are not generic enough . For example - 'I've never had a lobster roll quite like this before.'

2. Key points with strong sentiment generate a lot of matches , but these are a bit uninformative: "This is the worst meal I've ever had" 'This is one of the worst restaurant I have ever been to'

9 Contributions of group members

List what each member of the group contributed to this project here. For example:

- Akhil Ayyanki : fine-tuning of YELP pre-trained model to build Quality and Matching models, implementation and result analysis of Rule based Key point generation algorithm and model. Error analysis, writing.
- Chaitanya Rajesh: Worked on the Pegasus model, Dataset creation for the task and fine-tuned the model. code for Algorithm to generate sentences from the model and finding top K kp's, organized all the resources used, report writing, flowchart creation.
- Praneeth Reddy: built and pre-trained models, done yelp dataset preprocessing, sentiment dataset creation, sentiment model fine-tuning, report writing.
- Veda Sree Bojanapally: Quality and Match-

ing model Development and finetuning, KP quality generation, Rule-based Key-Point Generation, Data pre-processing, error analysis and annotations. Report writing.

10 Conclusion

In this paper, we have applied models that are finetuned on a dataset from which we are learning features and using the model to predict on our review dataset. In order to able to do that we have pretrained the models on our dataset. This process of achieving good results have been demanding since we have to go through the process of understanding the original application of model and make it work for our dataset.

Our approach to generate keypoints using a sequence to sequence model using Pegasus was successful in generating novel keypoints that have very high meaning but hasn't been the case always. If the input reviews are too sparse then the model had difficulty generating good keypoints instead it generates a keypoint that is more relevant to ArgKP data than review data. This problem may be overcome if we have more review data specific keypoints or even more data for pretraining. This is one thing we intend to focus on in our future work. The quality of key points generated is at par with the baseline model and has good coverage that is comparable to the baseline model.

References

Alshomary, M., Düsterhus, N., and Wachsmuth, H. (2020). Extractive snippet generation for arguments. *SIGIR '20*, abs/2109.15086.

Alshomary, M., Gurke, T., Syed, S., Heinisch, P., Spliethöver, M., Cimiano, P., Potthast, M., and Wachsmuth, H. (2021). Key point analysis via contrastive learning and extractive argument summarization. *CoRR*, abs/2109.15086.

Bar-Haim, R., Eden, L., Friedman, R., Kantor, Y., Lahav, D., and Slonim, N. (2020a). From Arguments to Key Points: Towards Automatic Argument Summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4029–4039, Online. Association for Computational Linguistics.

Bar-Haim, R., Kantor, Y., Eden, L., Friedman, R., Lahav, D., and Slonim, N. (2020b). Quantitative argument summarization and beyond: Cross-domain key point analysis. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 39–49, Online. Association for Computational Linguistics.

Bar-Haim, R., Kantor, Y., Eden, L., Friedman, R., and Slonim, N. (2021). Every Bite Is an Experience: Key Point Analysis of Business Reviews. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, page 3376–3386.

Bražinskas, A., Lapata, M., and Titov, I. (2020). Unsupervised Opinion Summarization as Copycat-Review Generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5151–5169, Online. Association for Computational Linguistics.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding.

Ganesan, K., Zhai, C., and Han, J. (2010). Opinosis: A Graph Based Approach to Abstractive Summarization of Highly Redundant Opinions. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 340–348, Beijing, China. Coling 2010 Organizing Committee.

Gretz, S., Friedman, R., Cohen-Karlik, E., Toledo, A., Lahav, D., Aharonov, R., and Slonim, N. (2020a). A large-scale dataset for argument quality ranking: Construction and analysis. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):7805–7813.

Gretz, S., Friedman, R., Cohen-Karlik, E., Toledo, A., Lahav, D., Aharonov, R., and Slonim, N. (2020b). A large-scale dataset for argument quality ranking: Construction and analysis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7805–7813.

Honnibal, M. and Montani, I. (2017). spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., and Zettlemoyer, L. (2020). "BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension". In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2019). Exploring the Limits of Transfer Learning with Unified Text-to-Text Transformer. *CoRR*, abs/1910.10683.

Snyder, B. and Barzilay, R. (2007). Multiple Aspect Ranking Using the Good Grief Algorithm. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 300–307, Rochester, New York. Association for Computational Linguistics.

Titov, I. and McDonald, R. (2008). A Joint Model of Text and Aspect Ratings for Sentiment Summarization. In *Proceedings of ACL-08: HLT*, pages 308–316, Columbus, Ohio. Association for Computational Linguistics.

Zhang, J., Zhao, Y., Saleh, M., and Liu, P. J. (2019). PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization. *CoRR*, abs/1912.08777.