Information Graphic Summarization using a Collection of Multimodal Deep Neural Networks

Edward Kim
Department of Computer Science
Drexel University
Philadelphia, PA 19104
Email: ek826@drexel.edu

Connor Onweller
Computer and Information Sciences
University of Delaware
Newark, DE 19716
Email: onweller@udel.edu

Kathleen F. McCoy
Computer and Information Sciences
University of Delaware
Newark, DE 19716
Email: mccoy@udel.edu

***Abstract*—We present a multimodal deep learning framework that can generate summarization text supporting the main idea of an information graphic for presentation to a person who is blind or visually impaired. The framework utilizes the visual, textual, positional, and size characteristics extracted from the image to create the summary. Different and complimentary neural architectures are optimized for each task using crowdsourced training data. From our quantitative experiments and results, we explain the reasoning behind our framework and show the effectiveness of our models. Our qualitative results showcase text generated from our framework and show that Mechanical Turk participants favor them to other automatic and human generated summarizations. We describe the design and results of an experiment to evaluate the utility of our system for people who have visual impairments in the context of understanding Twitter Tweets containing line graphs.**

I. Introduction

The goal of our work is to provide a useful textual summary of an online information graphic to someone who is visually impaired or otherwise unable to view the graphic. Our objective is to produce a summary containing the salient information content of the graphic (as opposed to the way the graph looks or the data values from the graph). To do this we created a machine learning framework that analyzes an information graphic and generates a text summary of the content in the image. These information graphics, such as line graphs and bar graphs, often appear in popular or social media and are presented to the reader to support an intended message. However, these graphical images are often not accessible to a person with visual impairments who is unable to see the graph. For instance, a person who is blind and using a screenreader would only hear the "alt-text" associated with an image. However, in popular and social media, that alt-text is often either left empty or not useful. To highlight the extent of this issue, we conducted a background study to determine how frequently Tweets containing images lack meaningful alt-text. In a sample of 984 recent Tweets containing images from twitter's 20 most popular accounts, and 123 Tweets with images that matched keywords like line-graph, bar-graph, info-graphic, etc..., we found that the alt-text posted for those graphics was "Image". This means that none of the users who posted those Tweets set any meaningful alt-text (note "Image" is the default alt-text in twitter). In situations where no (meaningful) alt-text is present and no summary of the graphic exists elsewhere in the text, people with visual impairments will miss the information contained in a graphic.

To generate our summary, we trained a collection of multimodal deep learning models to detect, extract, and present the relevant information that supports the main idea of the graphic. Our implementation has thus far concentrated on line graphs with single lines. This area of research is useful for text-to-speech applications that contain a mixture of information graphics and text, assistive technology applications that require interpretation of visual data, and information retrieval applications based upon the characteristics of data depicted. A result of our system can be seen in Figure 1. We describe our experiment evaluating the efficacy of our summarization methodology against others in the context of Twitter Tweets containing information graphics.

II. Background

Information graphic interpretation and summarization is a difficult task and an open research question. Several works in the past have looked at different aspects of information summarization including, classifying salient textual entries in grouped bar charts [1] and improving accessibility to information graphics [2], [3], [4]. Recent work by [5], presented, "Chartsense", an interactive tool that extracts data from a chart using a semi-automatic, interactive extraction algorithm. Chartsense uses a deep learning network for classification and makes several assumptions about the color, size, and orientation of the chart. In contrast, we are fully automatic, utilize a collection of deep learning classifiers, make no assumptions about the presentation of the data within the line graphic, and present an overview of the message in the graphic rather than the data points themselves. We hypothesize that our summary will be useful to people with visual impairments reading for pleasure.

A. Background in Multimodal Deep Learning

Multimodal deep learning is a multi-disciplinary field that processes different modalities of communication including visual, linguistic, and auditory signals. Multimodal deep learning is particularly challenging, in that the model must learn how

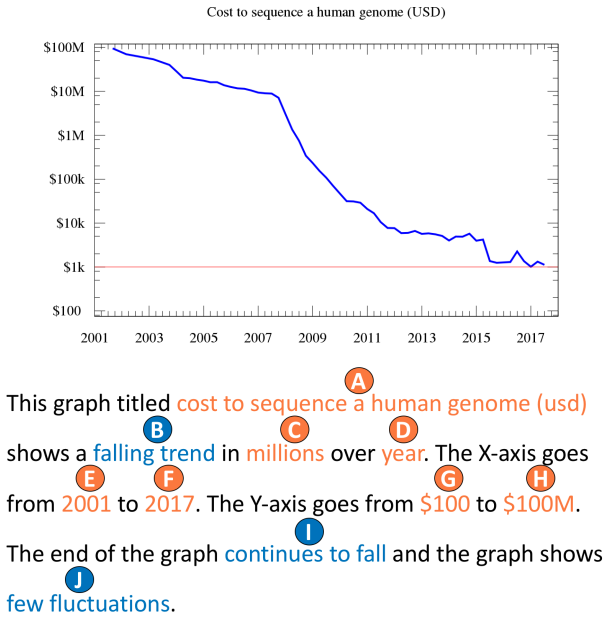


Fig. 1: Summarization result of a graph by our collection of deep neural networks (DNNs). 10 network models, labeled A-J, are used to extract the information needed to fill in the summary template. The networks labeled in blue use visual convolutional network features, whereas the networks labeled in orange use text and positional features.

to effectively integrate heterogeneous data in a common architecture. Early work in multimodal modeling was performed by Ngiam et al. [6] using audio and video deep autoencoder models in a sparse Restricted Boltzman Machine. Bengio et al. [7] illustrates the usefulness of representation learning (feature learning) and how the combination of multiple types of inputs helps with the generalization of the model.

Multimodal networks are ideal for image to text captioning, summarization, and translation. General image captioning tasks have been created with the seminal work by [8]. This multimodal deep learning model uses a convolutional neural network (CNN) to extract the image features and a long short term memory (LSTM) recurrent neural network to generate the caption. Antol et al. [9] combined CNNs with question and answer pairs to build the Visual Q&A system that enables users to ask questions about the visual content of an image. Follow up work [10], [11] improved on these methods by adding attention mechanisms and attributes. Similarly, our work fuses multimodal data together in our networks including visual, textual, positional, and size data. While these previous general image captioning methods can be used for information graphic summarization, we show that the output of these systems are much too general. Information graphics have an intended message; prior analysis (e.g., [12]) has shown that that intended message, along with other information that can be gleaned from the graphic, should be included in the summary for it to be useful. Finally, our previous work in information graphic classification describes a method for extracting the trends in a line graph [13]. Indeed, the trends in a line graph are very important for inclusion in a summary, but would form only part of a useful summary. In fact, the work in [13] describes how to create and validate a neural network similar to the one labeled as part "B" in Figure 1. In the following sections, we expound on the methodology for creating networks "A", and "C-J", and highlight the differences of our trend classification network, "B", in the context of this work.

III. Methodology

"B" Classes	Description of Graph
RT	**R**ising **T**rend in the line graph.
FT	**F**alling **T**rend in the line graph.
ST	**S**table **T**rend, generally no change in the line graph.
CT	**C**hanging **T**rend in the graph, e.g. up and down, or down and up, possibly multiple times.
BJ	**B**ig sudden **J**ump in the line graph.
BF	**B**ig sudden **F**all in the line graph.
"I" Classes	**Description of Graph**
RTT	**R**ising **T**rend at the line **T**ail.
FTT	**F**alling **T**rend at the line **T**ail.
STT	**S**tablizing **T**rend at the **E**nd of the line graph.
"J" Classes	**Description of Graph**
V	**V**olatile with many fluctuations.
NV	**N**on-**V**olatile graph.

TABLE I: Description of the classification categories of the information graphics from networks, B-Trend, I-Tail, and J-Volatility. The ground truth labels are crowdsourced from online workers using a majority rule to select the final label.

By studying human summaries of graphics [12], [4], we developed a template model that could generate a summary by extracting useful information from a line graph. For the extraction, we trained a collection of deep neural networks - each responsible for extracting a needed piece of content. We have labeled the slots in the template A through J. Each slot (essentially) corresponds to a piece of information that must be extracted from the graph. The overall template used:

> This graph titled <A: GRAPH-TITLE> shows a <B:GRAPH-TREND-DESCRIPTION> in <C: Y-AXIS-LABEL> over <D: X-AXIS-LABEL>. The x-axis goes from <E: X-AXIS-START-VALUE> to <F: X-AXIS-END-VALUE>. The y-axis goes from <G: Y-AXIS-START-VALUE> to <H: Y-AXIS-END-VALUE>. The end of the graph <I: END-GRAPH-DESCRIPTION> and the graph shows <J: FLUCTUATION-VALUE>.

The template and the corresponding networks were used to generate the summary in Figure 1.

A. Information Graphic Dataset

For our dataset, we use the 1000 line graph images obtained from [13]. These line graphs are non-scientific and representative images that one might see in popular media articles. The

ground truth data classifies graphics into one of six possible categories described in Table I, "B" classes. This classification provides the overall message of the graphic, however, for our summarization, we required much more information about the infographics. Thus, we augmented the data by using the crowdsourcing platform, Amazon Mechanical Turk, to collect additional ground truth for training our network. We asked five unique users to label the data with additional classes and extract additional ground truth text from every graph in our dataset.

To automatically extract text from the image, we use the latest version of Tesseract [14]. Tesseract is an OCR engine that is being developed by Google and the latest version[1] includes a recurrent neural network implementation using LSTMs for improved accuracy. We extract all the text from all of the training and testing images and utilize their corresponding GloVe vector [15] to project the words into a semantic embedding space. We use the entire GloVe vocabulary as well as the words found in our training set (initialized to the zero vector) for a total vocabulary size of 342,078. The dimensionality of the GloVe vector was set to 100.

B. Neural Network Architectures

As a first step, we have developed a screening process to determine the appropriateness of an image for our system. The screening network is based upon a fine-tuned VGG16 model with binary output: line graph or not. The network was pretrained on ImageNet, and the last layer was replaced with a single sigmoid activated output. The network was fine tuned with 5000 total images, 1000 line graphs, 4000 random images from the ILSVRC 2012 validation set, using a 80/20 train/test split. The accuracy of our screening process is nearly perfect in detecting whether or not an image is appropriate for our system, i.e. 99.9% accurate after 5 epochs of training.

Next, we organize the description of our deep neural networks that process the information graphics by the type of network and modality of information used to perform inference. The visual components of our network use a convolutional neural network (and on occasion text features), and other networks use text and positional elements.

C. Convolutional Neural Networks

To construct the summarization elements that require one to interpret the visual components of the graph, we use convolutional layers in our architecture.

Network "B" (Trend)

The trend network is similar to background work in [13]. For this model, we combine two modalities extracted from the information graphic. The first modality is the raw pixel data. Each image is scaled to an image size of 224x224x3 and passed through a series of convolutional and pooling layers. The kernel size for the convolutional layers is 7x7, and the pooling layers utilize a max operation with a stride of 2. There are a total of four convolutional/pooling layers resulting in an end feature vector dimension of 4,096. For the text data, we encode the text into a bag of words (BOW) binary representation, where the feature vector encodes whether or not the word is present in the information graphic. We create a dictionary of size 342,078 and then encoded to a 100 dimensional vector. Both modality streams are jointly embedded in a concatenation vector, that is connected to a 6 dimensional softmax output. An illustration can be seen in Figure 2. As referenced previously in Table I, "B" classes, the six possible outputs are rising trend, falling trend, changing trend, stable trend, big jump, and big fall.

[1]https://github.com/tesseract-ocr/tesseract

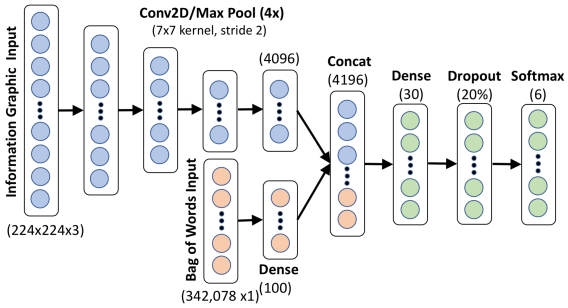


Fig. 2: Architecture of the Trend neural network. The network input is the information graphic and bag of words representation. The final output is a softmax for 6 possible classes.

Network "I"(Tail) and "J" (Volatility)

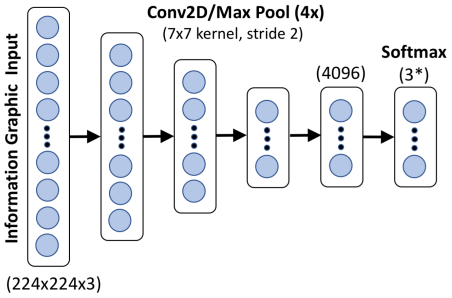


Fig. 3: Architecture of the Trend Tail and Volatility neural network. The final output is a softmax either 3 classes for the Trend Tail or 2 classes for the Volatility network (denoted by the * in the diagram).

The "I" - Tail and "J" - Volatility networks are purely visual and almost identical in structure to each other. Both networks have the image pixels as the input (resized to 224x224x3), and the data is passed through four successive convolutional and pooling layers. The "I" - Tail network analyzes the graph to see what happens at the end of the trend. It classifies the tail of the trend into the 3 categories labeled, rising trend tail, falling trend tail, or stabilizing. The purpose of the tail network is to provide important information missing from network "B". For example, network "B" could classify the trend as a changing trend, but we found that leaves much ambiguity to the user. Thus, the tail network further characterizes the trend, as does the volatility network. The volatility network looks

at the graph and classifies the line graph as containing either high fluctuations, or low fluctuations. This is somewhat of a subjective characterization that we left to the human annotators to assess. The architecture of these networks can be seen in Figure 3.

D. Text and Positional Networks

The following set of deep learning networks utilize the text extracted from the information graphic by an off-the-shelf OCR. The extraction process provides the words within the image as well as their x and y coordinates, text size, and confidence. We threshold the text extracted by the OCR system at 0.15 confidence. This is relatively low confidence, allowing much of the detected text (correct or not) to pass through to our framework.

Network "A" (Title), "C" (Y-axis label), and "D" (X-axis label)

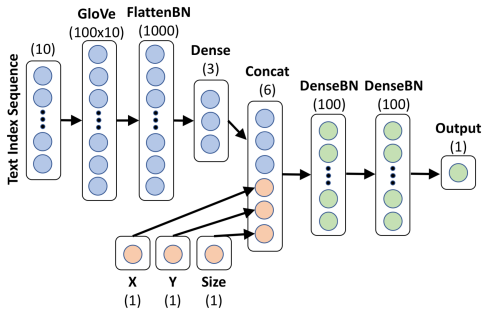


Fig. 4: Architecture of the Title, Y-axis label, and X-axis label neural networks. The network takes text sequences from the information graphic, as well as their relative x and y positions in the image, and relative text size. The final output is the approximated Levenshtein ratio.

All three of these networks are identical in structure and so we will describe only the "A" - Title network in detail. The other networks follow the same algorithmic steps. The architecture of these networks can be seen in Figure 4. The title network takes a text sequence (of maximum length 10 words) from the OCR data and embeds it into a 100x10 GloVe matrix, flattens and batch normalizes the data, then reduces it to a three node layer. This layer is concatenated with the scalar inputs that characterize the x and y position of the text sequence in the graphic, as well as the text size. This concatenated layer is passed through two more fully connected layers. The final output is the approximated Levenshtein ratio with a mean squared error (MSE) loss function. The idea of using the Levenshtein ratio stemmed from the fact that we had collected the ground truth title data from the mechanical turk workers and needed an approach that would select the title sequence from a list of sequences extracted from the graphic. A natural way to compare two strings was through the Levenshtein distance, and as such, we used a neural network to approximate this distance. The Levenshtein distance corresponds to the number of edits needed to change one string into the other. Thus, a perfect match between strings would have distance of zero, and a completely different string would have a distance of the length of the longer string, m. Mathematically, given two strings a and b, where the length of the string is denoted $|a|$ and $|b|$,

$$\mathrm{lev}_{a,b}(i,j) = \begin{cases} \max(i,j) & \text{if } \min(i,j) = 0, \\ \min \begin{cases} \mathrm{lev}_{a,b}(i-1,j)+1 \\ \mathrm{lev}_{a,b}(i,j-1)+1 \\ \mathrm{lev}_{a,b}(i-1,j-1)+1_{(a_i \neq b_j)} \end{cases} & \text{otherwise.} \end{cases} \tag{1}$$

Where $1_{(a_i \neq b_j)}$ is an indicator function equal to 1 when $a_i \neq b_j$. We normalize and invert this distance i.e. $(1 - \frac{\mathrm{lev}}{m})$ for our final Levenshtein ratio metric.

Backup "C" (Y-axis label) and "D" (X-axis label) -

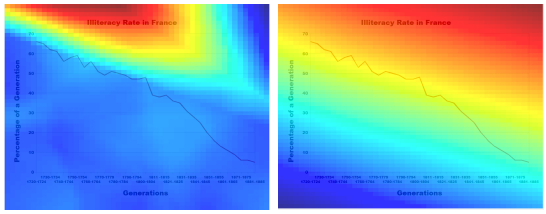
On many information graphics, the x-axis label, or y-axis label is omitted. This happens quite frequently in popular media where the axis representations are either inferred from other text in the image, or through common sense. For example, if we look at our first example, Figure 1, neither the x nor y axis are actually labeled. However, to a human, the represented information is clear and our network was able to successfully label the graph as showing a change in "millions over years". To accomplish this, we created two additional neural networks to handle the inference of axis labels in the event that they are missing. Backup network "C" and "D", look at every individual word, its x position, y position, and text size, and makes a label prediction as to what that word most likely represents on the x or y axis. The output is a softmax over every word in the vocabulary. For the final label, we pick the maximum word over all scores from all words in the image.

Network "E" (X domain low), "F" (X domain high), and "G" (Y range low), and "H" (Y range high)

Our final set of networks provide the extents for the x domain and y range. There are four total networks, one for each data point. The architecture of these networks is similar to the architecture of the other networks. The input to this net is the input word, as well as the x, y, and size data. The training data for this network was binary, either 1 for the correct word extracted from the OCR, or 0 for every other word. The output of the network predicts whether or not the input word with attributes is the correct domain or range item.

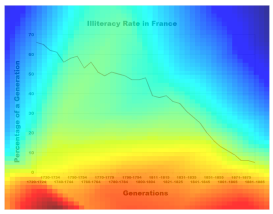
IV. Experiments

We present several qualitative and quantitative experiments evaluating our system. In the first set of experiments, we illustrate the selection of several hyper-parameters and the effect they have on the output of our networks. Following these results, we discuss 2 experiments involving human-subjects. In the first, we ask sighted participants to select among 3 different summaries (our DNN-generated summary, another machine generated summary, and a human-generated summary). A second experiment was designed to evaluate the efficacy of our system for generating helpful summaries presented to people with visual impairments.

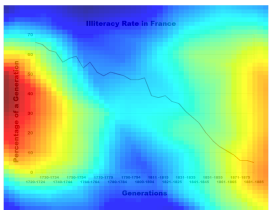


(a) Title

(b) Title (loss of positional sensitivity)

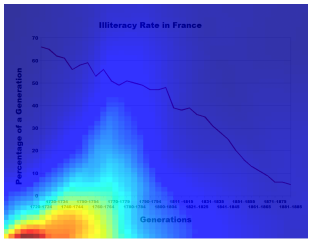


(c) X-Axis Label

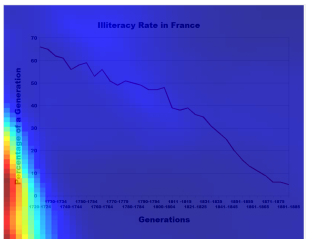


(d) Y-Axis Label

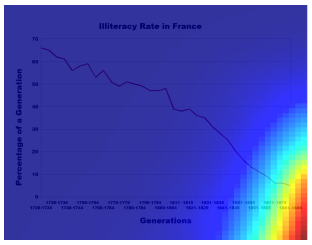
Fig. 5: Heat maps corresponding to the likelihood that the given attribute can be found in a certain area of the graph. When the network is not forced to utilize the coordinates, it looses positional sensitivity, as shown in (b). The red indicates high probability of finding the text sequence in the area, whereas the blue is low probability.



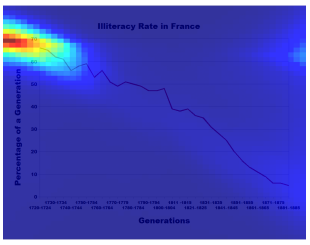
(a) X Domain (Low)



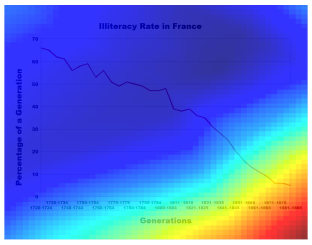
(b) Y Range (Low)



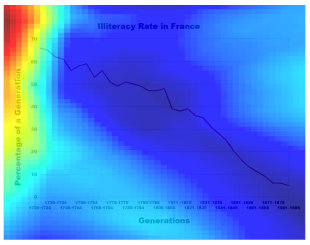
(c) X Domain (High)



(d) Y Range (High)



(e) X Domain High smear based on Levenshtein ratio



(f) Y Range High smear based on Levenshtein ratio

Fig. 6: Additional heat maps corresponding to the likelihood that the given attribute can be found in a certain area of the graph. Graphs (a)-(d) show a binary sigmoid output whereas graphs (e) and (f) show Levenshtein ratios.

A. Attribution of Multimodal Data

Image attribution is the concept of determining what parts of the image contribute to the classification, and how important these parts of the image are to the end result. Most attribution methods work by either perturbing the input signal in some way and observing the change in the output, or by backtracking the influence of the input via a modification of backpropagation.

We can visualize the attribution which provides insight into the classifier decision. These visualizations have been used to characterize which parts of an input are most responsible for the output. This lends some interpretability to the model, and can be used to explain the prediction result.

Thus, in our first experiment, we wanted to compute the relative contributions of the multimodal inputs into our system. Our intuition was that the image text characteristics, such as x, y position were very strong indicators of certain labels. For example, we normally assume that the title appears near the top-center of an information graphic. In order to ensure that our network was taking this information into account, we systematically changed the x and y coordinates of text sequences around a 50x50 equally spaced grid placed on the image. (This is similar to the occlusion perturbation methods in image attribution, where gray occlusion squares are moved across the image.) We then computed the effect of the positional changes on the Levenshtein ratio, see Figure 5, and Binary crossentropy loss, see Figure 6.

Interestingly, it turns out that when the concatenation vector contains the 100-dimensional word vector plus the x, y, and size scalars, (total size of 103), the word vector dominates the decision. The contributions of the position and size are weaker and loose sensitivity, as shown in Figure 5 (b). In order to increase the importance of the text characteristics in networks “A” and “C”-“H”, we bottlenecked the word data through a 3-dimensional, fully connected layer. Thus, the contributions of the word (3-dim) plus the contributions of the text characteristics (3-dim) were equal, for a total layer size of 6-dimensions. In our backup “C” and “D” network, we left the layer size at 103-dimensions as we were much more concerned about the semantic meaning of the words, rather than their position or text size.

Another interesting result was the location of the y-axis label. Our initial thought was the position of the y-axis label would be on the left hand side of the graph. However, our experiments show that there is also high probability that the y-axis label could appear on the right hand side of the graph. This result was confirmed by looking through the training data, as many of the graphs have non-standard y-axis label positions on both the left and right hand side of the graphic.

In Figure 6, we explain why we chose a sigmoid output with binary crossentropy loss rather than the Levenshtein ratio for the x domain and y range networks. It turned out that the

Levenshtein ratio regression output was too similar among the tick mark labels of a graph. For example, if the x axis goes from 2010 to 2019, and the ground truth for the x domain high number is 2019, the Levenstein ratio is 0.75 and 1.0, respectively. As one can see in Figure 6 (e)(f), since the ratio is still quite high along the tick mark labels, it tends to smear out the likelihood across the axis. In order to target specific numbers and areas within the graph, we chose a sigmoid output with binary crossentropy loss. This trains the network to be more selective.

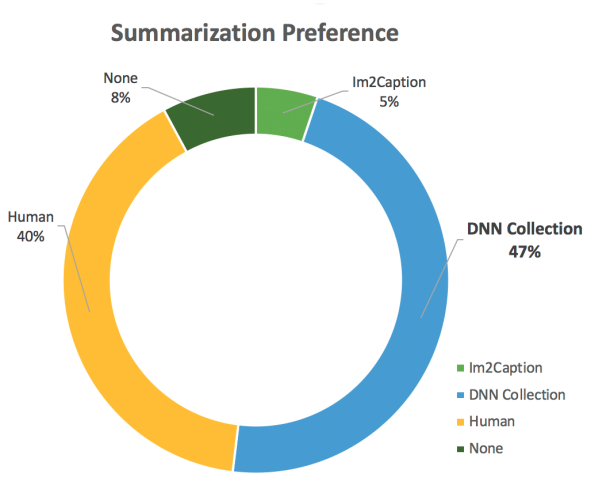


Fig. 7: Summarization preferences on our test dataset when comparing against three other summary generators. Our method was the most preferred summary at 47% whereas the Im2Caption was the least preferred at 5%.

B. Evaluations with Human Participants

1) Initial Evaluation: Mechanical Turker Summary Preferences: The purpose of this experiment was to show that our summaries were judged to be reasonable and accurate by sighted participants. To do this the materials included a graph along with summaries generated using three different methods: The first summarization we compared against was a human provided summarization obtained from mechanical turk. The second method was the Im2Caption model proposed in [8], i.e., Show and Tell model. This model uses a VGG16 CNN extraction module and LSTM for generating captions for the test information graphics. This model was trained on our dataset using 5 unique summaries obtained from human annotators per information graphic, for a total of 3,584 summaries. The model was trained on a NVidia Titan V over 500 epochs. The Im2Caption model generated an arbitrarily long summary until it outputted a special stop token. The third summarization was our method presented here using deep neural network collections (DNN Collection).

For this preference experiment, we built a survey on Mechanical Turk. A graph is presented along with these three generated summaries presented randomly side by side. The instructions for the survey asked participants to select the summary that best represents the given information graphic. In addition to the three generated summaries, the "none", option is given, such that if all of the options do not describe the graph well, the participant is not forced to select one.

The test data we used had on average 3.73 human summaries per image, after filtering out rejected and blank responses. Using these summaries, the total number of preference answers obtained was 1,080 (3.73 summaries x 145 test images x 2 redundant HITS). As shown in Figure, 7 our DNN collection is most preferred at 47% followed by human annotations at 40%. The Im2Caption technique rarely gives enough information and was preferred over all other methods at only 5%. Additionally, 8% of the time, none of the summaries given were deemed adequate by an online crowdsourcing worker. Some final examples of text generations can be seen in Figure 8, 9. Although, one can see that our method's responses are quite thorough and generally preferred over the crowdsourcing summaries, we acknowledge that a carefully crafted human summary that picks up all the nuances in the data is still unattainable by our method. The results of this experiment led us to believe our method was strong enough for evaluation with the target population: people with visual impairments.

2) Visually Impaired Study: To assess our system's usefulness to our target population, we ran a study with visually impaired participants. This study was conducted as an online survey sent out to visually impaired participants, and we were interested in determining whether the summaries generated by our DNN system were helpful to someone attempting to understand social media content containing line graphs. We presented Tweets that originally contained images of line graphs to participant with the images replaced by alt-text from 1 of 3 available sources: (1) the original alt-text with the image (which was "Image" in every case), (2) alt-text generated by our DNN system, (3) alt-text generated by Microsoft Cognitive Services.

We collected random twitter posts that included text and line graph data from the period between September and October of 2019. We then randomly selected 9 of these Tweets for our user study. We presented the text of each of these Tweets to our participants in a random order paired with 1 of the 3 differently-generated alt-texts, so that we could compare our system against other available technology. See Figure 10 for an illustration of what the first 3 Tweets text/alt-text pairs could have looked like for a single participant. One Tweet was shown at a time, and the participant then answered a series of questions about that Tweet before moving on to the next one. We also asked participants to complete a brief demographic questionnaire before taking the survey.

For each Tweet and alt-text, we asked the participants to rate a series of questions on 7 point Likert scales. The questions concerned whether the alt-text improved understanding, whether something was missing or confusing, etc...

For the study, we recruited 100 online participants from the "r/Blind" sub reddit channel. Participants filled out a consent form, a background information survey, and were given a $5 Amazon gift certificate for completing the survey.

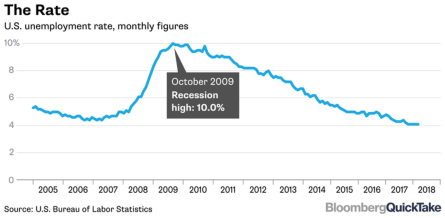


Fig. 8: Summarizations for this information graphic.
Our method: This graph titled the rate shows a changing trend in rate over year. The X-axis goes from 2005 to 2018. The Y-axis goes from 0 to 10%. The end of the graph is falling and the graph shows many fluctuations.
Human 1: This is a line graph showing a changing trend in the u.s unemployment rate.between 2005 to 2018,
Human 2: From 2005, unemployment rises to a recession high of 10.0% in October 2009, then falls to 4% in early 2018.
Human 3: A line graph which conveys a THE UNEMPLOYMENT RATE WILL BE INCREASED BETWEEN THE YEAR OF 2006-2018.
Im2Caption: a line graph which conveys a changing chart

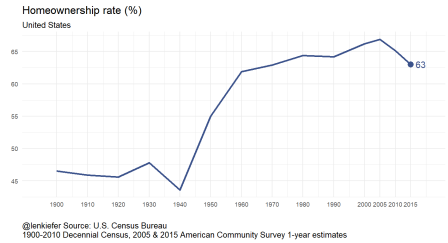
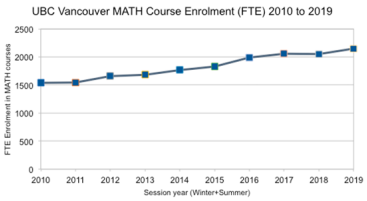
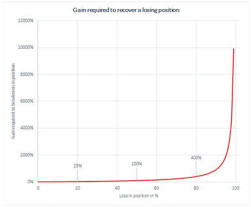


Fig. 9: Summarizations for this information graphic.
Our method: This graph titled homeownership rate (%) united states shows a rising trend in rate over year. The X-Axis goes from 1900 to 2015. The Y-Axis goes from 45 to 65. The end of the graph continues to rise and the graph shows few fluctuations.
Human 1: Between 19000 and 1940 the homeownership rate was relatively stable, with a large jump in 1940 into a steady incline through 2000.
Human 2: THE POPULATION OF DECREASE TO INCREASE.
Human 3: Line graph showing an upward trend in the percent of homeowners in the United States
Im2Caption: looks stable trend of u s concern interest among from 2012 2012.



tweet:I put this graph together for a short piece that will appear in the upcoming edition of the First Year Math and Stats in Canada newsletter. Even with hard caps on some of our courses, there is still an uptick for 2019. FTE faculty counts are almost constant over this time period.

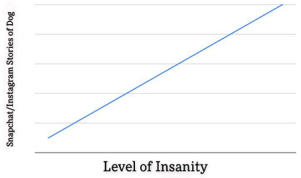
alt text (source DNN): This graph titled ubc vancouver math course enrolment (fte) 2010 to 2019 2500 shows a rising trend in enrolment in math courses over year. The X-Axis goes from 2010 to 201 8. The Y-Axis goes to 2500. The end of the graph continues to rise and the graph show s few fluctuations.



tweet: The importance of stop losses.

Whether we invest fundamentally or quantitatively or by looking at charts it makes sense to imprint this graph in our minds.

alt text (source MSCS): A close up of text on a black background



tweet: i've collected some data and created a graph to give u guys a visual aid

alt text (source Twitter): Image

Fig. 10: Example Tweets and alt text presented to visually impaired users. The survey randomly presented an alt text for the associated twitter image. The three conditions were, our DNN generated alt text, a caption generated from Microsoft Cognitive Services (MSCS), or the default Twitter caption.

The channel contains over 9k members and supports people who are blind, visually impaired, and those who work for the blind. While we expected participants to take about 20 minutes to complete the survey, some participants took a significantly shorter amount of time (and thus clearly were not responding to the survey content). In our survey analysis, we only considered participants that took 10 minutes or more to complete the survey. 23 responses met this criteria and were considered in our analysis.

Participants were 91% male, and 74% had a Bachelor's degree. All participants experienced some form of visual impairment and one participant was totally blind. Participants ranged in age from 18 to 45 years old. The median age group was between 26 and 35 years old.

In the analysis of responses, we looked at the relationship between the alt-text generators used and the perceived quality of that alt-text within the context of that Tweet. The most relevant prompts were ones in which participants assessed the extent to which alt-text improved their understanding. These were the following:

- The alt-text for the graphic improved my understanding of the graphic
- The alt-text for the graphic improved my understanding of the Tweet
- It seems that some of the information in the Tweet and some of the information in the alt-text don't match

Using the 7-point Likert scale responses, we rated the participant's agreement with above statements on a scale from 1 through 7, with a 1 indicating strong disagreement and a 7 indicating strong agreement. For each of these statements, our

DNN generated captions rated better than those provided by Twitter and those generated by Microsoft Cognitive Services; participants who were shown our DNN generated alt-text were more likely to agree that the alt-text improved their understanding of the graphic and Tweet, and were less likely to agree that the Tweet and the information in the alt-text didn't match. The agreement score means are shown in table II. The mean agreement ratings were shown to be significantly more favorable for alt-text provide by our DNN system than that provided by the other two systems using a two-sided Mann-Whitney rank test with $p < 0.05$.

These results indicate a correlation between the use of our DNN captioning system and participants' improved understanding of the Tweets containing text and line graphs in comparison participants' understanding of the Tweets captioned by existing technologies.

Prompt	**DNN**	Microsoft	Twitter
The alt-text for the graphic improved my understanding of the graphic	**5.232**	4.493	4.362
The alt-text for the graphic improved my understanding of the Tweet	**5.29**	4.362	4.406
It seems that some of the information in the Tweet and some of the information in the alt-text don't match	**3.957**	4.652	4.638

TABLE II: Mean agreement rating for prompts given alt-text source. For all three prompts, a significant difference was shown between mean agreement scores for Tweets with alt-text generated using DNN and mean agreement scores for Tweets with alt-text from either of the other two sources, using a two-side Mann-Whitney rank test with $p < 0.05$

V. Discussion

Through this research effort, we came to several important realizations. 1) The use of multiple specific networks worked better than a single network trained end-to-end. Originally, we created a single network and were unable to successfully optimize all the different targets while maintaining a high level of accuracy. We found it to be much more accurate if we focused a neural network on a single optimization, and used a collection of these specific networks to build pieces of the summary. 2) We showed that the relative capacity given to different modalities forced the network to consider different variables. Thus, balancing the contributions of different modalities could be controlled within the concatenation vector. 3) A general image captioning method cannot capture all the details needed. These general image captioning models, like the Im2Caption model or Microsoft Cognitive service, create a very high level, and short description that closely models the training data. With the templating and collection of DNNs, we were able to be much more specific and verbose.

VI. Conclusion

In conclusion, we created an information graphic summarization framework that utilizes a collection of neural networks to formulate a summary. The framework utilizes the visual, textual, positional, and size characteristics within several multimodal neural networks. In our experimentation, we showed the effectiveness of our models and were able to generate summaries that were preferred even over some human provided summaries. Importantly, people with visual impairments rated our summaries as more helpful in understanding Tweets containing line graphs over other alternatives. All the code, network models, dataset, and online demo will be provided online.

VII. Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. 1954364.

References

[1] R. Burns, S. Carberry, and S. E. Schwartz, "Classifying salient textual entities in the headlines and captions of grouped bar charts." in *FLAIRS Conference*, 2015, pp. 217–220.

[2] J. Gao, R. E. Carrillo, and K. E. Barner, "Image categorization for improving accessibility to information graphics," in *Proceedings of the 12th international ACM SIGACCESS conference on Computers and accessibility*. ACM, 2010, pp. 265–266.

[3] L. Ferres, G. Lindgaard, L. Sumegi, and B. Tsuji, "Evaluating a tool for improving accessibility to charts and graphs," *ACM Transactions on Computer-Human Interaction (TOCHI)*, vol. 20, no. 5, p. 28, 2013.

[4] P. S. Moraes, S. Carberry, and K. McCoy, "Providing access to the high-level content of line graphs from online popular media," in *Proceedings of the 10th International Cross-Disciplinary Conference on Web Accessibility*. ACM, 2013, p. 11.

[5] D. Jung, W. Kim, H. Song, J.-i. Hwang, B. Lee, B. Kim, and J. Seo, "Chartsense: Interactive data extraction from chart images," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. ACM, 2017, pp. 6706–6717.

[6] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal deep learning," in *Proceedings of the 28th international conference on machine learning (ICML-11)*, 2011, pp. 689–696.

[7] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[8] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*. IEEE, 2015, pp. 3156–3164.

[9] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick, and D. Parikh, "Vqa: Visual question answering," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2425–2433.

[10] T. Yao, Y. Pan, Y. Li, Z. Qiu, and T. Mei, "Boosting image captioning with attributes," in *IEEE International Conference on Computer Vision, ICCV*, 2017, pp. 22–29.

[11] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang, "Bottom-up and top-down attention for image captioning and vqa," *arXiv preprint arXiv:1707.07998*, 2017.

[12] S. Carberry, S. Elzer Schwartz, K. Mccoy, S. Demir, P. Wu, C. Greenbacker, D. Chester, E. Schwartz, D. Oliver, and P. Moraes, "Access to multimodal articles for individuals with sight impairments," *ACM Transactions on Interactive Intelligent Systems (TiiS)*, vol. 2, no. 4, p. 21, 2012.

[13] E. Kim and K. F. McCoy, "Multimodal deep learning using images and text for information graphic classification," in *Proceedings of the 20th International ACM SIGACCESS Conference on Computers and Accessibility*, 2018, pp. 143–148.

[14] R. Smith, "An overview of the tesseract ocr engine," in *Document Analysis and Recognition, 2007. ICDAR 2007. Ninth International Conference on*, vol. 2. IEEE, 2007, pp. 629–633.

[15] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.