Identification of Most Relevant Paragraph in an Article for a Bar Chart Using Machine Learning and Kullback-Leibler Divergence

Alyssa Zevallos and Stephanie Schwartz
Millersville University
{aezevall,stephanie.schwartz}@millersville.edu

ABSTRACT
This paper describes a project aimed at identifying the most relevant paragraph in an article given a bar chart. This project fits in with a larger effort to design a system that provides access to information graphics for the visually impaired.

KEY WORDS
Information retrieval, machine learning, information graphics, Kullback-Leibler divergence

1. Introduction

Information graphics such as bar charts and line graphs play an ever increasing role in popular media. Providing a visual illustration of data, they function as supplemental information to an article, conveying messages to be quickly interpreted without having to consult article text. Information graphics are useful for displaying comparisons, contrasts, trends, and other similar messages.

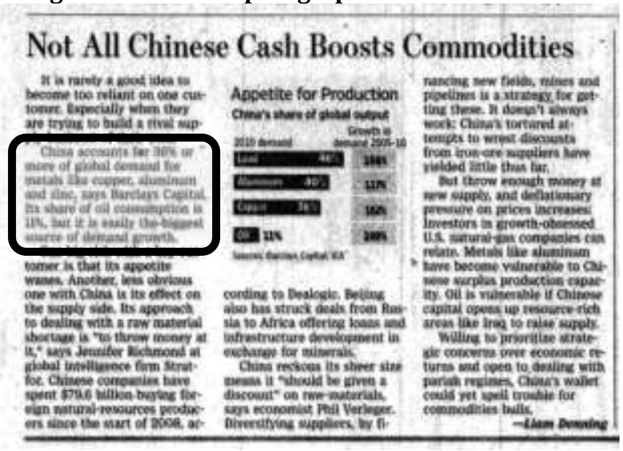
Interest in data visualization has seen a near threefold increase since 1990 based on the number of book references [1]. The goal of information graphics is to be able to provide ease of use and clarity; however, those who are blind or visually disabled are unable to take advantage of data visualization or be able to interpret those graphs.

Blind and visually disabled users have access to a variety of assistance technology such as haptic devices [2] and Braille printers [3]. While helpful, these methods require additional training to learn, are expensive, and aren't universally available. Screen readers are more widely available and work as an add-on to pre-existing software such as a computer's operating system or Internet browser. They audibly read text from a computer screen and provide ways to interact with different elements.

Navigation through screen readers eliminates the need to visually follow a cursor on screen. In terms of web pages specifically, users are at the discretion of the website developer's adherence to accessibility standards. Elements on the page may be read out of order or might be missed entirely such as an information graphic in an article. Information graphics pose a unique problem for screen readers.

For images and information graphics, screen readers rely on alternate text, typically called alt text. In many instances, alt text is unavailable [4], so blind or visually disabled users know that something exists, but have no indication as to what it may be. Inaccurate or missing alt text was considered as the fourth most frustrating problem according study conducted by WebAIM [5]. When available, the alt text description may not be sufficient to get an understanding of the object.

Figure 1: A salient paragraph close to a bar chart



Not All Chinese Cash Boosts Commodities

It is rarely a good idea to become too reliant on one customer. Especially when they are trying to build a rival sup- [illegible]

China accounts for 30% or more of global demand for metals like copper, aluminum and zinc, says Barclays Capital. Its share of oil consumption is 11%, but it is easily the biggest source of demand growth.

[illegible] tomer is that its appetite wanes. Another, less obvious one with China is its effect on the supply side. Its approach to dealing with a raw material shortage is "to throw money at it," says Jennifer Richmond at global intelligence firm Stratfor. Chinese companies have spent $79.6 billion buying foreign natural-resources producers since the start of 2008, according to Dealogic. Beijing also has struck deals from Russia to Africa offering loans and infrastructure development in exchange for minerals.

China reckons its sheer size means it "should be given a discount" on raw materials, says economist Phil Verleger. Diversifying suppliers, by financing new fields, mines and pipelines is a strategy for getting these. It doesn't always work: China's tortured attempts to wrest discounts from iron-ore suppliers have yielded little thus far.

But throw enough money at new supply, and deflationary pressure on prices increases: Investors in growth-obsessed U.S. natural-gas companies can relate. Metals like aluminum have become vulnerable to Chinese surplus production capacity. Oil is vulnerable if Chinese capital opens up resource-rich areas like Iraq to raise supply.

Willing to prioritize strategic concerns over economic returns and open to dealing with pariah regimes, China's wallet could yet spell trouble for commodities bulls.

—Liam Denning

Information graphics tend to expand on a concept in the text to provide a visual correlation. Typically, a graph is salient in one section of an article (Figure 1) although the graph isn't necessarily in close proximity to that section. The graph may even be on a different page. Graphs in print tend to be placed where they are more visually appealing. In many cases, printed articles will not have their graph displayed in their digital version as is the case with BusinessWeek articles.

Proximity of a graphic to its most relevant paragraph is not a common occurrence. Previous work on line graphs implemented a method using machine learning algorithms to rank paragraphs in an article in order of the most to the least relevant paragraph [6]. By comparing words in a paragraph to words used to describe the graph, the system attempted to identify the most relevant paragraph in an
]

article. It bases its accuracy score by determining if its top chosen paragraph matched any of the user-selected paragraphs. For this project, we wanted to expand that method to bar charts to determine if it would work as well.

The overall goal of the SIGHT project is to work in conjunction with a pre-existing screen reader to be able to identify an information graphic, create a short summary of its intended message, and provide the summary at an appropriate time [7]. The focus of this paper is on the problem of correctly identifying the most relevant paragraph for bar charts.

1.1 Importance

Figure 2: Visual representation of Figure 3

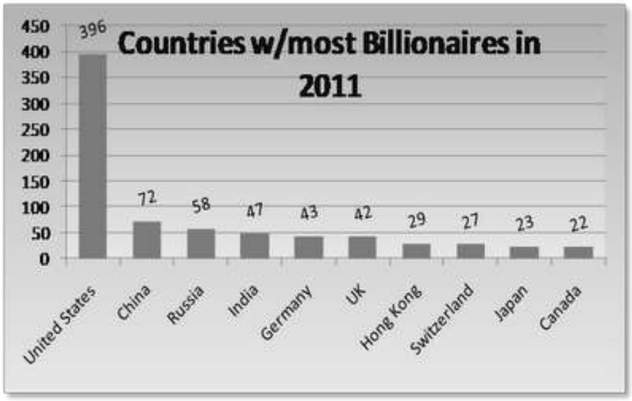
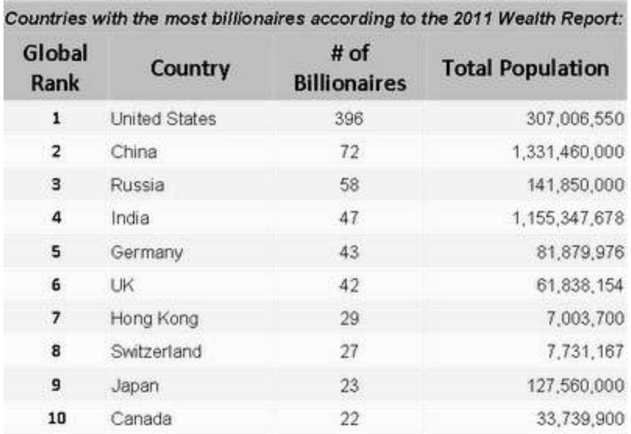


Figure 3: Text based data representation



Countries with the most billionaires according to the 2011 Wealth Report:

Global Rank	Country	# of Billionaires	Total Population
1	United States	396	307,006,550
2	China	72	1,331,460,000
3	Russia	58	141,850,000
4	India	47	1,155,347,678
5	Germany	43	81,879,976
6	UK	42	61,838,154
7	Hong Kong	29	7,003,700
8	Switzerland	27	7,731,167
9	Japan	23	127,560,000
10	Canada	22	33,739,900

Information graphics provide quick understanding. It only takes about 150ms to interpret an image and an additional 100ms to gather its meaning compared to about 100ms per printed word [8] [9]. For example, in the figures above, Figure 2 is the visual representation of Figure 3. While it can be interpreted from the text in Figure 3 that the United States is leader in the number of billionaires, Figure 2 emphasizes the impact of the data. Figure 2 would take around 250ms to interpret while Figure 3 may take up to 2 full seconds.

Graph accessibility has not matched the increased use of digital graphics [10]. Even though guidelines are in place for universal accessibility, 80% of all websites have accessibility problems. Insufficient use of alternate text poses the biggest accessibility issue for a website [4]. Providing users with an easy to understand message for any given information graphic at the appropriate time within an article is an important task in order to advance user accessibility [11].

1.2 Most relevant paragraph location

In order for an information graphic to be useful in helping a visually disabled user to comprehend an article, the information from the graphic must be supplied at an appropriate time. The physical location of the graph does not necessarily correlate to the most salient time in an article to provide the message. Because of this, work has to be done in order to identify the most relevant paragraph in an article. This is a non-trivial task because an information graphic is rarely mentioned directly by name within article text. Sometimes data points in a paragraph specifically correlate to the graphic, but usually when a graphic is focused on in a paragraph, it’s a generalization.

1.3 Difference between bar charts and line graphs

One of the key, non-aesthetic differences between line graphs and bar charts is the intended message. Line graphs can be separated into four basic categories: rising-trend, change-trend, change-trend-return, and big-jump. Bar charts can be grouped into categories such as relative differences (Figure 4), minimums or maximums (Figure 5), and ranks (Figure 7) in addition to trends (Figure 6). Generally speaking, line graphs depict ordinal data with a preference towards chronological data.

Trends are ordinal and are ordered chronologically (Figure 6) while ranks are labeled independently such as by goals (Figure 7). The goal of line graphs is to follow data over a period of time and determine if some kind of pattern exists. Bar charts, on the other hand, are more versatile and may provide comparisons between similar entities such as the number of filings to deregister securities (Figure 4). Because bar charts have a wider selection of intended message types, it becomes more difficult to find a set of words to encompass them. Line graphs are much simpler because all of the graphs are related in that they are ordered by time, so words like fall, jump, or increased would be more likely to be found and be relevant in a line graph article than a bar chart article.

Figure 4: Relative difference

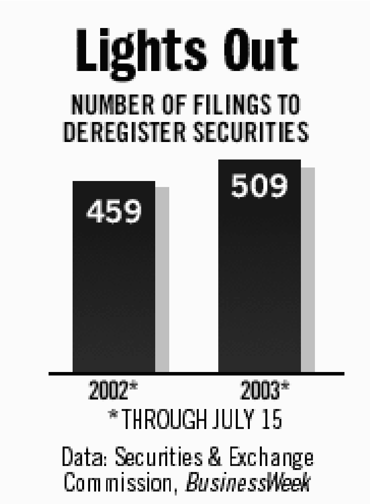


Figure 5: Minimum

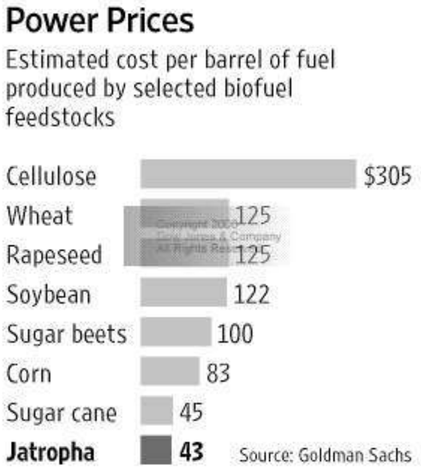


Figure 6: Trend

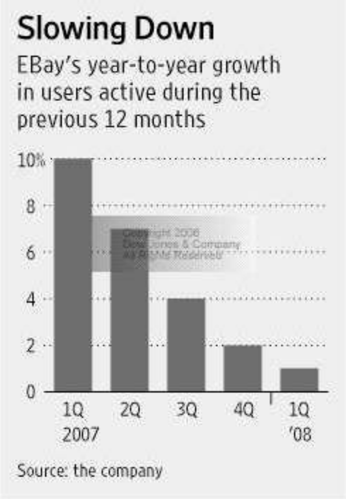
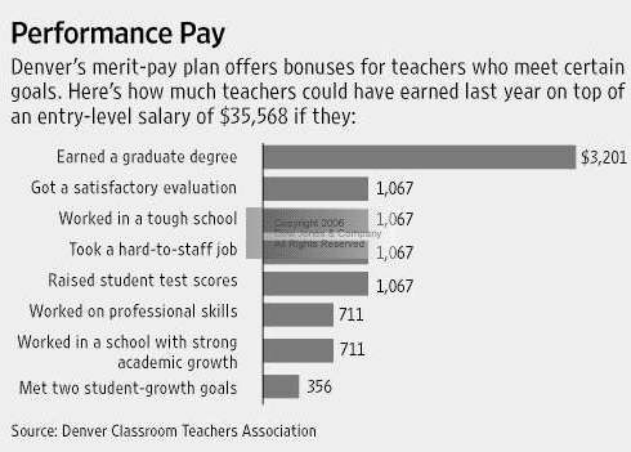


Figure 7: Rank



Related Work

1.4 Ranking methods

Our methods for relevant paragraph selection are similar to that of information retrieval and pseudo-relevance feedback. In one research project, the authors studied the effectiveness of query expansion terms based on term distribution which they found to be unsuccessful [12].

Christopher J.C. Burges researched a machine learning method to create a ranked structured output called LambdaRank. LambdaRank eliminates unnecessary work by the learning algorithm for when items are already in their proper place. He uses a Jacobian matrix instead of the typical kernel matrix [13].

For systems which require speed and simplicity, the authors of created a ranking system for retrieval using a neural network. Their system utilizes rank boundaries to approach it as an ordinal regression problem instead of a ranking problem [14].

1.5 Testing methods

There are many different types of similarity measures used for pattern recognition. Similarity measures are used in the computer science field as well as geology, physics, and psychology [15]. The main measure we use for relevant paragraph testing is KL divergence.

A project analyzing the effectiveness of query performance used KL divergence to measure a clarity score. The more ambiguous the query, the lower the clarity score and the less effective it was at obtaining relevant documents [16].

1.6 Graph analysis

A similar project, iGraph, focused on providing a natural language summary of a graphic. Similar to the SIGHT system which will be addressed later, iGraph can be used to navigate a graph and obtain more details than the given short description [17].

The Tangible Graph Builder uses a tangible user interface to create graphs. It uses various weighted items and audio feedback.. The researchers developed a method to digitally track changes in the graph in order to maintain a model of the graph [18].

2. Identifying Relevant Paragraphs

System

2.1 Architecture

The problem of most relevant paragraph identification is only a piece in a larger puzzle. Our project utilizes SIGHT's ability to identify bar charts within an article. SIGHT (Summarizing Information GrapHics Textually) is the overriding architecture encompassing the identification, breakdown, and summarization of an information graphic. It functions as a browser extension with the pre-existing JAWS sight reader software.

The graphs and articles used in the experiment were collected from a range of different national and local news outlets. All article data was stored in our MySQL database except for the original formatting.

The graph analysis process begins with the Visual Extraction Module. The image data is broken down into XML format is then augmented with supplementary information in the Preprocessing and Caption Tagging Module. From there, the intended message is determined by three factors: the effort required by the user to figure out the designer's intention, whether or not a particular bar or set of bars stands out due to coloring or caption, and particular words in a caption indicating the type of intended message. Once the intended message is derived, a textual summary is created and sent to the JAWS architecture to read to the user [19].

2.2 Key elements

For this project, there are two main features: the expansion word list generator and the evaluator. The system is built utilizing a form of machine learning. The machine learning algorithm aims to capture an all-encompassing generalization of data due to the difficulty in trying to account for the multitude of variations in data. Our project uses supervised machine learning due to the finite number of solutions and our main concern how a set of paragraphs or a bag of words will be ranked.

2.3 Word list generation

A key piece of the evaluation system is the expansion word list. The list contains the words most commonly found in articles for the type of graph being tested. The expansion word list is created using a vector space model (VSM), a widely used model in the information retrieval realm [20].

The thought behind using a VSM is for query expansion so that these words create a generalization for all graphs of the given type to aid in pattern recognition. By adding these words to the evaluation, the idea is that there will be a higher chance of accurately selecting a most relevant paragraph. A paragraph containing words from the graphic caption and expansion word list would score much higher than a paragraph with no related words.

It is important to make sure that the training set and the data set do not have overlapping graphics. Doing so creates the possibility of generating a false accuracy score since data would be duplicated. The training set should capture what the general graph will be like. For example, our training data models the distributions used in the original test using line graphs. The distribution of intended message as well as the average article size was replicated to match the original experiment as closely as possible. The importance of capturing a picture of the general bar chart is so that the expansion words generated can then be effective for an array of different bar chart types and styles.

2.4 Evaluation

In addition to using the test set of graphs, the evaluation process requires outside input to run. It requires six inputs: two separate human evaluators' picks for most relevant paragraphs, the closest paragraph to the graphic, the intended message and the x-axis bar labels, a filter file indicating which graphs to test on, and a shortened expansion word list.

Two human evaluators went through each graph and associated article to select what they believed to be the most relevant paragraph. Every graph in the corpus was analyzed so that different test sets could be used in the future without having to do any additional work.

The most relevant paragraph in an article can be determined by different factors. The placement of the graph in the article can be one of these indicators. The closest paragraph selection was completed manually for each graph. When a graph was within the article, a typical setup, the paragraph within the closest proximity by reading direction was selected which can be seen in Figure 1. For articles with linked graphs, we selected the paragraph closest to the link. Those with an unknown original format had the first paragraph in the article selected as the closest. Closest paragraph selection for bar charts varies from the original line graph experiment due to linked or missing original article formats. It is possible that this inconsistency may have caused some drop in closest paragraph accuracy in comparison to the original line graph experiment.

The graph annotations were chosen by a consensus of human evaluators instead of running the intended message generator to assure accuracy in the testing data. An important point to note is that even though the system requires intended message as an input, it is not used during the evaluation, a choice made by the program's original author. The bar labels, however, are used as arguments supplementing the graphic caption data. The original line graph test only used the first five graphs for intended message and arguments which is in contrast to ours which utilized every graph, resulting in slightly better scores.

Unlike the VSM which used a query to collect the graphs it needed, the evaluator requires an outside file stating which graphs need to be used. A query is run to collect each individual graph's graph name, caption, description, text in graphic, article title, article subtitle, and article content/text. The filter file defines the test set.

The easiest outside input to gather for the evaluation system is the shortened expansion word list, assuming the expansion word list generator already produced a full expansion word list. The top 25 words, which are ranked by most to least relevant automatically, make up the shortened expansion word list used for input. The words are used to augment the graphic caption data.

2.5 VSM Procedure

Once the training set is determined, a mySQL query to the database grabs the associated data for each graph: the graph name, graphic caption, graphic description, text in graphic, article title, article subtitle, and article text. As the most commonly used words are pulled out, there is also a set of stop words that are ignored which can be seen in Table 1. These stop words provide no assistance in narrowing down what specifies a bar chart. All words outside the list of stop words are placed into a ranked word list representing the word count distribution.

Table 1: Stop Words

a, able, about, across, after, all, almost, also, am, among, an, and, any, are, as, at, be, because, been, but, by, can, cannot, could, dear, did, do, does, either, else, ever, every, for, from, get, got, had, has, have, he, her, hers, him, his, how, however, i, if, in, into, is, it, its, just, least, let, like, likely, may, me, might, most, must, my, neither, no, nor, not, of, off, often, on, only, or, other, our, own, rather, said, say, says, she, should, since, so, some, than, that, the, their, them, then, there, these, they, this, tis, to, too, twas, us, wants, was, we, were, what, when, where, which, while, who, whom, why, will, with, would, yet, you, your

The process works by first creating a dictionary and distribution of all words in one article. Using the word distribution of the entire article, a pseudo-relevant paragraph is selected. The word distribution of all pseudo-relevant paragraphs, $W_{p,}$ is stored in one vector while the word distribution of all articles, W_a, is stored in a separate vector. Using those two vectors, the overall word frequency vector representing a graphic, W_g, can be computed. W_g represents the expansion word list.

$$W_g = \textbf{normalized}\left(W_p - \frac{< W_p, W_a >}{< W_a, W_a >} \cdot W_a\right)$$

The ranking process is iterative, modifying the word list already generated by all graphs in the test set. Subsequent iterations alter the word list, moving the more relevant words to the top of the list. Once the word list changes only minimally or not at all, the process can be stopped. Twenty iterations are typically more than enough. The final product is a ranked list of all words within the training set from most common to least common. With the word list created, the evaluation process can begin.

The expansion word list creator and the evaluation system both use the Perl programming language. To supplement text evaluation, a Perl API called WordNet aids by disambiguating words within a document as well as breaking words down using a stemmer. The corpus interface utilizes a model-view-controller setup built using PHP with the help of CodeIgniter, a PHP framework. A MySQL database is used to store all graphs, articles, annotations, and other relevant data. The front end interface for the database is limited to certain operations like uploading, adding annotations, editing an article, and other simple functions.

2.6 Evaluation Procedure

The evaluation system measures two types of results: random paragraph selection accuracy and Kullback-Leibler divergence accuracy. To clarify, when graphic caption information is referred to, this includes the graphic caption, description, text in graphic, and bar labels.

Random paragraph selection is as its name describes. The system selects an arbitrary paragraph in the article and checks to see if it matches any of the user-selected paragraphs. As the size of the article increases, the accuracy rates for random selection decreases.

Kullback-Leibler (KL) divergence is a similarity measure that compares two entities. Also known as relative entropy, KL divergence falls under Shannon's entropy family. Entropy is the concept of probabilistic uncertainty or the uncertainty of an unknown variable. Shannon's entropy measures a random variable's average amount of contained information. Once the outcome of that variable is known, it also measures the amount of uncertainty that was removed [21]. In the case of this experiment, KL divergence measures the word distribution for paragraphs in a document against the word distribution of the graphic caption data. If *p* and *q* represent the word distributions, KL divergence can be measured using the following

$$D_{KL}(p||q) = \sum_{i \in V} p(i) log \frac{p(i)}{q(i)}$$

where *V* is the dictionary of all words, *p(i)* and *q(i)* represent the count of word *i* in distributions *p* and *q* respectively. Scores are on a scale from 0 to 1. A score of 1 indicates a correctly chosen paragraph with decreasing scores representing less and less likely matches.

Within the KL divergence group, there are three sub-methods tested: P-KL, P-KLE, and P-KLEM. "P" represents the paragraph selection, "KL" represents the KL divergence, "E" represents the expansion word list, and "M" represents a mixed model using scores from sentence and paragraph. P-KL is the KL divergence based on sentences within a paragraph. Scores are given to the sentences in a paragraph based on how well they match to the graphic's caption information and the word distribution. P-KLE uses the same model as P-KL, but augmented with the shortened expansion word list. The sentence text is now being compared to both the graphic caption information and the word list. The purpose of adding more is that there will be, theoretically, a higher chance of having a word match since there are more words available to compare to. P-KLEM takes the P-KLE method one step further by using a mixed model of sentence in a paragraph scores like the previous methods in addition to words in a sentence scores to select the most relevant paragraph.

Results and testing

This section discusses the different methods and processes used to measure the effectiveness of most relevant paragraph identification for bar charts. The KL divergence measure, three methods were used: TOP, COVERED, and normalized discounted cumulative gain (nDCG). All three methods are measured on a scale of 0 to 1 with 0 meaning no similarity and 1 being the most likely match.

TOP measures whether the top system pick matches the human-evaluator-selected most relevant paragraphs. If the system pick matches an evaluator's top pick, it gets a score of 1. If there is no match or if it's anything beyond the system's top pick, that paragraph is given a score of 0. This is measuring the selection accuracy for the most relevant paragraph.

COVERED compares the top three system picks to determine if any match the human-evaluator-selected most relevant paragraphs. Using this method, if the top paragraph matches, it gets the highest score of 1. If the second pick matches, it gets a score of 0.63. If the third pick matches, it gets a score of 0.5. Anything beyond the top three for the COVERED method gets a score of 0. Since more paragraphs are available to try to match the evaluators' choices, this method should theoretically perform better than TOP since it is measuring the selection accuracy for any relevant paragraph, not the most relevant paragraph.

nDCG compares how closely the system's ranking matches the evaluators' ranking. The *nDCG* equation used in the experiment is slightly modified to account for all documents. The original equation does not accurately measure the discount assigned to the second ranked document. The following was used for our needs:

$$nDCG_p = \frac{DCG_p}{IDCG_p}$$

$$\text{where } DCG_p = \sum_{i=1}^{p} \frac{rel_i}{log_2(i+1)}$$

$$\text{and } IDCG_p \text{ is the highest possible } DCG_p$$

where rel_i represents the gain from paragraph retrieval and $1/log_2(i + 1)$ represents the discount based on position ***i***. *p* is the paragraph rank which is set at a max of $p = 3$ meaning that it will only look at the top three highest ranked paragraphs. As the original author explains,

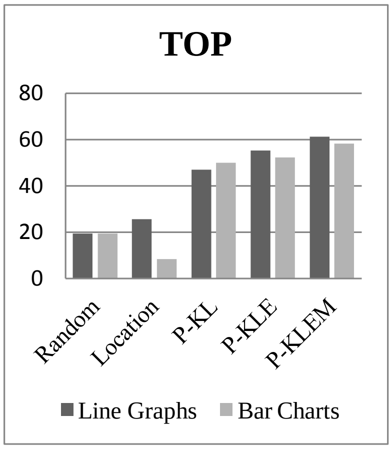
"The value of rel_i depends on p and the number of relevant paragraphs identified by the human evaluator. If the human evaluator identifies k paragraphs as relevant (where $k \leq p$), then $rel_i = k$ if the i-th ranked paragraph by the system matches the top-ranked paragraph by the human evaluator and is equal to $k - 1$ or $k - 2$ if it matches the paragraph ranked second or third by the human evaluator, respectively. Ranking a good paragraph higher gets less discount with the same gain, and ranking a better paragraph at the same position gets higher gain with the same discount. They both achieve a better nDCG score." [6].

Unlike the original experiment, the bar labels were included as extra input to supplement the graphic caption data. For bar charts, a few percentage points in accuracy were gained. The evaluation file is built to take these labels as input, so it's not clear as to why the original experiment did not take advantage of this feature.

2.7 Testing with Original Conditions

The testing set contained 100 graphs and the training set utilized the remaining 329 graphs for the expansion word list. Our training set contained 62 more bar charts than the original line graphs test. The test set was broken down by intended messages: 59 were trend-type messages, 17 were rank-type messages, 8 were relative-difference-type messages, and 16 were maximums or minimums. The average number of paragraphs per article was 15.38, matching the original experiment.

Figure 8: Comparison between line graphs and bar charts using the TOP measurement



Two human evaluators selected what they believed to be the most relevant paragraphs in an article. Evaluator-1 selected an average of 1.22 paragraphs and Evaluator-2 selected an average of 1.21 paragraphs. The two evaluators agreed 85% of the time on their top ranked paragraph. That means that 15% of the graphics show that the most relevant paragraph is not always clear. However, this is an improvement over the line graphs which only saw an agreement of 66%. It is unknown if this improvement had anything to do with our evaluators working in the same room. The conditions for the line graph evaluators are unknown.

Figure 9: Comparison between line graphs and bar charts using the COVERED measurement

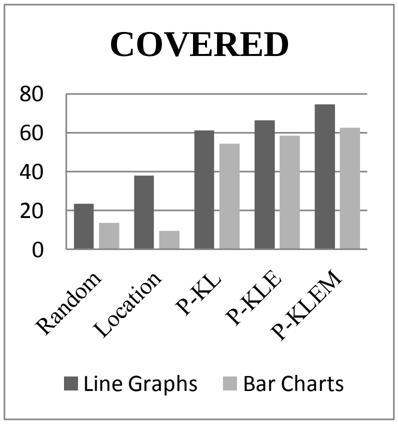
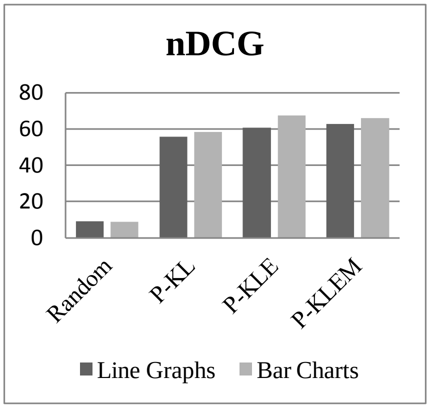


Figure 10: Comparison between line graphs and bar charts using the nDCG measurement



When looking at the expansion words, while there is some noise like “according” or “come” there are also words that would indicate trends such as “rise,” “grow,” “decline,” “gain,” “rising,” and “fall.” As mentioned earlier, bar charts represent a wider range of intended messages beyond trends alone, making it more difficult to capture a single, generalized image of a bar chart. A noisy word like “according” is common, yet ineffective, due to articles citing information sources such as the phrase “according to budgetary figures.” The expansion words for bars and lines can be seen below in Table 2.

Table 2: Shortened expansion word list

Bar Charts	rise expect grow average according long global demand look large low same decline come economic gain slow real domestic foreign chart gross rising fall national
Line graphs	up down rise according fall high expect low late grow decline hit federal buy free drop worry dip long august jump risky back average surge

Tables 3 and 4 and Figures 8 and 9 show the results of our experiment in contrast to the original experiment using line graphs, Table 4. The results show two baseline figures, random and location, as well as the results for KL divergence. The baseline test “random” is a randomly selected paragraph in the article and the “location” baseline test is the paragraph located in closest proximity to the graph. They are considered as baseline because no outside information or comparisons are needed.

Table 3: Results of bar charts

	Random	Location	P-KL	P-KLE	P-KLEM
TOP	19.3%	8.3%	50.0%	55.2%	58.3%
COVERED	13.5%	9.4%	54.2%	58.3%	62.5%
NDCG	8.8%	--	58.5%	67.6%	66.0%

Table 4: Original results of line graphs

	Random	Location	P-KL	P-KLE	P-KLEM
TOP	19.4%	25.5%	46.9%	55.1%	61.2%
COVERED	23.4%	37.8%	61.2%	66.3%	74.5%
NDCG	9.1%	--	55.8%	60.6%	62.9%

Our test used the same evaluation as the original experiment, but our COVERED results were not as comparable. Though they show the basic improvement of P-KLEM and P-KLE over P-KL, the difference between TOP and COVERED is minimal. One possible theory to this discrepancy would be that bar chart articles typically only use a small, pertinent section in relation to a graph. Due to this, there would be less chance of there being more than one relevant paragraph, so COVERED wouldn’t have as significant of an increase to TOP.

Another poor performer in our test set was the “location” baseline. This may have been caused by lacking original article format data. Another potential culprit could be the types of sources used. Between the test and training set, there were 26 unique article sources used for bar charts and 24 unique sources for line graphs. The distributions of the articles are more even for line graphs than bar charts which can be seen in Tables 5 and 6. BusinessWeek has the highest distribution of articles in bar charts and they are also the source that does not retain their graphs in digital editions.

Table 5: Line graph article sources

	Distribution
BusinessWeek	25.9%
New York Times	14.8%
USA Today	21.1%
The Wall Street Journal	25.0%
Other (20)	13.1%

Table 6: Bar chart article sources

	Distribution
BusinessWeek	40.9%
The Wall Street Journal	38.1%
Other (24)	20.9%

The best results for bar charts came from the normalized discounted cumulative gain (nDCG) scoring method. As can be seen in Figure 10, this method is the closest performer to line graphs.

2.8 Testing by Intended message

Since the results from our tests did not meet or exceed those from the original test, further investigation was necessary. The next step was to separate bar charts into two intended message-based groups: trends and non-trends. Non-trends are composed of relative differences, ranks, and maximums and minimums.

Due to dividing the corpus into smaller groups, the traditional evaluation was not feasible. After taking out 100 graphs for a test set, there were typically less than 100 graphs left to train on. To counter this problem, we used a leave-one-out cross validation approach.

Cross validation is used for machine learning when there is not a large amount of data available. Leave-one-out works by using a single graph as the test set and using the remaining graphs as the training set. This is done for every graph within the group [22]. By using leave-one-out, the training set has the ability to capture a generalization of the rest of the group since it's no longer limited by having 100 graphs removed, only 1. Unfortunately, this also meant that an expansion word list had to be created for every graph used in the corpus, a total of 429. This is a very time expensive approach even when parallelized.

Within the intended message groupings, evaluator-1 chose an average of 1.14 to 1.31 paragraphs, evaluator-2 chose an average of 1.13-1.46 paragraphs, and agreement between the evaluators ranged from 75.9% to 84.7% with a kappa statistic ranging from 0.738 to 0.834. There were 29 relative differences, 92 ranks, 72 maximums and minimums, and 236 trends.

Table 7 shows the results of trends versus Table 8's non-trends. "Trends" outperformed "non-trends" as expected since the "trends" group is more closely related to the original test's line graphs.

Table 7: Results of bar chart trends

	Random	Location	P-KL	P-KLE	P-KLEM
TOP	19.0%	7.0%	52.0%	56.0%	56.0%
COVERED	12.2%	10.0%	52.0%	58.0%	59.0%
NDCG	8.5%	--	58.9%	68.8%	68.1%

Table 8: Results of bar chart non-trends

	Random	Location	P-KL	P-KLE	P-KLEM
TOP	18.6%	14.9%	34.0%	44.7%	44.7%
COVERED	13.9%	14.9%	38.3%	46.8%	46.8%
NDCG	8.4%	--	47.0%	54.1%	52.5%

One last test was conducted using the leave-one-out method. Its test set used the same graphs used in the traditional format test set while using the expansion words created for the individual intended message groups. Doing this produced the best results of the leave-one-out tests and was comparable to the traditional test. Results for random, location, and P-KL remained the same since there were no changes to affect these scores. The differences start when the expansion word list is added. In all three testing methods, the traditional method outperformed the mixed model in P-KLE with a range of 4.1% to 7.2% difference in favor of the traditional method. P-KLEM was a bit closer with a range of difference from 0% to 3.5%. While the mixed model and the traditional test have similar results, not only are the results from traditional slightly better, the work and time involved to prepare the expansion word lists for the mixed model make the traditional method a more feasible and better option to use.

2.9 Testing by Article/Paragraph Size

Since no significant improvement could be seen using a leave-one-out method, we decided to look into the articles themselves and break them up according to article length. The graphs used were the same in the original conditions test set. The size grouped test sets were designed to mirror the original line graphs test which had 31 small graphs, 35 medium graphs, and 34 large graphs.

Articles with 2 to 10 paragraphs are considered as the "small" category. Articles with 11 to 15 paragraphs fall under the "medium" category. Articles with more than 15 paragraphs belong to the "large" category. In the original test with line graphs, the average number of paragraphs was 6.65, 13.67, and 25.09 for small, medium, and large articles respectively. Our test set was able to match these averages within 0.05 paragraphs per article with the average number of paragraphs for small, medium, and large articles being 6.65, 13.66, and 25.12 respectively. The same expansion word list used for the traditional method is used for all size subsets.

As the size of the article gets smaller, the higher the accuracy for most relevant paragraph selection becomes.

This makes sense since a shorter article will have fewer options to choose from, following basic probability rules. If the number of choices increases, then the probability of selecting a paragraph accurately decreases. In our case, a choice represents a singular paragraph with the article containing and the entire selection of choices.

The small subset, Table 11, performed the best out of the three sizes, performing the best with the nDCG method by topping out at 78.4% for the P-KLE method. The next best performing group is the overall 100 test set, Table 3, then medium, Table 10, then large, Table 9. Between the best and worst size groups, small and large respectively, there is a significant difference. TOP and COVERED both see a drastic difference of around 23% for both P-KL and P-KLE.

Table 9: Results of bar charts with large articles

	Random	Location	P-KL	P-KLE	P-KLEM
TOP	9.9%	9.1%	33.3%	45.5%	42.4%
COVERED	7.7%	12.1%	39.4%	51.5%	45.5%
NDCG	5.0%	--	44.8%	55.8%	55.8%

Table 10: Results of bar charts with medium articles

	Random	Location	P-KL	P-KLE	P-KLEM
TOP	14.8%	11.4%	37.1%	42.9%	40.0%
COVERED	9.6%	11.4%	42.9%	48.6%	45.7%
NDCG	6.4%	--	53.7%	67.8%	60.8%

Table 11: Results of bar charts with small articles

	Random	Location	P-KL	P-KLE	P-KLEM
TOP	34.2%	13.3%	56.7%	63.3%	66.7%
COVERED	23.6%	16.7%	63.3%	70.0%	73.3%
NDCG	15.2%	--	69.1%	78.4%	77.5%

3. Conclusion

Information graphics are a powerful tool in media used to convey messages in a succinct way. With a rise in the use of digital media, the use of data visualization has followed. Blind and visually disabled users are unable to use these graphs as the designers intended. Many sight assistance software programs do not have the means to be able to interpret a graph which is where our project steps in. To aid in developing the SIGHT software for blind and visually disabled users, our project works to find the most relevant point in the article where it would be best to present the user with bar chart data.

The project followed the structure of a similar experiment used for line graphs. By using the line graph results as a baseline, bar charts were tested to measure if the same method could be used. Overall, bar charts were unable to duplicate or exceed the line graph results, so further work would be required before there could be implementation of most relevant paragraph selection for bar charts.

Future work on the project may include using testing methods different than Kullback-Leibler such as cosine similarity. While they proved effective for line graphs, bar charts did not meet the same results. Another method within the inner product family of similarity measures would be useful as they are commonly used within the information retrieval realm. The overall goal of the project aims for accuracy over efficiency, so a universal method for all graphs is unnecessary as long as there is at least one method that consistently works for a given graph type.

Other changes that could be made to the project include modification to the expansion word list. The expansion word lists used were taken directly from the generator output. It’s possible that a human evaluator could pick out words from that list that seem like they don’t belong. The word list could also be modified by combining top results from different tests. A more salient word list would, theoretically, produce better results.

Future work could also include applying some of the ranking methods to create a way to query the information graphics. Very little has been done for graph querying and the work completed on this project so far would serve as a helpful contribution or staring place.

Another way to augment the text for evaluation would be through the use of ontologies. Ontologies are groupings of concepts with relationships between words so if, for example, a bar label says “France,” related concepts within an ontology might include “country” or “Europe.” It provides a way to expand on a given word and provide flexibility for similar concepts. The current system limits comparisons to being an exact word match. There are plans to implement the use of ontologies on this project.

4. Acknowledgements

Firstly, I would like to thank Dr. Stephanie Elzer-Schwartz, my mentor and project leader. Without her, I would have never pursued such an endeavor. I would also like to thank our fellow researchers at University of Delaware and Dr. Sandra Carberry for her guidance there. I have another big thank you to Todd Echterling and Chris Medalis for putting up with me over the summers. Lastly, I’d like to thank the National Science Foundation for funding our work through grant #1016900.

References:

[1] J. Michel, Y. Shen, A. Aiden, A. Veres, M. Gray, W. Brockman, The Google Books Team, J. Pickett, D. Hoiberg, D. Clancy, P. Norvig, J. Orwant, S. Pinker, M. Nowak, & E. Aiden, Quantitative Analysis of Culture Using Millions of Digitized Books, *Science, 331*, 2010, 176-182.

[2] Y. Yang, Y. Zhang, Z. Hou & B. Lemaire-Semail, Adding Haptic Feedback to Touch Screens at the Right Time, *ICMI '11 Proceedings of the 13th international conference on multimodal interfaces*, Alicante, Spain, 2011, 73-80.

[3] T. Limna, C. Sae-tang, C. Jantaraprim, P. Tandayya & W. Niyompol, Linux User Interface and Front-End Operation for the Visually Impaired, *i-CREATe '07 Proceedings of the 1st international convenction on Rehabilitation engineering & assistive technology: in conjunction with 1st Tan Tock Seng Hospital Neurorehabilitation Meeting*, Singapore, Singapore, 2007, 179-84.

[4] T. McEwan & B. Weerts, ALT Text and Basic Accessibility, *Proceedings of the 21st BCS HCI Group*, Lancaster University, UK, 2007, 3-7.

[5] C. Rowland, J. Smith, J. Whiting, T. Galloway, D. Knight, D. Hernandez & P. McGuire, Screen Reader User Survey #4 Results, *webaim.org*, May 2012.

[6] Peng Wu, Recognizing the Intended Message of Line Graphs: Methodology and Applications, PhD Thesis, University of Delaware, 2011.

[7]S. Carberry, S. Elzer Schwartz, K. Mccoy, S. Demir, P. Wu, C. Greenbacker, D. Chester, E. Schwartz, D. Oliver & P. Moraes, Access to Multimodal Articles for Individuals with Sight Impairments, *ACM Transactions on Interactive Intelligent Systems (TiiS) – Special issue on highlights of the decade in interactive intelligent systems*, 2, 2012, 21:1-21:49.

[8] P. Holcomb & J. Grainger, On the Time Course of Visual Word Recognition: An Event-related Potential Investigation using Masked Repetition Priming, *Journal of Cognitive Neuroscience*, 18, 2006, 1631-43.

[9] S. Thorpe, D. Fize & C. Marlat, Speed of Processing in the Human Visual System, *Nature, 381*, 1996, 520-22.

[10] L. Ferres, P. Verkhogliad, G. Lindgaard, L. Boucher, A. Chretien & M. Lachance, Improving accessibility to statistical graphs: the iGraph-Lite system, *Assets '07 Proceedings of the 9th International ACM SIGACCESS Conf. on Computers and Accessibility*, Tempe, AZ, 2007, 67-74.

[11] S. Elzer, E. Schwartz, S. Carberry, D. Chester, S. Demir & and Peng Wu, A Browser Extension for Providing Visually Impaired Users Access to the Content of Bar Charts on the Web, *WEBIST'2007 Proceedings*, Barcelona, Spain, 2007.

[12] G. Cao, J. Nie, J. Gao & S. Robertson, Selecting Good Expansion Terms for Pseudo-Relevance Feedback, *SIGIR '08 Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, Singapore, Singapore, 2008, 243-50.

[13] C. Burges, Ranking as Learning Structured Outputs (2005), *Proceedings of the NIPS 2005 workshop on Learning to Rank*, Vancouver, Canada, 2005.

[14] C.J.C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, & G. Hullender, Learning to Rank using Gradient Descent, *Bonn 22nd International Conference on Machine Learning*, Bonn, Germany, 2005.

[15] S.H. Cha, Comprehensive Survey on Distance/Similarity Measures between Probability Density Functions, *International Journal of Mathematical Models and Methods in Applied Sciences, 1*, 2007, 300-7.

[16] S. Cronen-Townsend, Y. Zhou & W.B. Croft, Predicting Query Performance, *SIGIR'02*, Tampere, Finland, 2002, 299-306.

[17] L. Ferres, P. Verkhogliad and L. Boucher, (Natural Language) Interaction with Graphical Representations of Statistical Data, *W4A '07 Proceedings of the 2007 international cross-disciplinary conference on Web accessibility*, Banff, Canada, 2007, 132-33.

[18] D. McGookin, E. Robertson, S. Brewster, Clutching at Straws: Using Tangible Interaction to Provide Non-Visual Access to Graphs, *CHI 2010 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Atlanta, Georgia, 2010, 1715-24.

[19][S. Elzer, E. Schwartz, S. Carberry, D. Chester, S. Demir & Peng Wu, A Browser Extension for Providing Visually Impaired Users Access to the Content of Bar Charts on the Web, *Proceedings of Third International Conference on Web Information Systems and Technology (WebIST)*, Barcelona, Spain, 2007, 59-6.

[20] G. Tsatsaronis & V. Panagiotopoulou, A Generalized Vector Space Model for Text Retrieval Based on Semantic Relatedness, *EACL '09 Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, Athens, Greece, 2009, 70-8.

[21] K. Shum, Shannon's Entropy, *planetmath.org*, 2013.

[22] C. M. Bishop, *Pattern Recognition and Machine Learning* (New York, NY: Springer, 2006).