A Novel Methodology for Retrieving Infographics Utilizing Structure and Message Content

Zhuo Li
ivanka@udel.edu

Sandra Carberry
carberry@udel.edu

Hui Fang*
hui@udel.edu

Kathleen F. McCoy
mccoy@udel.edu

Kelly Peterson
keldryc@udel.edu

Matthew Stagitis
mattstag@UDel.Edu

Department Computer and Information Science,
*Department of Electrical and Computer Engineering
University of Delaware

January 9, 2015

Abstract

Information graphics (infographics) in popular media are highly structured knowledge representations that are generally designed to convey an intended message. This paper presents a novel methodology for retrieving infographics from a digital library that takes into account a graphic's structural and message content. The retrieval methodology can be summarized thus: 1) hypothesize requisite structural and message content from a natural language query, 2) measure the relevance of each candidate infographic to the requisite structural and message content hypothesized from the user query, and 3) integrate these relevance measurements via a linear combination model in order to produce a ranked list of infographics in response to the user query. The methodology has been implemented and evaluated, and it significantly outperforms a baseline method that treats queries and graphics as bags of words.

1 Introduction

Information graphics (infographics) commonly appearing in popular media, such as bar charts and line graphs, are effective visual representations of a relationship between data entities. Designers of such graphics generally construct them using well-known communicative signals (e.g., coloring a bar differently to highlight it) to convey a high-level *intended message*. For example, the bar chart in Figure 1 ostensibly conveys the message that Toyota

Figure 1: Infographic Example

has the highest profit compared to the other car manufacturers listed. Although it is possible to describe the content of an infographic by paragraphs of written explanations, it is easier for a reader to absorb the information quickly from a graphic [35], making infographics an important and unique knowledge source that should be accessible and retrievable based on their content. As a take-off from the proverb "a picture is worth a thousand words", we can similarly say that "a graphic is worth a thousand words" since it contains a multitude of information.

Yet compared to the retrieval of text documents [36, 58] and pictorial images [1,57], scant attention has been given to the retrieval of infographics. Conventional search engines rely on the document text that contains the infographic, including the infographic's file name, the image tag from the webpage html source file, and words in the accompanying article appearing near the infographic in the source file. These approaches ignore the content of the infographic itself.

Consider the query, *"How does the net profit of major car manufacturing companies compare?"* This query is requesting infographics that convey a comparison of car manufacturing companies according to their net profit, as suggested in part by the use of the verb *"compare"* and the plural form of *"companies"* in the query. When this query was entered into Google Image Search on December 10th, 2014, no satisfactory graphics appeared among the top 10 infographics returned. The infographic deemed most relevant was the bar chart shown in Figure 2, which displays a car model (Volkswagon Golf) and its sales trend, not a comparison of car companies; the terms *"profit"*, *"car"*, and *"company"* appeared near the infographic in the accompanying text article and may account for its retrieval. On the other hand, a much more relevant graph is the one shown in Figure 1; it presents a comparison of twelve car manufacturing companies by representing each

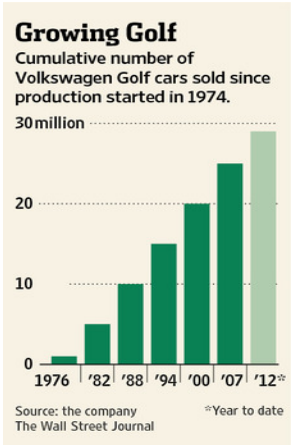

Figure 2: Top Retrieved Infographic by Google Image

company by a bar on the independent axis and ordering the bars according to the net profit of the companies that they represent. Although the retrieval process of Google Image Search is unclear, it is fair to conclude that little or no consideration is given to the content of infographics themselves.

Image search engines specialized for infographics, such as Zanran numeric data search (http://www.zanran.com) and SpringerImages (http://www.springerimages.com), also search by means of text around and near a graphic's image. In response to the same query as above, neither SpringerImages nor Zanran returned any relevant infographics

In contrast, this paper presents a novel methodology for infographics retrieval, determining how well information needs gleaned from a user query have been fulfilled by taking into account a candidate infographics's structure and message content. This paper is an extended version of the initial work presented in [38]. The methodology has been implemented and tested on a corpus of infographics and user queries, and has been shown to significantly outperform a baseline approach that treats queries and graphics as bags of words.

Section 2 identifies unique characteristics of infographics and describes how user queries suggest the kind of infographics that are desired. Section 3 outlines the main components of our retrieval methodology for infographics, followed by sections that present three major modules in our system: query processing (Section 4), graphic preprocessing (Section 5), and rank-ordering (Section 6). Section 7 presents an evaluation of our methodology. Section 8 discusses related work, including text document retrieval, content based image retrieval, semi-structured data retrieval, and query processing. Section 9 summarizes the contributions of our work and discusses future research.

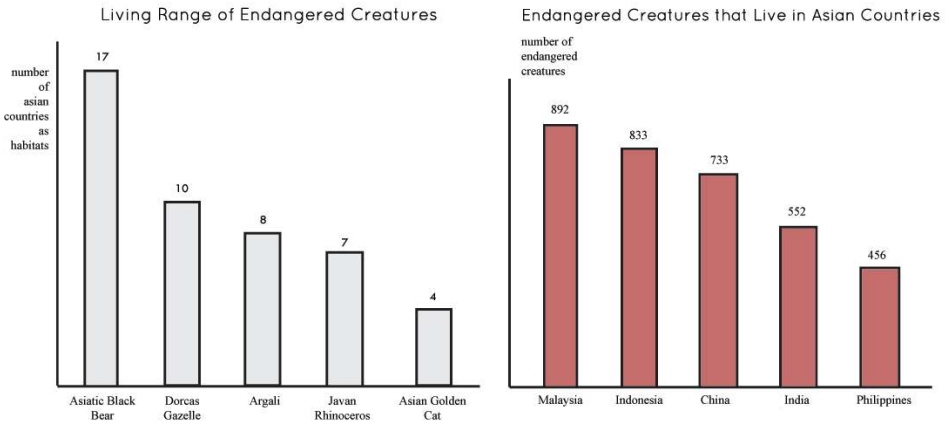

Figure 3: Two Infographics with Similar Keywords but Different Content

2 Motivation for a Novel Retrieval Methodology

2.1 Structural Content of Infographics

Infographics have more structure than pictorial images. Infographics visualize the relationship between at least two types of entities, one on the independent axis (or X-axis) (such as the bar entities in a bar chart) and another on the dependent axis (or Y-axis) (such as what is measured about the bar entities in a bar chart). For example, the graphic on the left in Figure 3 contains names of endangered species on its X-axis and measures their number of habitat countries on its Y-axis.

The structure of an infographic conveys much of its content. The content captured by a word in an infographic greatly depends on the positioning of that word within the graphic's frame. For example, the words "asian countries" appear on the Y-axis of the left bar chart in Figure 3, capturing its Y-axis content; these same two words, "asian countries", appear in the caption of the right bar chart shown in Figure 3, and in this case they capture the general concept of the graph's X-axis bar labels. Thus the role "asian countries" plays in the two bar charts is different. On the other hand, the words "endangered creatures" appear in the caption of both graphics in Figure 3. However in the bar chart on the left, "endangered creatures" describes the general concept of its X-axis entities, whereas in the bar chart on the right, "endangered creatures" appears both in the caption and above the Y-axis and is the entity being measured on the Y-axis. Based on these observations, we conclude that ignoring the structure of infographics by treating all the words in a graphic as a single bag of words is not sufficient when describing the graphic's content.

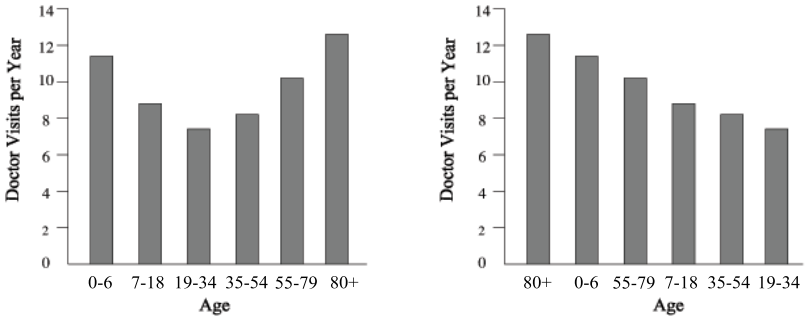

Figure 4: Graphics Displaying the Same Data with Different Messages

2.2 Message Content of Infographics

In addition to having structure, infographics often convey a high-level message [10]. To convey this message, a graphic designer makes design choices that serve as communicative signals.

Composition and layout are one kind of communicative signal. Consider the bar charts in Figure 4. Both bar charts contain the same data (that is, a person's age on the X-axis and the number of doctor visits on the Y-axis), but their layout is different. The bar chart on the left ostensibly conveys the changing trend in doctor visits throughout a person's life, while the bar chart on the right side is designed differently to convey the ranking of age groups by the number of visits. This correlates with an observation by Larkin and Simon [35] that infographics may be informationally equivalent (that is, they contain the same information) but not computationally equivalent (that is, it may be more difficult to perceive the information from one graphic than from the other).

Salience is another kind of communicative signal. An entity in a bar chart might become salient because the bar representing it is much taller than the other bars, or is colored differently from other bars. For example, Figure 1 highlights the bar entity *Toyota* to emphasize the importance of it, thereby helping convey the message that Toyota ranks highest among the listed car manufacturing companies.

In these examples, the infographics convey a high-level message through the visual signals in the graphic. Thus infographics are a form of language since, according to Clark (Clark and Curran, 2007), language is any deliberate signal that is intended to convey a message.

Our research group identified and noted the importance of the high level message of infographics [10] and referred to it as the *intended message* of the graphic. For simple bar charts [21] and single line graphs [9], our group's previous work identified the major categories of intended messages:

- *Trend* messages: This category captures messages that convey a trend over some ordinal entity, such as *Rising-trend*, *Falling-trend*, *Changing-*

trend, etc. Note that while a graphic might convey a rising trend, a query would be much more likely merely to request the trend of some entity since the user would not know a priori whether the trend is rising, falling, or stable.

- *Comparison* messages: This class is comprised of five intended message categories that compare entities according to some criteria. Members of this class are:
  - *Min* and *Max*: convey the entity that has the smallest or largest value with respect to other entities.
  - *Rank-all:* convey the relative rank of a set of entities
  - *Rank:* convey the rank of a specific entity with respect to other entities.
  - *Relative-Difference:* convey the relative difference between two entities
- General: convey no specific message and just display data

The entity (or entities) that is specifically focused on by the intended message is referred to as the *focused entity* (or focused entities), denoted as G_{fx}. For example, Figure 1 conveys a *Max* intended message, which conveys that the highlighted entity *Toyota* has the largest net profit among the listed car manufacturing companies. Our representation of the intended message indicates whether there are focused entities and what the focused entities are. Entities on the X-axis which are not specifically focused on by the intended message are referred to as G_{nx}, such as the car manufacturing companies that are not highlighted in Figure 1.

2.3 Queries and their Information Needs

Most text retrieval systems work with keyword queries, since the probability or frequency of query keywords in an unstructured text document could indicate the overall relevance of the document to the query. When retrieving semi-structured or structured text documents, systems usually require the input to be either in a particular query language format [30,51] (such as SQL or XQL), or full sentence natural language queries [13, 16]. This is because the information need of users for semi-structured or structured data is more specific than that for unstructured data.

Similarly, keywords cannot fully express a user's information need for the content of infographics. For example, a keyword query such as *"endangered animal Asian countries"* can only indicate that the requested infographics are about endangered animals and asian countries. However, within this broad domain, there are numerous infographics with distinct content; this

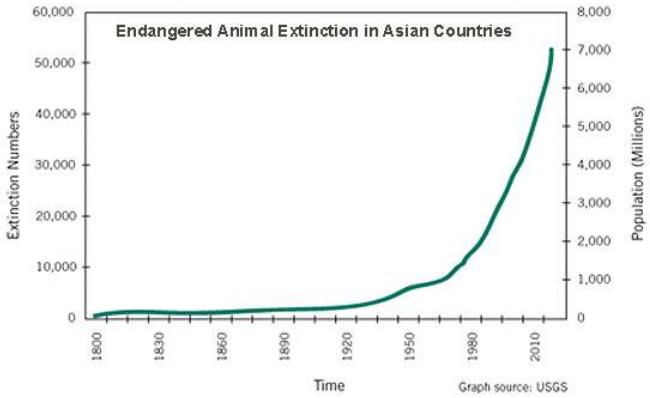

Figure 5: An Infographic Showing the Dramatic Increasing Trend of Animal Extinction in Asian Countries

is especially true when the domain is about popular topics or events. For example, both infographics shown earlier in Figure 3 are in this domain, as is the infographic shown in Figure 5 demonstrating a sharp increase in the trend of animal extinction. Moreover, the keyword query might also retrieve graphics such as a line graph that demonstrates a decreasing trend in government spending to protect endangered animals in Asia and a bar chart that compares the number of endangered animals in Asia with that of other continents. The content of these infographics vary greatly even though they are all in the domain of the keyword query. Thus infographic retrieval cannot be done effectively on the basis of keyword queries.

In contrast, a full sentence query provides important semantic clues about the structure and message content of the desired infographic. Consider the following two full sentence natural language queries:

Q_1: *"Which Asian countries have the most endangered animals?"*
Q_2: *Which endangered animals are found in the most Asian countries?*

The structure of query Q_1 indicates that it is asking for a comparison of *"Asian countries"* (X-axis) in terms of their population of *"endangered animal"* (Y-axis) whereas the structure of query Q_2 indicates that it is seeking a comparison of *"endangered animals"* (X-axis) according to the number of *"Asian countries"* they dwell in (Y-axis). These two queries contain almost identical words but are asking for very different graphics; the two queries are asking for graphics where the two mentioned entities, *"Asian countries"* and *"endangered animals"*, are completely flipped around, just by organizing the query in a different way. Moreover, the use of the superlative *"the most"*, along with the plural *"countries"* in query Q_1 and *"animals"* in query Q_2, suggests a *Rank All* message conveying a ranking of Asian countries (query

Q_1) or endangered animals (query Q_2), as opposed to a different kind of message such as a Trend message.

Thus we have developed an infographic retrieval system where the input is full sentence interrogative queries whose semantics are analyzed to identify the desired characteristics of relevant graphs: what is expected on the X-axis and Y-axis, the type of preferred intended message, and whether there should be a focused entity emphasized by the intended message. Our retrieval system then measures how infographics satisfy these four aspects of query information need and ranks them for retrieval.

3 A Novel Retrieval Methodology: Background and Overview

This paper presents a new methodology for retrieving relevant infographics from a digital library. We assume that user queries are full sentences, are grammatically correct, and do not contain spelling errors. Methods exist or could be devised for correcting misspellings and grammatical errors and that is not the focus of this research. We further assume a digital library that contains a collection of infographics along with their XML representations; the XML representation of an infographic gives its content, including the content of the X-axis G_x, the content of the Y-axis G_y, other descriptive text in the graphic, the category of intended message G_{IM}, and the message focused entities G_{fx} (if any) and non-focused entities G_{nx}. Note that G_x is comprised of G_{fx} and G_{nx}.

We assume that all of the above components of each infographic G have been correctly extracted and stored in the graphic's XML representation in the digital library. A number of research efforts have focused on parsing a graphic into its constituent pieces. Futrelle et al. devised a graphic parser for vector-based PDF documents [54]. Information extraction from graphics in raster form, such as bitmap graphics, is a harder problem than extraction from vector-based graphics. For raster graphics, both graphical component extraction and text recognition are needed: low-level image features are used for image segmentation and extracting the underlying data; optical character recognition (OCR) techniques are employed to extract the textual information in the infographics. Yokokura et al. used a network structure to represent the layout information of bar chart images based on vertical and horizontal projections [61]. Zhou and Tan applied Hough transform to extract bars from bar charts; their system was able to deal with bar charts lying in any orientation and even hand-drawn bar charts [63]. Savva et al. developed a system for redesigning poorly designed infographics; as the first step, their system extracts the graphical marks and infers the underlying data [53]. Other research provides methodologies to improve upon traditional OCR technologies and transform raw raster infographics into

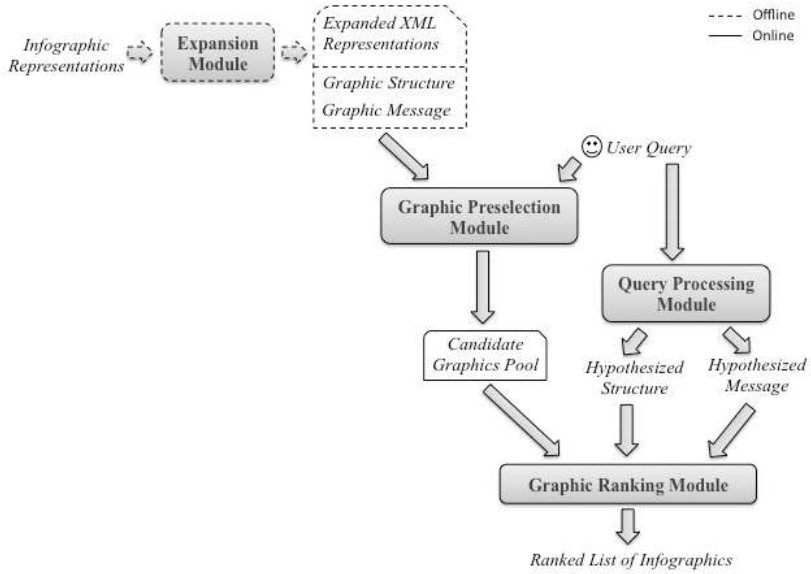

Figure 6: Overall Infographic Retrieval System Flowchart

semi-structured textual representations [12, 29].

Although the axes of the infographics are essential for retrieval, many infographics often do not explicitly label the Y-axis with what is being measured in the infographic. Previous work on this project developed a methodology for hypothesizing what is being measured by the dependent axis of a bar chart [19], by using heuristics to extract and meld together words from different pieces of the graphic. We assume a similar methodology (or a program such as this) has extracted the Y-axis content G_y. Prior work on our project [21, 59] developed systems that extract communicative signals from the XML representation of a bar chart or line graph and use them in a Bayesian system that identifies the infographic's intended message and focused entity (if any). Again, we assume that the intended message has been extracted and stored in the XML representation of infographics in our digital library.

Figure 6 gives an overview of our system. When an infographic is stored in the digital library, the words in the infographic are expanded as discussed in Section 5, resulting in an expanded XML representation of each graphic (Expansion Module). Given a query, a candidate set of infographics are first preselected by matching words from the expanded representation to the words in the user query (Graphic Preselection Module). Our retrieval methodology then analyzes the query to identify the requisite characteristics of infographics that will best satisfy the user's information need (Query

Processing Module). Then the candidate set of infographics are rank-ordered according to how well their structure and content satisfy this information need as hypothesized from the user's query (Graphic Ranking Module).

Section 4 discusses the processing of user queries, Section 5 discusses our expansion technique to tackle text sparsity issues, and Section 6 describes the ranking of infographics for retrieval in response to a user query.

4 Query Processing

As illustrated earlier, full sentence queries provide clues to the structural and message content of requisite infographics. Figure 7 outlines the *Query Pro-*

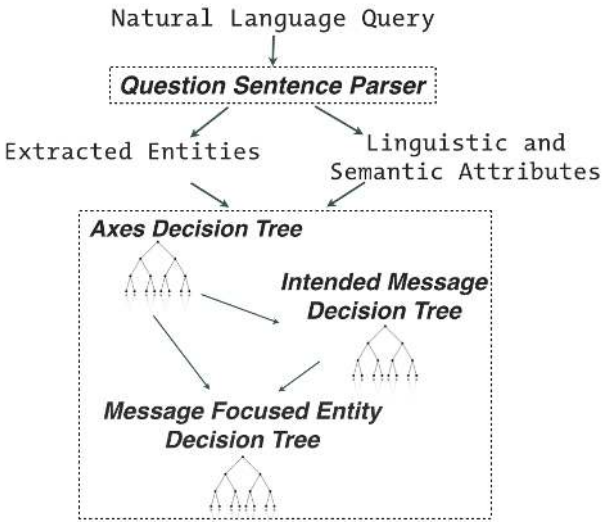

Figure 7: Query Processing Module

cessing Module. Our approach to query processing was described in [39,40]. This section expands on that work. Given a new query, the system first passes it to a CCG parser [14] to produce a parse tree. The parser we use was developed by Clark and Curran and trained specifically for questions whereas most parsers are trained entirely or mostly on declarative sentences. From the parse tree, the *Query Processing Module* populates a set of noun phrases as candidate entities E_1, E_2, ..., E_n, and extracts a set of linguistic attributes associated with each query-entity pair, Q-E_i. Each query-entity pair is input to a decision tree for determining whether the entity represents the content of the X-axis, or the content of the Y-axis, or none of the axes [40]. Then the *Query Processing Module* uses the content of the axes in a second decision tree to identify the category of intended message requested by the user's query [39]. A third decision tree utilizes the results of both the axes decision tree and the message decision tree to decide whether an entity

is the message focused entity.

In order to build and evaluate decision trees, we need a corpus of queries. To construct a corpus of full-sentence queries oriented toward retrieval of infographics, a human subject experiment was conducted by an undergraduate research assistant. Each subject was shown a set of infographics on a variety of topics such as adoption of children, oil prices, etc. For each displayed graphic, the subject was asked to construct a query that could be best answered by the displayed graphic. After dropping off-target queries, this resulted in a total of 192 queries. [1]

A second human subject experiment was conducted in addition to the first experiment. In the second experiment, each participant was given several sets of infographics; each set consisted of four graphics on similar topics but with different intended messages. For each graphic in a set, the subjects were asked to write a query where that graph would be more relevant than the other graphs in the set. The two experiments together produced a corpus of 324 queries.

4.1 Hypothesizing Axes Content from a User Query

4.1.1 Enumeration of Candidate Axes Entities

To hypothesize the content of the axes from a user query, we first need to generate a set of candidate axis entities that will be considered by the decision tree as possible content of the axes. First, phrases that describe a period of time are extracted by an automata and included in the candidate entities. Then the query is parsed using a CCG parser [14] and noun phrases that are not part of time intervals are extracted from the parse tree and added to the set of candidate entities; noun-noun phrases are treated as a single noun. The set of entity candidates is filtered to remove certain categories of simple noun phrases which are used to describe a graph rather than to refer to the content of the graph. These include nouns such as "trend" or "change" that are part of the trend category of words and "comparison" and "difference" which are part of the comparison category of words.

For example, Figure 8 shows an abbreviated version of the parse tree for the following query:

> Q_5: *How does the revenue of Discover compare to American Express in 2010?*

First, a specific time point phrase, "in 2010", is extracted. Then the noun phrase "the revenue" and "Discover" are extracted from the parse tree and added to the set of candidate entities. The noun phrase "American Express"

[1] The link to the experiment's online SQL database is http://www.eecis.udel.edu/∼stagitis/ViewAll.php

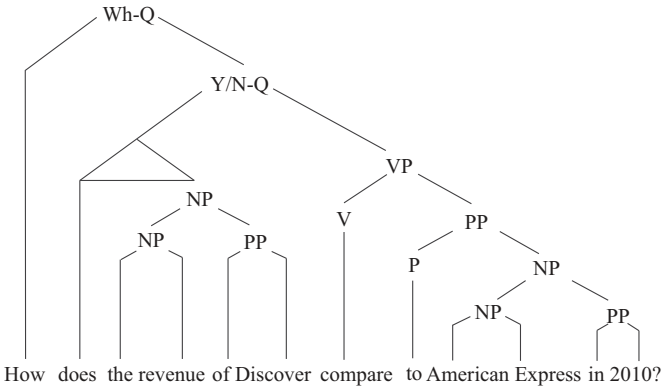

Figure 8: Abbreviated Parse tree for query Q_9

is added to the candidate list as a single noun phrase. The final set of query-entity pairs for query Q_5 are:

Q_5-E_1: *the revenue*
Q_5-E_2: *Discover*
Q_5-E_3: *American Express*
Q_5-E_4: *in 2010*

4.1.2 Cues from User Query for Axes Identification

We identified a set of attributes that might suggest whether a candidate entity reflects the content of the X or Y-axis of a relevant graphic. These clues can be divided into two classes:

1. *Global query attributes* that are features of the whole query and are independent of any specific entity.

2. *Specific entity attributes* that are particular to each specific candidate entity.

For example, the question type of the query sentence is a global query attribute. Consider the following two queries:

Q_6: Which country has the highest amount of exports?
Q_7: How many students enter college each year in the United States?

"Which" or "What" queries are often followed by a noun phrase that indicates the class of entities (such as *"country"* in Q_6) that should appear on the X-axis. On the other hand, "How many" and "How much" queries are often followed by a noun phrase that indicates what quantity (such as number of students in Q_7) should be measured on the Y-axis.

Comparatives and superlatives also provide clues. For example, the presence of a comparative or superlative, such as "highest" in query Q_6, often suggests that the Y-axis should capture the noun phrase modified by the comparative or superlative. Moreover, in query Q_6, the entity "exports" is modified by the noun phrase "the amount of", indicating that "exports" is a quantitative entity that could be captured on the Y-axis.

Certain categories of phrases provide strong evidence for what should be displayed on the X-axis. For example, consider the following queries:

> Q_8: *How does CBS differ from NBC in terms of viewers?*
> Q_9: *How does CBS compare with other networks in terms of viewers?*

The presence of a comparative verb such as "differ" or "compare" suggests that the entities preceding and following it capture the content of the X-axis. Furthermore, the plurality of the noun phrases is another clue. If both the noun phrases preceding and following the comparative verb are singular, as in query Q_8, then the noun phrases suggest entities that should appear on the X-axis; on the other hand, if one is plural (as in Q_9), then it suggests the class of entities to be displayed on the X-axis, of which the singular noun phrase is a member.

Similar to comparative word sets, certain words, such as "trend" or "change" in a query such as "How have oil prices changed from January to March?", suggest that the entity (noun phrase) that is the subject of the verb is likely to be on the Y-axis. On the other hand, time interval entities, such as "from January to March" or "in the past three years", are likely to capture the content of the X-axis. The set of attributes for identifying requisite structural content from a user query is enumerated in Appendix A.

For each query-entity pair, we determine the value for each of the attributes. This is accomplished by analyzing the parse tree and the part-of-speech tags of the elements of the parse tree, and by matching regular expressions for time intervals against the query-entity pair.

Each query-entity pair, along with its attributes, is processed by the axes decision tree in order to categorize the entities into one of three classes: whether this query entity describes the content of the X-axis, Y-axis, or none of the axes in the desired infographic. Consider query Q_5 as a working example.

- Query Q_5 is of the "How does" question type, causing the global query attribute indicating whether the query is of *How do* question type to be set to *True* for every query-entity pair derived from Q_5.
- The system finds that the query contains a word from the *comparison* word category; therefore the global query attribute for presence of a *Comparison* category word is set to *True* for all the query-entity pairs from Q_5.

- Regular expressions detect that Q_5 contains a phrase describing a specific time point, namely query-entity pair Q_5-E_4 "in 2010", so the attribute indicating whether a specific query-entity pair contains a time interval is set to *True* for Q_5-E_4, and *False* for the other query-entity pairs from Q_5.

- For query-entity pairs Q_5-E_2 and Q_5-E_3, the attribute designating that the entity is on the left and right side respectively of the comparison verb is set to *True*.

- Since E_1 ("the revenue") is the leftmost noun phrase following the question head "How does" in the parse tree (Figure 8), the attribute reflecting the leftmost noun phrase is set to *True* for Q_5-E_1 and to *False* for the other query-entity pairs.

- Since all entities are tagged as singular nouns by the part-of-speech tagger, the plurality attribute is set to *False* for each query-entity pair.

4.1.3 Learning and Evaluating Axes Hypotheses

A learned decision tree model is trained for hypothesizing whether an entity in a query captures content for the independent axis, for the dependent axis, or neither. To construct the training set, candidate entities were automatically extracted from each query, a set of query-entity pairs was constructed, and values for each of the attributes were extracted. The correct classification annotation of each query-entity pair was assigned by one researcher and then verified by another researcher with the final annotation of each query-entity pair indicating both researchers' consensus [2]. Human annotated classification is used as the ground truth. The evaluation measure is accuracy, measured as the proportion of instances in which the annotated correct classification matches the system's decision (*X-axis*, *Y-axis*, or *None*). Note that the overall process for hypothesizing requisite query axes is evaluated (not just the decision tree classification) since the test query is parsed and its entities, along with the values of the attributes, are automatically computed from the parse tree, its part-of-speech tags, and the use of regular expressions. WEKA [2], an open source machine learning toolkit, is used to construct the decision trees.

Since each query can produce more than one query-entity pair and since global attribute values (those based on the whole query) are identical for all query-entity pairs extracted from the same sentence, it would be unfair to have some of a sentence's query-entity pairs in the training set and some in the test set. Thus we use a variant of leave-one-out cross validation

[2] http://www.cs.waikato.ac.nz/ml/weka/

which will be referred to as *leave-one-query-out* cross-validation. In *leave-one-query-out* cross-validation, all of the query-entity pairs extracted from one query will be bundled together and used as the test set while all of the remaining query-entity pairs will be used to construct the classifier. A total of n repetitions of this custom cross-validation are performed, where n is the number of unique queries in the data set.

Prediction of the x and y axis has a baseline accuracy of 52.14%; that is, if the system always predicts the majority class of *Y-axis* for every query-entity pair, its success rate would be 52.14%. Our methodology does considerably better than the baseline, achieving an overall accuracy of 85.45%. For identifying the *X-axis*, the query processing module achieved precision of 85.16% and recall of 86.27%. For identifying the *Y-axis*, the module achieved precision of 85.79% and recall of 88.87%.

4.2 Hypothesizing Message Category and Focused Entity

We use the classification result from the axes decision tree as part of the attributes used to build a second decision tree that hypothesizes the preferred intended message category of graphics that might be relevant to the user query. For example, for a given query, if the axes decision tree identifies a time interval entity as on the X-axis, then the intended message of graphics relevant to this query is likely to fall into the *trend* category. Similarly, the plurality of X-axis entities in a query is another clue about the preferred intended message. Consider the following example queries:

> Q_{10}: *How does the revenue of Google compare with that of other technology companies?*
> Q_{11}: *How does the revenue of Google compare with that of Facebook?*

Knowing that *Google* and *technology companies* are components of the X-axis, and that one is singular (*Google*) while the other is plural (*technology companies*), suggests that query Q_{10} might be asking for a graphic whose intended message falls into the *Rank* category, namely the rank of *Google* among all technology companies. On the other hand, knowing that *Google* and *Facebook* are the entities on the X-axis and that both are singular suggests that query Q_{11} might be asking for a graphic whose intended message falls into the *Relative-difference* category, namely a comparison between only *Google* and *Facebook*. Although an infographic that shows the revenue of many technology companies, including Google and Facebook, could provide the information requested by query Q_{11}, the user can extract that information more easily from a graphic specifically devoted to Google and Facebook without other technology companies distracting the reader's attention. Thus attributes for building the second decision tree include whether the X-axis represents a time interval, the number of X-axis entities, and their plurality.

The class of the main verb in the user's query is also useful in hypothesizing the intended message of relevant graphs. For example, *comparison* main verbs, such as *differ* and *compare*, suggest that relevant graphics will have a *Relative-difference*, *Rank*, or *Rank-all* intended message; on the other hand, main verbs in the *change* class suggest a trend message. When a comparison verb is identified in a query, the number of extracted X-axis entities and their positions with respect to the comparison verb are all useful attributes for building the intended message decision tree. For example, when there are two or more X-axis entities extracted in a query, it increases the possibility that this query requests a *Rank*, *Rank-all*, or *Relative-Difference* intended message. Consider the following two queries from our corpus:

> Q_{12}: *"How do Ford, BMW, Toyota, and Honda compare in terms of revenue?"*
> Q_{13}: *"How does the revenue of Ford and Toyota compare to that of other car manufacturing companies?"*

Our system correctly identifies four X-axis entities in query Q_{12}, and three X-axis entities in query Q_{13}. The fact that all four X-axis entities in query Q_{12} are on the left side of the comparison verb "compare" indicates that this query requests an overall comparison of all four entities, thus requesting a graphic with a *Rank-all* message; in query Q_{13}, entities "Ford" and "Toyota" are on the left side of the comparison verb, while entity "car manufacturing companies" is on the right side of the comparison verb, suggesting that query Q_{13} is asking for a *Rank* message focused on the rank of "Ford" and "Toyota".

Superlatives, such as the word *highest*, suggest that relevant graphics will have a *Maximum* or *Minimum* intended message. Time intervals together with a *trend* main verb suggest that a *Trend* message is preferred. If an entity is modified by certain types of words, such as "all of", "each", and "every", this entity is likely to refer to the general concept of a group of entities that are compared against each other, thereby suggesting that the query is requesting a graph with a comparison-based intended message.

Using these attributes (Appendix B) along with a subset of the attributes in Appendix A, a decision tree is learned for taking a user query as input and hypothesizing the category of intended message of potentially relevant graphs. Using leave-one-out cross validation, the model had a success rate of 89.51%, which is much higher than the baseline of 61.42% that is achieved by simply selecting the most prevalent category (namely *Trend*).

A third decision tree to determine whether an extracted entity from the user query is the message focused entity uses the classification output from the previous two decision trees (the axes decision tree and the intended message category decision tree), a subset of the attributes described in Section 4.1.2, and a few additional attributes. For example, if exactly two entities from a query are classified as X-axis entities, then identification of

the query as asking for a *Rank* intended message suggests that one of the X-axis entities is likely to be a focused entity. If one of these two X-axis entities is singular, it is likely to be the focused entity of the *Rank* message whereas a plural entity is likely to describe the category of the X-axis entities. On the other hand, identification of the query as asking for a *Relative-difference* intended message suggests that both of these X-axis entities are highly likely to be focused entities. If there are more than two entities classified as X-axis entities, identification of the query as asking for a *Relative-difference* intended message suggests that the focused entities are the entities located directly on either side of the comparison main verb.

Noun phrases modifying an X-axis entity also provide clues about whether that entity is focused. Phrases such as *"the other"* (in query Q_{10}) and *"the rest of"* are likely to modify a general concept describing the entire X-axis, not a focused entity; adjectives such as *"each"* and *"every"* behave similarly.

Using leave-one-out cross validation, the focused entity decision tree model achieved a success rate of 95.98%, with precision of 96.96% and recall of 96.60%. Since most entities in a query are not focused entities, the baseline success rate, 86.25%, is achieved by always predicting the entity to be a non-focused entity.

5 Expanded Representation of Infographics

The terms in a user query often do not match the terms in a relevant infographic. Several problems must be addressed:

1. the query uses a word such as *"income"* whereas a relevant infographic uses the term *"revenue"*. Thus either the term *"income"* from the query must be expanded to include the term *"revenue"* or the term *"revenue"* must be expanded to include the term *"income"*.

2. the query uses a term such as *"credit card company"* representing a class of entities, whereas relevant infographics contain bar labels for individual credit card companies but no mention of the class *"credit card companies"*. Either the general class mentioned in the query must be expanded to a list of all of its members or the set of labels in an infographic must be generalized to its ontological class.

3. the query uses a common term for a focused entity but a relevant infographic uses a synonym or abbreviation of the term used in the query. These must be disambiguated to reference the same entity.

Each of these issues is discussed in the next subsections.

5.1 Text Expansion

Short text expansion is a commonly used strategy in information retrieval that bridges the vocabulary gap between terms in a query and those in short documents. The basic idea is to expand the limited document text with terms that are semantically similar. This addresses the problem encountered when the query uses the word *car* but the document uses the word *automobile*. Similar to short documents, such as tweets, microblogs, and short speech documents, text extracted from infographics is limited in length and therefore yields little in the way of term frequency information. Document and query expansion helps in such contexts [43]. Recent research on document retrieval has shown that the relevance model (RLM) works well with short documents [20, 23], where a model of relevance for every word in the vocabulary is computed assuming each word would appear in pseudo-relevant documents as well as the given short document. Consistent and significant improvements in retrieval performance have also been shown using Wikipedia-based document expansion [3].

We expand the text in each infographic using Wikimantic [7, 8], a term expansion method that uses Wikipedia articles as topic concepts. Given a sequence s of the text in a graphic, Wikimantic extracts all Wikipedia articles and disambiguation articles whose titles contain a subsequence of s; each of these articles is viewed as a Wikimantic *Atomic Concept* and is weighted by the likelihood that the concept generates sequence s. The weights of these extracted atomic concepts are then normalized so that their weights sum up to 1. Wikimantic then builds a unigram distribution for words from the articles representing these weighted Atomic Concepts.

To expand the text in an infographic using Wikimantic, our graph expansion module treats all of the text in the graphic as a string $G_t = \{t_i\}$ of terms and gets a weighted vector of Wikimantic concepts $M_G = \{\omega_j \cdot A_j\}$, where M_G is referred to as a *Mixture Concept* and w_j captures the likelihood of atomic concept A_j being the correct interpretation for a subsequence of G_t. Wikimantic collects all of the words in the constituent weighted atomic concepts to build a combined unigram distribution U_G for M_G. The combined frequency tf_i of each term t_i in U_G is a weighted sum of frequencies tf_{ij} of that term in the unigram distribution of each constituent concept $A_j \in M_G$: $tf_i = \sum_j \omega_j \cdot tf_{ij}$.

The words in the unigram distribution are ranked by their term-frequency-inverse-document-frequency value (tf-idf). For each word w_i in the unigram distribution, $tf\text{-}idf_i = tf_i \cdot log(\frac{|W|+1}{df_i+1})$, where tf_i is the term frequency of word w_i in the unigram distribution, $|W|$ is the estimated total number of Wikimantic concepts and df_i is the estimated number of Wikimantic concepts that contain word w_i. The words with the highest $tf\text{-}idf$ values are included in the expansion of the entire graph text.

Our methodology also expands the Y-axis of infographics within the

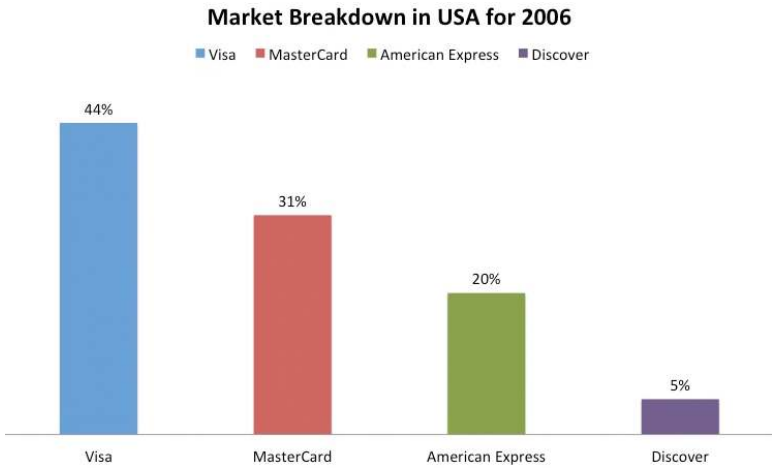

Figure 9: An Infographic without the Ontological Category of X-axis Entities

context of the entire infographic. From the interpreted concept M_G of the entire text in an infographic G_t, constituent atomic concepts A_y that are interpretations of $G_y \in G_t$ are extracted as concepts for the mixture concept M_y for the Y-axis. The top 30 terms in the unigram distribution of M_y with the highest $tf\text{-}idf$ value are used as the expansion of the graphic's Y-axis. For an infographic whose Y-axis measures *"revenue"*, this approach expands the Y-axis to include words such as *"profit"* and *"interest"*.

However, infographics present an additional problem. A query from our human subject experiment asks: *"Which credit card company gained the most market share in 2010?"*. The infographic shown in Figure 9 satisfies this query's information need. A set of credit card company names are listed on the X-axis (Visa, Mastercard, American Express, Discover) but nowhere in Figure 9 does the term "credit card company" or a synonym appear. Identifying the ontological category, such as *credit card company*, of these labels is crucial for infographics retrieval since the user query often generalizes the entities on the independent axis rather than listing them.

Our methodology expands the textual components of the infographics (rather than expanding the words in the user query) for two reasons: first, infographic expansion can be done off-line; more importantly, it is more feasible to expand specific entity words to include hypernyms than to expand general words to all possible specific entities that comprise the category. For example, it is more feasible to recognize that *Visa*, *Mastercard*, and *American Express* are all *credit card companies* than it is to expand the term *credit card company* from the query to all possible entities that are credit card companies, hoping to be able to include *Visa*, *Mastercard*, *American Express*, and all other credit card company names that might appear in

other relevant infographics.

The goal of expanding graphic X-axis labels (G_x) is to be able to include words that describe a more general concept that captures the specific instances listed on the graphic's X-axis, such as the general concept of *credit card companies* for specific instances *Visa*, *Mastercard* , *American Express*, etc.. Given the context concept M_G of the entire text in the graphic, we extract the Wikimantic concepts M_{Xe} ($e = 1, 2, ...$) for each corresponding X-axis entity X_e, construct its unigram distribution U_{Xe}, and consider the top 30 words with the highest tf-idf values in the unigram distribution U_{Xe}. For example, the top 30 words of the X-axis entity "Visa" include:

> ***card***, ***debit***, ***bank***, *bankamericard*, ***credit***, *bofa*, *ipo*, ***payment***, *electronic*, *issue*, *san*, *mateo*, *inc*, *merchant*, *business*, *hologram*, ***corporation***, *paywave*, *country*, *francisco*, *brand*, *secure*, *america*, *transaction*, *barclaycard*, ***financial***, ***company***

Using the same approach, the top 30 words of the X-axis entity "Mastercard" include:

> ***card***, ***bank***, *paypass*, ***debit***, *priceless*, ***payment***, ***credit***, *poland*, ***company***, *hsbc*, *datacell*, *purchase*, *intern*, ***financial***, *worldwide*, ***corporation***, *european*, *banknet*, *global*, *york*, *interbank*, *chief*, *network*, *unit*, *headquarter*

Then the X-axis entities, $X_1, X_2, ... \in G_x$, vote to identify the most commonly expanded words. The words that receive a majority vote are included in the expansion of the X-axis. The words "credit", "card", and "company" appear in the expansion of both the Visa entity and the Mastercard entity; therefore they receive votes from both entities as words to be included in the expansion of the overall X-axis. Other general terms that receive votes from both entities include "bank", "debit", "payment", "financial", and "corporation". Intuitively, the general concept words, such as "company", would appear in most (if not all) of the Wikipedia entries for specific instances of company, such as Visa and Mastercard. However, words that are relevant to only one entity will not be expanded as general terms since they receive fewer votes.

5.2 Interpretation of Message Focused Entity

Unlike the expansion of graphic X-axis content, there is no need for generalization when expanding an infographic's message focused entity G_{fx}. However, users are likely to choose synonyms of G_{fx} when forming their queries. For example, suppose an infographic lists several technology companies on its X-axis and the focused entity is labeled with "FB"; then it is important to disambiguate "FB" to match queries that request a focused entity "facebook".

Our approach is to extract the most heavily weighted (i.e., the most likely) atomic concept among the weighted vector of concepts produced by Wikimantic for G_{fx} as its disambiguation concept; the query focused entity will also be disambiguated in the same fashion. For example, the most heavily weighted atomic concept for the bar entity "FB" in the context of a graphic listing technology companies on its X-axis is likely to be the atomic concept of the company *Facebook Inc.*. If a query contains "Facebook" as its focused entity, this entity will also likely be disambiguated into the same concept *Facebook Inc.*.

6 Rank Ordering Infographics for Retrieval

Each infographic G in the candidate pool P must be ranked in terms of its relevance $R(Q, G)$ to the query Q. Our methodology measures the relevance between the different components of the infographic and the corresponding elements of the query, as depicted in Figure 10, and then combines each component relevance measurement through a *mixture model* to estimate the overall relevance $R(Q, G)$. This approach is denoted as the *component approach*. The baseline approach we compare our methodology against simply treats the entire text in the graphic as one bag of words, and the query as another bag of words, without giving consideration to the structural and message content of the infographic.

1. Baseline approach (bag of words): measures the relevance $R_{baseline}(Q, G) = R(Q_t, G_t)$, where G_t is all the words in a graphic, and Q_t is all the words in a query.

2. Component approach: measures the relevance of various graphic components to the corresponding query components, as illustrated in Figure 10, and then combines the relevance measurements of these components in a mixture model.

In the component approach, each of the following relevance measurements estimates the relevance of a component of a candidate infographic to the corresponding component specified in the user's query:

- X Axis Relevance $R(Q_x, G_x)$: relevance of the graphic's X-axis content G_x to the requisite X-axis content Q_x extracted from the user's query.

- Y Axis Relevance $R(Q_y, G_y)$: relevance of the graphic's Y-axis content G_y to the requisite Y-axis content Q_y extracted from the user's query.

- Intended Message Category Relevance $R(Q_{IM}, G_{IM})$: relevance of the category of intended message G_{IM} of the infographic to the category of intended message Q_{IM} preferred by the query.

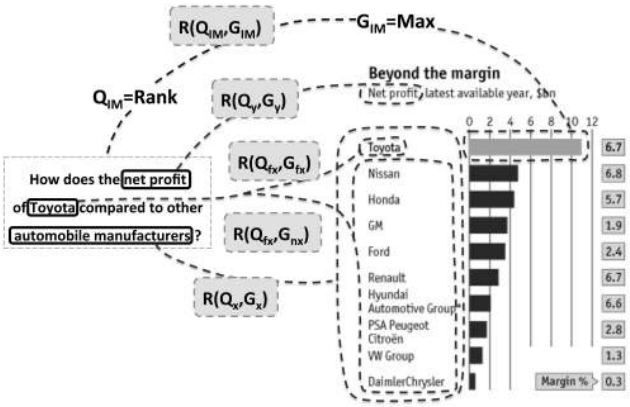

Figure 10: Relevance Measurements for Component Approaches

- Intended Message Focused Entity Relevance $R(Q_{fx}, G_{fx})$ and $R(Q_{fx}, G_{nx})$: relevance of the graphic's focused entity G_{fx} (if any) to the focused entity Q_{fx} (if any) extracted from the user's query. In cases where Q_{fx} appears on the X-axis of a graphic but is not focused, such graphics may address the user's information need, though less so than if the graphic also focused on Q_{fx}. Therefore we also measure the relevance of the non-focused X-axis entities $G_{nx} \in G_x$ to Q_{fx} as $R(Q_{fx}, G_{nx})$.

For comparison purposes, we consider three mixture models which respectively capture structural relevance, message relevance, and both structural and message relevance. Since the results of query processing are not always correct, we add to each model a back-off relevance measurement $R(Q_t, G_t)$ which measures the relevance of all the words in the query to all the words in a candidate infographic.

Model-1 (structural components): relevance of the structural components (the X-axis and the Y-axis) computed as:

$$R_1(Q, G) = \omega_0 \cdot R(Q_t, G_t) + \omega_1 \cdot R(Q_x, G_x) + \omega_2 \cdot R(Q_y, G_y) \tag{1}$$

Model-2 (message components): relevance of intended message components (message category and message focused entity, if any) computed as:

$$\begin{aligned} R_2(Q, G) = \omega_0 \cdot R(Q_t, G_t) + \omega_3 \cdot R(Q_{IM}, G_{IM}) \\ +\omega_4 \cdot R(Q_{fx}, G_{fx}) + \omega_5 \cdot R(Q_{fx}, G_{nx}) \end{aligned} \tag{2}$$

Model-3 (both structural and message components): relevance of both structural and intended message components, computed as:

$$\begin{aligned} R_3(Q, G) = \omega_0 \cdot R(Q_t, G_t) + \omega_1 \cdot R(Q_x, G_x) + \omega_2 \cdot R(Q_y, G_y) \\ +\omega_3 \cdot R(Q_{IM}, G_{IM}) + \omega_4 \cdot R(Q_{fx}, G_{fx}) + \omega_5 \cdot R(Q_{fx}, G_{nx}) \end{aligned} \tag{3}$$

The weighting parameters, ω_i, are learned using multi-start hill climbing to find a set of parameters that yields a local maximal retrieval evaluation

metric. Such hill-climbing search has been used successfully to learn parameters in other problems where the available dataset is small [44]. The next subsections discuss how relevance is measured for each of the terms in the above relevance equations.

6.1 Measuring Textual Relevance

The relevance between the words from the query and words from the graphic, such as $R(G_t, Q_t)$, $R(G_x, Q_x)$, $R(G_y, Q_y)$, and $R(G_{fx}, Q_{fx})$, are textual relevances, measured by relevance function R_{text}. We use a modified version of Okapi-BM25 [22] for measuring textual relevance R_{text}:

$$R_{text}(Q_c, G_{c'}) = \sum_{w_i \in Q_c} log(\frac{|D|+1}{gf_i+1}) \cdot \frac{tf_i \cdot (1+k_1)}{tf_i + k_1}$$

where Q_c is a query component and $G_{c'}$ is a graphic component, $|D|$ is the total size of our graphic collection, gf_i is the number of graphics that contain the word w_i, tf_i is the term frequency of w_i in $G_{c'}$, and k_1 is a parameter that is set to 1.2, a widely used value. This version of Okapi-BM25 has replaced the original inverse document frequency in Okapi with the regular inverse document frequency ($idf = log(\frac{|D|+1}{gf_i+1})$) to address the problem of negative idf. Our version of Okapi also does not take graphic text length into consideration, since text in graphics usually have similar limited lengths; moreover, a graph component, such as the message focused entity or the Y-axis, only consists of a noun entity and therefore normalizing the length of such a component does not have the same affect as for documents. Our version of Okapi also does not take query term frequency into consideration, since most terms in the query occur only once.

In the graph preprocessing stage, as discussed in Section 5.2, G_{fx} is disambiguated into a Wikimantic concept. Once the query processing module extracts a focused entity from a user query, the query focused entity Q_{fx} is also disambiguated into a Wikimantic concept. When measuring $R(Q_{fx}, G_{fx})$, the disambiguated concept name for Q_{fx} and the disambiguated concept name for G_{fx} are used to compute Okapi-BM25. Thus, if a query focused entity is disambiguated into the concept of "Facebook Inc.", a high Okapi-BM25 score is computed for an infographic whose focused entity is also disambiguated into the concept of "Facebook Inc.". The reason for using Okapi-BM25 instead of an exact matching of concept names is to be able to yield a non-zero $R(Q_{fx}, G_{fx})$ score when Q_{fx} and G_{fx} are disambiguated into similar but not exactly the same concepts. For example, if the Q_{fx} is disambiguated into the concept of "Facebook Messenger" (an instant messaging service and software application developed by Facebook Inc.), an exact matching of concept names would produce zero as the relevance score whereas Okapi-BM25 will not result in a zero relevance measurement because of the overlapping term "Facebook".

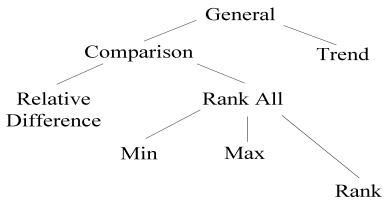

Figure 11: Intended Message Category Similarity

6.2 Measuring Relevance of the Intended Message Category

Intended message relevance measures the relevance of the category of the intended message, such as *Rank*, (along with the message focused entity of a query as described above) to those of an infographic.

We abstract a concept hierarchy containing the seven general intended message categories, as shown in Figure 11. Our methodology uses *relaxation* as the paradigm for ranking infographics according to how well an infographic's category of intended message G_{IM} satisfies the requisite category of intended message Q_{IM} hypothesized from the user query.

A six degree relevance measurement for $R(Q_{IM}, G_{IM})$ is computed based on this hierarchy. When G_{IM} matches Q_{IM}, little perceptual effort is required for the user to get the message information he or she wants; this infographic is deemed fully relevant to the query in terms of message category relevance. However, when G_{IM} differs from Q_{IM}, the amount of perceptual effort that the user must expend to satisfy his information need depends on G_{IM}. By moving up or down the intended message hierarchy from $Q_{IM} \rightarrow G_{IM}$, Q_{IM} is relaxed to match different G_{IM} with different degrees of penalties for the relaxation. The greater the amount of relaxation involved, the less relevant the message category of the infographic is to the query, and the more points penalized for message relevance.

At the top of the hierarchy is the *General* intended message category, which captures the least information message-wise. Message categories lower in the hierarchy contain more specific information. When Q_{IM} is lower in the hierarchy than G_{IM}, Q_{IM} requires more specific information than provided by G_{IM} . By relaxing $Q_{IM} \xrightarrow{up} G_{IM}$, perceptual effort is needed for the user to get the desired information; this infographic will be penalized for not having specific enough information. For example, consider two graphics, one whose intended message is the *Rank* of France with respect to other European countries in terms of cultural opportunities (and thus France is highlighted or salient in the graphic) and a second graphic whose intended message is just a ranking (category *Rank-all*) of all European countries in terms of cultural opportunities. If the user's query requests the rank of France with respect to other countries, then the first graphic matches the user's information need whereas the second graphic requires a relaxation of message category from (Q_{IM} = Rank) $\xrightarrow{up}$ (G_{IM} = Rank-all); in this latter case, user effort is required to search for France among the countries

listed and thus the second infographic is penalized for message relevance. If G_{IM} = General, the user must expend even more perceptual effort to extract the desired *Rank* message from this infographic. This is reflected in the hierarchy in that relaxing (Q_{IM} = Rank) $\xrightarrow{up}$ (G_{IM} = General) requires more upward movement than does relaxing (Q_{IM} = Rank) $\xrightarrow{up}$ (G_{IM} = Rank-all). However, relaxing the information need of (Q_{IM} = Min/Max) $\xrightarrow{up}$ Rank-all is penalized less than (Q_{IM} = Rank) $\xrightarrow{up}$ (G_{IM} = Rank-all). This is because it is comparatively easier to identify the minimum or maximum entity from a ranked bar chart (since it will appear first or last) than looking for a specific bar entity among all the entities.

On the other hand, if Q_{IM} is higher in the hierarchy than G_{IM}, then the graphic provides a more specific message than requested by the user's query; Q_{IM} requires less specific information than provided in G_{IM}. In this case, Q_{IM} is relaxed to a category lower in the hierarchy ($Q_{IM} \xrightarrow{down} G_{IM}$), and the infographic is penalized for containing extraneous information that might be distracting to the user. For example, if the query only requests a ranking of countries without focusing on a specific country (a *Rank-all* message), then an infographic that focuses on the rank of a specific country (G_{IM} = Rank) distracts the reader's attention to the highlighted country in the graphic. We contend that relaxing $Q_{IM} \xrightarrow{down} G_{IM}$ should be penalized less than $Q_{IM} \xrightarrow{up} G_{IM}$, since it requires less effort to ignore the distraction than to look for information that has not been presented explicitly. Therefore relaxing $Q_{IM} \xrightarrow{down} G_{IM}$ is penalized one point, while relaxing $Q_{IM} \xrightarrow{up} G_{IM}$ is penalized two points.

7 Experiments and Evaluation

7.1 Corpus Construction and Relevance Judgments

To evaluate the retrieval methodology, we used queries from the second experiment discussed in Section 4, in which each participant is shown five sets of made-up infographics; infographics in each set are constructed from data on the same topic and contain similar words, but they convey different intended messages; these five sets of artificial infographics are on five different domains. To get a collection of graphics, we used the 152 queries from the second experiment to search on popular commercial image search engines to get more infographics from the same domains. These commercial search engines include: Google Image, Microsoft Bing Image Search, and Picsearch. This produced in total 257 infographics that are in the domains of the collected queries.

Each query-infographic pair was assigned a relevance score on a scale of 0-3 by an undergraduate researcher. A query-infographic pair was assigned three points if the infographic was considered highly relevant to the query

and 0 points if it was irrelevant. Infographics that were somewhat relevant to the query were assigned 1 or 2 points, depending on the judged degree of relevance.

7.2 Experiment Results

Section 6 discussed three combinations of relevance measurements in the linear combination model, and a baseline ranking model. Textual relevance, as discussed in Section 6.1, could be measured with the original text in the graphics, or with Wikimantic expansion of the text as discussed in Section 5.1. This section presents experiments that compare the four ranking functions, with and without Wikimantic expansion of the words in the infographics. Thus there are a total of eight sets of experiments to evaluate the proposed methods.

The Bootstrapping method [45] has proven to be an effective evaluation method for small data sets. For each run of the bootstrapping method, the training query set is constructed by randomly selecting N queries with replacement from the entire query collection, where N is the number of queries in the collection. Queries that have not been selected for the training set comprise the testing set. Thus the training set contains approximately $(1-\frac{1}{e}) \approx 63.2\%$ of the overall query corpus, and the test set contains approximately $\frac{1}{e} \approx 36.8\%$ distinct queries. We average together the results of 10 runs with the Bootstrapping method.

Evaluation of retrieval systems usually follows the paradigm designed by Cleverdon and colleagues [15], in which users give judgements of the relevance of a set of documents given a set of queries. Standard evaluation metrics following this paradigm include precision, recall, and Mean Average Precision (MAP), which judge documents as either relevant or not relevant to a given query. Normalized Discounted Cumulative Gain (NDCG) [31] is an evaluation metric that improves upon the binary judgements so that documents are judged at multiple levels (for example, highly relevant, relevant, somewhat relevant, and not relevant). Within the top K ranked documents, Cumulative Gain (CG) is simply the sum of the graded relevance values regardless of the position of documents. Discounted CG (DCG) introduces a logarithmic penalty proportional to the position of documents. Normalized DCG normalizes a query's DCG by its ideal DCG, which is the DCG value when the retrieval result is ordered by relevance. We use Normalized Discounted Cumulative Gain (NDCG) [31] to evaluate the retrieval performance of our methods. It is between 0 and 1 and measures how well the rank-order of the graphs retrieved by our method agrees with the rank order of the graphs identified as relevant by our evaluator. We use a Student's t-test for computing significance.

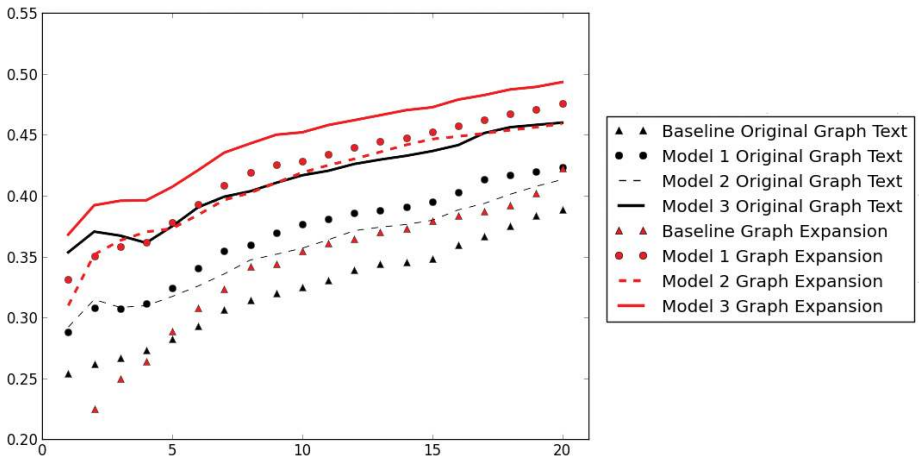

Figure 12: Linear Combination Mixture Model - NDCG@1-20 Plot using Learned Decision Tree Models for Query Processing

Graphic Expansion	Baseline	Model-1: Structural	Model-2: Message	Model-3: Message and Structural
No Expansion	0.3245	0.3766	0.3568	0.4168
With Expansion	0.3544	0.4280	0.4191	0.4520

Table 1: NDCG@10 Results Using Learned Decision Tree Model for Query Processing

Table 1 presents NDCG@10 results, with the second column of Table 1 giving the results for the baseline and the next three columns giving the results for our three models (structural, message, and structural+message). Furthermore, the first row show results when textual relevance is computed using exact match of query words with graph words, whereas the second row gives results when query words are matched with words in the expansion of the graph text via Wikimantic. The experimental results show that utilizing structural relevance (Model-1) and utilizing message relevance (Model-2) each provide significantly better results than the baseline approach ($p \leq 0.0001$). Furthermore, the combination of structural and message relevance improves upon either alone ($p=0.00005$). The results also show that Wikimantic graph expansion improves the retrieval performance consistently throughout all of the approaches. Figure 12 plots the nDCG@1-20 values using each retrieval method.

A question that arises is how much infographic retrieval is impacted by errors in extracting requisite structural and message content from user queries, since the learned models are not perfect, Table 2 compares NDCG@10 results when the decision tree models are used to process queries against the results when correct hand-labelled query data is used. Overall,

Query Q_x, Q_y Q_{IM}, Q_{fx}	Graph Expansion	Model-1: Structural	Model-2: Message	Model-3: Structural & Message
Learned Model	No Expansion	0.3766	0.3568	0.4168
	With Expansion	0.4280	0.4191	0.4520
Hand Labeled	No Expansion	0.4348	0.3881	0.4576
	With Expansion	0.4782	0.4433	0.4866

Table 2: NDCG@10 Comparison - Using Hand Labeled Query Data v.s. Using Learned Decision Tree Model for Query Processing

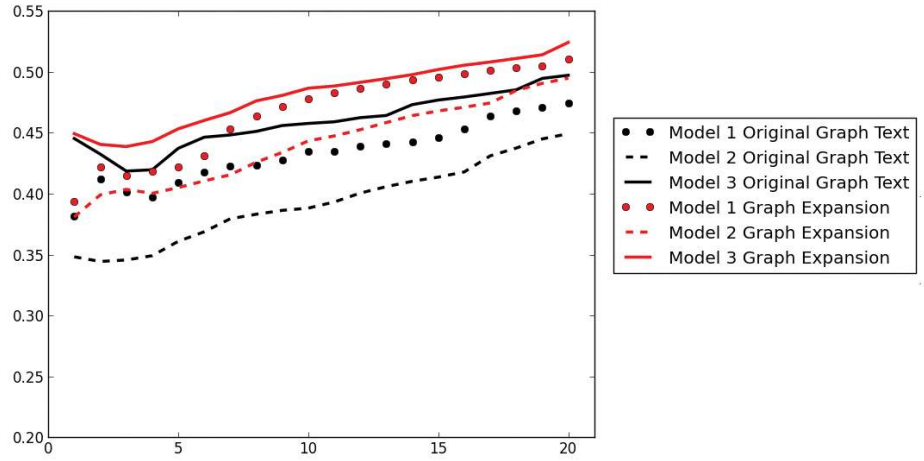

Figure 13: Linear Combination Mixture Model - NDCG@1-20 Plot using Hand Labeled Query Data

all of the six retrieval approaches show improvement when using hand labeled query data. Message category similarity is weighted much higher when the models are trained using hand labeled query data. Using hand-labeled query data for training, the average weight of message category similarity in the mixture model is 0.15 whereas it is only 0.07 when training uses the results produced by the decision tree for axis content, message category, and focused entity. On the other hand, the weight assigned to the smoothing relevance factor, $R(Q_t, G_t)$, is lower when training with hand labeled query data than when training with the results produced by the decision trees. These observations suggest that by improving the accuracy of the query processing module, it is possible to achieve better retrieval performance. Figure 13 plots the nDCG@1-20 values for each retrieval method when trained on the hand-labelled query data.

8 Related Work

8.1 Document Retrieval Research

Document retrieval research is concerned with retrieving and ranking relevant text documents given a user query. Three major types of document retrieval models have been developed: 1) vector space models from the earlier stages of information retrieval [52,55] that compare the distance between the vector representation of the query and the vector representation of each document in the term-document space, 2) classical probabilistic models [48–50] that assume there is a set R of relevant documents to the given query and estimate the probability of a document being in R by the occurrence of query words in the document, and 3) language models [47,62] that estimate the likelihood of a document generating the query.

The first step of retrieval in a vector space model is indexing, where both documents and queries are represented by term vectors; indexing models include the 2-Poisson probabilistic indexing model [26], latent semantic indexing model [18], and probabilistic latent semantic indexing model [28].

Classic probabilistic retrieval models rely on heuristic retrieval functions, such as BM25 [49] that approximate a 2-Poisson mixture model of document generation, combining term frequency and document length. Its later version, Okapi-BM25 [50], is recognized as one of the most effective and robust retrieval functions.

Retrieval models that apply language models rank documents in descending order of the probability of query words being generated from documents [33]. The query likelihood scoring method [6, 47] is the earliest attempt at applying language models in information retrieval. To better estimate the language model of each document, various smoothing strategies are applied, ranging from simple additive smoothing [11] and Good Turing smoothing [25] to more complex smoothing methods.

One can imagine treating the words in the graphics like a text document and apply these retrieval methods directly; however, regarding queries and documents as *a single bags of words* fails to take into consideration the most important characteristics of infographics: their structure and message content.

8.2 Infographic Processing and Image Retrieval

One branch of content-based image retrieval research consists of taking an image as an input query and extracting visual features and/or segmenting the provided query image, and searching in the space of visual features extracted from a collection of images for "similar" images [17,42,56]. Another direction of image retrieval research follows the keyword-based paradigm: an index is built on the textual annotations of images; such image retrieval systems take keywords or free-text queries as input (instead of an image) and

retrieve images using the constructed index. Other image retrieval systems allow users to provide both image query and text query and combine searching techniques on both visual features and text. Text-based approaches usually rely primarily on the surrounding text from the multimedia document [34] or user-provided metadata tags in social media such as Flickr and Youtube [24]. Li et al. used a learned statistical model to automate textual annotations with the most statistically significant terms for images in a database [37]. These image retrieval techniques are insufficient for infographic retrieval. Rarely would a user input an infographic and request similar ones. Furthermore, images do not usually have an intended message and they lack the structure of infographics.

8.3 Long Query Processing

Natural language queries are used in many retrieval systems to allow users to fully describe their complex information need. These information retrieval systems include question answering systems (QA systems), structured-data retrieval systems (such as XML data retrieval), and linked data retrieval (such as querying ontologies).

The degree of specificity of a search query corresponds roughly to the length of that query [5, 46]. Bendersky presented a probabilistic model for selecting the key concepts that will have the most impact on retrieval effectiveness of text documents from verbose (long) queries [4]. Linguistic characteristics, such as hierarchical structures and semantic relationships, have been utilized in other research to augment verbose query understanding, especially natural language queries [41].

Research on retrieval of structured data, such as linked data and ontologies, also relies on the syntax and semantics of natural language queries [16, 32]. For example, the query *"Who owns the biggest department store in England?"* specifies that the requisite type of attribute is a *"person"* and the relationship of that person to department stores in England is *"the owner of"*. In such queries, attributes and relationships in the data are explicitly given in the ontology being searched, and the queries specify the desired attribute and/or relationship. Our research is similar to these efforts in that we also use full sentence queries. However, infographic search queries do not explicitly state the desired attribute and/or relationship. Therefore extracting structural and message content from an infographic query is more complex. Moreover, the objective of our research is retrieval of an infographic as opposed to searching linked data, where the retrieval unit is a single attribute or concept.

8.4 Semi-structured Data Retrieval

Linear combination model has been an effective and flexible data fusion method for combining multiple information retrieval results [60]. Hiemstra proposed several language models to estimate the relevance of each XML element of semi-structured XML data to query unigrams, and proposed a linear combination mixture model and a translation mode to combine the relevance from each element to estimate the overall document relevance [27]. Our ranking methodology also adopts a linear combination mixture model to combine an arbitrary number of relevance components. A similar linear combination mixture model is used for a different problem in the work of Metzler et al., where parameters are greedily learned with a simple coordinate-level hill climbing search instead of expectation maximization given a small dataset [44]. Our methodology uses a similar greedy algorithm to set weights in the linear combination model so that a maxima in retrieval performance is achieved.

9 Conclusion and Future Work

This paper has presented a novel methodology for infographics retrieval based on the unique and fundamental characteristics of infographics: 1) their two-dimensional structure – displaying two groups of entities, one on the independent axis (X-axis) and another on the dependent axis (Y-axis); 2) the high-level intended message that their graphic designer intended to convey through specific communicative signals.

Our work is the first to analyze a user query and hypothesize the requisite structural and message content of infographics that might satisfy the user's information needs. The overall relevance of each candidate infographic to a user query is estimated by how well the infographic satisfies each of the different aspects of query information needs. A linear combination model is used to integrate structural (axes) relevance measurements and message relevance measurements and produce a rank ordering of the candidate infographics for retrieval.

Our retrieval methodology has been implemented and evaluated on a corpus of infographics and queries. Our experiments show that utilizing structural relevance and utilizing message relevance each significantly outperform a baseline method that treats queries and infographics as bags of words. Furthermore, utilizing a combination of both structural and message relevance produces significantly better performance than either alone. The results also show that our text expansion techniques improve retrieval performance consistently throughout all of the approaches.

In the future, we will further improve the accuracy of our query processing module. We will also explore a greater diversity of relevance measurements on top of Okapi-BM25. Instead of trying to integrate more relevance

measurements into a single linear function, we will investigate non-linear retrieval models such as learning-to-rank models. In addition, we will extend the retrieval methodology presented in this paper to other types of infographics, such as grouped bar charts and multiple line graphs.

10 Acknowledgements

This work was supported by the National Science Foundation under grant III-1016916 and IIS-1017026.

References

[1] Faruq A. Al-Omari and Mohammad A. Al-Jarrah. Query by image and video content: a colored-based stochastic model approach. *Data & Knowledge Engineering*, 52(3):313 – 332, 2005.

[2] Jeremy Ang, Rajdip Dhillon, Ashley Krupski, Elizabeth Shriberg, and Andreas Stolcke. Prosody-based automatic detection of annoyance and frustration in human-computer dialog. In *INTERSPEECH'02*, 2002.

[3] Jaime Arguello, Jonathan L Elsas, Jamie Callan, and Jaime G Carbonell. Document representation and query expansion models for blog recommendation. *ICWSM*, 2008(0):1, 2008.

[4] Michael Bendersky and W Bruce Croft. Discovering key concepts in verbose queries. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 491–498. ACM, 2008.

[5] Michael Bendersky and W Bruce Croft. Analysis of long queries in a large scale search log. In *Proceedings of the 2009 workshop on Web Search Click Data*, pages 8–14. ACM, 2009.

[6] Adam Berger and John Lafferty. Information retrieval as statistical translation. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 222–229. ACM, 1999.

[7] Christopher Boston, Sandra Carberry, and Hui Fang. Wikimantic: disambiguation for short queries. In *Natural Language Processing and Information Systems*, pages 140–151. Springer, 2012.

[8] Christopher Boston, Hui Fang, Sandra Carberry, Hao Wu, and Xitong Liu. Wikimantic: Toward effective disambiguation and expansion of queries. *Data & Knowledge Engineering*, 2013.

[9] Richard Burns, Sandra Carberry, Stephanie Elzer, and Daniel Chester. Automatically recognizing intended messages in grouped bar charts. In *Diagrammatic Representation and Inference*, pages 8–22. Springer, 2012.

[10] Sandra Carberry, Stephanie Elzer, and Seniz Demir. Information graphics: an untapped resource for digital libraries. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 581–588. ACM, 2006.

[11] Stanley F Chen and Joshua Goodman. An empirical study of smoothing techniques for language modeling. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, pages 310–318. Association for Computational Linguistics, 1996.

[12] Daniel Chester and Stephanie Elzer. Getting computers to see information graphics so users do not have to. In *Foundations of Intelligent Systems*, pages 660–668. Springer, 2005.

[13] Philipp Cimiano, Peter Haase, Jörg Heizmann, Matthias Mantel, and Rudi Studer. Towards portable natural language interfaces to knowledge bases–the case of the orakel system. *Data & Knowledge Engineering*, 65(2):325–354, 2008.

[14] S. Clark and J.R. Curran. Wide-coverage efficient statistical parsing with ccg and log-linear models. *Computational Linguistics*, 33(4):493–552, 2007.

[15] Cyril Cleverdon, Jack Mills, and Michael Keen. Aslib cranfield research project: factors determining the performance of indexing systems. Technical report, Cranfield University, 1966.

[16] Danica Damljanovic, Milan Agatonovic, and Hamish Cunningham. Freya: An interactive way of querying linked data using natural language. In *The Semantic Web: ESWC 2011 Workshops*, pages 125–138. Springer, 2012.

[17] Ritendra Datta, Dhiraj Joshi, Jia Li, and James Z Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys (CSUR)*, 40(2):5, 2008.

[18] Scott C. Deerwester, Susan T Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. Indexing by latent semantic analysis. *JASIS*, 41(6):391–407, 1990.

[19] Seniz Demir, Sandra Carberry, and Stephanie Elzer. Effectively realizing the inferred message of an information graphic. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP)*, pages 150–156, 2007.

[20] Miles Efron, Peter Organisciak, and Katrina Fenlon. Improving retrieval of short texts through document expansion. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 911–920. ACM, 2012.

[21] Stephanie Elzer, Sandra Carberry, and Ingrid Zukerman. The automated understanding of simple bar charts. *Artificial Intelligence*, 175(2):526–555, 2011.

[22] Hui Fang, Tao Tao, and ChengXiang Zhai. A formal study of information retrieval heuristics. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '04, pages 49–56, New York, NY, USA, 2004. ACM.

[23] Debasis Ganguly, Johannes Leveling, and Gareth JF Jones. An lda-smoothed relevance model for document expansion: a case study for spoken document retrieval. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 1057–1060. ACM, 2013.

[24] Y. Gao, M. Wang, H. Luan, J. Shen, S. Yan, and D. Tao. Tag-based social image search with visual-text joint hypergraph learning. In *Proceedings of the 19th ACM international conference on Multimedia*, pages 1517–1520. ACM, 2011.

[25] Irving J Good. The population frequencies of species and the estimation of population parameters. *Biometrika*, 40(3-4):237–264, 1953.

[26] Stephen P Harter. A probabilistic approach to automatic keyword indexing. part ii. an algorithm for probabilistic indexing. *Journal of the American Society for Information Science*, 26(5):280–289, 1975.

[27] Djoerd Hiemstra. Statistical language models for intelligent xml retrieval. In *Intelligent Search on XML Data*, pages 107–118. Springer, 2003.

[28] Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57. ACM, 1999.

[29] Weihua Huang and Chew Lim Tan. A system for understanding imaged infographics and its applications. In *Proceedings of the 2007 ACM symposium on Document engineering*, pages 9–18. ACM, 2007.

[30] Sayyed Kamyar Izadi, Mostafa S. Haghjoo, and Theo Hrder. S3: Processing tree-pattern XML queries with all logical operators. *Data & Knowledge Engineering*, 72(0):31 – 62, 2012.

[31] Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446, October 2002.

[32] Nicolas Kuchmann-Beauger and Marie-Aude Aufaure. A natural language interface for data warehouse question answering. In *Natural Language Processing and Information Systems*, pages 201–208. Springer, 2011.

[33] John Lafferty and Chengxiang Zhai. Probabilistic relevance models based on document and query generation. In *Language modeling for information retrieval*, pages 1–10. Springer, 2003.

[34] Mirella Lapata. Image and natural language processing for multimedia information retrieval. In *Advances in Information Retrieval*, volume 5993 of *Lecture Notes in Computer Science*, pages 12–12. Springer Berlin Heidelberg, 2010.

[35] Jill H Larkin and Herbert A Simon. Why a diagram is (sometimes) worth ten thousand words. *Cognitive science*, 11(1):65–100, 1987.

[36] Johannes Leveling. On the effect of stopword removal for sms-based faq retrieval. In Gosse Bouma, Ashwin Ittoo, Elisabeth Mtais, and Hans Wortmann, editors, *Natural Language Processing and Information Systems*, volume 7337 of *Lecture Notes in Computer Science*, pages 128–139. Springer Berlin Heidelberg, 2012.

[37] Jia Li and James Ze Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(9):1075–1088, 2003.

[38] Zhuo Li, Sandra Carberry, Hui Fang, Kathleen F McCoy, and Kelly Peterson. Infographics retrieval: A new methodology. In *Natural Language Processing and Information Systems*, pages 101–113. Springer, 2014.

[39] Zhuo Li, Matthew Stagitis, Sandra Carberry, and Kathleen F. McCoy. Towards retrieving relevant information graphics. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '13, pages 789–792, New York, NY, USA, 2013. ACM.

[40] Zhuo Li, Matthew Stagitis, Kathleen F. McCoy, and Sandra Carberry. Towards finding relevant information graphics: Identifying the independent and dependent axis from user-written queries. In *FLAIRS Conference'13*, 2013.

[41] Jingjing Liu, Panupong Pasupat, Yining Wang, Scott Cyphers, and Jim Glass. Query understanding enhanced by hierarchical parsing structures. In *ASRU*, pages 72–77. IEEE, 2013.

[42] Ying Liu, Dengsheng Zhang, Guojun Lu, and Wei-Ying Ma. A survey of content-based image retrieval with high-level semantics. *Pattern Recognition*, 40(1):262–282, 2007.

[43] Donald Metzler and Congxing Cai. Usc/isi at trec 2011: Microblog track. In *TREC*, 2011.

[44] Donald Metzler and W Bruce Croft. A markov random field model for term dependencies. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 472–479. ACM, 2005.

[45] Tan Pang-Ning, Michael Steinbach, Vipin Kumar, et al. Introduction to data mining. *WP Co*, 2006.

[46] Nina Phan, Peter Bailey, and Ross Wilkinson. Understanding the relationship of information need specificity to search query length. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 709–710. ACM, 2007.

[47] Jay M Ponte and W Bruce Croft. A language modeling approach to information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–281. ACM, 1998.

[48] Stephen E Robertson and K Sparck Jones. Relevance weighting of search terms. *Journal of the American Society for Information science*, 27(3):129–146, 1976.

[49] Stephen E Robertson and Steve Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 232–241. Springer-Verlag New York, Inc., 1994.

[50] Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. Okapi at trec-3. *NIST SPECIAL PUBLICATION SP*, pages 109–109, 1995.

[51] Jonathan Robie. XML query language (XQL). *http://www. w3. org/TandS/QL/QL98/pp/xql. html*, 1998.

[52] Gerard Salton, Anita Wong, and Chung-Shu Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[53] Manolis Savva, Nicholas Kong, Arti Chhajta, Li Fei-Fei, Maneesh Agrawala, and Jeffrey Heer. Revision: automated classification, analysis and redesign of chart images. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 393–402. ACM, 2011.

[54] M. Shao and R. Futrelle. Recognition and classification of figures in pdf documents. *Graphics Recognition. Ten Years Review and Future Perspectives*, pages 231–242, 2006.

[55] Amit Singhal, Chris Buckley, and Mandar Mitra. Pivoted document length normalization. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 21–29. ACM, 1996.

[56] Arnold WM Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain. Content-based image retrieval at the end of the early years. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(12):1349–1380, 2000.

[57] V. P. Subramanyam Rallabandi and S. K. Sett. Image retrieval system using r-tree self-organizing map. *Data & Knowledge Engineering*, 61(3):524–539, June 2007.

[58] Edwin Thuma, Simon Rogers, and Iadh Ounis. Exploiting query logs and field-based models to address term mismatch in an hiv/aids faq retrieval system. In Elisabeth Mtais, Farid Meziane, Mohamad Saraee, Vijayan Sugumaran, and Sunil Vadera, editors, *Natural Language Processing and Information Systems*, volume 7934 of *Lecture Notes in Computer Science*, pages 77–89. Springer Berlin Heidelberg, 2013.

[59] Peng Wu, Sandra Carberry, Stephanie Elzer, and Daniel Chester. Recognizing the intended message of line graphs. In *Diagrammatic Representation and Inference*, pages 220–234. Springer, 2010.

[60] Shengli Wu. Linear combination of component results in information retrieval. *Data & Knowledge Engineering*, 71(1):114 – 126, 2012.

[61] Naoko Yokokura and Toyohide Watanabe. Layout-based approach for extracting constructive elements of bar-charts. In *Graphics Recognition Algorithms and Systems*, pages 163–174. Springer, 1998.

[62] Chengxiang Zhai and John Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 334–342. ACM, 2001.

[63] YanPing Zhou and Chew Lim Tan. Hough-based model for recognizing bar charts in document images. In *Photonics West 2001-Electronic Imaging*, pages 333–340. International Society for Optics and Photonics, 2000.