Proceedings of the Twenty-Sixth International Florida Artificial Intelligence Research Society Conference

Towards Finding Relevant Information Graphics: Identifying the Independent and Dependent Axis from User-Written Queries

Zhuo Li and **Matthew Stagitis** and **Kathleen McCoy** and **Sandra Carberry**
Department of Computer and Information Science
University of Delaware
Newark, DE 19716

Abstract

Information graphics (non-pictorial graphics such as bar charts and line graphs) contain a great deal of knowledge. Information retrieval research has focused on retrieving textual documents and on extracting images based on words appearing in the accompanying article or based on low-level features such as color or texture. Our goal is to build a system for retrieving information graphics that reasons about the content of the graphic itself in deciding its relevance to the user query. As a first step, we aim to identify, from a full sentence user query, what should be depicted on the independent and dependent axes of potentially relevant graphs. Natural language processing techniques are used to extract features from the query and machine learning is employed to build a model for hypothesizing the content of the axes. Results have shown that our models can achieve accuracy higher than 80% on a corpus of collected user queries.

Introduction

The amount of information available electronically has grown dramatically. Although information retrieval research has addressed the need to be able to identify and access information relevant to a user's needs, these research efforts have focused on the text of documents and to some extent on their pictorial images. Unfortunately, information graphics (non-pictorial graphics such as bar charts and line graphs) have been largely ignored. Such graphics are prevalent in popular media such as newspapers, magazines, blogs, and social networking sites and, unlike graphs in scientific articles where the article text explicitly refers to and explains the graphics, the information conveyed by graphics in popular media is often not repeated in the article's text (Carberry, Elzer, and Demir 2006). Yet these graphics are a significant knowledge source.

Consider, for example, an author who is constructing a report and wishes to make a point concerning how social media such as Twitter can give predictive insights to the public opinions on popular events such as the United States presidential election. The author could try to describe in words that the comparative number of tweets mentioning each candidate generally corresponds to the popularity of that candidate. On the other hand, imagine that the author has at his/her disposal a query system that enables him/her to put out a query to retrieve an information graphic that compares the number of tweets for the two candidates (President Barack Obama and Governor Mitt Romney) in the weeks leading up to the election. The following, which we will refer to as Q_1, might be such a query:

> Q_1: *How many tweets mentioned Obama compared to Romney from October to November?*

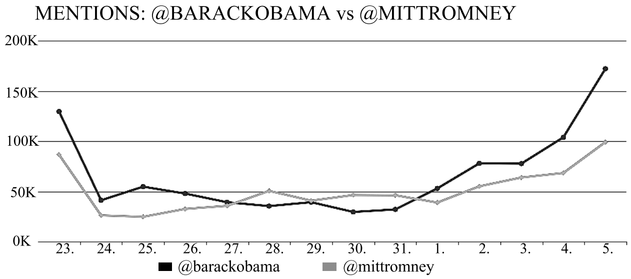

Figure 1: Comparative number of tweets mentioning Barack Obama and Mitt Romney

Figure 1, found on the official website of HootSuite (social media management system), would be an ideal response to query Q_1. Imagine how much more powerfully and convincingly the author's point can be made by including this graphic in their article because the comparative trends shown in this graphic track remarkably well with opinion polls leading up to the election (which ultimately was won by President Obama).

Unfortunately, most commercial search engines are unable to respond effectively to requests for information graphics. This is largely due to their inability to understand the content of the graphic; instead they rely on the text of the document containing the graphic, with special attention paid to the image tag information, image file name, and nearby text paragraphs surrounding the image. For example, if query Q_1 is input to Google with a request for images, the first returned graphic compares Canadian tweets for the two candidates on the three debate days and the second returned graphic compares the sentiment of Obama tweets with Gallup polls. All of the top returned graphics are similarly off-target. This is due to the retrieval system's reliance on the appearance of the query words in the text of the webpage source file, not on the graphic's content and whether it is relevant to the query.

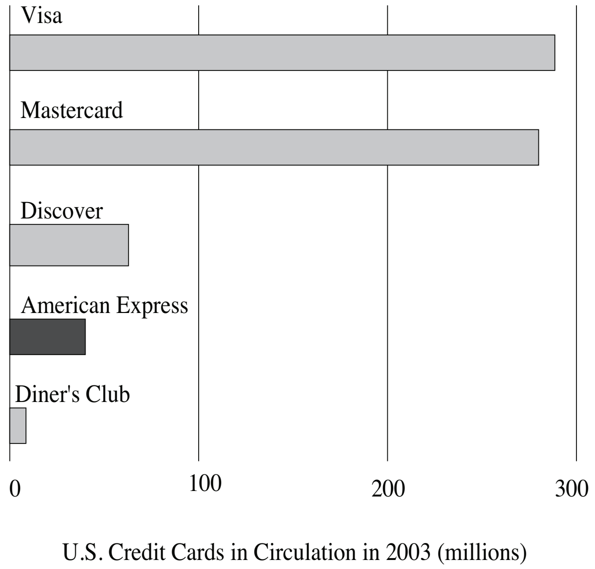

Figure 2: Bar Chart showing the rank of American Express

The long-term goal of our research is to build a system for retrieving relevant information graphics in response to user queries. Our prior work (Wu et al. 2010; Elzer, Carberry, and Zukerman 2011; Burns et al. 2012) has produced a system for recognizing the overall message of an information graphic by reasoning about the content of the graphic, including its communicative signals such as one bar in a bar chart being colored differently from other bars. For example, our system recognizes that the bar chart in Figure 2 is conveying the rank of American Express with respect to the other credit card companies in terms of US credit cards in circulation, what we categorize as a Rank message, and which can be formally represented as *Rank(American Express,* $\{$*Visa, Mastercard, Discover, Diner's Club*$\}$*, US credit cards in circulation)*. This recognized message will be used in our retrieval system to represent the graphic's high-level informational content.

In order to retrieve graphics in response to a user query, it is necessary to identify requisite characteristics of relevant graphics. These include the category of the intended message (such as a Rank message), any focused parameters (such as American Express), the entities on the independent axis (such as credit card companies), and the entities on the dependent axis (such as number of credit cards in circulation). We present in this work our methodology for developing a learned model that takes as input a natural language query and hypothesizes the content of the independent and dependent axes of relevant graphics. Future work is designed to extend this methodology to hypothesize the category of intended message and any focused parameters.

Related Work

Most image retrieval systems use various kinds of text annotation of the image, such as automatic annotation learned from a manual annotation training set (Jeon, Lavrenko, and Manmatha 2003), co-occurring text generated from the multimedia document (Lapata 2010) and user-provided metadata tags in social media like Flickr and Youtube (Gao et al. 2011). State of the art content-based image retrieval looks into the image and uses low-level features to recognize entity objects such as cars and tables, human faces, and background scenery from natural images (Yue et al. 2011; Doersch et al. 2012; Kapoor et al. 2012). Research on information graphics has focused on identifying the type of graphic such as bar charts and line graphs (Shao and Futrelle 2006; Mishchenko and Vassilieva 2011b). Others have focused on information extraction by converting information graphics into tabular form data, such as XML representations (Huang and Tan 2007; Mishchenko and Vassilieva 2011a; Gao, Zhou, and Barner 2012). To the best of our knowledge, our work is the only project which attempts to recognize the high-level message or knowledge conveyed by an information graphic and use it in determining the relevance of a graphic to a user query.

Motivation for Identifying Axis Content from Queries

One approach to retrieving information graphics given the graphic's caption, axis labels, and intended message would be to match the words in the query to the words in the graphic directly. However, in many situations the majority of the words in the query are not actually contained in the graphic. Even when they are, simply taking graphics with a high overlapping word count would produce poor search results as illustrated in the following examples.

Consider the following two queries:

Q_2: *Which Asian countries have the most endangered animals?*

Q_3: *Which endangered animals are found in the most Asian countries?*

These two queries contain almost identical words but are asking for completely different graphics. Query Q_2 is asking for a comparison of Asian countries (independent axis) according to the number of endangered animals (dependent axis) in each country. On the other hand, query Q_3 is asking for a comparison of different endangered animals (independent axis) according to the number of Asian countries they dwell in (dependent axis). The two queries are asking for graphics where the two mentioned entities, Asian countries and endangered species, are completely flipped around, just by organizing the query in a different way. This difference results in two completely different content graphs to be retrieved.

In addition, a query may contain more than two entities. Consider the following two queries where a third entity is mentioned in the query:

Q_4: *How many major hurricanes occurred in each east coast state in 2002?*
Q_5: *How many east coast states have had major hurricanes since 2002?*

This third entity could be a confusing factor causing irrelevant graphics to be retrieved. Query Q_4 is asking for a comparison of states on the east coast according to the number of major hurricanes in 2002, where *"2002"* should only be part of the label on the dependent axis. Query Q_5, on the other hand, is asking for the trend in the number of east coast states affected by major hurricanes during the years since 2002, where the independent axis should contain a listing of *the years since 2002* and the dependent axis should give the number of east coast states with major hurricanes in each of the years listed.

Based on the above concerns, we contend that preprocessing the query Q is needed to identify words that can be matched against descriptions of the independent and dependent axes of candidate graphs. This will eliminate many irrelevant graphics and reduce the number of graphs that need to be given further consideration.

Methodology for Hypothesizing Axis Content

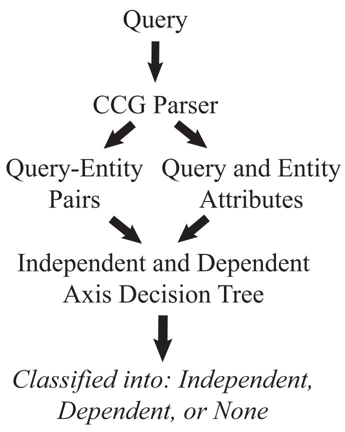
Query
CCG Parser
Query-Entity Pairs
Query and Entity Attributes
Independent and Dependent Axis Decision Tree
Classified into: Independent, Dependent, or None

Figure 3: System Model

Our methodology is to extract clues from the user's query and use these clues to construct a learned model for hypothesizing the content of the independent and dependent axes of relevant graphs. Figure 3 outlines the application of the learned model to hypothesizing the content of the axes of potentially relevant graphs. Given a new query, we first pass it to a CCG parser (Clark and Curran 2007) to produce a parse tree. We use the parser developed by Clark and Curran since it is trained expressly for questions whereas most parsers are trained entirely or mostly on declarative sentences. From the parse tree, we populate a set of candidate entities E_1, E_2, $\ldots$, E_n, and extract 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 independent axis, or the content of the dependent axis, or none of the axes.

In the following subsections, we will discuss how entities that might represent axis content are extracted, the clues in a user query that we use as attributes in the decision tree, how these attributes are extracted from a user query, and the construction and evaluation of our models for hypothesizing the requisite content of the axes of potentially relevant graphs.

Candidate Entity Enumeration

To hypothesize the content of the independent and dependent axes from a user query, we first need to generate a set of candidate entities that will be considered by the decision tree as possible content of the axes. To do this, first of all, phrases that describe a period of time are extracted as time intervals and included in the candidate entities. Afterwards, the parse tree is analyzed and noun phrases that are not part of time intervals are also extracted and added to the set of candidate entities.

The set of 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 and *"comparison"* and *"difference"* which are part of the comparison category of words. Pronouns such as *"that"* or *"it"* are also filtered from the entity candidate list since they do not carry content themselves but rather refer back to entities previously introduced. Simple compound noun phrases that do not contain any prepositions or conjunctions are considered as a single entity, such as *"American Idol"*. In contrast, compound noun phrases containing prepositions and (or) conjunctions, such as *"the revenue of Ford and Toyota"*, are broken down into component pieces such as *"the revenue"*, *"Ford"*, and *"Toyota"*.

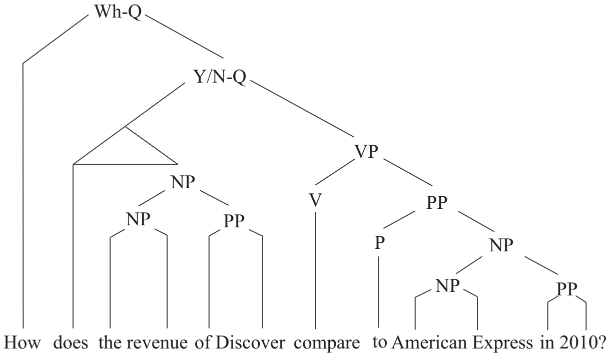

Figure 4: Abbreviated Parse tree for query Q_6

For example, Figure 4 shows an abbreviated version of the parse tree for the following query[1]:

Q_6: *How does the revenue of Discover compare to American Express in 2010?*

First of all, a specific time point phrase, *"in 2010"*, is detected from this query sentence. Then the noun phrase *"the revenue"* and *"Discover"* are extracted from the parse tree

[1] While the parser we use is a CCG parser, we have drawn the tree using more traditional syntactic node names for clarity.

and added to the set of candidate entities. The noun phrase *"American Express"* is added to the candidate list as a single noun phrase given that it does not contain any prepositions or conjunctions. The final set of query-entity pairs for query Q_6 are:

Q_6-E_1: *the revenue*
Q_6-E_2: *Discover*
Q_6-E_3: *American Express*
Q_6-E_4: *in 2010*

Cues from the User's Query

We are considering only full sentence queries, such as the examples presented above. While most text retrieval systems work with keyword queries, they are retrieving documents each of which can fulfill a variety of information needs. We wish to retrieve information graphics that are relevant to a user who has a specific information need in mind that the graphics are to fulfill. We wish to therefore develop mechanism to analyze the semantics of full sentence queries to identify characteristics of graphics relevant to the user's information need.

The content and structure of such queries provide clues to the content of relevant graphics. We have identified a set of attributes that might suggest whether a candidate entity reflects the content of the independent or dependent axis of a relevant graphic. These clues can be divided into two classes: *query attributes* that are features of the whole query and are independent of any specific entity, and *entity attributes* that are particular to each specific candidate entity.

One query attribute is the question type of the query sentence. For example, *"Which"* or *"What"* queries, such as Q_2 and Q_3, are often followed by a noun phrase that indicates the class of entities (such as *countries*) that should appear on the independent axis. On the other hand, *"How many"* and *"How much"* queries, such as Q_4 and Q_5, are often followed by a noun phrase that indicates what quantity (such as number of *major hurricanes*) should be measured on the dependent axis.

Comparatives and superlatives also provide clues. For example, the presence of a superlative, such as *"highest"* in the query *"Which countries had the highest GDP in 2011?"*, often suggests that the dependent axis should capture the noun phrase modified by the superlative. Similarly, comparatives such as *"higher"* in the query *"Which countries have a higher GDP than the U.S. in 2011?"* suggests that the dependent axis should capture the following noun phrase, which in this case is also *"GDP"*.

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

Q_7: *How does CBS differ from NBC in terms of viewers?*
Q_8: *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 independent 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_7, then the noun phrases suggest entities that should appear on the independent axis; on the other hand, if one is plural (as in Q_8), then it suggests the class of entities to be displayed on the independent 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?"*, indicate a change in the dependent variable, which is *"oil prices"*. Such words suggest that the entity (noun phrase) dominated by them is likely to be on the dependent axis. On the other hand, the characteristic of representing a time interval, such as the entity *"from January to March"* or *"in the past three years"*, suggests that the entity captures the content of the independent axis.

Extracting Attributes for each Query-Entity Pair

Table 1 describes the categories of attributes we have included in our analysis for identifying the content of the independent and dependent axes. Recall that query attributes are based on features of the whole query whereas entity attributes are particular to the entity in the query entity pair. 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.

For example, let us consider query Q_6. Query Q_6 is of *"How does"* question type, causing the query attribute from the question type attribute category to be set to *True* for every query-entity pair derived from Q_6. Regular expressions detect that Q_6 contains a phrase describing a specific time point, which is query-entity pair Q_6-E_4 that is *"in 2010"*, so the attribute from the associated with the presence of a time interval in the query is set to *True* for this query-entity pair, and *False* for the other query-entity pairs from Q_6. In the rest of the query, the system finds the presence of a word from the *comparison* word category therefor setting the query attribute of presence of *Comparison* category words to be *True* for all the query-entity pairs from Q_6; for query-entity pair Q_6-E_2 and Q_6-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 is the leftmost noun phrase following the question head *"How does"* in the parse tree (Figure 4), the attribute reflecting the leftmost noun phrase is set to *True* for query-entity pair Q_6-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.

Constructing the Decision Tree

In order to construct a corpus of full-sentence queries oriented toward retrieval of information graphics, a human subject experiment was conducted. Each participant was at least 18 years of age and was either a native English speaker or was fluent in English. Each subject was shown a set of information graphics on a variety of topics such as adoption of children, oil prices, etc. For each displayed graphic, the

Whole Query Attributes	Presence of superlative or comparative in the query sentence.
	Question types, including *Which*, *What*, *How many* or *How much*, *How do* or *How have*, *What is*.
	Presence of *Trend* category words or *Comparison* category words in the query sentence.
	Main verb is a *trend verb* or *comparison verb*.
	If main verb is a *comparison verb*, this comparison verb is followed by a preposition, such as *with*, *to*, *against*, *from*, or *among* and *amongst*, or *between*. Otherwise, the comparison verb has no preposition and is used at the end of the query.
Specific Entity Attributes	Whether the entity contains a superlative or comparative.
	Whether the entity describes a time interval, a specific time point, or neither of them.
	Whether the entity contains or is modified by *gradient* category phrases, such as *in terms of*, or *quantity* category words, such as *the number*, or *comparison* category words, or *trend* category words.
	Whether the entity is modified by *inclusive* words such as *all* and *total*, or by *exclusive* words such as *other* and *the rest*, or by words that indicates enumeration such as *each* and *every*.
	Whether the entity is the direct noun phrase following each question type.
	Whether the entity is singular or plural.
	If the main verb belongs to either the *comparison* or *trend* category, whether the entity is on the left or right side of the main verb.

Table 1: A description of attribute categories

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. [2]

To construct a training set, candidate entities are extracted from each query, a set of query-entity pairs is constructed, and values for each of the attributes are extracted. Our decision tree takes in the query-entity pairs, along with their attribute values, and returns a decision of independent axis entity, dependent axis entity, or neither. The classification of each query-entity pair for training is assigned by one researcher and then verified by another researcher with the final annotation of each query-entity pair indicating both researchers' consensus. To construct the decision tree for identifying the content of the dependent and independent axes, this training set is then passed to WEKA, an open-source machine learning toolkit.

Evaluation of the Methodology

Leave-one-out cross validation holds out one instance from the full dataset of n instances. The classifier is constructed from the other n-1 instances and tested on the held-out instance. This procedure is repeated n times with each of the n instances serving as the held-out test instance, and the results are averaged together to get the overall accuracy of the classifier. This strategy maximizes the size of the training set.

In our case, each query can produce more than one query-entity pair. This leads to an unfair advantage in evaluating our classifier, since the training set contains query-entity pairs extracted from the same query as the held-out test instance. This would mean that all query attribute values (those based on the whole query) would exist identically in both the training and testing sets at the same time. In order to prevent this from happening, we use a variant of leave-one-out cross validation which we refer to as *leave-one-query-out* cross-validation. In *leave-one-query-out* cross-validation, all of the query-entity pairs extracted from the same query will be bundled together and used as the testing 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.

Note that we are evaluating the overall system (not just the decision trees) since the held-out 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.

We evaluated our methodology using accuracy as the evaluation metric: the proportion of instances in which the annotated correct classification matches the system's decision (*Independent*, *Dependent*, or *None*). This is depicted in Table 2, in which the system is awarded 1 point each time it matches the system annotation.

		System Output		
		Ind.	Dep.	None
Annotation	Ind.	+1	+0	+0
	Dep.	+0	+1	+0
	None	+0	+0	+1

Table 2: Standard Accuracy Scoring

Axis Extraction

There are 76 entities that are considered as on neither of the axes, which is only 7.1% of the total entities generated. Therefore 92.9% of the entities generated by our system are helpful in identifying the independent and dependent axis of the hypothesized graph. Prediction of the independent and dependent axis has a baseline accuracy of 54.15%, that is, if the system always predicts the majority class of *Independent*

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

Axis for every query-entity pair. However, our methodology did considerably better than the baseline, achieving an overall accuracy of 81.51%, with overall precision of 79.26%, overall recall of 73.96%, and overall F-measure of 76.5%. For identifying the *Independent Axis*, we achieved precision of 81.61%, recall of 78.07%, and F-measure of 79.77%. For identifying the *Dependent Axis*, we achieved precision of 82.14%, recall of 87.24%, and F-measure of 84.58%.

The confusion matrices for the axis classification is displayed in Tables 3.

		System Output		
		None	Ind.	Dep.
Annotation	None	43	9	24
	Ind.	5	324	86
	Dep.	10	64	506

Table 3: Axes Classification Confusion Matrix

Conclusion and Future Work

As presented in this paper, we have achieved the first step toward successful retrieval of relevant information graphics in response to a user query. Our methodology for identifying the content of the independent and dependent axes uses natural language processing and machine learning and has a success rate of 81.51% range, substantially exceeding the baseline accuracy. Our future work will include extending our methodology to recognize the message category (such as a Rank message or a Trend message) of graphics that are potentially relevant to a user query and any focused parameters of the graphic's message. We will then use these in a mixture model that matches requisite features identified from the user's query with features of candidate graphs in order to rank graphs in terms of relevance. To our knowledge, this work is the only ongoing effort that attempts to use the informational content of a graphic in determining its relevance to a user query.

References

Burns, R.; Carberry, S.; Elzer, S.; and Chester, D. 2012. Automatically recognizing intended messages in grouped bar charts. *Diagrammatic Representation and Inference* 8–22.

Carberry, S.; Elzer, S.; and Demir, S. 2006. 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*, 581–588. ACM.

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

Doersch, C.; Singh, S.; Gupta, A.; Sivic, J.; and Efros, A. 2012. What makes paris look like paris? *ACM Transactions on Graphics (TOG)* 31(4):101.

Elzer, S.; Carberry, S.; and Zukerman, I. 2011. The automated understanding of simple bar charts. *Artificial Intelligence* 175(2):526–555.

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

Gao, J.; Zhou, Y.; and Barner, K. E. 2012. View: Visual information extraction widget for improving chart images accessibility. In *IEEE International Conference on Image Processing (ICIP)*. IEEE.

Huang, W., and Tan, C. 2007. A system for understanding imaged infographics and its applications. In *Proceedings of the 2007 ACM symposium on Document engineering*, 9–18. ACM.

Jeon, J.; Lavrenko, V.; and Manmatha, R. 2003. Automatic image annotation and retrieval using cross-media relevance models. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, 119–126. ACM.

Kapoor, A.; Baker, S.; Basu, S.; and Horvitz, E. 2012. Memory constrained face recognition. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 2539–2546. IEEE.

Lapata, M. 2010. Image and natural language processing for multimedia information retrieval. *Advances in Information Retrieval* 12–12.

Mishchenko, A., and Vassilieva, N. 2011a. Chart image understanding and numerical data extraction. In *Digital Information Management (ICDIM), 2011 Sixth International Conference on*, 115–120. IEEE.

Mishchenko, A., and Vassilieva, N. 2011b. Model-based chart image classification. *Advances in Visual Computing* 476–485.

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

Wu, P.; Carberry, S.; Elzer, S.; and Chester, D. 2010. Recognizing the intended message of line graphs. *Diagrammatic Representation and Inference* 220–234.

Yue, J.; Li, Z.; Liu, L.; and Fu, Z. 2011. Content-based image retrieval using color and texture fused features. *Mathematical and Computer Modelling* 54(3):1121–1127.