Knowledge Resource Development for Identifying Matching Image Descriptions

Alicia Sagae and Scott E. Fahlman
Language Technologies Institute, Carnegie Mellon University, Pittsburgh, PA, U.S.A.

Keywords: Image Retrieval, Textual Similarity, Textual Inference.

Abstract: Background knowledge resources contribute to the performance of many current systems for textual inference tasks (QA, textual entailment, summarization, retrieval, and others). However, it can be difficult to assess how additions to such a knowledge base will impact a system that relies on it. This paper describes the incremental, task-driven development of an ontology that provides features to a system that retrieves images based on their textual descriptions. We perform error analysis on a baseline system that uses lexical features only, then focus ontology development on reducing these errors against a development set. The resulting ontology contributes more to performance than domain-general resources like WordNet, even on a test set of previously unseen examples.

1 INTRODUCTION

This paper describes experiments to retrieve images based on matching their descriptive English labels. As in ad-hoc document retrieval, a baseline system using term vectors to represent these labels performs reasonably well (>80% MRR, Mean Reciprocal Rank). However, error analysis reveals that the most challenging examples for this task require a richer feature space, allowing the system to capture more of the deep semantic similarities that humans seem to notice when they make comparisons between images and their descriptions. As a result, we present a solution that uses knowledge-based features for identifying when two English descriptions refer to the same image.

Object labels assigned by humans typically consist of short multi-word phrases. These phrases exhibit syntactic and semantic structure that is not always modeled by information retrieval systems. Nonetheless, humans rely on this structure, along with background knowledge, when generating and interpreting labels. These characteristics place our task in the class of Applied Textual Inference (ATI) problems. ATI tasks depend on some level of text understanding and background knowledge, but they are designed to abstract away from system-specific representational choices. They include summarization, question answering, and recognizing textual entailment, among other problems. *Image-identity* is a relation that holds between two texts, A and B, when they refer to the same image. In our current work, we focus on the ATI problem of recognizing image-identity between a description that serves as a query to a retrieval system, and a description that labels a known image in a collection.

2 RELATED WORK

This work draws on related research in image retrieval and knowledge-based textual inference. Digital images are commonly associated with textual metadata, including tags, titles, descriptions, or a textual context such as a web page where an image has been embedded. As a result, query topics and indexed images can be represented by textual features (*text-based image retrieval*) or by features derived from computer-vision analysis of the image (*content-based image retrieval*). Other approaches explore a combination of the two (*multimedia* or *multimodal image retrieval*). Multimodal retrieval techniques have shown promise in retrieving images in response to a textual query, even when the images in the test set were not annotated (Blei and Jordan, 2003). To achieve this, a joint model of visual features and keywords was learned from a training set before running the test queries. Similar joint models have been used for generating image descriptions (Farhadi et al., 2010) and for mea-

Sagae A. and Fahlman S..
Knowledge Resource Development for Identifying Matching Image Descriptions.
DOI: 10.5220/0004550601000108
In *Proceedings of the International Conference on Knowledge Engineering and Ontology Development* (KEOD-2013), pages 100-108
ISBN: 978-989-8565-81-5

suring semantic relatedness between words and images (Leong and Mihalcea, 2011). However, these models capture only the relationship between text in a query and visual features in an indexed image. Annotations on the indexed image are not leveraged, even when available.

Other approaches to multimodal retrieval allow the models to take advantage of text-only features in addition to visual and joint textual-visual features. In the Wikipedia Image Retrieval Task at ImageCLEF 2011 (Tsikrika et al., 2011), the best-performing system applied Late Semantic Combination to leverage features from text and visual modalities (Csurka et al., 2011). Under this combination strategy, text-only features and visual-only features can be developed and improved independently, and still contribute to better combined multimodal performance.

As a result, text-only image retrieval features like the ones explored in this work can be applied on their own, as we show here, and may also be combined in a multimodal system. In addition, although multimodal retrieval represents a growing research area of interest, visual features of a query may not be available in a typical real-world image retrieval scenario. The organizers of the ImageCLEF 2011 Wikipedia Image Search task, while encouraging participants to develop multimodal systems, acknowledge that "a text-only query… is likely to fit most users searching digital libraries or the Web." (Tsikrika et al., 2011). In addition, text-only features have been shown to outperform multimodal features for some tasks like automatic image tagging (Leong et al., 2010).

Constraining ourselves to text-based representations of images, we find that the central problem of image retrieval is to compare two texts and determine whether the image-identity relation holds between them. This problem is an instance of Applied Textual Inference (ATI). Taking the PASCAL RTE Challenge as an example, we can see that deep semantic representations and knowledge-based techniques play an important role in state-of-the-art ATI systems. Of 16 research teams participating in the first challenge in 2005[1], 7 used features from WordNet, 3 applied some kind of world knowledge, and 7 applied logical inference engines (9 systems out of 16 used at least one of the three). In 2007 the number of participants and the variety of techniques expanded, with the vast majority of these systems relying on some combination of WordNet, syntactic matching/alignment, and machine learning algorithms; the most successful system in that year applied all of these techniques in addition to a logical inference engine (Hickl and Bensley, 2007).

[1]http://pascallin.ecs.soton.ac.uk/Challenges/RTE

Our approach applies knowledge from general-purpose semantic resources like WordNet and Dolce, in addition to developing new custom resources for our task and for our training set. This approach to ontology development is consistent with (Montazeri and Hobbs, 2011), however we perform an additional step of error analysis before beginning ontology development. One contribution of our work is the methodology for using this error analysis to drive ontology development, described in Section 6.2. In addition, many applications of formal ontology use manually-constructed rules to operate over ontology concepts and produce an analytical result. A second contribution of our work is the architecture for gathering relevant ontological features and combining them with syntactic features in a learned reranking function, described in Section 7.

3 PROCEDURE

In our experiments, we developed an ontology to help identify the description-identity relation among texts. To evaluate, we performed retrieval on a data set where images are associated with multiple descriptions. For each image, one description is held out as a development query, another is held out as a test query, and the remainder make up the "document" that represents the image in a collection.

First, we constructed a baseline system that uses only lexical features (Section 5.2) and performed error analysis on the baseline, in order to identify the most promising conceptual space for ontology development. Next, we used a set of training queries to construct the ontology (Section 6.2) and to provide ontological features to a perceptron-based classifier, training it to decide image-identity. Finally, we applied the ontology and the classifier to a held-out set of test queries in order to rerank the baseline retrieval results (Section 7). Our results show an improvement in performance as measured by Mean Reciprocal Rank (MRR).

4 DATA

We evaluate on the Phetch data set, collected by (von Ahn and Dabbish, 2004). It was collected in the context of an online game where multiple participants compete in teams to identify an image based on one teammate's typed description. The exercise was repeated for each image in a large collection of JPEG files harvested from the web. In this data set, a single description is a short paragraph written by a single

Annotator 1 [training query]:	"stadium; pageant girls in the foreground"
Annotator 2 [testing query]:	"olympic stadium; women in foreground; with sashes"
Annotator 3 [indexed document]:	"people at sporting event; pageant contestants"

Figure 1: Image from the Phetch data set with descriptions. One description is used for training, another for testing, and remaining descriptions are concatenated to form the document representation in the retrieval index.

annotator/participant about a single image. Each image in the collection has multiple descriptions. Each description is composed of one or more phrases, short segments that contribute to the overall description and are usually connected rhetorically to each other. An example is shown in Figure 1.

The total number of images represented in the corpus is over 50,000. Of these, approximately 6,000 were labeled with 5 or more descriptions. Our experiments were conducted on a subset of 700 images that have 5 descriptions or more and that do not contain text in the image itself. This subset is shown as partition *5A-notext subset* in Table 1.

The availability of relevance judgments and the level of detail in textual annotations set the Phetch corpus apart from other data sets that are commonly used to evaluate image retrieval, including the Corel image collection and the ImageCLEF evaluation sets. The Corel data set is most appropriate for evaluating systems that use a sample image as a query, rather than using text. Keywords, but not descriptions, are available for the images in Corel. Some difficulties with this data set for standardized evaluation are discussed by (Müller et al., 2002).

The ImageCLEF evaluation sets have evolved over the years; one set was derived from the IAPR TC-12 Benchmark of 20,000 images (Grübinger et al., 2006). This set has been used in the ImageCLEF image retrieval evaluations since 2006[2]. This data includes image descriptions, but provides fewer examples labeled with relevance judgments than Phetch. In the Phetch data set, every image is labeled with descriptions that can be used as a query/document pair, where relevance judgments are binary: if the document contains descriptions that come from the same image as the description being used as a query, relevance is 1. Otherwise, relevance is 0. In our experiments, we test on all 700 images from the 5A-notext subset of Phetch. ImageCLEF, in contrast, contains a subset of 60 images that have been labeled with relevance judgments.

The MIRFLICKR-1M data set (Huiskes and Lew, 2008) contains 1 million images with their Flickr tags, published under a Creative Commons license. These images do not include full-phrase descriptions of the type associated with each image in the Phetch collection. However they are annotated with content-based visual descriptors. As a result, MIRFLICKR has been used for image annotation and retrieval by visual example, but is not sufficient for testing retrieval by phrasal description.

A smaller data set with structure similar to Phetch was collected in 2010, using Amazon's Mechanical Turk as a source for annotators (Rashtchian et al., 2010). This data set includes 8000 images from Flickr.com, annotated with multiple full-sentence descriptions. Although it contains far fewer examples, this set was developed with natural language processing applications in mind. As a result more attention was paid to annotation quality than in the Phetch data, and this data is likely to be more free of noise due to misspellings and other annotator errors.

5 BASELINE

5.1 Parameterization

In our document collection, each document is a set of descriptions for one image. Since each of the 700 images in the 5A-notext subset have 5 descriptions or more, we use one description from each image as a development query. One additional description is

[2]http://ir.shef.ac.uk/imageclef/2006/

Table 1: Phetch corpus partitions. Partitions are created based on the number of descriptions associated with each image.

Section	Descriptions per Image	Total Images	Total Words
1A	1	17,237	470,924
1B	1	17,237	470,924
2A	2	7,264	371,313
2B	2	7,265	371,879
3A	3	5,171	367,284
3B	3	5,171	367,999
4A	4	3,084	283,142
4B	4	3,084	282,486
5A	$\geq$5	2,946	357,582
5A-notext subset	$\geq$5	700	68,867
5B	$\geq$5	2,946	355,853

held out as a test query. The remaining 3 descriptions are taken together to be the document representing the image in our collection. To retrieve an image, we compare the query to each document in the collection and calculate a relevance score. The image from which the query description was taken is interpreted as the only relevant document. The rest of the documents in the collection are taken to be non-relevant for that query. This interpretation casts the retrieval task as a way to identify the image-identity relation among two texts (the query and an image document).

To establish a baseline for retrieval performance on the Phetch data, we perform indexing and retrieval with version 2.5 of the Indri search engine (Strohman et al., 2005), a component of the Lemur Toolkit for Language Modeling and Information Retrieval[3]. Indri implements a retrieval model that combines the language modeling approach (Ponte and Croft, 1998), which estimates word probabilities, with the inference network approach (Turtle and Croft, 1991) for combining beliefs into a single document-level retrieval score. Indri supports a complex query syntax that includes ordered and unordered windows, along with field-specific language models. In our initial experiments, the best-performing query formulation was the `#combine` operator, which treats the terms in the query and document as an unordered bag-of-words, and calculates document relevance as a function only of the overlap in terms between the query and a given image description document. This keyword parameterization does not take word order or other syntactic structure into account. Results for this setting are given in Table 4 as *Kwds+spell*, or retrieval with keyword features that have undergone a spell-checking pass. On the development queries, the baseline achieves MRR 0.8295; on the test queries, it achieves 0.8216.

[3]http://www.lemurproject.org/

5.2 Error Analysis

In the retrieval setting described above, the main criterion for success is returning the single image of interest at the lowest rank possible. Since each query describes precisely one image, we seek a metric that measures where in the results list that image appeared. This metric is Mean Reciprocal Rank (MRR). MRR is defined as the average, over all queries, of 1 divided by the rank where the correct document was found.

When the system assigns the relevant image a rank of $N > 1$, at least two errors have occurred. First, the system compared a non-relevant image description to the query and determined that they describe the same image, when in fact they do not. This error is a type of false-positive judgement, which we refer to as a precision error, because it dilutes the result list with a non-relevant image at a high rank. Second, the system compared the relevant image description to the query and determined that they did *not* describe the same image, at least not confidently enough to rank the document first in the result list. This is a type of false-negative judgement, which we refer to as a recall error, since it implies that the system failed to recognize the relevant image when it appeared. We have developed a vocabulary of error classes that trigger both precision and recall errors. The most frequent of these classes are described in Table 2.

The vocabulary of error classes is motivated by the hypothesis that the bag-of-words representation for image descriptions leads to errors because it fails to recognize certain types of textual similarity. Specifically, the bag-of-words model fails to capture semantic similarities that are obscured by surface features

Table 2: Classes of error in the baseline retrieval system.

Error Class	Context	Description and Examples
Ontology	Precision	The wrong word meaning triggered a false match "on the bank" $\neq$ "at the bank"
	Recall	Failed to match words that would be similar Ontological classes "shawl" $\sim$ "wrap"
Faulty Inference	Precision	Match in spite of conflicting relations among words "green bandana" $\neq$ "green shirt", "man skating" $\neq$ "girl skating"
	Recall	Failed to make a relevant inference "lips are puckered" $\sim$ "getting ready to kiss"
Contradiction	Precision	Failed to recognize contradictions "black background" $\neq$"blue background"
	Recall	Failed to match due to faulty contradiction "black or blue background" $\sim$ "black background"
Missing Elements	Precision	Failed to penalize for missing major elements "globe on a stand" $\neq$ "globe"
	Recall	Over-penalized for minor missing elements "guy smiling with glasses" $\sim$ "a guy smiling"

like word choice, and it fails to follow the inferential chains of reasoning that human annotators envision between their descriptions and the content of an image. As a result, our error classes are composed of specialized cases of semantic and inferential mismatch that we expect to see in the errorful retrieval runs.

We arrived at this vocabulary of error types in an iterative fashion, based on observations in a sample of 50 retrieval results from the Phetch 3A data set (a sample that does not overlap with the 5A-notext subset on which we test). In a first pass, we annotated this development sample with free-text descriptions of the evidence that a human might use to correct the errors made by the baseline system. In a second pass, these annotations were distilled by hand into a set of 14 phenomena that result in retrieval error. These 14 classes were used to re-annotate the sample. After this pass, a final revision of the annotation classes was made to focus on the most frequent and clearly-defined classes. The resulting vocabulary contains 8 classes with precise definitions in the precision-error and recall-error contexts. These classes are not mutually exclusive; rather, a given retrieval error can be annotated with all classifications that apply.

Table 3 shows the most frequently occurring error types in the annotated sample from section 3A. For each query we make two comparisons: we compare the query description to the indexed description of the same image in order to annotate the recall errors. We also compare the query description to the indexed description that was retrieved at rank 1 for this query, in order to annotate the precision errors. Although not all of the errorful results returned by the baseline system involve errors from these classes, most of them do (90% of precision errors and 86% of recall errors). Recall errors were annotated with 3.3 of these classes, on average, and precision errors were annotated with an average of 2.5 classes.

The most frequently-appearing class in the case of recall errors were Ontology-related; that is, surface-level mismatch between concepts that would be identical or closely linked in an ontological representation of background knowledge. The most frequently-occurring classification of precision errors relates to contradiction. In these cases, the baseline system failed to recognize an explicit contradiction between the query and the image description that was retrieved at rank 1. A system with a model for recognizing contradiction might be able to correct the baseline system in nearly 60% of the cases where it currently makes Precision errors.

Given this background knowledge about the nature of retrieval errors in a baseline retrieval run, we can identify some specific strategies for improving retrieval performance. We have established a set of error classes and textual features that contribute to retrieval error under the bag-of-words model. In the next Section we will establish a more knowledge-rich model for representing image descriptions and use that model to implement handlers for the error classes described here: Ontology-based matching and inference, handling of quantification over major content elements, negation, analogy, and a model of media types.

Table 3: Highest-frequency error classes.

Error Type	Freq in Recall Errors	Freq in Precision Errors
Ontology	36 (72%)	7 (14%)
Quantification	34 (68%)	26 (52%)
Faulty Inference	29 (58%)	20(40%)
Missing Elements	23 (46%)	27(54%)
Any	43 (86%)	45 (90%)
Total	165	126
Average	3.3	2.52

6 ONTOLOGY

6.1 Framework and Development

To address the errors in Section 5.2, we developed an ontology using the freely available Scone Knowledge Base System[4] (Scone) as our ontology framework. The Scone engine supports adding, searching, and evaluating logical statements based on marker-passing inference (Fahlman, 2006). To support the experiments described in Section 7, we implemented additional Common Lisp components that extend Scone engine functions. These include new ontologies, inference routines, and APIs that use Scone to annotate text and measure the semantic distance between concepts. In the remainder of this paper we use "SconeImage" to refer to the extended software suite.

Knowledge bases in Scone use a frame-semantics formalism to represent a network of concepts, or Scone *elements*. Scone supports taxonomic relationships, like "a *flower* is a *plant*" as well as role-filling relationships, like "a *flower* has *scent*". New non-taxonomic relations can be defined as well, with instances of such relations being encoded as statements, like "a *bird flies*". Exceptions can be marked to handle relations that apply to most, but not all, instances of a class, as in "a *penguin* is a *bird* that *does not fly*". Scone also includes a lexical lookup function that allows multiple strings to be attached to any element.

Given such a knowledge base as input, routines defined in the Scone engine calculate the answer to queries like "Is a *rose* a *flower*?". Extensions in SconeImage make higher-level calculations that depend on these answers, like "what is the relationship between *bouquet* and *rose*?". In all cases, the answers returned by these calculations depend on the knowledge bases that are currently loaded.

To build these knowledge bases (KBs), we used a combination of automatic and manual processes. We first attempted to leverage existing ontological knowledge by importing concepts and relations from WordNet (Fellbaum, 1998), to which domain-specific knowledge can later be added. Using this knowledge alone to annotate image descriptions with semantic information (using the first nominal synset available for each content word in the description), we achieved an improvement over the baseline system, but with small statistical significance ($p >= .4$ using single-factor ANOVA). This result supports the hypothesis that knowledge may help on this task; however, our baseline error analysis indicated that some classes of error can only be corrected by a system that applies reasoning and inference over its knowledge base representations. The ad-hoc network structure of WordNet was not intended to support this type of reasoning.

We hypothesized that a task-specific knowledge base, structured with the challenges from Section 5.2 in mind, could improve performance even more. The intuition that KB structure, particularly at the upper levels of the ontology, plays an important role in its usability and effectiveness is supported by related work on textual inference and knowledge engineering. (Fan et al., 2003), for example, describe the effect of ablating layers of the KB in a system for resolving noun-noun compounds. Their finding was that concepts from the upper levels of the ontology were critical to performance on that task, and that they had a larger impact on performance than concepts near the frontier. We leverage these findings in our work by selecting an existing upper-level ontology and re-connecting a subset of WordNet concepts to that ontology, while also adding several concepts specific to the task of image retrieval as well as concepts specific to our training data. In the SconeImage KB, we use the DOLCE upper ontology (Masolo et al., 2003) for the top levels, and then apply a mapping from DOLCE to WordNet following (Gangemi et al., 2003), with some task-specific modifications.

6.2 Acquiring Knowledge from Training Data

After the new upper-level SconeImage KB has been constructed, we perform a round of knowledge acquisition based on the training queries from the Phetch 5A data set. The baseline retrieval run on this data set resulted in 50 retrieval failures, cases where no relevant image was found in the result list. These failures were classified according to the procedure described in Section 5.2. To expand the knowledge base, a developer examined each of the retrieval failures annotated as *Ontology* or *Faulty Inference* errors. The training query and collection document were com-

[4]http://www.cs.cmu.edu/~sef/scone/

pared, and terms were added to the ontology to compensate for the error.

New elements are selected to correspond with an appropriate term from WordNet, but their arrangement in the ontology may differ significantly from their placement in WordNet. This development strategy is consistent with the analysis that WordNet is most useful for associating strings with lexical-semantic concepts, while the arrangement of concepts into logical structures can be improved through connection to an ontology like DOLCE and an inference platform like Scone. For example, while the terms *man, boy, woman,* and *child* all appear in the WordNet hierarchy, the structure connecting "man" and "boy" is different from the structure connecting "woman" and "girl". This type of inconsistency means that an inference rule of the form *if the query mentions a subtype of person, expand with sister terms* would correct one of these lexical mismatches but not the other. We prefer an ontology structure that uses parallel structures for conceptually parallel relationships among concepts.

7 EXPERIMENTAL RESULTS

In our error analysis exercise, we identified retrieval failures in the training set that could be attributed to lack of ontological knowledge in the baseline system. We hypothesized that a simple annotation approach can reduce these errors. To test this hypothesis, we implemented a SconeImage module in Lisp that performs a single-pass search over the words in an image description for elements in the SconeImage ontologies that are triggered by each word. All elements are added without performing word sense disambiguation. The resulting list of element identifiers is concatenated with the original description to form the annotated query or collection description. Indexing and retrieval are performed using Indri under the same model described in Section 5.1.

In the 3A development subset, which was used to estimate the frequency of error types in Section 5.2, the knowledge-augmented system reduces the ontological errors by 25%, a difference that is statistically significant[5] with 95% confidence ($p = 0.05$). This result supports our hypothesis that knowledge base annotation could reduce ontology-related retrieval errors. When we turn to the full development set 5A, which has many more examples, we see a reduction in error of 17%. This result is marginal statistically but is still an encouraging finding in support of the hypothesis that correcting ontology errors leads to better overall performance. Improvement on the full test set is similar, with an error reduction of 15% compared with the baseline. Even with marginal statistical significance of $p = 0.1$, this result shows some support for the conclusion that these improvements will generalize well to unseen queries.

We further hypothesized that the class of errors related to inference can be reduced by using our knowledge base in combination with syntactic information. To test this hypothesis, we first annotated each query and result description from the baseline run with a dependency structure, storing the result in a semantic graph. Such a graph links vertices in the syntactic dependency tree (i.e. words from the description) with concepts from the knowledge base. As a result, we can calculate similarity features between the query graph and the graph for any image in the candidate list, taking both semantic and syntactic similarity into account.

To combine these features into a relevance score for a document, given a query, we define a combination function for the features and perform both manual and learned weighting of features in this function. The manually-set weighting allows our intuition about the relative importance of knowledge-base concepts to play a role. However, the best performance is achieved when we make this hand-tuned score (*gphSim score* in Table 4) into another input feature for the learned weighting function. We train an off-the-shelf perceptron classifier based on (Collins, 2002) for this purpose. The classifier distinguishes graph pairs that describe the same image from graph pairs that do not. To train the classifier, we generate similarity features with SconeImage for every query-candidate pair in the baseline retrieval output. When more than one description is available for an indexed image in the candidate list, we generate the features for every such description independently.

For every query in the training set, at most one candidate contains descriptions of the same image as the query. These descriptions are positive training examples. The remaining descriptions are negative training examples. The classifier learns a set of weights $\lambda_1 \cdots \lambda_N$ for a linear combination over all of the features $f_1 \cdots f_N$ that we calculate over graph pairs:

$$\text{learnedGraphSim}(g, g') = \sum_{n=1}^{|F|} \lambda_n \times f_n(g, g') \quad (1)$$

where F is the set of all similarity features, and λ_n is the weight of feature f_n.

At test time, we use a fresh set of queries that were never seen by the classifier during training (the

[5] single-factor ANOVA

Table 4: Results of retrieval using combinations of syntactic and semantic features for graph-based reranking. ANOVA significance is shown for improvement over the *keywords* baseline.

Ranking Features					MRR (Test)	MRR (Train)
Kwds+ spell	Synt-graph	KB annot.	Sem-graph	gphSim score		
√					0.8295	0.8216
√	√				0.8404	0.8383
√		√			0.8508	0.8494
√	√	√			0.8535	0.8559
√	√	√	√		0.8531	0.8584
√	√	√		√	0.8528	0.8543
√	√	√	√	√	0.8567 ($p = 0.05$)	0.8575 ($p = 0.02$)

Kwds=keywords, spell=spell-correction, Synt-graph=syntactic graph features for reranking, KB-annot=query expansion with KB concepts (before reranking), Sem-graph=semantic graph features for reranking, gphSim score=value of the graphSim hand-tuned distance function

test query descriptions from Phetch Section 5A). The test queries are run through the baseline retrieval system. The results from this run are sent to SconeImage for graph-feature extraction and scoring using the hand-tuned similarity function. The output of the test run is one verdict and similarity measurement for every description of every candidate in the result list.

Our aim was to address inference-related errors by including dependency information in our parameterization of the retrieval problem. We ran the learned-GraphSim reranker on the annotated 3A subset in order to observe the effect on annotated inference errors. In comparison with the baseline, inference errors were reduced by 24%. This represents a large absolute improvement on inference errors as a result of adding dependency information. Because the number of such examples is small (50), this number is only marginally significant ($p = 0.07$); however, results on the full test set (5A-notext-subset) confirm that this reduction contributes to better performance overall, underscoring the importance of these gains.

Table 4 gives results from runs of the retrieval system using a variety of features, including the baseline (*Kwds + spell*), concepts from the task-specific knowledge base (*KB annot*), and graph structures described above (*Sem-graph, gphSim score*).

8 CONCLUSIONS AND FUTURE WORK

Background knowledge resources contribute to the performance of many current systems for textual inference tasks (QA, textual entailment, summarization, retrieval, and others). However, it can be difficult to assess how additions to such a knowledge base will impact a system that relies on it. This paper describes the incremental, task-driven development of an ontology that provides features to a system that retrieves images based on their textual descriptions. We perform error analysis on a baseline system that uses lexical features only, then focus ontology development on reducing these errors against a development set. The resulting ontology contributes more to performance than domain-general resources like WordNet, even on a test set of previously unseen examples.

This paper describes the development of a knowledge base as a principled response to errors found in a baseline image retrieval system. These errors were found to be triggered by shortcomings in the bag-of-words representation of image descriptions. We apply the knowledge base for simple term annotation and for learning measures of distance between graphs constructed in the semantic space of the KB. Our experiments support the hypotheses that textual inference techniques can lead to improved retrieval performance, in particular on the most interesting types of images: images whose descriptions can only be interpreted with the application of ontological knowledge and inferential knowledge. In addition, these improvements can complement the strengths of a strong bag-of-words baseline to achieve better overall performance on all image types.

The current paper addresses two of the error types identified in Section 5.2. It would be an interesting extension of this work to test specific techniques that could reduce the remaining error types. For example, co-reference resolution might be beneficial in reducing errors associated with quantification mismatch, in particular as it relates to the number of people in an image description. Techniques for contradiction detection have been developed and tested for other textual inference problems, including recognizing tex-

tual entailment and question answering. These techniques could also apply to retrieval errors caused by real or perceived contradictions between a query and an index description.

REFERENCES

Blei, D. and Jordan, M. (2003). Modeling annotated data. In *the 26th annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, page 127134. ACM Press.

Collins, M. (2002). Discriminative training methods for hidden markov models: theory and experiments with perceptron algorithms. In *EMNLP '02: Proceedings of the ACL-02 conference on Empirical methods in natural language processing*, pages 1–8, Morristown, NJ, USA. Association for Computational Linguistics.

Csurka, G., Clinchant, S., and Popescu, A. (2011). XRCE and CEA LIST's Participation at Wikipedia Retrieval of ImageCLEF 2011. In Petras, V., Forner, P., and Clough, P., editors, *Working Notes of CLEF 2011*, Amsterdam, The Netherlands.

Fahlman, S. E. (2006). Marker-passing inference in the scone knowledge-base system. In *the First Annual International Conference on Knowledge Science, Engineering, and Management (KSEM 2006)*, Guilin, China.

Fan, J., Barker, K., and Porter, B. (2003). The knowledge required to interpret noun compounds. Technical Report UT-AI-TR-03-301, University of Texas at Austin.

Farhadi, A., Hejrati, M., Sadeghi, M. A., Young, P., Rashtchian, C., Hockenmaier, J., and Forsyth, D. (2010). Every picture tells a story: generating sentences from images. In *Proceedings of ECCV 2010*, Greece.

Fellbaum, C. (1998). *WordNet An Electronic Lexical Database*. Bradford Books.

Gangemi, A., Guarino, N., Masolo, C., and Oltramari, A. (2003). Sweetening wordnet with dolce. *AI Magazine*, 24(3):13 – 24.

Grübinger, M., Clough, P., Mller, H., and Deselaers, T. (2006). The IAPR TC-12 benchmark: A new evaluation resource for visual information systems. In *International Workshop OntoImage2006 Language Resources for Content-Based Image Retrieval, held in conjuction with LREC'06*, pages 13–23, Genoa, Italy.

Hickl, A. and Bensley, J. (2007). A discourse commitment-based framework for recognizing textual entailment. In *ACL 2007 Workshop on Textual Entailment and Paraphrasing*, Prague. ACL.

Huiskes, M. J. and Lew, M. S. (2008). The mir flickr retrieval evaluation. In *MIR '08: Proceedings of the 2008 ACM International Conference on Multimedia Information Retrieval*, New York, NY, USA. ACM.

Leong, C. W. and Mihalcea, R. (2011). Measuring the semantic relatedness between words and images. In *IWCS '11 Proceedings of the Ninth International Conference on Computational Semantics*, pages 185–194. Association for Computational Linguistics.

Leong, C. W., Mihalcea, R., and Hassan, S. (2010). Text mining for automatic image tagging. In *Coling 2010: Posters*, pages 647–655, Beijing, China. Coling 2010 Organizing Committee.

Masolo, C., Borgo, S., Gangemi, A., Guarino, N., Oltramari, A., and Schneider, L. (2003). The wonderweb library of foundational ontologies. Technical Report WonderWeb Deliverable D17, National Research Council, Institute of Cognitive Sciences and Technology (ISTC-CNR).

Montazeri, N. and Hobbs, J. R. (2011). Elaborating a knowledge base for deep lexical semantics. In Bos, J. and Pulman, S., editors, *Proceedings of the Ninth International Conference on Computational Semantics (IWCS 2011)*, pages 195–204.

Müller, H., Marchand-Maillet, S., and Pun, T. (2002). The truth about corel - evaluation in image retrieval. In Lew, M. S., Sebe, N., and Eakins, J. P., editors, *Lecture Notes In Computer Science*, volume 2383, pages 38–49. Springer-Verlag, London.

Ponte, J. M. and Croft, W. B. (1998). A language modeling approach to information retrieval. In *SIGIR 1998*, pages 275–281.

Rashtchian, C., Young, P., Hodosh, M., and Hockenmaier, J. (2010). Collecting image annotations using amazons mechanical turk. In *NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazons Mechanical Turk*. Association for Computational Linguistics.

Strohman, T., Metzler, D., Turtle, H., and Croft, W. B. (2005). Indri: A language model-based search engine for complex queries. In *International Conference on Intelligence Analysis (ICIA) (poster)*, McLean, VA.

Tsikrika, T., Popescu, A., and Kludas, J. (2011). Overview of the wikipedia image retrieval task at imageclef 2011. In *Working Notes for the CLEF 2011 Labs and Workshop*, Amsterdam, The Netherlands.

Turtle, H. and Croft, W. B. (1991). Evaluation of an inference network based retrieval model. *Trans. Inf. Syst.*, 9(3):187–222.

von Ahn, L. and Dabbish, L. (2004). Labeling images with a computer game. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 319–326. ACM Press, Vienna, Austria.