

# Classifying Tags using Open Content Resources

Simon Overell<sup>\*</sup>  
Multimedia and  
Information Systems  
Imperial College London  
London, UK  
seo01@doc.ic.ac.uk

Börkur Sigurbjörnsson  
Yahoo! Research  
Barcelona, Spain

Roelof van Zwol  
Yahoo! Research  
Barcelona, Spain

borkur@yahoo-inc.com

roelof@yahoo-inc.com

## ABSTRACT

Tagging has emerged as a popular means to annotate on-line objects such as bookmarks, photos and videos. Tags vary in semantic meaning and can describe different aspects of a media object. Tags describe the content of the media as well as locations, dates, people and other associated meta-data. Being able to automatically classify tags into semantic categories allows us to understand better the way users annotate media objects and to build tools for viewing and browsing the media objects. In this paper we present a generic method for classifying tags using third party open content resources, such as Wikipedia and the Open Directory. Our method uses structural patterns that can be extracted from resource meta-data. We describe the implementation of our method on Wikipedia using WordNet categories as our classification schema and ground truth. Two structural patterns found in Wikipedia are used for training and classification: categories and templates. We apply our system to classifying Flickr tags. Compared to a WordNet baseline our method increases the coverage of the Flickr vocabulary by 115%. We can classify many important entities that are not covered by WordNet, such as, *London Eye*, *Big Island*, *Ronaldinho*, *geocaching* and *wii*.

## Categories and Subject Descriptors

H.3.1 [Information Storage and Retrieval]: Content Analysis and Indexing

## Keywords

Multimedia annotation, Flickr, Categorisation, Wikipedia, User Generated Content

## 1. INTRODUCTION

The collaborative efforts of users participating in social media services such as Flickr, YouTube, Wikipedia, and

<sup>\*</sup>research conducted while on internship at Yahoo! Research Barcelona

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

WSDM '09 Barcelona, Spain

Copyright 2009 ACM 978-1-60558-390-7 ...\$5.00.

Del.icio.us have led to an explosion in user-generated content [7, 8, 20, 23]. A popular way of organizing the content is through folksonomy-style tagging. The flexibility of such a tagging mechanism clearly addresses the user's need to index and navigate the large amount of information that is being generated. As a result, an uncontrolled vocabulary emerges that by far exceeds the semantics of a hierarchical ontology or taxonomy such as WordNet [21]. At the same time, it imposes the problem of semantically categorizing and exploring a potentially infinite tag space.

In this paper we address the task of classifying tags into semantic categories. Consider this problem in terms of an example. Figure 1 shows a Flickr photo annotated with tags. The tags in this example have clear facets: they describe the subject of the photo, indicate where and when it was taken, and what camera was used (this is typical for a tagged photo in Flickr). Our task is to automatically classify these tags in order to help users better understand the image annotation or to enhance photo browsing tools (E.g. [18]). Using WordNet, the tags *skyscraper*, *august* and *vacation* can be classified as representing respectively an artifact, time, and act (See Figure 2 (a)). The tags *chrysler building*, *nyc*, *2006*, *olympus x200* and *william van alen* cannot be matched to WordNet lemmas and as far as WordNet knows their semantic category is thus unknown.

To overcome the limited coverage of WordNet we present a system called ClassTag which classifies tags using structural patterns of Wikipedia articles. We build a classifier to classify Wikipedia articles into semantic categories. We map Flickr tags to Wikipedia articles using anchor texts in Wikipedia. Since we have classified Wikipedia articles we can thus categorize the Flickr tags using the same classification. For example the tag *nyc*, in Figure 1, may be mapped to the anchor text *NYC*. The most common target page for this anchor text is the Wikipedia article *New York City*. Our classifier classifies the Wikipedia article *New York City* as a *location*. Consequently, we can argue that the Flickr tag *nyc* is referring to a *location*.

Figure 2 (b) illustrates how ClassTag can extend the coverage of WordNet. The tags *chrysler building*, *nyc*, *william van alen* and *2006* do not appear in WordNet, however they can be matched to Wikipedia anchor texts and can thus be classified by our system as an artifact, location, person and time, respectively. Finally, the tag *olympus x200* cannot be matched to either a WordNet lemma or a Wikipedia anchor text. Categorizing this type of tag is not in the scope of this paper, but could potentially be covered by incorporating different [open] content resources.



**Tags**  
 chrysler building  
 skyscraper  
 nyc  
 august  
 2006  
 vacation  
 olympus x200  
 william van alen

**Figure 1: Example photo with user-defined tags, extracted from Flickr**

<i>What:</i> skyscraper (artifact)	<i>Where:</i> nyc (location)
<i>When:</i> august (time) vacation (act)	<i>What:</i> chrysler building (artifact) skyscraper (artifact) william van alen (person)
<i>Unknown:</i> chrysler building nyc 2006 olympus x200 william van alen	<i>When:</i> august (time) 2006 (time) vacation (act)
(a) WordNet classification	<i>Unknown:</i> olympus x200
	(b) ClassTag classification

**Figure 2: Classification of the tags in Figure 1 using WordNet (a) and ClassTag (b)**

The contribution of this paper is twofold. First, we present a highly generic system for classifying a hierarchical corpus deploying the structural patterns of the corpus. These patterns can be derived from various open content resources or web directories, such as Wikipedia or the Open Directory<sup>1</sup>. The system can be trained using any classification taxonomy, given a classified lexicon. In this paper we use Wikipedia as our hierarchical corpus and WordNet as our lexicon of classified terms. Second, we present ClassTag – a system for classifying tags using a classified corpus. We train ClassTag such that it will be able to extend the WordNet lexicon using the structural patterns of Wikipedia. Furthermore, we show that ClassTag is a highly tunable system that can be optimized for various applications by trading precision for recall and vice versa.

We compare the performance of ClassTag to a baseline: using WordNet only. We show that by deploying ClassTag we can improve the classified portion of the Flickr vocabulary by 115%. Considering the full volume of Flickr tags – i.e., taking tag frequency into account – we show that with ClassTag we can classify nearly 70% of Flickr tags.

The remainder of this paper starts by presenting the related work in Section 2. Section 3 describes the architecture of the ClassTag system and describes the resources that are used for building and evaluating the system. Section 4 describes the classification of Wikipedia articles: feature selection, and optimization. An empirical evaluation of classifying Wikipedia articles is then presented in Section 5. In Section 6 we describe and evaluate our method for classifying Flickr tags. Finally, we will present our conclusions and future work in Section 7.

<sup>1</sup><http://www.dmoz.org>

## 2. RELATED WORK

There are two approaches to categorizing tags: corpus-based approaches and knowledge-based approaches. In a corpus-based approach statistics and clusters are inferred from the data being classified. In knowledge-based approaches an external knowledge base is used to classify tags. It is also possible to combine these approaches into a semi-supervised method. In this case an external knowledge base is used to classify a portion of the corpus, and properties and statistics of the corpus are used to propagate those classifications. The semi-supervised method is reliant on the initial knowledge-based approach to classify a portion of the corpus. This has motivated us to concentrate on a knowledge-based approach for this paper where categorized Wikipedia articles provide our knowledge base.

In our related work, we begin with a brief examination of how tags can be categorized directly. We then move on to a detailed examination of categorizing articles and anchor texts in Wikipedia.

### 2.1 Categorizing multimedia tags

Schmitz [15] recognizes that people should not have to choose between a hierarchical ontology or unrestricted tags and proposes a probabilistic unsupervised method for inferring an ontology from data. Their results are promising but leave room for improvement.

Rattenbury et al. [13] cluster tags from Flickr using temporal and spatial meta data, to assign event and place semantics. Their approach has a high precision however a large proportion of tags remain unclassified. Sigurbjörnsson and van Zwol [16] map Flickr tags onto WordNet semantic categories using straight forward string matching between Flickr tags and WordNet lemmas. They found that 51.8% of the tags in Flickr can be assigned a semantic category using this mapping.

### 2.2 Categorising Wikipedia anchor texts

There are two mappings which are necessary when categorising Wikipedia anchor texts:

1. Anchor text → Wikipedia article, and
2. Wikipedia article → Category.

The task of mapping an anchor text to a Wikipedia article is studied in several papers. Generally a model of how articles are referred to by specific anchor texts is built from Wikipedia, this model can then be applied to classify entities in an external corpus. This method takes the links in Wikipedia as ground truth. The accuracy of the links and relational statements in Wikipedia are quantified in a study by Weaver et al. [19]. They measure the accuracy of internal links in Wikipedia as 99.5% and the accuracy of relational statements as 97.2%. This method assumes Wikipedia is representative of the external corpus.

Bunescu and Paşca [2] approach this task by learning the textual context of specific categories using a Support Vector Machine. A 55 word window makes up the context of each anchor text, and a mapping is learnt from contexts to the categories of articles (for example the word “conducted” appearing in an entity’s context would provide evidence for the Wikipedia article of the entity being in the category “Composers”). They found substantial improvement taking advantage of the Wikipedia category tree structure over tex-

tual features alone. However, their system is not scalable to the whole of Wikipedia due to the volume of features used.

Cucerzan [5] presents an approach that scales to the whole of Wikipedia. They reduce the amount of contextual information extracted from text by only using links occurring in the first paragraph of a Wikipedia article where a reciprocal link is contained in the target page. Contexts are represented in a vector space and compared to ambiguous entities using the scalar product. Their system disambiguates multiple entities by simultaneously maximizing their category agreement and contextual similarity. The sparsity of category data is partially solved by using Wikipedia list pages to add additional categories.

In this paper, we are concerned with the sub task of mapping Wikipedia articles to categories, and the remainder of this section discusses this task. Various ontologies and gazetteers have been used for this task to provide a set of possible classifications making comparisons between techniques difficult. We consider the WordNet noun syntactic categories as our classification scheme.

Overell and Rüger [12] disregard textual context altogether, instead using only the categories and templates of an article for classification. Their paper is concerned only with Wikipedia articles describing locations; entities in the Getty Thesaurus of Geographic Names form their classification classes. They use a series of heuristics to gather evidence supporting a mapping from an article to a location.

Buscaldi et al. [3] also attempt to classify whether a Wikipedia article describes a location. They use Wikipedia as a filter for geographic terms, classifying simply whether an article refers to a location or not. They extract a set of geographic trigger words from WordNet and compare this set to the text of Wikipedia articles using Dice's coefficient. Any article greater than a set threshold is classified as a location.

Ruiz-Casado et al.'s paper [14] was the first to map Wikipedia articles to WordNet synsets. Mapping Wikipedia articles to WordNet semantic categories (the focus of this paper), can be seen as a sub task of this. They map Wikipedia articles in the Simple English Wikipedia to WordNet lemmas based on string matching the subject of the article. When there is only one lemma for a synset no disambiguation is necessary. However when multiple senses exist, an extended glossary entry for each potential synset is constructed. A synset's extended glossary entry is the original glossary extended with synonyms and hypernyms. These extended glossaries are then mapped into a vector space with tf-idf term weights. The Wikipedia article is mapped into the same feature space and disambiguated as the most similar sense with respect to the dot product of the vectors. In the case of ties, the extended glossary entries are iteratively increased. This method is similar to that presented by Buscaldi et al. [3]: both build a bag-of-words from WordNet for each classification class, which is expected to be similar to the corresponding Wikipedia article.

Suchanek et al. [17] present a method of recognizing Wikipedia articles describing entities (referred to as individuals) and relationships between them using the YAGO ontology. Wikipedia categories are split into administrative, relational, thematic and conceptual classes. The class of a category is identified by parsing its name. Articles in conceptual categories are considered entities, the *type* of the entity is extracted from the conceptual category. WordNet synsets are also mapped into the YAGO ontology with *hyper-*

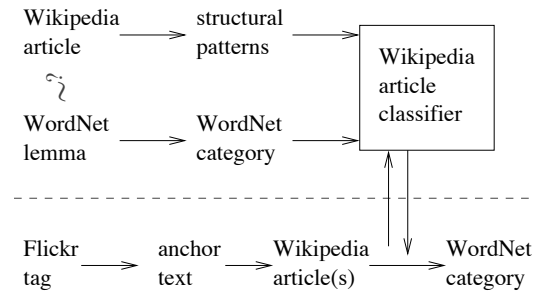


Figure 3: Overview of the ClassTag system.

*nym* relationships mapping to *subclassof* relationships. The *subclassof* hierarchy is further expanded to include Wikipedia categories using heuristic processing of the titles. The YAGO classifications are not directly comparable with the categorizations made in this paper because it does not enforce the same strict hierarchy as WordNet.

The most extensive mapping of Wikipedia articles to WordNet synsets has been built by DBpedia and released under the GNU Free Documentation License [6]. DBpedia stores the structured information contained in Wikipedia templates and uses it as a knowledge base [1]. The entities and relations are stored in an RDF format and linked to other external knowledge sources such as geographic gazetteers and US census data. Their mapping of Wikipedia articles to WordNet synsets was generated by manually associating individual synsets with specific templates.

### 3. THE CLASSTAG SYSTEM

Flickr is one of the world's largest photo sharing sites [8]. Tagging is one of the main means of annotating photos in Flickr, resulting in a large uncontrolled annotation scheme. Being able to classify Flickr tags into semantic categories would allow us to understand better how users annotate photos and to implement tools for better browsing of photos – for example by applying faceted browsing [22].

Sigurbjörnsson and van Zwol [16] mapped Flickr tags onto WordNet semantic categories using straight forward string matching between Flickr tags and WordNet lemmas. They found that 51.8% of the tags in Flickr can be assigned a semantic label using this mapping. They showed that the most common semantic categories of Flickr tags were locations, artifacts, objects, people and groups. In this paper we take their approach as a baseline and show that we can significantly improve the coverage using the ClassTag system. In the remainder of this section we will give a high level overview of the ClassTag system and introduce the resources used for development and evaluation of the system.

#### 3.1 System overview

An overview of the ClassTag system can be found in Figure 3. The system is composed of two components:

1. A classifier for classifying Wikipedia articles using structural patterns as features and WordNet semantic categories as a classification scheme (top part of Figure 3).
2. A pipeline for mapping (Flickr) tags to WordNet semantic categories, using the classifier (bottom part of Figure 3).

We will now describe – at a high level – the two components. Detailed descriptions can be found in Sections 4 and 6.

**Classifying Wikipedia articles.** We build the feature space of our classifier by extracting structural patterns from Wikipedia articles – more precisely using category and template structures. We can map a subset of the Wikipedia articles to WordNet semantic categories using simple string matching between the Wikipedia article titles and WordNet lemmas. We use the successful matches as training instances. Next we use the trained model to classify the remaining Wikipedia articles. More details on the Wikipedia article classifier can be found in Section 4 and its evaluation in Section 5.

**Classifying Flickr tags.** Having classified Wikipedia articles we can use the classification results to classify Flickr tags. We do that using a simple pipeline of mappings. First we map the Flickr tag to Wikipedia anchor texts. Next we map Wikipedia anchor texts to Wikipedia articles. This mapping is the same as described by Mihalcea [11]. Then we can determine the appropriate classification of the Flickr tag using the Wikipedia article classification procedure (described above). The mapping pipeline – including the resolution of ambiguous mappings – is described in more detail in Section 6. The evaluation of the mapping is presented in the same section.

### 3.2 Resources

In the remainder of this section we describe the three corpora used in this paper: Flickr, WordNet and Wikipedia. Our methods are not limited to these corpora, they were chosen as they are in the public domain.

**Flickr.** We use a snapshot of the Flickr database consisting of metadata from 52 million public photos uploaded between 2004 and 2007. The metadata was gathered using the Flickr API [9].

**WordNet.** WordNet is a publicly available English lexicon. Words (lemmas) are mapped to multiple synsets, each synset representing a distinct concept. Synsets are split into 45 semantic categories. Semantic categories are classified further by part-of-speech into adjective, adverb, verb and noun classes [21]. The 25 noun semantic categories are considered as our classification scheme when we classify Flickr tags. WordNet also contains extensive information on the relations between synsets including antonym, hyponym, instance etc. We use the WordNet 3.0 database for the experiments described in this paper.

**Wikipedia.** Wikipedia is the largest reference web site on the Web. The content is collaboratively written by volunteers; to date there are over 2 million articles in the English language version and 10 million articles across all languages [20]. We match Flickr tags with the union of WordNet lemmas and Wikipedia anchor texts to get an upper bound on the number of tags we can classify using ClassTag. In our setting this upper bound is 78.6%.

Each Wikipedia article has a unique title and is assigned to at least one category. Categories form a directed graph and can be navigated in a hierarchy. Articles can optionally contain any number of templates. Templates contain struc-

tured data and article formatting information. Wikipedia templates can transclude other templates forming a similar network to categories. The first paragraph of a Wikipedia article contains the subject of the article in bold and a brief description / definition. Wikipedia contains over 100 million internal links. Each internal link is made up of 2 parts: the target article's title and the anchor text that appears in the source article (these are often the same).

We use the WikiXML download of the English Language Wikipedia provided by University of Amsterdam<sup>2</sup>. This was generated from the 4 November 2006 Wikipedia dump. The dump of Wikipedia contains 1.5 million articles and a total of 3.8 million pages including redirects, categories and templates.

## 4. CLASSIFYING WIKIPEDIA ARTICLES

The aim behind ClassTag was to build a generic and scalable system. It must be generic so that later versions of Wikipedia can be included and full advantage can be taken of new data, also it must be fully applicable to versions of Wikipedia provided in languages other than English and additional open-content resources. Scalability is important because our motivating aim is to maximize coverage. To do this we need to cover the whole of Wikipedia and be able to process updated versions of Wikipedia periodically.

Because of these requirements, we decided to avoid a full semantic interpretation of Wikipedia. Experiments by Buscaldi et al. [3] have shown Wikipedia articles are too heterogeneous to take advantage of shallow textual features, and Bunescu and Paşca [2] show representing the context of every link is difficult to scale.

We follow an approach similar to Overell and R ger [12], and Suchanek et al. [17], using only Wikipedia article meta-data, specifically the structural patterns of categories and templates. Our approach differs from these by using a supervised classifier rather than a set of constructed heuristic rules. This is because we want a scalable approach that will be compatible with future versions of Wikipedia and alternate resources. Articles form our objects, WordNet noun semantic categories form classification classes, and Wikipedia categories and templates form features.

We have selected a Support Vector Machine (SVM) as our classifier. The details of the optimization problem within an SVM will not be discussed here. We use the SVM<sup>light</sup> package for learning and classification<sup>3</sup> and refer the reader to Joachims [10] for details. We train a binary SVM classifier for each class. Each article is classified by each classifier and assigned to the class of the classifier outputting the highest confidence value.

### 4.1 Ground truth

We use the WordNet corpus as a ground truth to train the mapping of Wikipedia articles to WordNet semantic categories. We match WordNet lemmas to Wikipedia article titles and re-directs. The Wikipedia articles are assigned to the class of the matched words. When multiple senses exist for a word, the class of the highest ranked sense is taken. For example the WordNet lemma *Manhattan* is classified as a location in WordNet and is matched to the corresponding Wikipedia article titled *Manhattan*.

<sup>2</sup><http://ilps.science.uva.nl/WikiXML/>

<sup>3</sup><http://svmlight.joachims.org/>

The ground truth is formed of all the Wikipedia articles where the titles match WordNet nouns. For each WordNet semantic category the ground truth is partitioned into a training and test set. The test set is made up of 100 articles from each category (or 10% of the articles from a category where less than 1000 examples exist). The final ground truth consists of 63,664 Wikipedia articles matched to WordNet lemmas, 932 of which are partitioned as a test set.

## 4.2 Sparsity of data

With respect to data sparsity, two problems occur. First is WordNet categories that are under represented in the ground truth; second is articles that have very few features.

### Under represented categories

There are 25 noun syntactic categories in WordNet (not including the “Top” noun category). Of these only 10 are represented with enough articles in Wikipedia matched to WordNet words to train an SVM that will not significantly over fit: act, animal, artifact, food, group, location, object, person, plant and substance. We can additionally include the Time category by artificially adding to WordNet 457 days and years categorized as times. We added the 366 days of the year in numerical day, full month format (e.g. “01 November”) and 121 years in numerical format (from 1887 – 2007 inclusive).

### Sparsity of features

There are a total of 39,516 templates and 167,583 categories in our dump of Wikipedia. The majority of these categories or templates occur in less than 10 articles. We select the categories and templates that occur in more than 50 articles to form our features list, giving us the 25,000 most commonly occurring categories and templates. This is a small enough number of features to allow relatively fast learning and classification for an SVM.

Most articles in Wikipedia have very few categories and templates (in fact the majority of articles have no templates and only one category). Because of this sparsity of features, we wanted to increase the number of categories and templates each article contains. We did this using the category network and template transclusion. As explained in Section 3.2, Wikipedia categories and templates are linked in a directed network. We navigated backwards through the network to increase the number of categories and templates each article has. The disadvantage of this method of enhancing the number of features is that additional features are not independent. For example consider the Wikipedia article *Chrysler Building* describing the art-deco skyscraper in New York City. Suppose we consider traversing 2 category arcs and 1 template arc. The article, “Chrysler Building,” is in categories: *Buildings and Structures in Manhattan* and *Skyscrapers in New York City*; and has one template: *InfoBox Skyscraper*. An additional category arc needs to be traversed adding the parent categories of *Buildings and Structures in Manhattan* and *Skyscrapers in New York City* as 2nd level categories. These additional categories are *Manhattan*, *Buildings and structures in New York City*, *Skyscrapers in the USA*, *Skyscrapers by city* and *Landmarks in New York*. This tree is partially illustrated in Figure 4.

Experiments detailing our choice on how many levels to navigate in these graphs and the weighting function used to

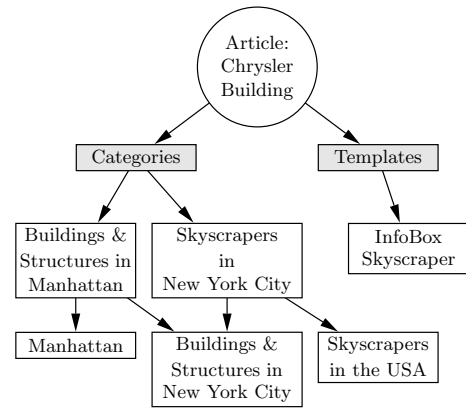


Figure 4: Partial category and template network

determine the scalar values of the features are detailed in Section 4.4.

## 4.3 Removing noise

A significant proportion of Wikipedia categories are actually related to Wikipedia administration rather than article content. ClassTag identifies these categories by navigating every possible path through the category tree back to the root category node for each article. If every path for a category passes through the Wikipedia Administration category we add that category to a *black list* of categories not considered as features. We found 12,271 categories through this method.

Similarly there exist templates that contain only page formatting information and contribute nothing to article content. We identify these by pruning all templates that occur in over 30,000 articles. 11 templates were identified with this method. This is analogous to stop word removal.

## 4.4 System optimisation

As explained in Sections 4.1 and 4.2, our ground truth consists of WordNet nouns matched to Wikipedia articles and our features for classification are 25,000 categories and templates. We partitioned this ground truth into training and test sets to allow us to select the optimum values for variables governing the feature weights. The variables we optimized were:

- the number of arcs traversed in the category network;
- the number of arcs traversed in the template network;
- and the choice of weighting function.

We considered between 0 and 5 arcs for both categories and templates. Taking category arcs as an example: 0 category arcs means we ignore the article’s categories, 1 category arc means we include an article’s categories as features, 2 category arcs means we include the article’s categories and the categories of article’s categories as features etc.

By traversing more arcs we increase the number of features a document contains. The scalar value of each feature is determined by a weighting function. The same weighting function is used by both category and template features. We considered three weighting functions:

- **Term Frequency (tf):** The scalar value of each feature is the number of times it occurs for this article.

Feature	tf	tf-idf	tf-il
c:Buildings and Structures in Manhattan	1	0.51	1
c:Skyscrapers in New York City	1	0.53	1
t:InfoBox Skyscraper	1	0.49	1
c: <i>Manhattan</i>	1	0.48	0.5
c: <i>Buildings and structures in New York City</i>	2	1.16	1
c: <i>Skyscrapers in the USA</i>	1	0.59	0.5
c: <i>Skyscrapers by city</i>	1	0.59	0.5
c: <i>Landmarks in New York</i>	1	0.60	0.5

Table 1: Weighting functions example

- **Term Frequency - Inverse Document Frequency (tf-idf):** The scalar value of each feature is the number of times it occurs for this article divided by the log of the number of times it occurs in the document collection.
- **Term Frequency - Inverse Layer (tf-il):** The scalar value of each feature is the number of times it occurs for this article divided by the number of arcs that had to be traversed in the category / template network to reach it.

Referring back to the *Chrysler Building* example in Figure 4. Table 1 shows how the scalar values of the features vary with the choice of weighting function. The *c* or *t* prefix specifies whether a feature is a category or a template. The features added by traversing an additional category arc are shown in *italics*. Notice how the problem of data sparsity has been reduced, as we have added an additional 5 features to a document that originally had only 3.

#### 4.4.1 Selecting variables

We performed an exhaustive search of every combination of variables evaluated against our ground truth test set. Our criteria for choosing the *best* method was that it must achieve a precision of more than 80% for each category. 80% was selected as an acceptable precision with large recall. This is a sufficient level of precision for our application, which also allows us to categorize significantly more articles than WordNet alone. Of the methods that achieved this we selected the one with the greatest  $F_1$ -measure. The optimal results were achieved traversing 3 arcs for both categories and templates, and weighting function tf-il.

Table 2 shows how, with respect to the *best* method, varying the number of arcs traversed in the category and template networks, and changing the weighting function affects the precision and  $F_1$ -measure. Notice that there is in fact minimal difference in performance as template arcs and the weighting function vary. For categories, when no category data is used, the data is too sparse to perform any correct classification (this is due to many articles having no templates and only one or two categories). Conversely when more than four category arcs are traversed the data becomes far too noisy.

We conclude that the features chosen are fairly robust provided the value selected for category arcs traversed produces training data that is neither too sparse nor too noisy.

#### 4.4.2 Selecting a threshold for classifying

The SVM binary classifiers output the values of their decision functions. The output of the decision function can be

Variable	Value	Prec	$F_1$
Cat.	0	0%	0
	1	87.1%	0.694
	2	87.3%	0.694
	3	<b>87.0%</b>	<b>0.696</b>
	4	0%	0
Arcs	5	0%	0
	0	86.7%	0.695
	1	86.8%	0.693
	2	86.9%	0.696
	3	<b>87.0%</b>	<b>0.696</b>
Temp.	4	87.0%	0.696
	5	86.9%	0.692
Weight. Func.	tf	86.7%	0.623
	tf-idf	87.6%	0.668
	tf-il	<b>87.0%</b>	<b>0.696</b>

Table 2: Varying feature values

interpreted as the confidence with which an article is correctly classified as a member of a category. If there exists no prior knowledge about the distribution of the data one can simply classify articles as the category of the classifier that outputs the greatest value above 0. If no classifiers output a value above 0, one can consider the article unclassified.

However if there exists prior knowledge about the data, for example if one knows a significant proportion of Wikipedia articles can be classified as one of our 11 categories, the threshold could be set lower than 0. On the other hand, if one has prior knowledge that the data is particularly noisy, the threshold could be set greater than 0.

We performed a training experiment where 250 Wikipedia articles were selected at random. Each article was classified as the WordNet semantic category of the classifier outputting the greatest decision function. An assessor then marked each classification as correct or incorrect by hand. We then varied the threshold for the minimum acceptable output value between -1 and 1. Articles where the maximum output value from a classifier were below the threshold were considered unclassified. Figure 5 shows how precision, recall and the  $F_1$ -measure vary with the threshold value. As our motivation is to maximize the coverage of tags we select the method that maximizes the recall given a minimum acceptable precision. We have selected the minimum acceptable precision across all categories as 90%; this gives a recall of 51% and a threshold value of -0.4. Were we to maximize the precision instead of recall within an allowable precision range, we would take the threshold as 0.3 giving a precision of 98% and recall of 33%.

Figure 6 shows how varying the threshold affects the proportion of articles classified and the proportion of ambiguous articles (articles with multiple positive classifications). When the threshold is -0.4, 39% of all articles are classified. 5.7% of those classified are ambiguous. When the threshold is 0.3, 21% of all articles are classified, 0.5% of which are ambiguous.

## 5. EVALUATION OF WIKIPEDIA CLASSIFICATION

In the following experiment we compare the performance of ClassTag with the performance of the mapping of Wikipedia articles to WordNet synsets provided for download from DBpedia [6].

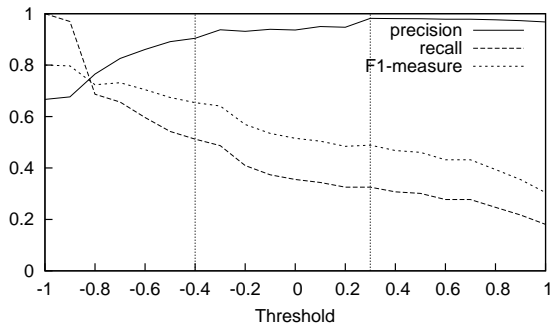


Figure 5: Threshold – F<sub>1</sub>-Measure

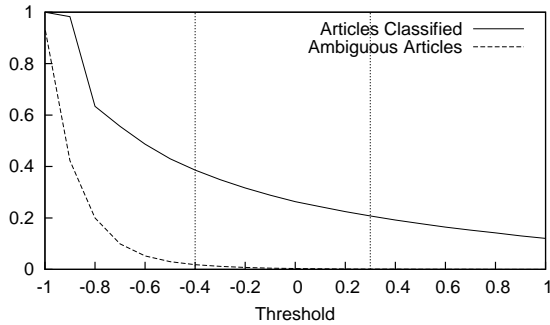


Figure 6: Threshold – Prop. of articles Classified

## 5.1 Experimental setup

An evaluation set of 300 Wikipedia articles were selected at random from the union of articles classified by DBpedia and articles classified by ClassTag. ClassTag classifies a total of 664,770 Wikipedia articles. DBpedia classifies a total of 338,061 articles, however we only consider the 206,623 articles that also exist in our dump of Wikipedia taken on 4 November 2006<sup>4</sup>. ClassTag classifies 258 of the articles in the evaluation set, while DBpedia classifies 88 articles. There is an overlap of 38 articles<sup>5</sup>.

DBpedia’s classifications are optimised for precision while ClassTag is optimized to maximise recall given a minimum precision requirement. We have produced an alternative version of ClassTag, which we shall refer to as ClassTag<sup>+</sup>, with the threshold for the SVM decision function set to 0.3 (as detailed in Section 4.4.2). ClassTag<sup>+</sup> is optimized for precision. ClassTag<sup>+</sup> classifies a total of 344,539 articles and 125 articles in the evaluation set.

### 5.1.1 Assessments

Three assessors assessed the Wikipedia articles. A randomly selected 50 articles were assessed by all assessors to measure assessor agreement. All remaining articles were only assessed by a single assessor. Assessments were performed blind. The assessors had no knowledge of which systems had classified the article or what the classifications were. The evaluation interface presented the user with the Wikipedia article that had been classified, a checkbox for each of the 25 semantic categories, and the semantic category brief descriptions taken from the WordNet web site [21].

<sup>4</sup>DBpedia.org’s data is based on a dump of Wikipedia taken from 16 July 2007.

<sup>5</sup>The evaluation set and classifications can be obtained for academic purposes by contacting the authors.

	ClassTag	DBpedia	ClassTag <sup>+</sup>
Prec.	72%	58%	<b>86%</b>
Recall	<b>81%</b>	17%	38%
Acc.	<b>62%</b>	16%	36%

Table 3: System evaluation results

Assessors were told to select all semantic categories they considered as correct classifications for each article.

### 5.1.2 Assessor agreement

We measure 2 values for assessor agreement: *partial agreement* and *total agreement*. With partial agreement there exists an article classification that all assessors agree on. Total agreement is where assessors agree on all classifications. For **86%** of articles assessors had partial agreement. For **78%** of articles assessors had total agreement.

## 5.2 Results

Our experimental results are reported in Table 3. In previous papers classifying Wikipedia articles only the accuracy of the classified set is reported [14, 17]. As we have built our sample set from the pool of articles classified by both ClassTag and DBpedia we can also consider articles not classified. We use the standard information retrieval measures of precision, recall and accuracy. Precision can be considered the accuracy of the classified set.

An assessor was selected at random, and their assessments were considered ground truth for the Wikipedia articles with multiple judgements. As we consider a system classification correct if it matches *any* of assessor classifications, the gold standard accuracy can be considered equal to the assessor partial agreement: 86% (This is the point where the judgements provided by the system become as accurate as those provided by a human). ClassTag<sup>+</sup> reaches gold standard precision of 86% but at a significant recall trade off, classifying less than half as many articles as ClassTag. ClassTag has a particularly high recall of 81%.

### 5.2.1 Per category results

The top four most commonly occurring categories in the evaluation set were (in order): person, location, artifact and group. Figure 7 shows the per-category precision of ClassTag and ClassTag<sup>+</sup>. *Artifact* is noticeably worse than the other three categories (over 12% lower than the second lowest) with a precision of 63.3%. This difference is even more pronounced for ClassTag<sup>+</sup> where the precision of the person, location and group categories significantly increases to between 89% and 100%, while the precision of the artifact category barely changes. We attribute the artifact category’s low precision to the huge variation in the types of artifacts in Wikipedia. WordNet defines an artifact as “nouns denoting man-made objects,” this ranges from a paper clip to the Empire State Building.

### 5.2.2 Summary

In Section 4 we identified the goal of ClassTag to be a generic, scalable system that maximizes recall while keeping as high a precision as possible. ClassTag classifies 39% of articles in Wikipedia with a precision of 72%. The system is flexible enough that we can also optimize for precision as demonstrated with ClassTag<sup>+</sup>. In our evaluation, both ClassTag and ClassTag<sup>+</sup> out performed DBpedia in

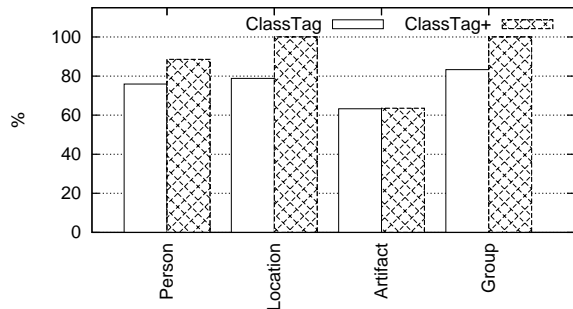


Figure 7: Per Category Precision

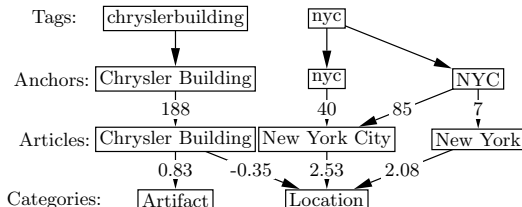


Figure 8: Tag  $\rightarrow$  Category Example

all our performance measures. For example with respect to precision, ClassTag outperforms DBpedia by 14%, while ClassTag<sup>+</sup> outperforms DBpedia by 30%.

## 6. CLASSIFYING FLICKR TAGS

We will now turn our attention to our original task outlined in Section 1 – categorizing Flickr tags according to semantic categories. We start by discussing the mapping of tags to categories in Section 6.1, and evaluate this mapping in Section 6.2. Finally, we discuss multilingual tags and context in Sections 6.3 and 6.4.

### 6.1 Mapping tags to semantic categories

Figure 8 displays the steps taken by ClassTag when mapping a tag to a semantic category. The mapping consists of three steps:

1. Tag  $\rightarrow$  Anchor text,
2. Anchor text  $\rightarrow$  Wikipedia article, and
3. Wikipedia article  $\rightarrow$  Category.

The tags from Figure 1 with the Chrysler Building example are used to illustrate the process. There are 4 tags that are covered by Wikipedia but not by WordNet: *chrysler building*, *nyc*, *william van alen* and *2006*. *2006* is covered by our extension of the time category of WordNet, leaving *chrysler building*, *nyc* and *william van alen* to be categorised by ClassTag. In the following paragraphs we will demonstrate how the tags *chrysler building* and *nyc* are mapped to semantic categories. Two of the mappings are weighted. The weights on the “Anchor text  $\rightarrow$  Wikipedia article” arcs represent the frequency of the mapping (e.g. the number of times “NYC” refers to “New York City” in Wikipedia). The weights on the “Wikipedia article  $\rightarrow$  Category” arcs represent the output of the SVM decision function.

Mapping from tags to anchors is a straight forward string matching process [2, 5, 11, 12]. Some ambiguity is intro-

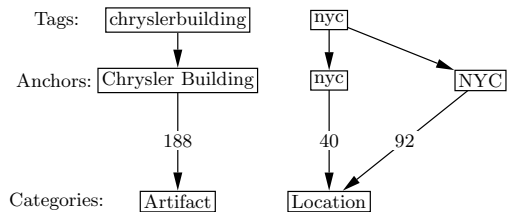


Figure 9: Tag  $\rightarrow$  Category Example (reduced ambiguity)

	WordNet	ClassTag	Diff.
Vocabulary	89,902	193,444	+115%
Vocabulary (%)	2.4%	5.2%	
Full volume	106,215,397	130,049,982	+22%
Full volume (%)	56.5%	69.2%	

Table 4: Coverage of the Flickr tags, both in terms of vocabulary coverage and full volume coverage.

duced because tags are commonly lower case and often contain no white space or punctuation.

Mapping from Wikipedia articles to categories also introduces relatively little ambiguity. As detailed in Section 4.4.2, only 5.7% of classified articles result in multiple positive classifications. In these cases ClassTag simply classifies the article as the category corresponding to the classifier with the greatest confidence value.

Mapping from anchors to Wikipedia articles is more complex. To reduce the number of mappings considered we remove *outlier* mappings. We define an outlier mapping as a mapping from an anchor text to an article that is referred to less than 5 times, or a mapping that makes up less than 5% of the total mappings from a specific anchor. After removing outliers we further reduce the ambiguity by first categorizing Wikipedia articles and grouping all articles in the same category together.

Continuing with the running example: the “Chrysler Building” article is disambiguated as an artifact, since that mapping has the largest weight. The “Chrysler Building” anchor is unambiguous, and can now map straight to artifact. The “NYC” anchor is ambiguous in Figure 8, but both articles map to location so can be combined into a single mapping. The resulting mapping is displayed in Figure 9. Observe that the tag *chrysler building* is disambiguated as an artifact and *nyc* a location.

### 6.2 Evaluation

In Section 3 we introduced a baseline approach which used a mapping from Flickr tags to WordNet semantic categories using string matching between Flickr tags and WordNet lemmas [16]. In this section we will show the results of extending this baseline approach with our ClassTag system to improve the coverage of the semantic labeling<sup>6</sup>.

Table 4 shows the performance of our WordNet baseline approach and the extension using the ClassTag system, in terms of how many of Flickr tags they are able to classify. Using ClassTag to extend the baseline WordNet coverage we increase the coverage of the vocabulary by 115% – from

<sup>6</sup>Here we use a slightly improved baseline to the one described in [16]. Our improvement includes the categorization of plural nouns.



	WordNet	ClassTag	Diff.
Act	4,445	8,694	96%
Animal	6,480	9,248	43%
Artifact	12,648	33,320	163%
Food	2,748	3,665	33%
Group	2,302	7,096	208%
Location	4,035	30,444	654%
Object	1,898	7,265	283%
Person	15,719	61,696	292%
Plant	7,394	7,421	0%
Substance	2,342	2,903	24%
Time	1,173	5,715	387%

**Table 5: Coverage of the Flickr vocabulary in terms of different semantic categories.**

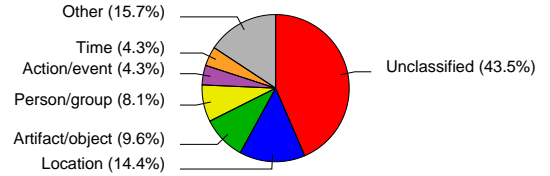
Category	Examples
<b>Act</b>	Triathlon, geocaching, mountain biking, kendo.
<b>Animal</b>	Jack Russell Terrier, Australian Shepherd.
<b>Artifact</b>	Notre Dame, London Eye, Sagrada Familia, nikon, nokia, pentax, leica, wii, 4x4.
<b>Food</b>	BBQ, Churrasco, Japanese food, Ramen, Asado.
<b>Group</b>	Live8, G8, NBA, SIGGRAPH, Tate Modern.
<b>Location</b>	NYC, Philly, Phuket, Big Island, Nottingham.
<b>Object</b>	Blue Mountains, Point Reyes, Half Dome, Lake Titicaca, Jungfrau.
<b>Person</b>	Norman Foster, Ronaldinho, Britney Spears, Chris, Alex, Dave, Emily, Laura, Lisa, Jen.
<b>Plant</b>	Guadua, Chelsea Flowershow, red rose.
<b>Substance</b>	Wheatpaste, O2, biodiesel
<b>Time</b>	New Year's Eve, 4th of July, Valentine's day.

**Table 6: Examples of tags covered by ClassTag but not covered by WordNet.**

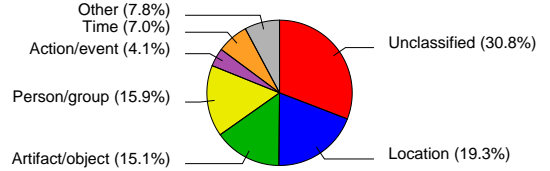
89,902 to 193,444 unique tags. Measured in terms of the full volume of tags – i.e., taking tag frequency into account – we can classify 69.2% of the Flickr tags. This is an improvement of 22% compared to our WordNet baseline.

Let us now look in more detail at the types of tags the ClassTag system can classify but were not classified by WordNet. Table 5 shows the coverage of Flickr tags in terms of different semantic categories. We see that the ClassTag system improves coverage considerably for all types of tags, except plants. The largest absolute increase in coverage is for the Person category where coverage is increased by almost 46,000 unique tags. For the Location and Artifact categories the coverage is, respectively, increased by over 26,000 and 20,000 unique tags. Having a better coverage of locations, artifacts, and people is certainly useful for any system analyzing multimedia annotations. As illustrated in Figure 2 the extended coverage of ClassTag enables us to give a more informed presentation of Flickr tags.

Let us now look at some examples of tags covered by ClassTag that were not covered by WordNet, Table 6 shows some examples of such tags. We see the ClassTag system is able to add some frequently photographed artifacts and objects such as Notre Dame, London Eye, Lake Titicaca and Half Dome; as well as some less famous ones such as Hundertwasserhaus and Strokkur. The ClassTag system is able to classify abbreviations of popular locations such as NYC and Philly. Furthermore, we add some popular tourist locations such as Phuket and Big Island, neither of which were covered by WordNet. Last but not least, ClassTag is able to classify correctly names of famous people such as Norman Foster and Ronaldinho, as well as a large set of frequent first



(a) Using our baseline WordNet based approach.



(b) Using the ClassTag system.

**Figure 10: Classification of Flickr tags. (a) Using our baseline WordNet based approach. (b) Using the ClassTag system.**

names such as Chris, Alex, Emily, Laura, etc.

We have shown that we can considerably extend our coverage of Flickr tag classification using ClassTag. We will now demonstrate how this can be used to provide improved analysis of Flickr tagging behavior. Figure 10 shows the distribution of Flickr tags over different semantic categories – both using our baseline system and the ClassTag system. When comparing the two charts we see the effect of being able to classify a larger portion of tags. We believe that this gives us a better understanding of the way people annotate their photos. Using the baseline system the size of the Location, Artifact, and People classes were underestimated since we were not able to recognize locations such as NYC, Phuket and Big Island; artifacts such as Notre Dame, London Eye and Starbucks; and common first names such as Alex, Emily, Laura etc.

### 6.3 A discussion of multilingual classification

Flickr tags are unrestricted and as such appear in a mixture of languages. Clough et al. [4] provide an estimate of the language distribution of Flickr tags. They estimate approximately 80% of tags are in English, 7% in German and 6% in Dutch. We expect a large portion of the 21.4% of unclassifiable tags (tags not covered by Wikipedia anchors  $\cup$  WordNet lemmas) fall into this category.

Wikipedia is currently available in 253 languages, the top 14 of which have over 100,000 articles each [20]. ClassTag contains no language specific elements. There are two possible methods of creating an alternate language classification of Wikipedia articles:

1. We can run ClassTag with an alternate language version of Wikipedia and a corresponding lexicon; or
2. the English language classifications we generate can be translated into an alternate language using Wikipedia's Interlanguage links.

Both methods would allow us to further increase the coverage of Flickr tags.

## 6.4 A discussion of context

Bunescu and Paşca [2] and Cucerzan [5] have worked on context-aware disambiguation of free text based on models built from Wikipedia. Photos are generally annotated with very few tags; because of this, there is often very little context available when classifying images.

Neither of the methods described by Bunescu and Paşca or Cucerzan are applicable in these circumstances. The method described by Bunescu and Paşca specifies a 55 word window for disambiguation, while Cucerzan attempts to disambiguate all named entities in a document simultaneously, shrinking their window only as small as the sentence level when ties occur.

We consider the area of context-aware disambiguation an essential part of our future work; when given a tag with multiple possible classifications such as “Java” we would like to be able to classify whether it refers to a location, food or artifact based on context. However, we do not consider this a priority as many tags exist with little or no context (Sigurbjörnsson and van Zwol [16] observe that 64% of tagged photos have 3 tags or less).

## 7. CONCLUSIONS

In this paper we have presented a method of categorising Flickr tags as WordNet semantic categories. We do this by first categorizing Wikipedia articles and then mapping Flickr tags onto these categorized articles. Our Wikipedia article categorization method can be configured to optimize either precision or recall. In either configuration this method outperforms the categorizations provided by DBpedia in our blind evaluation. When optimized for recall nearly **40%** of Wikipedia articles are classified with a precision of **72%**. When optimized for precision **21%** of Wikipedia articles are classified with a precision of **86%**.

We have shown that with our approach we can categorize a **115%** larger share of the Flickr vocabulary, compared to a baseline using WordNet. Consequently we are able to categorize **69.2%** of the total volume of Flickr tags – i.e., when we take tag frequency into account. We can classify many important entities that are not covered by WordNet, such as, *London Eye*, *Big Island*, *Ronaldinho*, *geocaching* and *wii*.

In future work we want to exploit this improved categorization of Flickr tags to implement tools that allow semantic browsing of Flickr content. We also plan to continue improving the precision of ClassTag and its coverage of Flickr tags. To do this, we would like to investigate how other Wikipedia meta-data and open content resources, such as the Open Directory, can be exploited.

## Acknowledgments

This research is partially supported by the European Union under contract FP6-045032, “Search Environments for Media — SEMEDIA” (<http://www.semedia.org>). Attendance to WSDM partially supported by True Knowledge.

## 8. REFERENCES

- [1] S. Auer and J. Lehmann. What have Innsbruck and Leipzig in common? In *Proc. of ESWC*, pages 503–517, 2007.
- [2] R. Bunescu and M. Paşca. Using encyclopedic knowledge for named entity disambiguation. In *Proc. of EACL*, pages 9–16, 2006.
- [3] D. Buscaldi, P. Rosso, and P. García. Inferring geographic ontologies from multiple resources for geographical information retrieval. In *Proc. of the SIGIR workshop on GIR*, pages 53–55, 2006.
- [4] P. Clough, A. Al-Maskari, and K. Darwish. Providing multilingual access to Flickr for arabic users. In *Proc. of CLEF*, 2006.
- [5] S. Cucerzan. Large-scale named entity disambiguation based on Wikipedia data. In *Proc. of EMNLP-CoNLL*, pages 708–716, 2007.
- [6] DBpedia. <http://dbpedia.org/>. Accessed 14 April 2008.
- [7] Delicious. <http://del.icio.us/>. Accessed 14 April 2008.
- [8] Flickr. <http://www.Flickr.com/>. Accessed 14 April 2008.
- [9] FlickrAPI. <http://www.flickr.com/services/api/>. Accessed 14 April 2008.
- [10] T. Joachims. Making large-scale SVM learning practical. In *Advances in Kernel Methods - Support Vector Learning*, pages 41–56, 1998.
- [11] R. Mihalcea. Using wikipedia for automatic word sense disambiguation. In *Proc. of NAACL*, pages 196–203, 2007.
- [12] S. Overell and S. Rüger. Geographic co-occurrence as a tool for GIR. In *Proc. of the CIKM workshop on GIR*, 2007.
- [13] T. Rattenbury, N. Good, and M. Naaman. Towards automatic extraction of event and place semantics from flickr tags. In *Proc. of SIGIR*, pages 103–110, 2007.
- [14] M. Ruiz-Casado, E. Alfonseca, and P. Castells. Automatic assignment of Wikipedia encyclopedic entries to WordNet synsets. In *Proc. of AWIC*, pages 380–386, 2005.
- [15] P. Schmitz. Inducing an ontology from flickr tags. In *Proc. of the Workshop on Collaborative Web Tagging at WWW’06*, 2006.
- [16] B. Sigurbjörnsson and R. van Zwol. Flickr tag recommendation based on collective knowledge. In *Proc. of WWW’08*, pages 327–336, 2008.
- [17] F. Suchanek, G. Kasneci, and G. Weikum. YAGO: A core of semantic knowledge unifying WordNet and Wikipedia. In *Proc. of WWW’07*, pages 697–706, 2007.
- [18] TagExplorer. <http://sandbox.yahoo.com/TagExplorer>. Accessed 14 November 2008.
- [19] G. Weaver, B. Strickland, and G. Crane. Quantifying the accuracy of relational statements in Wikipedia: A methodology. In *Proc. of JCDL*, pages 358–358, 2006.
- [20] Wikipedia. <http://www.wikipedia.org/>. Accessed 14 April 2008.
- [21] WordNet. <http://wordnet.princeton.edu/>. Accessed 14 April 2008.
- [22] P. Yee, K. Swearingen, K. Li, and M. Hearst. Faceted metadata for image search and browsing. In *Proc. of ACM CHI*, pages 401–408, 2003.
- [23] YouTube. <http://youtube.com/>. Accessed 14 April 2008.