Coarse to Fine Grained Sense Disambiguation in Wikipedia

Hui Shen
School of EECS
Ohio University
Athens, OH 45701, USA
hui.shen.1@ohio.edu

Razvan Bunescu
School of EECS
Ohio University
Athens, OH 45701, USA
bunescu@ohio.edu

Rada Mihalcea
Department of CSE
University of North Texas
Denton, TX 76203, USA
rada@cs.unt.edu

Abstract

Wikipedia articles are annotated by volunteer contributors with numerous links that connect words and phrases to relevant titles. Links to general senses of a word are used concurrently with links to more specific senses, without being distinguished explicitly. We present an approach to training coarse to fine grained sense disambiguation systems in the presence of such annotation inconsistencies. Experimental results show that accounting for annotation ambiguity in Wikipedia links leads to significant improvements in disambiguation.

1 Introduction and Motivation

The vast amount of world knowledge available in Wikipedia has been shown to benefit many types of text processing tasks, such as coreference resolution (Ponzetto and Strube, 2006; Haghighi and Klein, 2009; Bryl et al., 2010; Rahman and Ng, 2011), information retrieval (Milne, 2007; Li et al., 2007; Potthast et al., 2008; Cimiano et al., 2009), or question answering (Ahn et al., 2004; Kaisser, 2008; Ferrucci et al., 2010). In particular, the user contributed link structure of Wikipedia has been shown to provide useful supervision for training named entity disambiguation (Bunescu and Pasca, 2006; Cucerzan, 2007) and word sense disambiguation (Mihalcea, 2007; Ponzetto and Navigli, 2010) systems. Articles in Wikipedia often contain mentions of concepts or entities that already have a corresponding article. When contributing authors mention an existing Wikipedia entity inside an article, they are required to link at least its first mention to the corresponding article, by using *links* or *piped links*. Consider, for example, the following Wiki source annotations: *The [[capital city | capital]] of Georgia is [[Atlanta]]*. The bracketed strings identify the title of the Wikipedia articles that describe the corresponding named entities. If the editor wants a different string displayed in the rendered text, then the alternative string is included in a piped link, after the title string. Based on these Wiki processing rules, the text that is rendered for the aforementioned example is: *The capital of Georgia is Atlanta*.

Since many words and names mentioned in Wikipedia articles are inherently ambiguous, their corresponding links can be seen as a useful source of supervision for training named entity and word sense disambiguation systems. For example, Wikipedia contains articles that describe possible senses of the word "capital", such as CAPITAL CITY, CAPITAL (ECONOMICS), FINANCIAL CAPITAL, or HUMAN CAPITAL, to name only a few. When disambiguating a word or a phrase in Wikipedia, a contributor uses the context to determine the appropriate Wikipedia title to include in the link. In the example above, the editor of the article determined that the word "capital" was mentioned with the political center meaning, consequently it was mapped to the article CAPITAL CITY through a piped link.

In order to use Wikipedia links for training a WSD system for a given word, one needs first to define a sense repository that specifies the possible meanings for that word, and then use the Wikipedia links to create training examples for each sense in the repository. This approach might be implemented using the following sequence of steps:

*Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task*, pages 22–31, Atlanta, Georgia, June 13-14, 2013.

In global climate models, the state and properties of the [[atmosphere]] are specified at a number of discrete locations
General = **Atmosphere**; *Specific* = Atmosphere of Earth; *Label* = A → A(S) → AE

The principal natural phenomena that contribute gases to the [[Atmosphere of Earth|atmosphere]] are emissions from volcanoes
General = Atmosphere; *Specific* = **Atmosphere of Earth**; *Label* = A → A(S) → AE

An aerogravity assist is a spacecraft maneuver designed to change velocity when arriving at a body with an [[atmosphere]]
General = **Atmosphere**; *Specific* = Atmosphere ▷ *generic*; *Label* = A → A(G)

Assuming the planet's [[atmosphere]] is close to equilibrium, it is predicted that 55 Cancri d is covered with water clouds
General = **Atmosphere**; *Specific* = Atmosphere of Cancri ▷ *missing*; A → A(G)

Figure 1: Coarse and fine grained sense annotations in Wikipedia (**bold**). The proposed hierarchical *Label* (right). A(S) = Atmosphere (S), A(G) = Atmosphere (G), A = Atmosphere, AE = Atmosphere of Earth.

1. Collect all Wikipedia titles that are linked from the ambiguous anchor word.
2. Create a repository of senses from all titles that have sufficient support in Wikipedia i.e., titles that are referenced at least a predefined minimum number of times using the ambiguous word as anchor.
3. Use the links extracted for each sense in the repository as labeled examples for that sense and train a WSD model to distinguish between alternative senses of the ambiguous word.

Taking the word "atmosphere" as an example, the first step would result in a wide array of titles, ranging from the general Atmosphere and its instantiations Atmosphere of Earth or Atmosphere of Mars, to titles as diverse as Atmosphere (unit), Mood (psychology), or Atmosphere (music group). In the second step, the most frequent titles for the anchor word "atmosphere" would be assembled into a repository $\mathcal{R}$ = {Atmosphere, Atmosphere of Earth, Atmosphere of Mars, Atmosphere of Venus, Stellar Atmosphere, Atmosphere (unit), Atmosphere (music group)}. The classifier trained in the third step would use features extracted from the context to discriminate between word senses.

This Wikipedia-based approach to creating training data for word sense disambiguation has a major shortcoming. Many of the training examples extracted for the title Atmosphere could very well belong to more specific titles such as Atmosphere of Earth or Atmosphere of Mars. Whenever the word "atmosphere" is used in a context with the sense of "a layer of gases that may surround a material body of sufficient mass, and that is held in place by the gravity of the body," the contributor has the option of adding a link either to the title Atmosphere that describes this general sense of the word, or to the title of an article that describes the atmosphere of the actual celestial body that is referred in that particular context, as shown in the first 2 examples in Figure 1. As shown in bold in Figure 1, different occurrences of the same word may be tagged with either a general or a specific link, an ambiguity that is pervasive in Wikipedia for words like "atmosphere" that have general senses that subsume multiple, popular specific senses. There does not seem to be a clear, general rule underlying the decision to tag a word or a phrase with a general or specific sense link in Wikipedia. We hypothesize that, in some cases, editors may be unaware that an article exists in Wikipedia for the actual reference of a word or for a more specific sense of the word, and therefore they end up using a link to an article describing the general sense of the word. There is also the possibility that more specific articles are introduced only in newer versions of Wikipedia, and thus earlier annotations were not aware of these recent articles. Furthermore, since annotating words with the most specific sense available in Wikipedia may require substantial cognitive effort, editors may often choose to link to a general sense of the word, a choice that is still correct, yet less informative than the more specific sense.

2 Annotation Inconsistencies in Wikipedia

In order to get a sense of the potential magnitude of the general vs. specific sense annotation ambiguity, we extracted all Wikipedia link annotations for the words “atmosphere”, “president”, “game”, “dollar”, “diamond” and “Corinth”, and created a special subset from those that were labeled by Wikipedia editors with the general sense links Atmosphere, President, Game, Dollar, Diamond, and Corinth, respectively. Then, for each of the 7,079 links in this set, we used the context to manually determine the corresponding more specific title, whenever such a title exists in Wikipedia. The statistics in Tables 1 and 2 show a significant overlap between the general and specific sense categories. For example, out of the 932 links from “atmosphere” to Atmosphere that were extracted in total, 518 were actually about the Atmosphere of Earth, but the user linked them to the more general sense category Atmosphere. On the other hand, there are 345 links to Atmosphere of Earth that were explicitly made by the user. We manually assigned *general* links (G) whenever the word is used with a generic sense, or when the reference is not available in the repository of titles collected for that word because either the more specific title does not exist in Wikipedia or the specific title exists, but it does not have sufficient support – at least 20 linked anchors – in Wikipedia. We grouped the more specific links for any given sense into a special category suffixed with (S), to distinguish them from the general links (generic use, or missing reference) that were grouped into the category suffixed with (G).

For many ambiguous words, the annotation inconsistencies appear when the word has senses that are in a subsumption relationship: the Atmosphere of Earth is an instance of Atmosphere, whereas a Stellar Atmosphere is a particular type of Atmosphere. Subsumed senses can be identified automatically using the category graph in Wikipedia. The word “Corinth” is an interesting case: the subsumption relationship between Ancient Corinth and Corinth appears because of a temporal constraint. Furthermore, in the case of the word “diamond”, the annotation inconsistencies are not caused by a subsumption relation between senses. Instead of linking to the Diamond (gemstone) sense, Wikipedia contributors often link to the related Diamond sense indicating the mineral used in the gemstone.

A supervised learning algorithm that uses the extracted links for training a WSD classification model to distinguish between categories in the sense repository assumes implicitly that the categories, and hence their training examples, are mutually disjoint. This assumption is clearly violated for words like “atmosphere,” consequently the learned model will have a poor performance on distinguishing between the overlapping categories. Alternatively, we can say that sense categories like Atmosphere are ill defined, since their supporting dataset contains examples that could also belong to more specific sense categories such as Atmosphere of Earth.

atmosphere	Size
Atmosphere	932
Atmosphere (S)	*559*
Atmosphere of Earth	*518*
Atmosphere of Mars	*19*
Atmosphere of Venus	*9*
Stellar Atmosphere	*13*
Atmosphere (G)	*373*
Atmosphere of Earth	345
Atmosphere of Mars	37
Atmosphere of Venus	26
Stellar Atmosphere	29
Atmosphere (unit)	96
Atmosphere (music group)	104
president	Size
President	3534
President (S)	*989*
Chancellor (education)	*326*
President of the United States	*534*
President of the Philippines	*42*
President of Pakistan	*27*
President of France	*22*
President of India	*21*
President of Russia	*17*
President (G)	*2545*
Chancellor (education)	210
President of the United States	5941
President of the Philippines	549
President of Pakistan	192
President of France	151
President of India	86
President of Russia	101

Table 1: Wiki (Caps) and manual (*italics*) annotations.

We see two possible solutions to the problem of inconsistent link annotations. In one solution, specific senses are grouped together with the subsuming general sense, such that all categories in the resulting repository become disjoint. For “atmosphere”, the general category Atmosphere would be augmented to contain all the links previously annotated

dollar	Size
DOLLAR	379
Dollar (S)	*231*
United States dollar	*228*
Canadian dollar	*3*
Australian dollar	*1*
Dollar (G)	*147*
UNITED STATES DOLLAR	3516
CANADIAN DOLLAR	420
AUSTRALIAN DOLLAR	124
DOLLAR SIGN	290
DOLLAR (BAND)	30
DOLLAR, CLACKMANNANSHIRE	30
game	Size
GAME	819
Game (S)	*99*
Video game	*55*
PC game	*44*
Game (G)	*720*
VIDEO GAME	312
PC GAME	24
GAME (FOOD)	232
GAME (RAPPER)	154
diamond	Size
DIAMOND	716
Diamond (S)	*221*
Diamond (gemstone)	*221*
Diamond (G)	*495*
DIAMOND (GEMSTONE)	71
BASEBALL FIELD	36
MUSIC RECORDING SALES CERT.	36
Corinth	Size
CORINTH	699
Corinth (S)	*409*
Ancient Corinth	*409*
Corinth (G)	*290*
ANCIENT CORINTH	92
CORINTH, MISSISSIPPI	72

Table 2: Wiki (CAPS) and manual (*italics*) annotations.

as ATMOSPHERE, ATMOSPHERE OF EARTH, ATMOSPHERE OF MARS, ATMOSPHERE OF VENUS, or STELLAR ATMOSPHERE. This solution is straightforward to implement, however it has the disadvantage that the resulting WSD model will never link words to more specific titles in Wikipedia like ATMOSPHERE OF MARS.

Another solution is to reorganize the original sense repository into a hierarchical classification scheme such that sense categories at each classification level become mutually disjoint. The resulting WSD system has the advantage that it can make fine grained sense distinctions for an ambiguous word, despite the annotation inconsistencies present in the training data. The rest of this paper describes a feasible implementation for this second solution that does not require any manual annotation beyond the links that are already provided by Wikipedia volunteers.

3 Learning for Coarse to Fine Grained Sense Disambiguation

Figure 2 shows our proposed hierarchical classification scheme for disambiguation, using "atmosphere" as the ambiguous word. Shaded leaf nodes show the final categories in the sense repository for each word, whereas the doted elliptical frames on the second level in the hierarchy denote artificial categories introduced to enable a finer grained classification into more specific senses. Thick dotted arrows illustrate the classification decisions that are made in order to obtain a fine grained disambiguation of the word. Thus, the word "atmosphere" is first classified to have the general sense ATMOSPHERE, i.e. "a layer of gases that may surround a material body of sufficient mass, and that is held in place by the gravity of the body". In the first solution, the disambiguation process would stop here and output the general sense ATMOSPHERE. In the second solution, the disambiguation process continues and further classifies the word to be a reference to ATMOSPHERE OF EARTH. To get to this final classification, the process passes through an intermediate binary classification level where it determines whether the word has a more specific sense covered in Wikipedia, corresponding to the artificial category ATMOSPHERE (S). If the answer is no, the system stops the disambiguation process and outputs the general sense category ATMOSPHERE. This basic sense hierarchy can be replicated depending on the existence of even finer sense distinctions in Wikipedia. For example, Wikipedia articles describing atmospheres of particular stars could be used to further refine STELLAR ATMOSPHERE with two additional levels of the type Level 2 and Level 3. Overall, the proposed disambiguation scheme could be used to relabel the ATMOSPHERE links in Wikipedia with more specific, and therefore more informative, senses such as ATMOSPHERE OF EARTH. In general, the Wikipedia category graph could be used to automatically create hierarchical structures for re-

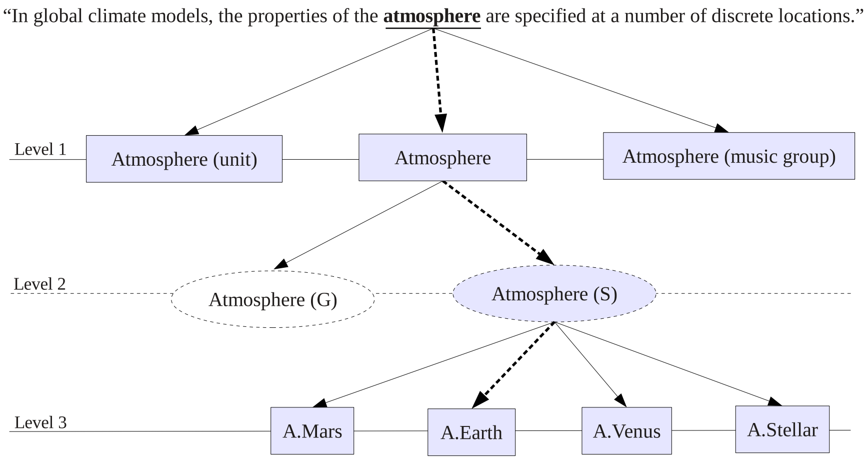

Figure 2: Hierarchical disambiguation scheme, from coarse to fine grained senses.

lated senses of the same word.

Training word sense classifiers for Levels 1 and 3 is straightforward. For Level 1, Wikipedia links that are annotated by users as ATMOSPHERE, ATMOSPHERE OF EARTH, ATMOSPHERE OF MARS, ATMOSPHERE OF VENUS, or STELLAR ATMOSPHERE are collected as training examples for the general sense category ATMOSPHERE. Similarly, links that are annotated as ATMOSPHERE (UNIT) and ATMOSPHERE (MUSIC GROUP) will be used as training examples for the two categories, respectively. A multiclass classifier is then trained to distinguish between the three categories at this level. For Level 3, a multiclass classifiers is trained on Wikipedia links collected for each of the 4 specific senses.

For the binary classifier at Level 2, we could use as training examples for the category ATMOSPHERE (G) all Wikipedia links that were annotated as ATMOSPHERE, whereas for the category ATMOSPHERE (S) we could use as training examples all Wikipedia links that were annotated specifically as ATMOSPHERE OF EARTH, ATMOSPHERE OF MARS, ATMOSPHERE OF VENUS, or STELLAR ATMOSPHERE. A traditional binary classification SVM could be trained on this dataset to distinguish between the two categories. We call this approach *Naive SVM*, since it does not account for the fact that a significant number of the links that are annotated by Wikipedia contributors as ATMOSPHERE should actually belong to the ATMOSPHERE (S) category – about 60% of them, according to Table 1. Instead, we propose treating all ATMOSPHERE links as unlabeled examples. If we consider the specific links in ATMOSPHERE (S) to be positive examples, then the problem becomes one of *learning with positive and unlabeled examples*.

3.1 Learning with positive and unlabeled examples

This general type of semi-supervised learning has been studied before in the context of tasks such as text classification and information retrieval (Lee and Liu, 2003; Liu et al., 2003), or bioinformatics (Elkan and Noto, 2008; Noto et al., 2008). In this setting, the training data consists of positive examples $x \in P$ and unlabeled examples $x \in U$. Following the notation of Elkan and Noto (2008), we define $s(x) = 1$ if the example is positive and $s(x) = -1$ if the example is unlabeled. The true label of an example is $y(x) = 1$ if the example is positive and $y(x) = -1$ if the example is negative. Thus, $x \in P \Rightarrow s(x) = y(x) = 1$ and $x \in U \Rightarrow s(x) = -1$ i.e., the true label $y(x)$ of an unlabeled example is unknown. For the experiments reported in this paper, we use our implementation of two state-of-the-art approaches to Learning with Positive and Unlabeled (LPU) examples: the *Biased SVM* formulation of Lee and Liu (2003) and the *Weighted Samples SVM* formulation of Elkan and Noto (2008). The original version of Biased SVM was designed to maximize the product between precision and recall. In the next section we describe a modification to the Biased SVM approach that can be used to maximize accuracy, a measure that is often used to evaluate WSD performance.

3.1.1 The Biased SVM

In the Biased SVM formulation (Lee and Liu, 2003; Liu et al., 2003), all unlabeled examples are considered to be negative and the decision function $f(x) = \mathbf{w}^T\phi(x) + b$ is learned using the standard soft-margin SVM formulation shown in Figure 3.

$$\begin{aligned}
\text{minimize:} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C_P \sum_{x \in P} \xi_x + C_U \sum_{x \in U} \xi_x \\
\text{subject to:} \quad & s(x)\left(\mathbf{w}^T\phi(x) + b\right) \geq 1 - \xi_x \\
& \xi_x \geq 0, \ \ \forall x \in P \cup U
\end{aligned}$$

Figure 3: Biased SVM optimization problem.

The capacity parameters C_P and C_U control how much we penalize errors on positive examples vs. errors on unlabeled examples. Since not all unlabeled examples are negative, one would want to select capacity parameters satisfying $C_P > C_U$, such that false negative errors are penalized more than false positive errors. In order to find the best capacity parameters to use during training, the Biased SVM approach runs a grid search on a separate development dataset. This search is aimed at finding values for the parameters C_P and C_U that maximize pr, the product between precision $p = p(y = 1|f = 1)$ and recall $r = p(f = 1|y = 1)$. Lee and Liu (2003) show that maximizing the pr criterion is equivalent with maximizing the objective $r^2/p(f = 1)$, where both $r = p(f = 1|y = 1)$ and $p(f = 1)$ can be estimated using the trained decision function $f(x)$ on the development dataset.

Maximizing the pr criterion in the original Biased SVM formulation was motivated by the need to optimize the F measure in information retrieval settings, where $F = 2pr(p+r)$. In the rest of this section we show that classification *accuracy* can be maximized using only positive and unlabeled examples, an important result for problems where classification accuracy is the target performance measure.

The accuracy of a binary decision function $f(x)$ is, by definition, $acc = p(f = 1|y = 1) + p(f = -1|y = -1)$. Since the recall is $r = p(f = 1|y = 1)$, the accuracy can be re-written as:

$$acc = r + 1 - p(f = 1|y = -1) \tag{1}$$

Using Bayes' rule twice, the false positive term $p(f = 1|y = -1)$ can be re-written as:

$$\begin{aligned}
p(f = 1|y = -1) &= \frac{p(f = 1)p(y = -1|f = 1)}{p(y = -1)} \\
&= \frac{p(f = 1)}{p(y = -1)} \times (1 - p(y = 1|f = 1)) \\
&= \frac{p(f = 1)}{p(y = -1)} - \frac{p(f = 1)}{p(y = -1)} \times \frac{p(y = 1)p(f = 1|y = 1)}{p(f = 1)} \\
&= \frac{p(f = 1) - p(y = 1) \times r}{p(y = -1)}
\end{aligned} \tag{2}$$

Plugging identity 2 in Equation 1 leads to:

$$\begin{aligned}
acc &= 1 + r + \frac{r \times p(y = 1) - p(f = 1)}{p(y = -1)} \\
&= 1 + \frac{r - p(f = 1)}{p(y = -1)}
\end{aligned} \tag{3}$$

Since $p(y = -1)$ can be assimilated with a constant, Equation 3 implies that maximizing accuracy is equivalent with maximizing the criterion $r - p(f = 1)$, where both the recall r and $p(f = 1)$ can be estimated on the positive and unlabeled examples from a separate development dataset.

In conclusion, one can use the original Biased SVM formulation to maximize $r^2/p(f = 1)$, which has been shown by Lee and Liu (2003) to maximize pr, a criterion that has a similar behavior with the F-measure used in retrieval applications. Alternatively, if the target performance measure is accuracy, we can choose instead to maximize $r - p(f = 1)$, which we have shown above to correspond to accuracy maximization.

3.1.2 The Weighted Samples SVM

Elkan and Noto (2008) introduced two approaches for learning with positive and unlabeled data. Both approaches are based on the assumption that labeled examples $\{x|s(x) = 1\}$ are selected at random from the positive examples $\{x|y(x) = 1\}$ i.e., $p(s = 1|x, y = 1) = p(s = 1|y = 1)$. Their best performing approach uses the positive and unlabeled examples to train two distinct classifiers. First, the dataset $P \cup U$ is split into a training set and a validation set, and a classifier $g(x)$ is trained on the labeling s to approximate the label distribution i.e. $g(x) = p(s = 1|x)$. The validation set is then used to estimate $p(s = 1|y = 1)$ as follows:

$$p(s{=}1|y{=}1) = p(s{=}1|x, y{=}1) = \frac{1}{|P|} \sum_{x \in P} g(x) \quad (4)$$

The second and final classifier $f(x)$ is trained on a dataset of weighted examples that are sampled from the original training set as follows:

– Each positive example $x \in P$ is copied as a positive example in the new training set with weight $p(y = 1|x, s = 1) = 1$.

– Each unlabeled example $x \in U$ is duplicated into two training examples in the new dataset: a positive example with weight $p(y = 1|x, s = 0)$ and a negative example with weight $p(y = -1|x, s = 0) = 1 - p(y = 1|x, s = 0)$.

Elkan and Noto (2008) show that the weights above can be derived as:

$$p(y{=}1|x, s{=}0) = \frac{1{-}p(s{=}1|y{=}1)}{p(s{=}1|y{=}1)} \times \frac{p(s{=}1|x)}{1{-}p(s{=}1|x)} \quad (5)$$

The output of the first classifier $g(x)$ is used to approximate the probability $p(s = 1|x)$, whereas $p(s = 1|y = 1)$ is estimated using Equation 4.

The two classifiers g and f are trained using SVMs and a linear kernel. Platt scaling is used with the first classifier to obtain the probability estimates $g(x) = p(s = 1|x)$, which are then converted into weights following Equations 4 and 5, and used during the training of the second classifier.

4 Experimental Evaluation

We ran disambiguation experiments on the 6 ambiguous words *atmosphere*, *president*, *dollar*, *game*, *diamond* and *Corinth*. The corresponding Wikipedia sense repositories have been summarized in Tables 1 and 2. All WSD classifiers used the same set of standard WSD features (Ng and Lee, 1996; Stevenson and Wilks, 2001), such as words and their part-of-speech tags in a window of 3 words around the ambiguous keyword, the unigram and bigram content words that are within 2 sentences of the current sentence, the syntactic governor of the keyword, and its chains of syntactic dependencies of lengths up to two. Furthermore, for each example, a Wikipedia specific feature was computed as the cosine similarity between the context of the ambiguous word and the text of the article for the target sense or reference.

The Level$_1$ and Level$_3$ classifiers were trained using the SVM^{multi} component of the SVM^{light} package.[1] The WSD classifiers were evaluated in a 4-fold cross validation scenario in which 50% of the data was used for training, 25% for tuning the capacity parameter C, and 25% for testing. The final accuracy numbers, shown in Table 3, were computed by averaging the results over the 4 folds. Since the word *president* has only one sense on Level$_1$, no classifier needed to be trained for this case. Similarly, words *diamond* and *Corinth* have only one sense on Level$_3$.

	atmosphere	*president*	*dollar*
Level$_1$	93.1%	—	94.1%
Level$_3$	85.6%	82.2%	90.8%
	game	*diamond*	*Corinth*
Level$_1$	82.9%	95.5%	92.7%
Level$_3$	92.9%	—	—

Table 3: Disambiguation accuracy at Levels 1 & 3.

The evaluation of the binary classifiers at the second level follows the same 4-fold cross validation scheme that was used for Level$_1$ and Level$_3$. The manual labels for specific senses and references in the unlabeled datasets are always ignored during training and tuning and used only during testing.

We compare the Naive SVM, Biased SVM, and Weighted SVM in the two evaluation settings, using for all of them the same train/development/test splits of the data and the same features. We emphasize that our manual labels are used only for testing purposes – the manual labels are ignored during training and tuning, when the data is assumed to contain only positive and unlabeled examples. We implemented the Biased SVM approach on top of the binary SVM^{light} package. The C_P and C_U parameters of the Biased SVM were tuned through the c and j parameters of SVM^{light} ($c = C_U$ and $j = C_P/C_U$). Eventually, all three methods use the development data for tuning the c and j parameters of the SVM. However, whereas the Naive SVM tunes these parameters to optimize the accuracy with respect to the noisy label $s(x)$, the Biased SVM tunes the same parameters to maximize an estimate of the accuracy or

[1] http://svmlight.joachims.org

F-measure with respect to the true label $y(x)$. The Weighted SVM approach was implemented on top of the LibSVM[2] package. Even though the original Weighted SVM method of Elkan and Noto (2008) does not specify tuning any parameters, we noticed it gave better results when the capacity c and weight j parameters were tuned for the first classifier $g(x)$.

Table 4 shows the accuracy results of the three methods for Level$_2$, whereas Table 5 shows the F-measure results. The Biased SVM outperforms the Naive SVM on all the words, in terms of both accuracy and F-measure. The most dramatic increases are seen for the words *atmosphere*, *game*, *diamond*, and *Corinth*. For these words, the number of positive examples is significantly smaller compared to the total number of positive and unlabeled examples. Thus, the percentage of positive examples relative to the total number of positive and unlabeled examples is 31.9% for *atmosphere*, 29.1% for *game*, 9.0% for *diamond*, and 11.6% for *Corinth*. The positive to total ratio is however significantly larger for the other two words: 67.2% for *president* and 91.5% for *dollar*. When the number of positive examples is large, the false negative noise from the unlabeled dataset in the Naive SVM approach will be relatively small, hence the good performance of Naive SVM in these cases. To check whether this is the case, we have also run experiments where we used only half of the available positive examples for the word *president* and one tenth of the positive examples for the word *dollar*, such that the positive datasets became comparable in size with the unlabeled datasets. The results for these experiments are shown in Tables 4 and 5 in the rows labeled *president$_S$* and *dollar$_S$*. As expected, the difference between the performance of Naive SVM and Biased SVM gets larger on these smaller datasets, especially for the word *dollar*.

The Weighted SVM outperforms the Naive SVM on five out of the six words, the exception being the word *president*. Comparatively, the Biased SVM has a more stable behavior and overall results in a more substantial improvement over the Naive SVM. Based on these initial results, we see the Biased SVM as the method of choice for learning with positive and unlabeled examples in the task of coarse to fine grained sense disambiguation in Wikipedia.

Word	NaiveSVM	BiasedSVM	WeightedSVM
atmosphere	39.9%	**79.6%**	75.0%
president	91.9%	**92.5%**	89.5%
dollar	96.0%	**97.0%**	**97.1%**
game	83.8%	**87.1%**	84.6%
diamond	70.2%	74.5%	**75.1%**
Corinth	46.2%	**75.1%**	51.9%
president$_S$	88.1%	**90.6%**	87.4%
dollar$_S$	70.3%	**84.9%**	70.6%

Table 4: Disambiguation accuracy at Level$_2$.

Word	NaiveSVM	BiasedSVM	WeightedSVM
atmosphere	30.5%	**86.0%**	83.2%
president	94.4%	**95.0%**	92.8%
dollar	97.9%	**98.4%**	**98.5%**
game	75.1%	**81.8%**	77.5%
diamond	8.6%	**53.5%**	46.3%
Corinth	15.3%	**81.2%**	68.0%
president$_S$	90.0%	**92.4%**	89.5%
dollar$_S$	77.9%	**91.2%**	78.2%

Table 5: Disambiguation F-measure at Level$_2$.

In a final set of experiments, we compared the traditional flat classification approach and our proposed hierarchical classifier in terms of their overall disambiguation accuracy. In these experiments, the sense repository contains all the leaf nodes as distinct sense categories. For example, the word *atmosphere* would correspond to the sense repository $\mathcal{R}$ = {Atmosphere (G), Atmosphere of Earth, Atmosphere of Mars, Atmosphere of Venus, Stellar Atmosphere, Atmosphere (unit), Atmosphere (music group)}. The overall accuracy results are shown in Table 6 and confirm the utility of using the LPU framework in the hierarchical model, which outperforms the traditional flat model, especially on words with low ratio of positive to unlabeled examples.

	atmosphere	*president*	*dollar*
Flat	52.4%	89.4%	90.0%
Hierarchical	79.7%	91.0%	90.1%
	game	*diamond*	*Corinth*
Flat	83.6%	65.7%	42.6%
Hierarchical	87.2%	76.8%	72.1%

Table 6: Flat vs. Hierarchical disambiguation accuracy.

[2] http://www.csie.ntu.edu.tw/˜cjlin/libsvm

5 Future Work

Annotation inconsistencies in Wikipedia were circumvented by adapting two existing approaches that use only positive and unlabeled data to train binary classifiers. This binary classification constraint led to the introduction of the artificial specific (S) category on Level$_2$ in our disambiguation framework. In future work, we plan to investigate a direct extension of learning with positive and unlabeled data to the case of multiclass classification, which will reduce the number of classification levels from 3 to 2. We also plan to investigate the use of unsupervised techniques in order to incorporate less popular references of a word in the hierarchical classification.

Conclusion

We presented an approach to training coarse to fine grained sense disambiguation systems that treats annotation inconsistencies in Wikipedia under the framework of learning with positive and unlabeled examples. Furthermore, we showed that the true accuracy of a decision function can be optimized using only positive and unlabeled examples. For testing purposes, we manually annotated 7,079 links belonging to six ambiguous words [3]. Experimental results demonstrate that accounting for annotation ambiguity in Wikipedia links leads to consistent improvements in disambiguation accuracy. The manual annotations were only used for testing and were ignored during training and development. Consequently, the proposed framework of learning with positive and unlabeled examples for sense disambiguation could be applied on the entire Wikipedia without any manual annotations. By augmenting general sense links with links to more specific articles, such an application could have a significant impact on Wikipedia itself.

Acknowledgments

This work was supported in part by the National Science Foundation IIS awards #1018613 and #1018590, and an allocation of computing time from the Ohio Supercomputer Center.

[3] Data and code will be made publicly available.

References

D. Ahn, V. Jijkoun, G. Mishne, K. Muller, M. de Rijke, and S. Schlobach. 2004. Using Wikipedia at the TREC QA track. In *Proceedings of the 13th Text Retrieval Conference (TREC 2004)*.

Volha Bryl, Claudio Giuliano, Luciano Serafini, and Kateryna Tymoshenko. 2010. Using background knowledge to support coreference resolution. In *Proceedings of the 2010 conference on ECAI 2010: 19th European Conference on Artificial Intelligence*, pages 759–764, Amsterdam, The Netherlands.

Razvan Bunescu and Marius Pasca. 2006. Using encyclopedic knowledge for named entity disambiguation. In *Proceesings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL-06)*, pages 9–16, Trento, Italy.

Philipp Cimiano, Antje Schultz, Sergej Sizov, Philipp Sorg, and Steffen Staab. 2009. Explicit versus latent concept models for cross-language information retrieval. In *International Joint Conference on Artificial Intelligence (IJCAI-09*, pages 1513–1518, Pasadena, CA, july.

S. Cucerzan. 2007. Large-scale named entity disambiguation based on Wikipedia data. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 708–716.

Charles Elkan and Keith Noto. 2008. Learning classifiers from only positive and unlabeled data. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '08, pages 213–220.

David A. Ferrucci, Eric W. Brown, Jennifer Chu-Carroll, James Fan, David Gondek, Aditya Kalyanpur, Adam Lally, J. William Murdock, Eric Nyberg, John M. Prager, Nico Schlaefer, and Christopher A. Welty. 2010. Building watson: An overview of the deepqa project. *AI Magazine*, 31(3):59–79.

Aria Haghighi and Dan Klein. 2009. Simple coreference resolution with rich syntactic and semantic features. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 1152–1161, Singapore, August.

M. Kaisser. 2008. The QuALiM question answering demo: Supplementing answers with paragraphs drawn from Wikipedia. In *Proceedings of the ACL-08 Human Language Technology Demo Session*, pages 32–35, Columbus, Ohio.

Wee Sun Lee and Bing Liu. 2003. Learning with positive and unlabeled examples using weighted logistic regression. In *Proceedings of the Twentieth International Conference on Machine Learning (ICML*, pages 448–455, Washington, DC, August.

Y. Li, R. Luk, E. Ho, and K. Chung. 2007. Improving weak ad-hoc queries using Wikipedia as external corpus. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 797–798, Amsterdam, Netherlands.

Bing Liu, Yang Dai, Xiaoli Li, Wee Sun Lee, and Philip S. Yu. 2003. Building text classifiers using positive and unlabeled examples. In *Proceedings of the Third IEEE International Conference on Data Mining*, ICDM '03, pages 179–186, Washington, DC, USA.

R. Mihalcea. 2007. Using Wikipedia for automatic word sense disambiguation. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics*, pages 196–203, Rochester, New York, April.

D. Milne. 2007. Computing semantic relatedness using Wikipedia link structure. In *Proceedings of the New Zealand Computer Science Research Student Conference*, Hamilton, New Zealand.

Hwee Tou Ng and H. B. Lee. 1996. Integrating multiple knowledge sources to disambiguate word sense: An exemplar-based approach. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics (ACL-96)*, pages 40–47, Santa Cruz, CA.

Keith Noto, Milton H. Saier, Jr., and Charles Elkan. 2008. Learning to find relevant biological articles without negative training examples. In *Proceedings of the 21st Australasian Joint Conference on Artificial Intelligence: Advances in Artificial Intelligence*, AI '08, pages 202–213.

Simone Paolo Ponzetto and Roberto Navigli. 2010. Knowledge-rich word sense disambiguation rivaling supervised systems. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1522–1531, Stroudsburg, PA, USA. Association for Computational Linguistics.

Simone Paolo Ponzetto and Michael Strube. 2006. Exploiting semantic role labeling, wordnet and wikipedia for coreference resolution. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 192–199.

M. Potthast, B. Stein, and M. A. Anderka. 2008. Wikipedia-based multilingual retrieval model. In *Proceedings of the 30th European Conference on IR Research*, Glasgow.

Altaf Rahman and Vincent Ng. 2011. Coreference resolution with world knowledge. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, pages 814–824, Stroudsburg, PA, USA. Association for Computational Linguistics.

Mark Stevenson and Yorick Wilks. 2001. The interaction of knowledge sources in word sense disambiguation. *Computational Linguistics*, 27(3):321–349, September.