

# Word Sense Disambiguation for Ontology Learning

*Research-in-Progress*

**Hayden Wimmer**

Bloomsburg University  
University of Maryland Baltimore County  
hayden.wimmer@gmail.com

**Lina Zhou**

University of Maryland Baltimore County  
zhoul@umbc.edu

## ABSTRACT

Ontology learning aims to automatically extract ontological concepts and relationships from related text repositories and is expected to be more efficient and scalable than manual ontology development. One of the challenging issues associated with ontology learning is word sense disambiguation (WSD). Most WSD research employs resources such as WordNet, text corpora, or a hybrid approach. Motivated by the large volume and richness of user-generated content in social media, this research explores the role of social media in ontology learning. Specifically, our approach exploits social media as a dynamic context rich data source for WSD. This paper presents a method and preliminary evidence for the efficacy of our proposed method for WSD. The research is in progress toward conducting a formal evaluation of the social media based method for WSD, and plans to incorporate the WSD routine into an ontology learning system in the future.

## Keywords

Ontology, Ontology Learning, Word Sense Disambiguation, Social Media

## INTRODUCTION

The objective of this research is to explore the role of social media in ontology learning. Word Sense Disambiguation (WSD) is a crucial task involved in ontology learning. WSD provides a mechanism to determine the meaning of words. Current approaches rely on a text or document corpus, Word Net, or a combination of both (Zhou, 2007). Social media has emerged as a dynamic, context-oriented data source that has the potential to increase the effectiveness of WSD and in turn ontology learning. Understanding the context and semantics of words is a critical first step in the creation of ontologies. Typically, determining semantics is performed by human domain experts. Automated semantic determination of terms is a potential step toward meeting the scalability requirements of ontology learning. Additional approaches have included incorporating external data (Weichselbraun, Wohlgenannt, & A., 2010), text mining (Hiep, Susan, & Qiang, 2012), a corpus (Zouaq, Gasevic, & Hatala, 2011) as well as Wikipedia to build concepts hierarchies (Ahmed, Toumouh, & Malki, 2012). However, the aforementioned literature largely neglects the potential of the vast amounts of context oriented user-generated content associated with social media. To address this gap in the literature, this research explores the efficacy of social media in WSD for ontology learning.

This paper is organized as follows; first we review the state of the art research in ontology learning. Next, we discuss WSD as it applies to ontology learning. Then, we introduce design artifact that performs WSD by exploiting context oriented social media, specifically twitter. After reporting the preliminary results of a case study, we describe plans for formal evaluation of the design artifact. Finally, we conclude this paper and discuss our future research directions.

## BACKGROUND AND RELATED WORK

### Ontology Learning Processes

The current World Wide Web is based on web pages with semantic meanings which typically can only be interpreted by humans. The vision of the future will be a semantic web which will bring structure and machine understandable information to the web (Berners-Lee, Hendler, & Lassila, 2001). The cornerstone to creating machine understandable content is the concept of ontologies. An ontology may be defined as a shared conceptualization of a domain (Fensel, 2001). Ontologies will support the semantic web; however, ontology development is a challenge which is time consuming and error-prone. Ontology acquisition is considered a bottleneck of the semantic web (Omelayenko, 2001). There are many reasons to create ontologies which include developing shared understandings of the structure of the domain, domain knowledge reuse, and to analyze domain knowledge (Noy, 2001). Ontology learning is the process of using machine learning techniques to assist in ontology development (Omelayenko, 2001). Ontology learning has been an emerging discipline in the artificial intelligence

community. There are various techniques to automatically or semi-automatically learn ontologies from sources; however, each has its own strengths and weaknesses.

Ontology learning is associated with specific processes. Each tool generally follows a set of predefined steps to achieve automated or semi-automated ontology learning. Although individual ontology learning environments are unique in system architecture, they also share a common process, which starts with information collection and pre-processing, followed by information processing, relationship discovery, and finally evaluation and validation. This process is shown in Figure 1. Each of these processes is discussed in detail next. These processes are iterative and may be repeated until the resulting ontology is acceptable.

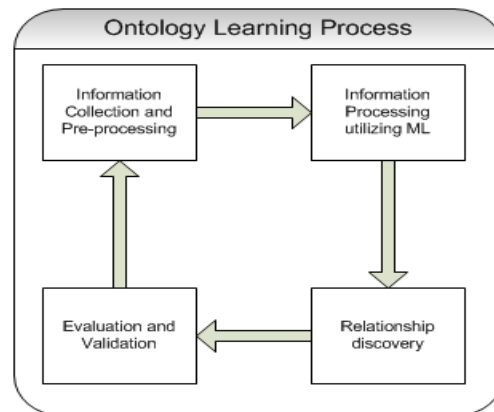


Figure 1 – Ontology Learning Process

#### Information Collection and Pre-processing

The first aspect of the ontology learning process is information collection and pre-processing. In this aspect of the ontology learning process it is necessary to identify information sources. Once sources are identified techniques are employed to extract the information. Sources may include internal sources such as a database and data warehouse and/or external sources such as web documents. Web documents are of particular interest as they apply to the vision of the semantic web. Frequently, natural language processing (NLP) techniques are employed for information pre-processing. These techniques may remove unnecessary or useless data.

#### Information Processing

NLP techniques typically parse each sentence looking to apply rules based on the syntax of the sentence and may learn rules by examples (Soderland, 1997). To learn and mine ontologies from text there are several processes that must be considered. A text processor that has multiple components is necessary. The processor must scan text and identify words, expressions, expand abbreviations, and possibly label the terms. Next, some form of lexical based system must be implemented to perform a lexical analysis of the terms and concepts. This is a critical step to prepare for relationship discovery and ontology construction.

#### Relationship Discovery

Relationship discovery is a key element to ontology learning. A statistical or heuristic system is implemented to find relationships (Maedche & Staab, 2000). Key techniques may include the Jaccard similarity index, the modified Jaccard similarity index, or PMI (point wise mutual information) (Aberer et al., 2007). There are also machine learning techniques that are employed in the automatic ontology learning process. Propositional rule learning techniques create rules and decision trees. Bayesian learning determines probabilistic values relating to the relationships between lexical terms. Clustering algorithms group instances together based on distance measures such as Euclidian distance (Omelayenko, 2001). Clustering algorithms may employ a top-down or bottom-up approach depending on the application area. Data mining approaches are also utilized in relationship discovery. Techniques such as frequency occurrence (Maedche, 2000) or *kNN* (k nearest neighbor) are frequently utilized with positive results (Maedche, Pekar, & Staab, 2002). Additionally, there are also tree based algorithms for ontology relationship discovery (Maedche, et al., 2002).

## Evaluation

Evaluation is a key component of ontology learning. Evaluation is typically performed by a human expert or knowledge engineer; however, it is important that standards and guidelines for ontology evaluation are available and utilized. Precision and recall are popular measures for evaluating ontology learning (Spyns & Reinberger, 2005). They are also referred to as lexical precision and recall and the latter reflects how well the learned terms cover the domain (Dellschaft & Staab, 2006). Similarly taxonomic precision and recall may be employed in the evaluation process. Taxonomic recall reflects how well the structure – or taxonomy – reflects the domain. It is also possible to conduct evaluations based on reference ontology (Dellschaft & Staab, 2006). In this scenario, a human expert creates domain ontology manually. The learned ontology is then compared with the manual ontology to determine the accuracy of the learned ontology. Ontologies may also be compared on instance data or on the schema. Instance based evaluation is concerned with the instances and classification of the instances, whereas schema based evaluation compares the ontologies at the schema level.

## Review of Ontology Learning Tools

There have been various automated and semi-automated ontology learning and ontology development techniques described in the literature (Zhou, 2007). Many of them have used natural language processing (NLP) techniques. Of these results we selectively discuss several popular tools that have used NLP in ontology learning, including ASIUM, Camille, Text-to-Onto, Onto Learn, Onto LT, OntoGain, and LexOnt.

ASIUM is a machine learning (*ML*) and natural language processing system used to learn sub-categorization frames of verbs as well as ontologies from text based sources using “domain dependence.” It is based on a clustering technique and is not fully automated. ASIUM begins with a syntactic parser (*SYLEX*) which gives interpretations of the parsed sentences. Classes are created using a bottom-up clustering method with domain expert validation at each level. To avoid over generalization there is a threshold as well as user intervention (Faure & Nedellec, 1999).

Camille is the contextual acquisition for incremental lexeme learning and is an extension of previous work (*Link*) in natural language processing. The novel concept behind Camille is that it infers meaning of words. Additionally, identifying verbs is a challenge that many systems do not support. Camille looks for examples of how the verb is used and attempts to infer a meaning. Tests suggest Camille has a 42% precision and 16% recall (Weimer-Hastings, Graesser, & Weimer-Hastings, 1998).

Text-to-Onto is a semi-automatic ontology learning architecture that learns ontologies from text based sources. The Text-to-Onto environment has a great deal of promise and usefulness as it uses web documents as its information source. There are five aspects to the conceptual architecture: text and processing management, text processing server, lexical database and domain lexicon, learning module, and ontology engineering environment Onto Edit (Maedche, 2000). The first aspect of the architecture is a data import and processing module which prepares the web documents for natural language processing. The source documents are HTML, PDF, and PostScript. The second aspect is the natural language processing system. A German based processor was utilized. The algorithm library contains algorithms for ontology extraction and maintenance. The learning algorithms perform the relationship discovery. Statistical and data mining techniques are employed in relationship discovery. Next an interface for presenting the results is implemented. Ontology extraction and maintenance are based on lexical entries, hierarchies, and lexical relations. Statistical and data mining techniques were employed to extract patterns. Ontology pruning and refinement were added as part of ontology maintenance (Maedche, 2001). The evaluation of the resulting ontology is performed via Onto Edit – an ontology engineering environment (Maedche, 2000, 2001).

The Onto Learn environment extracts domain specific terms from text and relates the terms to an ontology (Navigli, 2003). Onto Learn was implemented and evaluated in two European projects. Onto Learn utilizes Consys – a web based groupware package for validation purposes. The conceptual architecture of Onto Learn is diagrammed in a three phase process. First, Onto Learn extracts relevant terms from domain based sources. Second, the system semantically processes the terms and employs WordNet. Finally, taxonomic and similarities are computed. According to the authors, Onto Learn significantly and remarkably improved ontology building productivity (Missikoff, Navigli, & Velardi, 2002). Onto Learn extracts domain terminology by employing statistical and natural language processing techniques. Next, the system uses an online search to find definitions and increase the level of precision. The Wordnet lexicon is consulted. Once definitions are located non-relevant definitions are discarded and ambiguity issues are handled by an algorithm known as SSI or structural semantic interconnections (Navigli & Velardi, 2004). If definitions cannot be located then it is necessary to gain input from a human expert. Next trees are created and the ontology is constructed (Velardi, Navigli, Cucchiarelli, & Neri, 2005). There are four algorithms employed by Onto Learn which extract terms, extract natural language definitions, parse natural language definitions, disambiguation, and relationship identification (Velardi, et al., 2005). Finally, evaluation of the created ontology is required (Navigli & Velardi, 2004).

Onto LT can be used as a plug-in for the popular Protégé ontology engineering environment in order to assist ontology engineers by automatically extracting classes and attributes from text. Onto LT employs an extraction phase which parses information from source documents. Next, defining and learning of concepts are used to create mappings via linguistic structures. The ontology is then output into the Protégé environment where the validation and evaluation may be performed by a human expert. OntoGain (Drymonas, Zervanou, & Petrakis, 2010) was developed to learn ontologies from unstructured text by employing formal concept analysis (FCA) and rule-based algorithms to discover non-taxonomic relations. LexOnt (Arabshian, Danielsen, & Afroz, 2012) was developed to use a programmable web directory as a corpus; however, it also incorporated WordNet and Wikipedia as external domain knowledge. LexOnt is designed to support semi-automatic ontology generation.

The ontology learning systems that have been surveyed in this study are summarized in Table 1.

Processes	Information Collection	Information Processing	Relationship Discovery	Evaluation
Tools				
Text-to-Onto	Data Import Module for HTML, PDF, PostScript	NLP, Statistical, Data Mining, SMES	Learning Algorithms - Concepts, Lexical Entries, Taxonomy, Relationships	Manual using Onto Edit
Onto Learn	Data Import Module for HTML, PDF, PostScript	Statistical, NLP, WordNet	SSI, Tree building, kind-of relationship extraction	Manual via domain expert
Onto LT	Data Import Module - proprietary XML input	Chi-Square statistical preprocessing	Precondition Rule Language, SHUG rule-base system	Manual using Protégé
OntoGain	Unstructured Text	Formal Concept Analysis	Rule based Algorithms	
LexOnt	Programmable Web Directory	NLP, TF-IDF Term	External Knowledge Base	Manual using Protégé

Table 1- Processes of Selected Ontology Learning Tools

### Word Sense Disambiguation

Word sense disambiguation research has been dominated by methods that employ WordNet and Wikipedia. Li, Sun, & Datta (Li, Sun, & Datta, 2011) proposed a method that follows article titles and anchors through links on Wikipedia. This method was reported to perform as well as previous techniques with less computational cost. Hwang, Jeong, Lee, & Jung (Hwang, Jeong, Lee, & Jung, 2011) proposed a similar technique, Wikipedia Link-Based Measure (*WLM*). *WLM* examines Wikipedia links and category information for word sense disambiguation. Other recent literature follows more traditional routes. One such example is (Banerjee & Pedersen, 2002) where a modified Lesk algorithm was applied to a semantic network constructed from WordNet. The Lesk algorithm was extended by adding semantically tagged glosses into the algorithm. There are also hybrid approaches that integrate multiple sources. (Che & Zhang, 2011) employed four separate but well known corpora to perform disambiguation of Chinese terms. (Mandreoli & Martoglia) employed meta-data where structures such as relational schemas, taxonomies, and ontologies for disambiguation purposes.

### Ontology Learning from Social Media

There exists a vast amount of information on the web (*WWW*) that may be extracted in order to perform ontology learning functions. Wikipedia ([www.wikipedia.org](http://www.wikipedia.org)) is an online dictionary that contains articles that are edited by a large user community and contains vast information that may be mined and utilized in the ontology learning process. WordNet (Godby, 1999; Kilgarriff, 2000; Lin, 1999) is another source for web-based information –specifically hyponyms – and has been employed in many natural language processing systems. Harnessing information from social media and utilizing this information to create ontologies is a powerful concept that requires further research.

Various authors have researched employing technologies such as Wikipedia (Denoyer & Gallinari, 2006) and Wordnet (Godby, 1999; Kilgarriff, 2000; Lin, 1999). (Syed & Finin, 2010) describe an approach for discovering ontological elements from Wikipedia via automatic ontology discovery using slot labels and fillers, and creating a class hierarchy based on the similarity between the classes and slot hierarchies. (Ruiz-Casado, Alfonseca, & Castells, 2005, 2007) discover semantic

relationships by utilizing a technique that involved entry sense disambiguation, pattern extraction, pattern generalization, and identification of the new relations utilizing Wordnet and Wikipedia. (F. Wu & D. Weld, 2008; F. Wu & D. S. Weld, 2008) developed Kylin Ontology Generator (*KOG*) that uses machine learning techniques and statistical relation learning to build ontologies from Wikipedia and Wordnet. *KOG* contains three modules: 1) schema cleaning on Wikipedia, 2) subsumption detection using Markov logic, and 3) a schema mapping between concepts. Weber & Buitelaar (Weber & Buitelaar, 2006) developed *ISOLDE*, which involves three components: name-entity recognition, linguistic pattern analysis, and collecting web-based knowledge for the extracted classes and integrating them into a new ontology.

Learning ontologies from social network sources is the focus of recent research on ontology learning. The concept of folksonomies where users create and maintain tags using freely chosen keywords (Mika, 2005). Mika (Mika, 2005) extended the standard bipartite model to a tripartite model. The bipartite model consists of a two-part model while the tripartite model consists of a three-part model or graph which consists of actors, concepts and instances. There are some challenges with using folksonomies (Limpens, Gandon, & Buffa, 2008) which include ambiguous tags, potential misspellings, a lack of explicit representation of concepts, and tags in multiple languages. Limpens et al. (Limpens, et al., 2008) also present the approaches to extracting semantics from folksonomies as well as system to utilize folksonomies for ontology creation. This includes the GroupMe system as well as *SOIC* (semantically interlinked online communities). Angeletou, Sabou, Specia, & Motta (Angeletou, Sabou, Specia, & Motta, 2007) describe semantic enrichment of folksonomies by concept identification and subsequently relationship discovery. Folksonomies may be enriched by harvesting the semantic web. Other authors have proposed for users to group their web 2.0 content (Abel et al., 2007). GroupMe is described where social tagging and grouping may be performed by individuals using drag and drop operations. It is supported by RDF and attempts to merge Web 2.0 and the semantic web.

Patrick et al. (2001) develop an algorithm known as *COBWEB* which may be used to automatically generate ontologies by classifying objects with respects to an existing class, creating a new class, merging existing classes, or separating a class into multiple classes. *COBWEB* generates ontologies in RDF and RDFS format. The smart radio application is employed which is a system where users rate songs based on their individual preferences. The *COBWEB* algorithm attempts to construct ontologies from the smart radio application to increase the effectiveness of recommendations to the user.

Combining the social and semantic web is a natural extension of the current direction of web 3.0. Collective knowledge systems as described by (Gruber, 2008) are the intersection of the semantic web and social networking or user generated content and tags. Real Travel is presented which permits users to post pictures, stories, and itineraries of their travels as a collective knowledge system. Mori, Tsujishita, Matsuo, & Ishizuka (Mori, Tsujishita, Matsuo, & Ishizuka, 2006) employ similarity based measures to extract labels which describe relations between social networks which also extracts concepts and uses similarity to compute hierarchical based clusters. The authors extract the underlying relations between entities which are embedded into social networks. The method proposed is unsupervised and domain independent which extends existing ontology extraction techniques from social networks. To address the problem that domain ontologies are overly static, Monachesi & Markus (Monachesi & Markus, 2010) used social media to enrich existing domain ontologies. In this research, the authors also propose a method for disambiguation of social media based tags. Ontology learning from folksonomies entails the identification of well-defined terms and the creation of hierarchical relations between tags. Tang, Leung, Luo, Chen, & Gong (Tang, Leung, Luo, Chen, & Gong, 2009) propose a three stage approach that uses probabilistic models to determine correspondences between tags which then computes the possible relations among the tags and finally determines the relations and constructs a hierarchical structure. Flink is a system developed to exploit FOAF for social intelligence by creating a portal for the semantic web community (Mika, 2005). The authors argue that semantic web technologies may be employed to assist the knowledge extraction, representation, and ontology mapping processes.

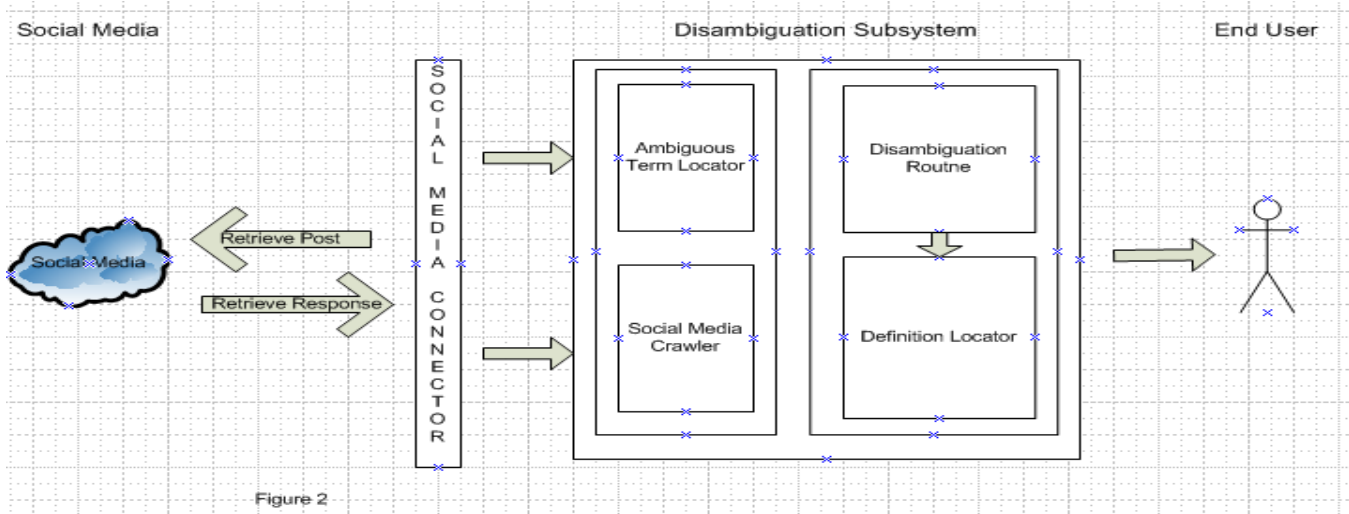
Despite of the progress made in the area of ontology learning from social media, with the exception of user tags, little research has focused on the issue of WSD in exploiting user generated content in ontology learning, particularly large-scaled running text generated by users of social media. To fill the gap, we design and implement an approach to WSD.

## **METHODOLOGY**

We followed the design science paradigm in this research. Design science in Information Systems involves some critical steps such as determining the relevance of the problem, designing an artifact to address the problem, evaluation of the design artifact and communicating research contributions (Hevner, March, Park, & Ram, 2004). We propose to evaluate our design artifact against other baseline methods for word sense disambiguation. In addition, human subjects will also be used to determine the efficacy of our system in relation to the baseline method.

## An Approach to WSD in Social Media

We illustrate the proposed method for WSD in social media with Twitter data, which is one of the most popular social media that collect use-generated data in real time. After a user generates a tweet and posts it to Twitter, other can post responses to that tweet. The responses will be context-oriented since they are responses to the original tweet. Based on this, the replies can be employed as a means for word sense disambiguation. We propose a design artifact which will take advantage of the context of the replies in order to disambiguate terms in the original tweet. Figure 2 presents the architecture of a WSD system that leverages social media data.



At the beginning, a connector is created to interact with social media. This connector is responsible for retrieving posts, issuing queries, posting to social media, and logging into social media. The social media connector interfaces with the core word sense disambiguation system. The first component of the disambiguation subsystem is the ambiguous term locator. The ambiguous term locator parses each word in a post and using a Jaccard similarity coefficient which is based on a lexical retrieval determines the most ambiguous term in a post. Once the ambiguous term is located it is sent back to the social media connector where the social media crawler retrieves up to 100 replies to the user of the original post. These replies are added into a one-dimensional array of a string data type. This array is sent back to the disambiguation subsystem. A multidimensional array of the ambiguous term is created which contains the synsets of all homonyms of the ambiguous term. Another array of type integer is created of the same size as the multidimensional array to be used as a counting mechanism. For each of the replies retrieved and stored in the array, the tweet is parsed and each noun, adverb, adjective, and verb is added into an array to be used in the disambiguation routine. The terms in this array are compared with the synsets of hyponyms of the original ambiguous term and each time a term appears in a synset the corresponding counter is incremented. The algorithm for determining the ambiguous term and the disambiguation routine is shown in *Equation 1*.

```

Find Ambiguous Term
1. For Each Word in the Tweet
   a) Count the all Synsets
   b) Compute Jaccard Coefficient
   c) Use Jaccard to determine Most Ambiguous Term
2. Return Most Ambiguous Term

Disambiguate Term
1. Get User ID of Originating Tweet
2. Create String Array with Size = 100
3. Get 100 Replies to User ID
4. Load Replies into Array from (2)
5. Create Multidimensional Array of all Hyponyms of Ambiguous Term
6. Create Integer Array Size = Size of Array from (5)
7. Parse Tweets from Array (2)
8. Compare Tweets from (2) and Array from (5)
9. Update Corresponding Counter in Array (6)

```

*Equation 1 - Algorithms for Locating and Disambiguating Terms*

The method was implemented using open source technologies. First, the system was built in Java using the Eclipse Helios environment. The lexical subsystem was based on the popular Word Net. In order to interface with WordNet the JAWS (Java API for WordNet Searching) system was utilized. The social media selected was Twitter (twitter.com) due to the dynamic nature of twitter. In order to interface with twitter the twitter4j API was utilized. Standard Java GUI components were employed for interacting with the user.

**Preliminary Results**

The preliminary evaluation was conducted with a randomly selected set of 40 tweets. Each tweet went through three processes, which were illustrated with a randomly selected tweet next. First, we identify the term *t* which is the most ambiguous in the selected tweet, as shown in Figure 3. The term that had the largest number of synsets in WordNet was considered the most ambiguous. Second, we collected 100 tweets from the context of the target tweet. Third, the 100 tweets were used to disambiguate the term *t*, as shown in Figure 4. Specifically, nouns, verbs, adverbs, and adjectives were retrieved from the related tweets. The synset that appeared most frequently among those words was selected as the sense of term *t*. Figure 5 shows the disambiguation of the ambiguous term.

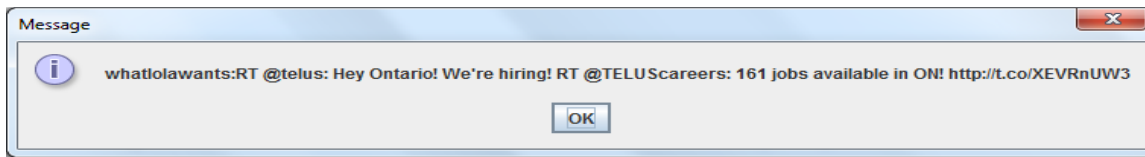


Figure 3 Tweet Retrieved from Twitter

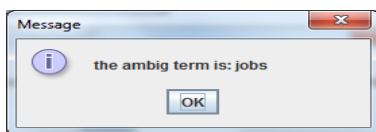


Figure 4- Ambiguous Term Identified

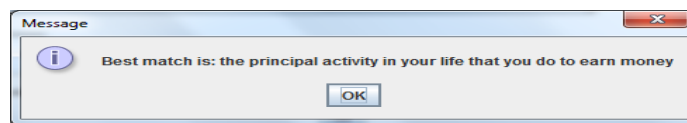


Figure 5 – Term Disambiguated using Social Media

The 40 disambiguation results were manually evaluated by a human evaluator. The results are reported in Table 2, and the overall precision was 70.6%.

Correctly Disambiguated	24
Incorrectly Disambiguated	10
Not Applicable (i.e. not in English)	6

Table 2 – Preliminary Results of WSD

To help us gain an understanding of the limitation of the proposed method we manually analyzed incorrectly disambiguated cases. For instance, the tweet shown in Table 3 is related to travel, specifically locating cheap flights. It turned out that another synset of the ambiguous term was more closely related to travel than the correct synset. The analysis highlights the important role of context in disambiguation, and it also suggests that we should have parsed the tweets to identify correct parts-of-speech of the ambiguous term (verb in this case) before feeding it to our disambiguation routine.

Tweet	CxxxCxxxx:Some cheap flights I found on the Skyscanner iPhone app @txxxl73 @DxxMxxxxUK @mxxxxlxxxxx 2 weeks bring it on <a href="http://t.co/3sP4Zsqv">http://t.co/3sP4Zsqv</a>
Ambiguous Term	Found
Definition of Term	food and lodging provided in addition to money

Table 3 – An Incorrectly Disambiguated Term with Identifiers Removed

**Evaluation**

The proposed WSD method will be evaluated by human subjects. A minimum of 30 subjects will be recruited from undergraduate and graduate students. Each human subject will be asked to evaluate the disambiguation results of a randomly

selected set of 20 ambiguous words extracted from different tweets. Each tweet containing an ambiguous word will be presented along with other 20 tweets in the preceding context and another 20 in the following context. The subjects will provide ratings for each system disambiguation result in terms of “accurate”, “highly relevant”, “somewhat relevant”, and “no relevance.” This will yield a total of 600 cases from human evaluation. Accuracy will be computed based on the percentage of correctly disambiguated words (summarization of accurate and high relevant cases). The results will be compared against other recent WSD methods such as (Li, et al., 2011).

## CONCLUSIONS AND FUTURE DIRECTIONS

Ontology learning is the process of utilizing machine learning techniques to automatically construct ontologies. Automated ontology learning is a complex task that requires many different systems in order to effectively learn ontologies. One critical aspect of ontology learning is word sense disambiguation (WSD). WSD is the process of determining the correct definition of an ambiguous term. Many approaches to WSD have been proposed; however, few have taken advantage of the power of social media. In this paper, a new approach to WSD was proposed and a prototype developed to automatically disambiguate words from twitter. We have provided preliminary evidence that social media can be a valuable tool to aid in word sense disambiguation and in turn ontology learning. Our future research includes conducting a large-scaled evaluation of the proposed method for WSD. Additionally, we will determine the sensitivity of the performance of WSD to the size of social media context. Further, development and testing of a full-fledged ontology learning system as well as assessing the impact of WSD on the effectiveness of ontology learning will be performed.

## REFERENCES

1. Abel, F., Frank, M., Henze, N., Krause, D., Plappert, D., & Siehndel, P. (2007). *GroupMe! - where semantic web meets web 2.0*. Paper presented at the Proceedings of the 6th international The semantic web and 2nd Asian conference on Asian semantic web conference.
2. Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L., et al. (2007). An Empirical Study of Instance-Based Ontology Matching *The Semantic Web* (Vol. 4825, pp. 253-266): Springer Berlin / Heidelberg.
3. Ahmed, K. B. S., Toumouh, A., & Malki, M. (2012). *Effective Ontology Learning: Concepts Hierarchy Building Using Plain Text Wikipedia*. Paper presented at the ICWIT, Hawaii.
4. Angeletou, S., Sabou, M., Specia, L., & Motta, E. (2007). *{Bridging the Gap Between Folksonomies and the Semantic Web: An Experience Report}*. Paper presented at the ESWC workshop. Bridging the Gap between Semantic Web and Web 2.0.
5. Arabshian, K., Danielsen, P., & Afroz, S. (2012). *LexOnt: A Semi-Automatic Ontology Creation Tool for Programmable Web*. Paper presented at the AAAI Spring Symposium Series.
6. Banerjee, S., & Pedersen, T. (2002). An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet. In A. Gelbukh (Ed.), *Computational Linguistics and Intelligent Text Processing* (Vol. 2276, pp. 136-145): Springer Berlin Heidelberg.
7. Berners-Lee, T., Hendler, J., & Lassila, O. (2001). The Semantic Web - A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities. *Scientific American*, 34+.
8. Che, L., & Zhang, Y. (2011). *Study on Word Sense Disambiguation Knowledge Base Based on Multi-Sources*. Paper presented at the 2011 3rd International Workshop on Intelligent Systems and Applications (ISA), Wuhan.
9. Dellschaft, K., & Staab, S. (2006). *On how to perform a gold standard based evaluation of ontology learning*. Paper presented at the ISWC 2006 LNCS 4273.
10. Denoyer, L., & Gallinari, P. (2006). The Wikipedia XML corpus. *SIGIR Forum*, 40(1), 64-69.
11. Drymonas, E., Zervanou, K., & Petrakis, Z. (2010). *Unsupervised ontology acquisition from plain texts: The OntoGain system*. Paper presented at the International Conference on Applications of Natural Language to Information Systems (NLDB).
12. Faure, D., & Nedellec, C. (1999). *Knowledge acquisition of predicate argument structures from technical texts using Machine Learning Learning: ASIUM*. Paper presented at the EKAW.
13. Fensel, D. (2001). Ontologies: Silver Bullet for Knowledge Management and Electronic Commerce.
14. Godby, C. (1999). WordNet: An electronic lexical database. *Library Quarterly*, 406-408.
15. Gruber, T. (2008). Collective knowledge systems: Where the Social Web meets the Semantic Web. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(1), 4-13.
16. Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Q.*, 28(1), 75-105.
17. Hiep, L., Susan, G., & Qiang, W. (2012). Ontology Learning Using Word Net Lexical Expansion and Text Mining.



18. Hwang, M., Jeong, D.-H., Lee, S., & Jung, H. (2011). *Measuring Similarities between Technical Terms Based on Wikipedia*. Paper presented at the 2011 International Conference on and 4th International Conference on Cyber, Physical and Social Computing.
19. Kilgarriff, A. (2000). WordNet: An electronic lexical database. *Language*, 706-708.
20. Li, C., Sun, A., & Datta, A. (2011). *A generalized method for word sense disambiguation based on wikipedia*. Paper presented at the ECIR'11 Proceedings of the 33rd European conference on Advances in information retrieval.
21. Limpens, F., Gandon, F., & Buffa, M. (2008). *Bridging ontologies and folksonomies to leverage knowledge sharing on the social Web: A brief survey*. Paper presented at the 2008 23rd IEEE/ACM International Conference on Automated Software Engineering - Workshops, L'Aquila, Italy.
22. Lin, D. (1999). WordNet: An electronic lexical database. *Computational Linguistics*, 292-296.
23. Maedche, A. (2000). *The Text-to-Onto Learning Environment*. Paper presented at the 8th Int. Conf. on Conceptual Structures Logical, Linguistic, and Computational Issues,.
25. Maedche, A. (2001). *The Ontology Extraction & Maintenance Framework Text-to-Onto*, New Orleans, Louisiana.
26. Maedche, A., Pekar, V., & Staab, S. (2002). *Ontology learning part one - on discovering taxonomic relations from the web*. Paper presented at the Proceedings of the Web Intelligence conference.
27. Maedche, A., & Staab, S. (2000). Mining ontologies from text. *Knowledge Engineering and Knowledge Management, Proceedings*, 189-202.
28. Mandreoli, F., & Martoglia, R. Knowledge-based sense disambiguation (almost) for all structures. *Journal Information Systems*, 36(2), 406-430.
29. Mika, P. (2005). Flink: Semantic Web Technology for the Extraction and Analysis of Social Networks. *Journal of Web Semantics*, 3, 211--223.
30. Missikoff, M., Navigli, R., & Velardi, P. (2002). Integrated approach to Web ontology learning and engineering. *Computer*, 60-+.
31. Monachesi, P., & Markus, T. (2010). Using Social Media for Ontology Enrichment. In L. Aroyo, G. Antoniou, E. Hyvönen, A. ten Teije, H. Stuckenschmidt, L. Cabral & T. Tudorache (Eds.), *The Semantic Web: Research and Applications* (Vol. 6089, pp. 166-180): Springer Berlin / Heidelberg.
32. Mori, J., Tsujishita, T., Matsuo, Y., & Ishizuka, M. (2006). Extracting Relations in Social Networks from the Web Using Similarity Between Collective Contexts. In I. Cruz, S. Decker, D. Allemang, C. Preist, D. Schwabe, P. Mika, M. Uschold & L. Aroyo (Eds.), *The Semantic Web - ISWC 2006* (Vol. 4273, pp. 487-500): Springer Berlin / Heidelberg.
33. Navigli, R. (2003). Ontology Learning and its application to automated terminology translation. *Intelligent Systems, IEEE*, 18, 21-33.
34. Navigli, R., & Velardi, P. (2004). Learning domain ontologies from document warehouses and dedicated web sites. *Computational Linguistics*, 151-179.
35. Noy, F. (2001). Ontology development 101: A guide to creating your first ontology. *Technical Report SMI-2001-0880*.
36. Omelayenko, B. (2001). *Learning of ontologies for the Web: the analysis of existent approaches*. Paper presented at the Proceedings of the international workshop on Web dynamics, London.
37. Ruiz-Casado, M., Alfonseca, E., & Castells, P. (2005). Automatic Extraction of Semantic Relationships for WordNet by Means of Pattern Learning from Wikipedia. In A. Montoyo, R. Muñoz & E. Métais (Eds.), *Natural Language Processing and Information Systems* (Vol. 3513, pp. 67-79): Springer Berlin / Heidelberg.
38. Ruiz-Casado, M., Alfonseca, E., & Castells, P. (2007). Automatising the learning of lexical patterns: An application to the enrichment of WordNet by extracting semantic relationships from Wikipedia. *Data & Knowledge Engineering*, 61(3), 484-499.
39. Soderland, S. (1997). *Learning to extract text-based information from the world wide web*. Paper presented at the International Conference on Knowledge Discovery and Data Mining.
40. Spyns, P., & Reinberger, M. (2005). Lexically evaluating ontology triples generated automatically from texts. *Semantic Web: Research and Applications, Proceedings*, 563-577.
41. Syed, Z., & Finin, T. (2010). *Unsupervised techniques for discovering ontology elements from Wikipedia article links*. Paper presented at the First International Workshop on Formalisms and Methodology for Learning by Reading, Los Angeles, CA.
42. Tang, J., Leung, H.-f., Luo, Q., Chen, D., & Gong, J. (2009). *Towards ontology learning from folksonomies*. Paper presented at the Proceedings of the 21st international joint conference on Artificial intelligence.
43. Velardi, P., Navigli, R., Cucchiarelli, A., & Neri, F. (2005). Evaluation of OntoLearn, a methodology for automatic learning of ontologies. In I. Press (Ed.), *Ontology Learning from Text, Methods, evaluation, and applications* (pp. 92-108).

44. Weber, N., & Buitelaar, P. (2006). *Web-based Ontology Learning with ISOLDE*. Paper presented at the Proceedings of the Workshop on Web Content Mining with Human Language, International Semantic Web Conference.
45. Weichselbraun, A., Wohlgenannt, G., & A., S. (2010). Refining Non-Taxonomic Relation Labels with External Structured Data to Support Ontology Learning. *Data & Knowledge Engineering*, 69(8), 763-778.
46. Weimer-Hastings, P., Graesser, A., & Weimer-Hastings, K. (1998). *Inferring the meaning of verbs from context*. Paper presented at the Proceedings of the Twentieth Annual Conference of the Cognitive Science Society.
47. Society.
48. Wu, F., & Weld, D. (2008). *Automatically refining the wikipedia infobox ontology*. Paper presented at the International World Wide Web Conference Committee, Beijing, CN.
49. Wu, F., & Weld, D. S. (2008). *Automatically refining the wikipedia infobox ontology*. Paper presented at the Proceeding of the 17th international conference on World Wide Web.
50. Zhou, L. (2007). Ontology learning: state of the art and open issues. *Information Technology & Management*, 241-252.
51. Zouaq, A., Gasevic, D., & Hatala, M. (2011). Towards Open Ontology Learning and Filtering. *Information Systems*, 36(7), 1064-1081.