Word Sense Disambiguation: An Investigation into Mechanisms for Sense Discrimination

Project Report

Submitted in partial fulfillment of the requirements for the degree of

Bachelor of Technology

by

DIPTESH KANOJIA

Roll No: 0944910004

Under the guidance of

Prof. (Dr.) Pushpak Bhattacharyya

Department of Computer Science and Engineering
Indian Institute of Technology, Bombay

August 2012

Acknowledgement

I would like to express my profound gratitude to **Prof. (Dr.) Pushpak Bhattacharyya** for his *invaluable guidance* and support for rigorous study of literature. His essential and unparallel inputs and suggestions during difficult phases were much required. I am highly indebted to him for his kind help. I thank my senior research fellows, **Salil Joshi** and **Arindam Chatterjee** for the minutely detailed discussions we had on Word Sense Disambiguation, and Human Annotation Techniques. I thank **Raj Dabre** for helping me out with PHP coding, whenever needed, and his immensely supportive nature during the course of this project. This project could not have been complete without their help.

I would also like to thank **Shri S.K. Jain (Chariman)**, K.S. Jain Institute of Engineering and Technology, for his constant moral support, and encouraging nature towards my research projects.

Finally, I would like to thank my dear friends **Nitin Kumar Singh, Ashish Sharma, Anu Gupta, Russell Quadros** and **Deck-16 people** for supporting me throughout all my tough times, and encouraging me to work harder every time I transgressed.

Diptesh Kanojia
(0944910004)

Abstract

Word Sense Disambiguation (WSD) is defined as the task of computationally finding the senses of words from a context. Our long time research on Word Sense Disambiguation shows that in almost all disambiguation algorithms, the sense distribution parameter *P(S|W)*, where P is the probability of the sense of a word W being S, plays the deciding role. The widely reported accuracy figure of around 60% for all-words-domain independent WSD is contributed to mainly by *P(S|W)*, as one ablation test after another reveals.

Our experience of working with human annotators who mark with WordNet sense ids, general and domain specific corpora brings to light the interesting fact that producing sense ids without looking at the context is a heavy cognitive load. Sense annotators do form hypothesis in their minds about the possible sense of a word (‘most frequent sense’ bias), but then look at the context for clues to accept or reject the hypothesis. Such clues are minimal, just one or two words, but are critical nonetheless. Without these clues the annotator is left in an indecisive state as to whether or not to put down the first sense coming to his mind. The task becomes all the more cognitively challenging, if the senses are fine grained and seem equally probable.

The development of a tool which allows for annotators to conveniently specify the clues that they use for distinguishing between the various senses of a word is quite crucial in the task of word sense disambiguation. It is further important to utilize these clues so as to build a structure or a framework which allows for reducing the uncertainty of the sense of a particular word.

We imagine that constructing a “Discrimination Net” in the form of a weighted graph will assist in calculating a score which will say something about this uncertainty. The underlying idea is that there are words with multiple senses as well as ones with unique senses present and by traversing this graph, we will eventually reach these unique senses and then determine the score.

Contents

Introduction 8
1.1 Word Sense Disambiguation (WSD) 8
1.1.1 WSD Definitions 8
1.1.1.1 General Definition 8
1.1.1.2 Formal definition 9
1.1.1.3 As a Classification Problem 9
1.1.2 Variants of WSD 9
1.1.2.1 Lexical Sample (Targeted WSD) 9
1.1.2.2 All-words WSD 9
1.1.3 WSD: Heart of the NLP Problems 9
1.2 The Sense Annotation Process 11
1.2.1 The Sense Marker Tool 11
1.3 Report Outline 12
Motivation 13
2.1 Human *vs.* Machine Annotation 13
2.1.1 Previous Work 13
2.2 Sense Discrimination Tool - Motivation 16
Sense Discrimination Tool 17
3.1 How to use “Sense Discrimination Tool”? 18
3.2 Navigation in Sense Discrimination Tool 20
3.2.1 Using “Go To Synset ID” 20
3.2.2 Using “Go To Synset Word” 20
Documentation 22
4.1 Tool Home 23
4.1.1 </admin/login.php> 23
4.1.2 </admin/login-action.php> 23
4.2 Logged in user interface 23
4.2.1 <index.php> 23
4.2.2 <addtocsv.php> 24
4.2.3 </admin/admin-class.php> 24

4.3 Administrator Center 25

4.3.1 </admin/admincenter.php> 25

4.3.1.1 Pending List 25

4.3.1.2 Registered Users 25

4.3.1.4 Super User 25

Conclusions 26

Future work 27

Bibliography 28

Chapter 1

Introduction

Major Natural Language Processing applications rely on Word Sense Disambiguation, which depends on sense tagged corpus. Sense tagging involves assignment of Part of Speech tag and Wordnet Sense ID to the words in the document provided. Sense marker tool helps lexicographers in sense tagging or better known as human annotation. It assumes one sense per discourse for faster tagging, and also assumes stems of unique words in corpora are available in absence of stemmer.

1.1 Word Sense Disambiguation (WSD)

Word Sense Disambiguation (WSD) is one of the toughest problems today, not only in Natural Language Processing (NLP) but also in Artificial Intelligence (AI). The problem of WSD dates back to 1950s. It basically refers to the automatic disambiguation of word senses and it has been an interest and concern since the earliest days of treating languages computationally.

According to experts, sense disambiguation is an "intermediate task", which is just a stepping stone to most Natural Language Processing tasks. Before moving into further detail, let us find out what exactly WSD is, and the various ways of looking at the problem, along with the various forms of the problem and *why the problem is favorite among NLP researchers*.

1.1.1 WSD Definitions

Initially it is important to know the basic definitions of the WSD problem and its different types. The following section deals with the evolution of WSD as a problem. In light of the motivation behind WSD, let us have a look at the colloquial and formal definitions of WSD.

1.1.1.1 General Definition

WSD is the ability to identify the meaning of words in context in a computational manner. Given a set of words (*e.g.*,, a sentence or a bag of words), a technique is applied which makes use of one or more sources of knowledge to associate the most appropriate senses with words in context. In light of the motivation behind WSD, let us have a look at the colloquial and formal definitions of WSD.

1.1.1.2 Formal definition

WSD is the task of assigning the appropriate sense(s) to all or some of the words in T, that is, to identify a mapping A from words to senses, such that $A(i) \in Senses_D(w_i)$, where $Senses_D(w_i)$ is the set of senses encoded in a dictionary D for word w_i, and A(i) is that subset of the senses of w_i which are appropriate in the context T. The mapping A can assign more than one sense to each word $w_i \in T$, although typically only the most appropriate sense is selected, that is, $|A(i)| = 1$. Where, T is a sequence of words (w_1, $w2$, . . . ,w_n).

1.1.1.3 As a Classification Problem

Word senses are the classes, and an automatic classification method is used to assign each occurrence of a word to one or more classes based on the evidence from the context and from external knowledge sources. Other classification tasks are part-of-speech tagging, named entity resolution, text categorization, etc. But important difference between these tasks and WSD is that the former use a single predefined set of classes (parts of speech, categories, etc.), whereas in the latter the set of classes typically changes depending on the word to be classified.

1.1.2 Variants of WSD

The problem of Word Sense Disambiguation can be broadly classified into two categories.

1.1.2.1 Lexical Sample (Targeted WSD)

Here, the system is required to disambiguate a restricted set of target words usually occurring one per sentence. It employs supervised techniques using hand-labeled instances as training set and then an unlabeled test set.

1.1.2.2 All-words WSD

Here, the system is expected to disambiguate all open-class words in a text (i.e., nouns, verbs, adjectives, and adverbs) therefore, they are termed as wide coverage systems to disambiguate all open-class words. Typically, such a system suffers from Data sparseness problem, as large knowledge sources are not available.

1.1.3 WSD: Heart of the NLP Problems

As mentioned earlier, due its applicability and hardness, WSD has become an area of keen interest and challenge for NLP researchers. Word Sense Disambiguation is truly the heart of all NLP problems. This very fact is illustrated in the figure below:

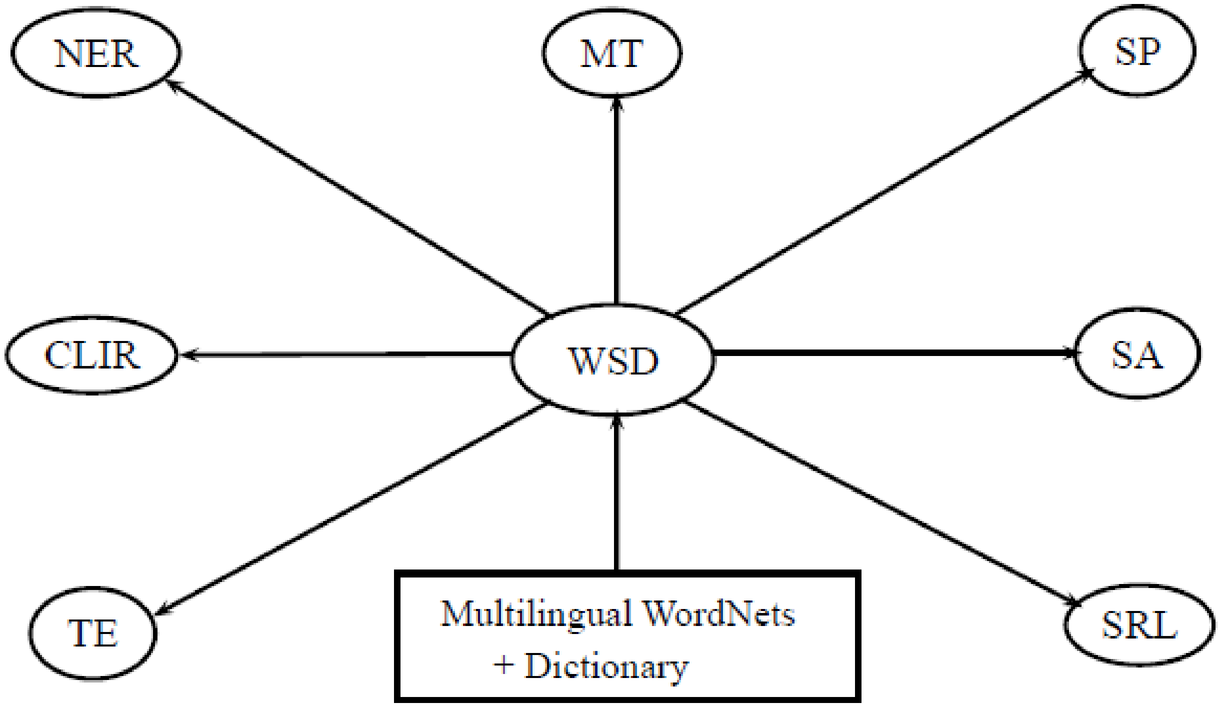


Figure 1.1: WSD - Heart of the NLP problems

- **SRL:** Semantic Role Labeling
- **TE:** Text Entailment
- **CLIR:** Cross Lingual Information Retrieval
- **NER:** Named Entity Recognition
- **MT:** Machine Translation
- **SP:** Shallow Parsing
- **SA:** Sentiment Analysis
- **WSD:** Word Sense Disambiguation

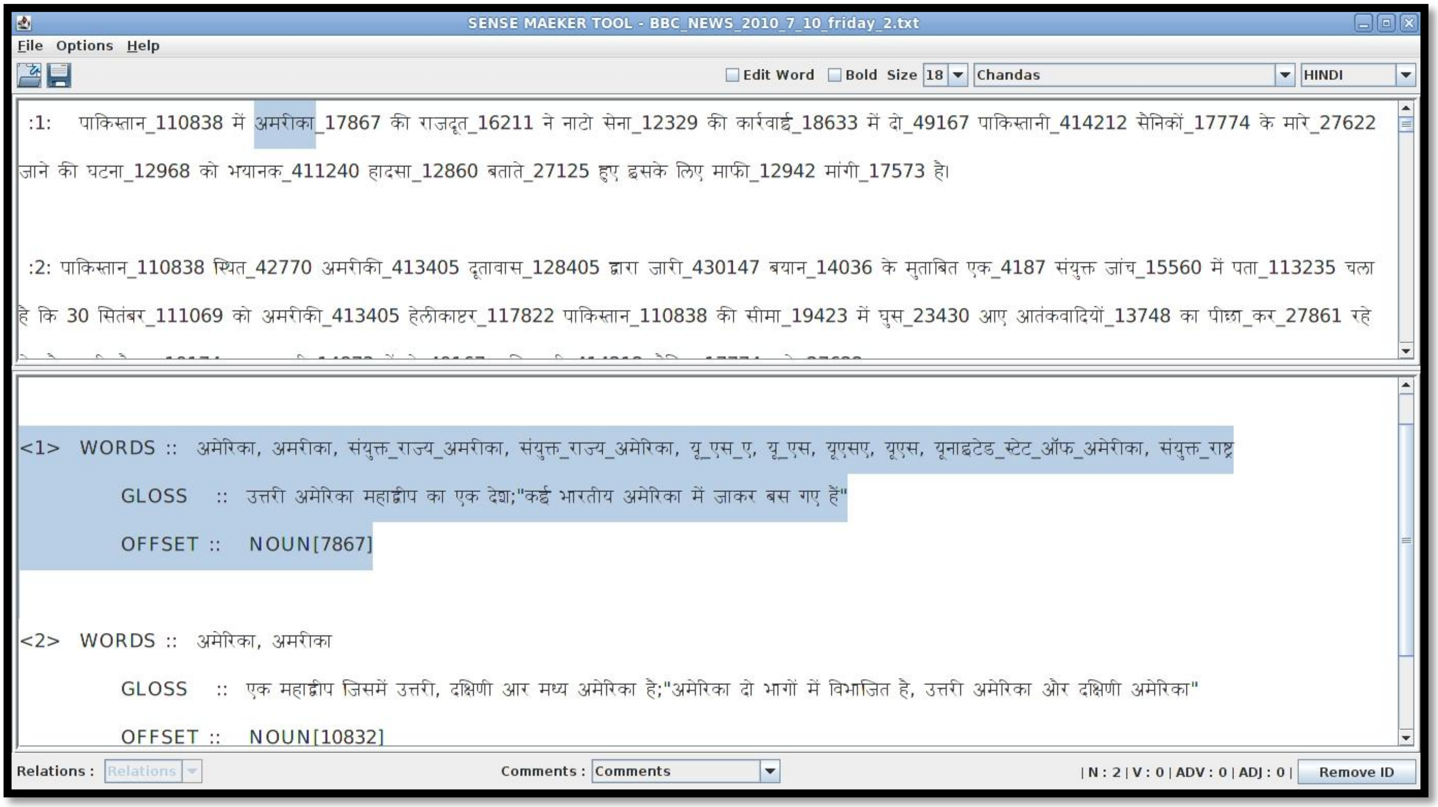
1.2 The Sense Annotation Process

In this section, we have demonstrated the backbone of our human annotation process, the sense marker tool. It serves as one single weapon to which helps our annotators mark the words with sense ids in 18 different languages. The machine needs to be trained by humans to understand the written language. Huge amount of accurately sense-marked data is supplied to the algorithm for its training.

1.2.1 The Sense Marker Tool

A word may have a number of senses and to identify and mark which particular sense has been used in the given context is known as sense marking.

At IIT Bombay, this work is being done in 3 languages – English, Hindi and Marathi. The corpus used so far have been taken from Tourism, Health, Environment and Travel review domains and the Princeton WordNet is used as the sense inventory for English text while the Hindi and Marathi WordNets have been used for Hindi and Marathi texts respectively.
The sense-marker tool developed by IITB supports 18 languages (English, Hindi, Marathi, Assamese, Bengali, Bodo, Gujarati, Kannada, Kashmiri, Konkani, Malayalam, Manipuri, Nepali, Oriya, Punjabi, Sanskrit, Telugu and Urdu).



SENSE MAEKER TOOL - BBC_NEWS_2010_7_10_friday_2.txt

□ Edit Word □ Bold Size 18 Chandas HINDI

:1: पाकिस्तान_110838 में अमरीका_17867 की राजदूत_16211 ने नाटो सेना_12329 की कार्रवाई_18633 में दो_49167 पाकिस्तानी_414212 सैनिकों_17774 के मारे_27622 जाने की घटना_12968 को भयानक_411240 हादसा_12860 बताते_27125 हुए इसके लिए माफ़ी_12942 मांगी_17573 है।

:2: पाकिस्तान_110838 स्थित_42770 अमरीकी_413405 दूतावास_128405 द्वारा जारी_430147 बयान_14036 के मुताबित एक_4187 संयुक्त जांच_15560 में पता_113235 चला है कि 30 सितंबर_111069 को अमरीकी_413405 हेलीकाप्टर_117822 पाकिस्तान_110838 की सीमा_19423 में घुस_23430 आए आतंकवादियों_13748 का पीछा_कर_27861 रहे

<1> WORDS :: अमेरिका, अमरीका, संयुक्त_राज्य_अमरीका, संयुक्त_राज्य_अमेरिका, यू_एस_ए, यू_एस, यूएसए, यूएस, यूनाइटेड_स्टेट_ऑफ़_अमेरीका, संयुक्त_राष्ट्र

GLOSS :: उत्तरी अमेरिका महाद्वीप का एक देश;"कई भारतीय अमेरिका में जाकर बस गए हैं"

OFFSET :: NOUN[7867]

<2> WORDS :: अमेरिका, अमरीका

GLOSS :: एक महाद्वीप जिसमें उत्तरी, दक्षिणी आर मध्य अमेरिका है;"अमेरिका दो भागों में विभाजित है, उत्तरी अमेरिका और दक्षिणी अमेरिका"

OFFSET :: NOUN[10832]

Relations : Relations

Comments : Comments

| N : 2 | V : 0 | ADV : 0 | ADJ : 0 |

Picture 1.2: The Sense Marker tool screenshot.

The Sense Marker tool is a Graphical User Interface based tool developed using Java which facilitates the task of manual sense marking. This tool displays the senses of the word as available in the Marathi, Hindi and Princeton (English) WordNets and allows the user to select the correct sense of the word from the candidate senses.

The table shown alongside is the statistics of sense-marking done for Tourism and Health files (for two languages: Marathi & Hindi):

Domain	Total Documents	Total Sentences	Tagged Words
Tourism Hindi	152	15,200	1,80,525
Tourism Marathi	152	15,200	1,25,387
Health Hindi	89	8,900	94,209
Health Marathi	72	7,200	51,415
Tourism English	152	15,200	1,81,964
Health English	140	14,000	1,49,259
Total	757	75,700	7,82,759

Table 1.1: *Statistics of sense marking*

1.3 Report Outline

This is an overview of the work done earlier. Clearly there is a need for the exploration of new mechnisms for tagging and thereby discriminating senses. This report describes our work done on developing a tool, which will assist in the process of clue marking for word sense disambiguation and also how its outputs can be harvested for the same. Chapter 2 outlines our motivation for going forward with a tool development. Chapter 3 gives a detailed explanation of what Sense Discrimination Tool is, and how is it used. Chapter 4 is built around the literature survey for Discrimination Net. Chapter 5 details how the tool was developed, and provides documentation for the tool. Chapter 6 includes help on clue marking and some examples of clue marking. While Chapter 7 encompasses of all our discussions, conclusion and insight from current work and possibilities of much needed future work.

Chapter 2

Motivation

2.1 Human *vs.* Machine Annotation

The process of sense annotation of words with senses is more accurate for humans than machines. The deciding parameter in the human sense disambiguation process is contextual evidence. Considering the principle of *weak AI*, the annotation procedure employed by the machine should make use of contextual evidence for disambiguation purposes in some form, which also conforms to the classical definition of WSD.

2.1.1 Previous Work

Our earlier work had presented the variance in difficulty levels of annotation across various POS and ontological categories, as recorded by the eye tracking device. The results in most cases were not stochastic, and conformed to the view of lexicographers. The second phase of the experiment, was difficult to analyze from the data recorded by the eye tracking device.

It formed a solid base for a future rule based framework, which would be self-sufficient or rather independent framework. It would be able to discriminate between polysemous word senses on its own, using context word set. We had discussed about developing a tool that would track contextual clues during annotation of a word with a particular sense would record the clue words aiding in the disambiguation from the lexicographers.

Using this we hope to develop a framework which uses the Discrimination Net constructed from the clue marked content, which would help us accurately disambiguate polysemous words, using the context word set.

The following observations were made previously from our experiments:

1. *Verbs* clearly take the highest amount of time among all the POS categories.
2. The average time taken by verbs is around 75% more than the time taken by other POS categories.
3. The average time taken for tagging a word varies greatly, depending on the skills, background and experience of a lexicographer *i.e.*, 1.619 to 5.848 sec.
4. Adjectives usually take smallest amount of time, followed by nouns and adverbs.

We had exhibited that **contextual evidence** is a *necessary attribute* for the human tagging process. Without contextual information, the human tagging process is crippled (Arindam Chatterjee, 2012). Machines, which use the *P(S|W)* statistic for WSD, take human context-sensitive information to learn the *P(S|W)* measure . This is an adaptation of the contextual evidence used by human beings. Hence the principle of *weak AI* holds for such WSD algorithms. Hence obtaining the *P(S|W)* values perfectly is of paramount concern for machines.

Supervised approaches to WSD deliver far better results, compared to knowledge-based or unsupervised methods (Navigli, Word Sense Disambiguation: A Survey, 2009). In a supervised framework, WSD is considered as a classification task, where senses of words are the classes. If we take a closer look at the state-of-the-art supervised algorithms for WSD, it will be evident that the parameters used by such algorithms are mostly statistical, *i.e., corpus-based evidence*.

WSD researchers have tried to incorporate contextual support in the form of syntactical features, co-occurrence statistics and so on, but these algorithms do not perform significantly better over the Most Frequent Sense baseline. WSD researchers have tried to incorporate contextual support in the form of syntactical features, co-occurrence statistics and so on, but these algorithms do not perform significantly better over the Most Frequent Sense baseline.

We successfully demonstrated:

- Contextual information is paramount for humans while disambiguating sense of a word.

- The annotation process of tagging without the context is cognitively strenuous and time consuming as compared to tagging with help of the context.

- In the case of machines, the *P(S|W)* measure can fetch high accuracies, provided that it has been correctly captured in the corpus by human beings, during annotation process. This in turn necessitates annotations with the help of context.

- In the case of machines, the *P(S|W)* measure can fetch high accuracies, provided that it has been correctly captured in the corpus by human beings, during annotation process. This in turn necessitates annotations with the help of context.

- WSD algorithms, if trained on corpus generated through Context Agnostic annotation process, would result in low accuracies, as the *P(S|W)* parameter is not efficiently captured in this case.

- Once the training process is over and *P(S|W)* statistic is captured, machines do not require further contextual information while annotating, unlike human annotation process. From this perspective, machines do not ape the human annotation technique but, through an adaptation of this technique provide high ac-curacies. Hence, machines conform to the principle of *weak AI* with respect to the an-notation process.

In case of machines, we have observed that the *P(S|W)* statistic is the machine's adaption of human context sensitive annotation process and the principle of weak AI is satisfied here. However, the accuracies for WSD algorithms are not yet at par with human annotation quality. For this, we would like to see if using better contextual parameters in the Iterative WSD scoring function and ranking the senses using a balanced formulation between statistical and contextual parameters.

WSD algorithms are mostly supervised and use the *P(S|W)* statistic for annotation. Besides, the *P(S|W)* statistic is obtained after training on a corpus in the context sensitive setting. Hence there is absorption of contextual information in the generation of the *P(S|W)* values from the context sensitive training data.

The need was felt for a tool, which could help us get a deeper insight into the human mind, while disambiguating polysemous words. In human mind, Sense disambiguation highly depends on finding clues in corpus text, which finally lead to a winner sense. Such clues to finding word meanings needed to be found out. Hence, We developed a tool which could help a lexicographer mark the clues for disambiguating a word. We called it, "Sense Discrimination Tool". In the current phase, this tool lets the lexicographer select the clues from the *gloss* and *example* fields in the synset, and adds them to a database.

2.2 Sense Discrimination Tool - Motivation

In Section 2.1 we have seen that human beings use contextual evidence as a prime parameter for sense disambiguation. This motivated us to look in depth at how human annotation is done.

Human annotators *form a hypothesis* as soon as they start reading the text, reaching the target word while reading might be sufficient to gain enough evidence to disambiguate it, in some cases even reading the whole text might not give sufficient clues to disambiguate a word. *Machines have no such facility*. **The paragraph** that the annotator is reading **always gives him a vague idea of the word sense**. In fact, **the domain** of the text being annotated gives away the most appropriate sense's idea (Mitesh Khapra, 2009). Also, being familiar with the text beforehand stimulates the idea of a winner sense in the mind. Hence, to assure genuineness of our experiment we separated line from different documents of the corpus and jumbled them up, such that, each sentence of text is taken from a different sort of contextual background.

The cognitive load on the human brain while annotating the text is much more than one can imagine. As our expert lexicographers narrate, the hypothesis formation and rejection, work hand in hand as the senses are first narrowed down to a few most probable senses and then the winner sense is selected on the basis of matching the word with the gloss provided along with the sense.

One of the more important factors is the replaceability of synonyms provided along with in the sense window, if somehow narrowing down to a few senses and gloss matching tests are not enough, replaceability of the synonyms give the annotator a better understanding of the sense, which also works as verification in many cases.

The above mentioned factors along with the rich knowledge background form firm sense identification basis in one's mind and decides on an appropriate winner sense. Humans have a more powerful very imaginative visual sense of thinking, hence reading text stimulates visual background in a mind and this is again a very helpful factor in disambiguating a word written within a piece of text.

Hence the process of human annotation differs from machine completely.

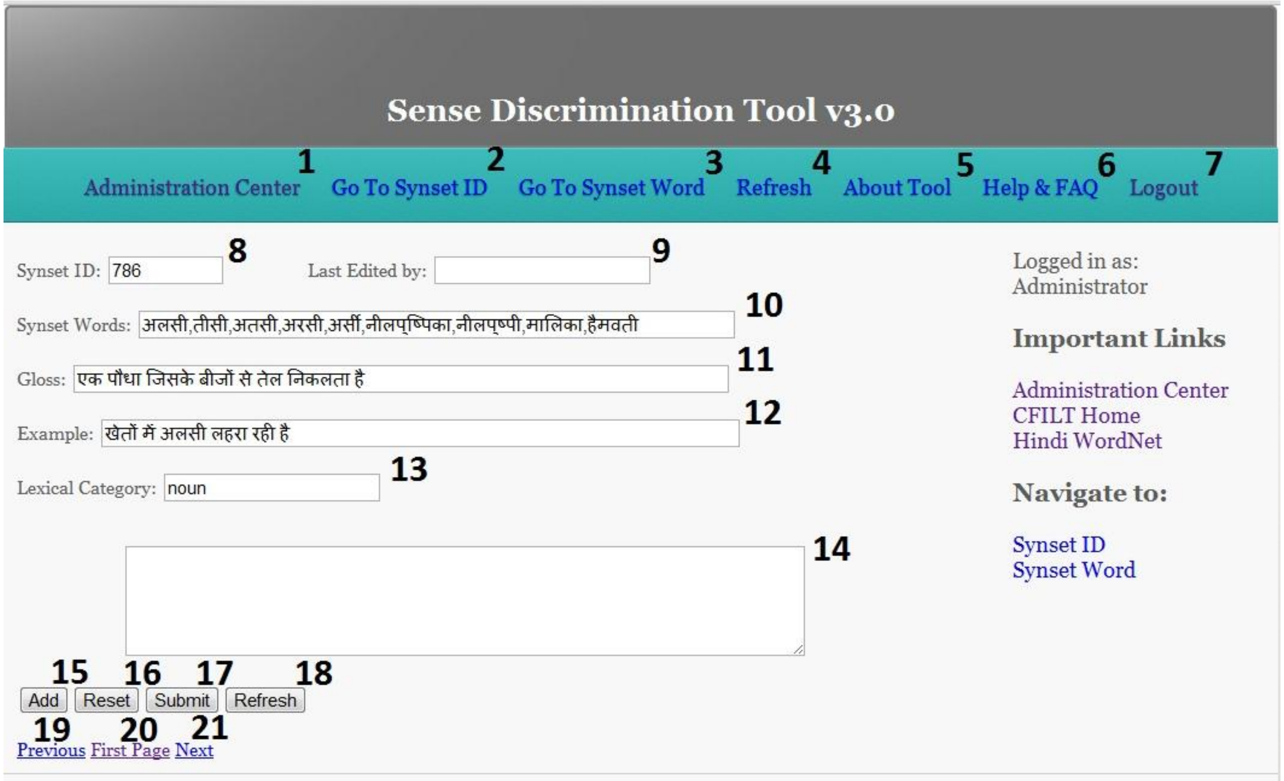
To study this process deeply, the clues which influence the decision of winner sense in a lexicographers mind need to be known to us. Hence, we went forth with the development of this tool, which lets us collect these clues, which would form base for a solid rule based framework in the future.

Chapter 3

Sense Discrimination Tool

Here we present an overview of the tool, and a detailed explanation of its current interface. In the current phase, we tend to collect clues and later user them for "Discrimination Net". Hence, in its current phase "Sense Discrimination Tool" adds manually entered sense clues to the database.

Given below is the detailed screenshot of the tool.



Synset ID: 786

Last Edited by:

Synset Words: अलसी,तीसी,अतसी,अरसी,अर्सी,नीलपुष्पिका,नीलपुष्पी,मालिका,हैमवती

Gloss: एक पौधा जिसके बीजों से तेल निकलता है

Example: खेतों में अलसी लहरा रही है

Lexical Category: noun

The Sense Discrimination Tool home page is shown above and it is described below:

1. **Administration Center**: For administrative users, Operations such as Approve, Reject, Ban and Super User and Delete user are present for an Administrator.
2. **Go To Synset ID**: Navigates to a particular Synset ID.
3. **Go To Synset Word**: Navigates to a particular Synset word, based on choice of user.
4. **Refresh**: Refreshes the page for showing updated clue words.
5. **About**: Opens a page explaining the tool
6. **Help**: Opens a page on how to use the tool, and who should use the tool.
7. **Logout**: Logs a user out.

Main Window:

8. Displays the **synset ID**.
9. Displays the **user id of the user who last edited the clue words** for this synset.
10. Displays the **Synset Words**.
11. Displays the **gloss** of the word in Hindi WN.
12. Displays the **example** of the word usage present in Hindi WN.
13. Displays the **Lexical Category** of the word.
14. **Clues text box** where user keeps adding the clues and can edit them before final submission.
15. **Adds the selected text** on the page to the Clues Text box[14].
16. **Resets the Clues Text Box[14]** for fresh clues addition.
17. **Submits the final entries in the Clues Text Box** in the database.
18. Refreshes the page.
19. Navigates to **Previous Synset** ID.
20. Navigates Back to **First Synset** ID.
21. Navigates to **Next Synset** ID.

3.1 How to use "Sense Discrimination Tool"?

Step One:

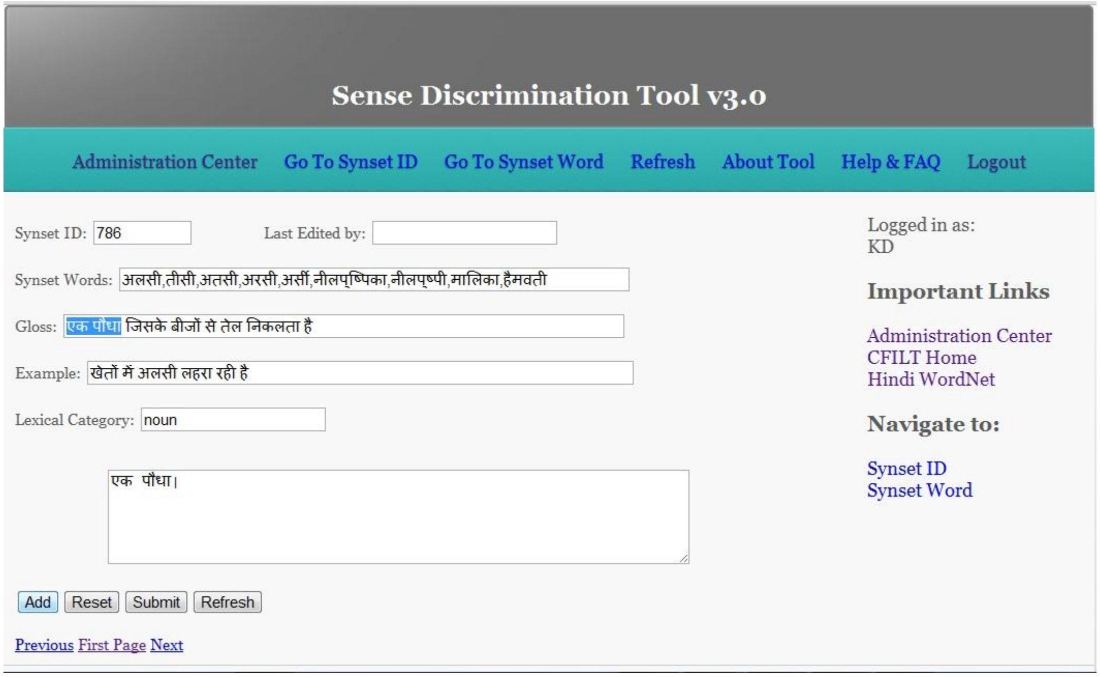
If you do not have a login ID and password, Kindly Create one by going to the Registration Page by clicking on the Create Login button on the Login Page.

Your ID has to be approved by the CFILT SysAd for a valid Email ID and only after approval you can login to work with the tool.

Step Two:

Once you have been approved, Login using your credentials and you will be taken to the tool Home Page show above.

Step 3: You have to identify the synset word in Synset Words[10] and select the word/phrase which you think helps disambiguate the word meaning and leads to the winner sense. Select that word/phrase using mouse or using SHIFT key on the keyboard. The clues will be available in gloss[12] and example[13].



Synset ID: 786

Last Edited by:

Logged in as:
KD

Synset Words: अलसी,तीसी,अतसी,अरसी,अर्सी,नीलपुष्पिका,नीलपुष्पी,मालिका,हैमवती

Gloss: एक पौधा जिसके बीजों से तेल निकलता है

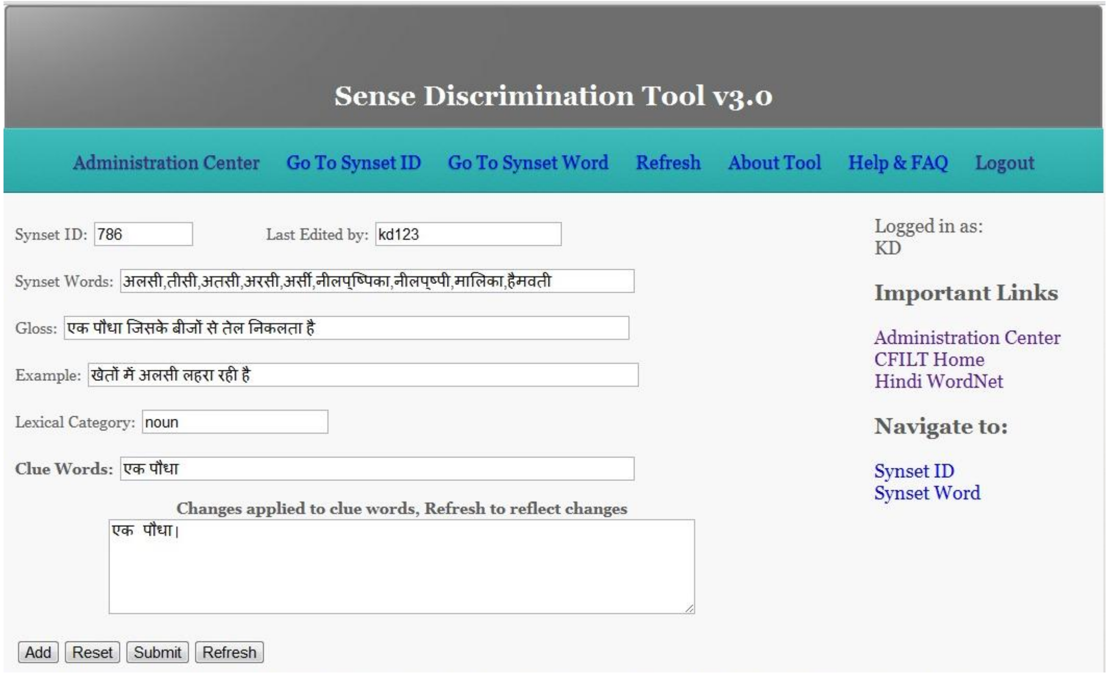
Example: खेतों में अलसी लहरा रही है

Lexical Category: noun

एक पौधा।

Step 4: Now, Click on add to add them to the Clues Text Box[14] and edit them for any changes, if needed. Make the clues set final for addition to database.

Step 5: Click on Submit to add the phrases to the database.



Synset ID: 786

Last Edited by: kd123

Logged in as:
KD

Synset Words: अलसी,तीसी,अतसी,अरसी,अर्सी,नीलपुष्पिका,नीलपुष्पी,मालिका,हैमवती

Gloss: एक पौधा जिसके बीजों से तेल निकलता है

Example: खेतों में अलसी लहरा रही है

Lexical Category: noun

Clue Words: एक पौधा

Changes applied to clue words, Refresh to reflect changes

एक पौधा।

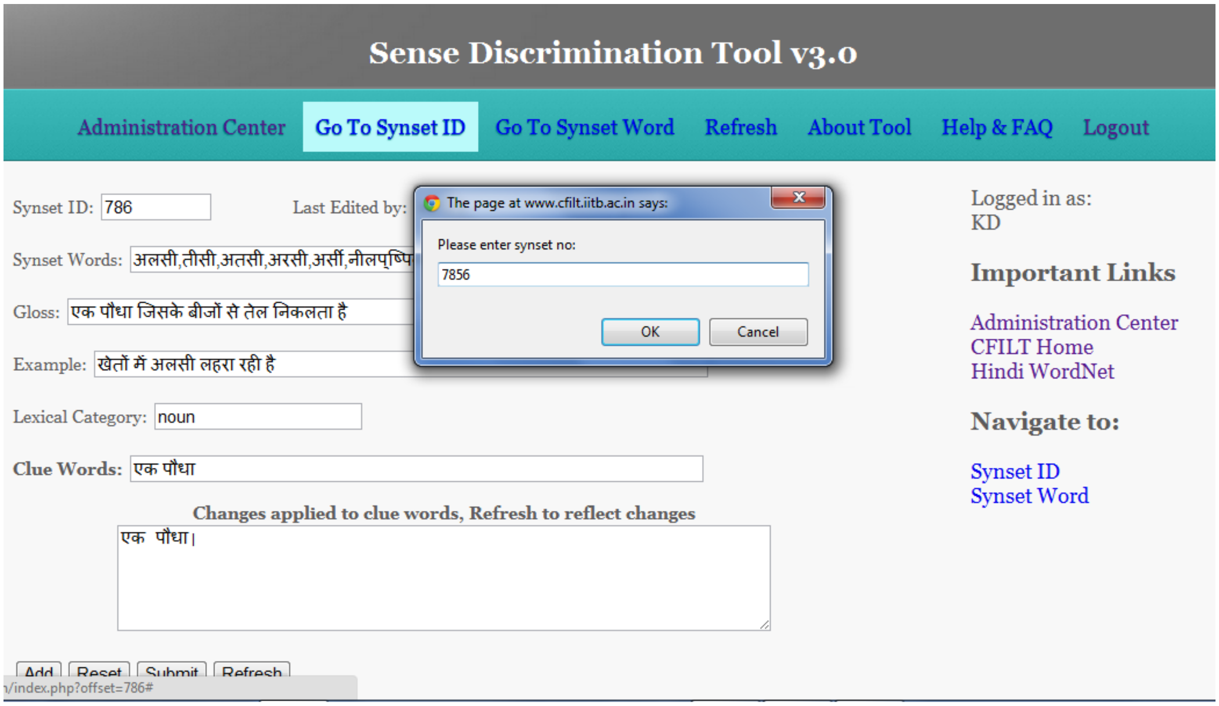
Step 6: After Submission, a new Text Field containing the added clue words will appear on the page. Navigate[2/3] to the next Synset ID, or to any Synset ID / Word as per your choice.

Step 7: After finishing, Click Logout[7] on the top right corner.

3.2 Navigation in Sense Discrimination Tool

3.2.1 Using "Go To Synset ID"

Clicking on Go To Synset ID in the navigation bar on top or in the sidebar, will pop up a small windows asking for target Synset ID.



Synset ID: 786

Last Edited by:

Synset Words: अलसी,तीसी,अतसी,अरसी,अर्सी,नीलपुष्पि

Gloss: एक पौधा जिसके बीजों से तेल निकलता है

Example: खेतों में अलसी लहरा रही है

Lexical Category: noun

Clue Words: एक पौधा

Changes applied to clue words, Refresh to reflect changes

एक पौधा।

The page at www.cfilt.iitb.ac.in says:

Please enter synset no:

7856

Logged in as:
KD

/index.php?offset=786#

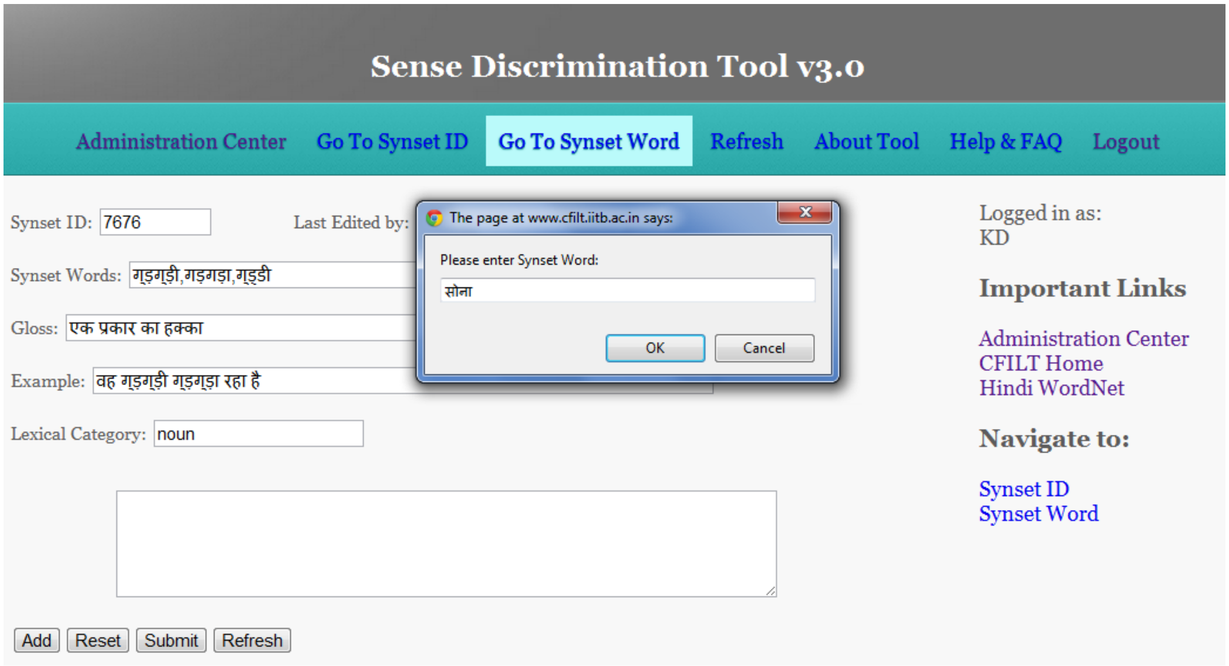
Enter the synset ID number in the text box pop up and click on OK to Navigate to the particular Synset ID.

3.2.2 Using "Go To Synset Word"

Clicking on Go To Synset Word in the navigation bar on top or in the sidebar, will pop up a small windows asking for target Synset Word. You can type in the synset word by changing your typing alternative to hindi, or you can copy a hindi word / its part from the Hindi WordNet.

For Convenience, Hindi WN link is provided in the sidebar.

The screenshot including the text box pop up is shown below:



Synset ID: 7676

Last Edited by:

Synset Words: गड़गड़ी,गड़गड़ा,गड़्डी

Gloss: एक प्रकार का हक्का

Example: वह गड़गड़ी गड़गड़ा रहा है

Lexical Category: noun

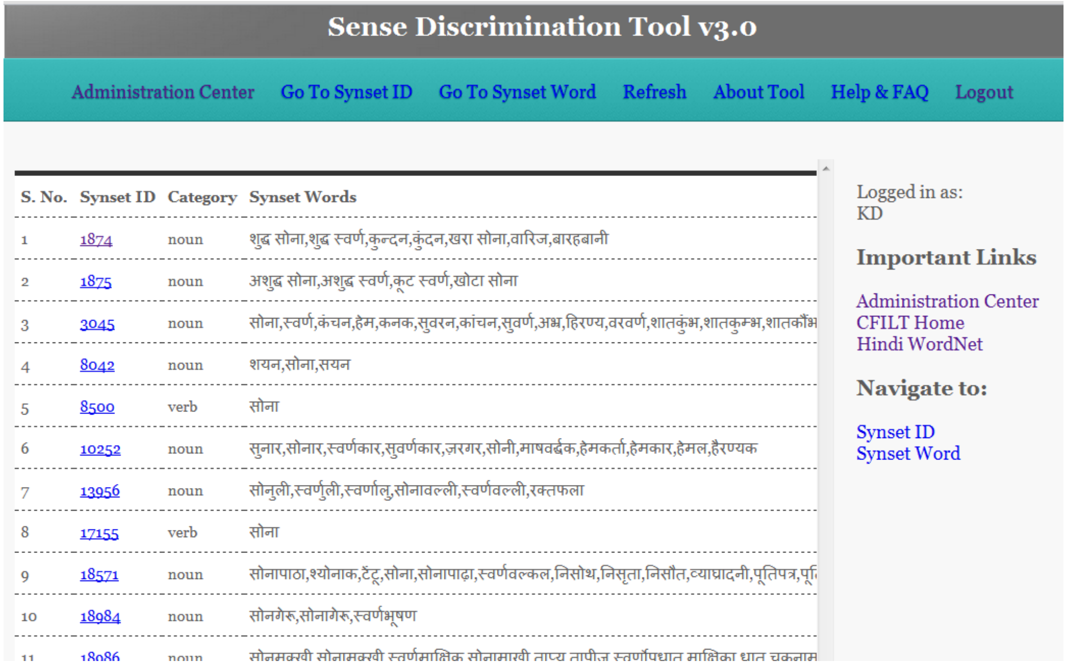
The page at www.cfilt.iitb.ac.in says:

Please enter Synset Word:

सोना

Logged in as:
KD

After entering the word in the text box, Click OK or Press Enter. The navigation will take you to a page where all the resulting instances of the input word in the Hindi WN database are present.



S. No.	Synset ID	Category	Synset Words
1	1874	noun	शुद्ध सोना,शुद्ध स्वर्ण,कुन्दन,कुंदन,खरा सोना,वारिज,बारहबानी
2	1875	noun	अशुद्ध सोना,अशुद्ध स्वर्ण,कूट स्वर्ण,खोटा सोना
3	3045	noun	सोना,स्वर्ण,कंचन,हेम,कनक,सुवरन,कांचन,सुवर्ण,अभ्र,हिरण्य,वरवर्ण,शातकुंभ,शातकुम्भ,शातकौंभ
4	8042	noun	शयन,सोना,सयन
5	8500	verb	सोना
6	10252	noun	सुनार,सोनार,स्वर्णकार,सुवर्णकार,ज़रगर,सोनी,माषवर्द्धक,हेमकर्ता,हेमकार,हेमल,हैरण्यक
7	13956	noun	सोनुली,स्वर्णुली,स्वर्णालु,सोनावल्ली,स्वर्णवल्ली,रक्तफला
8	17155	verb	सोना
9	18571	noun	सोनापाठा,श्योनाक,टेंटू,सोना,सोनापाढ़ा,स्वर्णवल्कल,निसोथ,निसृता,निसौत,व्याघ्रादनी,पूतिपत्र,पूि
10	18984	noun	सोनगेरू,सोनागेरू,स्वर्णभूषण
11	18086	noun	सोनमक्खी,सोनामक्खी,स्वर्णमाक्षिक,सोनामाखी,ताप्य,तापीज,स्वर्णोपधातु,माक्षिका धातु,चक्रनाम

Logged in as:
KD

From here, you can click on the target synset word you want to go to and the tool will open the page on that particular synset ID.

Chapter 4

Documentation

The documentation of this tool contains both black box and white box type documentation. First, we define the logic wise flow and what the tool entails. The tool is an online PHP based interface which works by importing the data from Hindi WN database. We have used MySQL as the back end for data manipulation and storage. The tool starts by displaying a login page where a user must enter his credentials to enter the tool. Unregistered users are required to click on the create login button to go create a login user id and password for them, and their login must be approved by a CFILT administrator or by any of the registered Super Users on the website.

Once the login id is created and approved, a user can log in to the tool and start using it from the home page itself. Home page of the tool is <index.php> contained in the root. It displays extracted data from the database on the first page with the first synset id i.e. <synset id = 1>. The user gets to start clue marking from here itself. Synset words and Synset ID is displayed on the top along with a text box displaying the username of the user who last edited the current clue words, if ever edited. If there is no text field labeled clue words present on the page, there are no entries for the clues of this synset in the database.

User now has to identify the clue words in the *gloss* and *example* fields of the page displayed. Clues can be words or word phrases depending on the users interpretation of the synset word along with the lexical category it belongs to. Once the user selects clues with mouse selection on the screen, He clicks the add button to put them down in the Clues Text Box given below. User adds as many clues as possible and when the clues are finally complete, He should click Submit button to submit the clues in the database. The clues are added to the database and changes are reflected immediately in most cases. Due to some problems in specific browsers, if the clue changes are not being reflected immediately, The user should click refresh to check the clue changes made in the database. Clicking on refresh will fetch entries of clues from the database immediately.

If there are some clues already added to the database, and user wants to edit them, the clues text box it editable and user can edit the clues present there, when done with the editing, clicking on the submit button is again required to update the clues entry in the database.

It is advised that user only edits the clues if he has a complete idea about the synset word he is presently editing.

The description of the flow of logic in our tool interface is give below:

4.1 Tool Home

Link to the tool:

www.cfilt.iitb.ac.in/~diptesh/

which in turn takes you to /index.php

Case 1: If the user is already authenticated [Session variable set from an earlier session in browser]

- /index.php is displayed and the tool can be used from here.

Case 2: If the user is unauthenticated

- User is taken to a page /admin/login.php to enter his login credentials, and then to /index.php.

4.1.1 </admin/login.php[1]>

The interface displays simple text boxes in form with action set to execute <login-action.php> and passes the values of username and password entered to the above mentioned file.

Create user button on the file takes user to </admin/register.php> which displays simple text fields which are used to create a user id and password for a user in the tool database.

4.1.2 </admin/login-action.php[1]>

It simply calls to include </admin/admin-class.php> and passes $admin variable to class itg_admin(), to be authenticated by function _login_action().

4.2 Logged in user interface

4.2.1 <index.php>

The tool interface is simple and the data is extracted and displayed on this one page in text fields.

Synset ID: $sid

Synset Words: $words

Gloss: $comb[0]

Example: $comb[1]

Lexical Category: $category

Clue Words: $clues

The data set consists of a field called clue words, which can be manipulated by the user based on his knowledge.

The interface consists of links to next, previous and first synset id which navigate the same page to extracting data of the target synset id and displaying it in the same interface.

After adding the clues to the Clues Text Box, when user clicks on Submit button, a call is made to </addtocsv.php> which adds the clues to the database.

4.2.2 <addtocsv.php>

A connection is made to the database 'cluemarker' and table 'tbl_all_synset' which contains column 'clues', which then is updated with a MySQL query replacing it with all what is within Clues Text Box.

4.2.3 </admin/admin-class.php[1]>

It calls to include </db/db.php> which makes a connection to the database to extract information about user id in the table 'user' and has all the fuctions related to registration of a user, login of a user, verification, authentication etc. The file contains proper comments in the source code, for further changes.

_login_action() – for login verification.

_register_action – for registration submission.

_check_db() – checks the database for correct username and password, private function.

_authenticate() – for authentication verification of a user, should be used before opening tool interface. Tool opens only if user is authenticated and this is done using this function.

[1] A simple version of tool login system was downloaded from www.intechgrity.com and the modified according to our pupose.

4.3 Administrator Center

Administration Center is the tool interface for administrator login, which opens up only when an authenticated administrator login in done through tool home. Administrator login redirects to </admin/admin-center.php> which is displayed only for Administrator status users.

This interface has all the user lists i.e. Pending Approval, Registered users

4.3.1 </admin/admincenter.php>

This file contains lists of users who are registered to use the tool i.e. approved by the admin, or a super user, and also the list of users whose registration is pending an approval.

4.3.1.1 Pending List

This list is only displayed if there are any users who have registered for the tool, and have not yet been approved by admin. Also, the users who are banned from using the tool, are brought here, and they cannot login again, until approved by the admin. It displays and option to "Approve" them or "Reject" them.

4.3.1.2 Registered Users

This list displays the user who are registered as normal users and are approved to view and use the tool. They have no access to administration center whatsoever. The list contains options to "Ban" them, "Delete" them, or promote them to "Super User" status.

4.3.1.4 Super User

Super users have access to administration center and can go to administration center through a link on index.php i.e. tool interface. Super users can approve or reject members in pending users list. They can also ban users who are creating nuisance in the database. Super users cannot disable other super users, that privilege lies with the admin.

Chapter 5

Conclusions

The development of a tool which allows for annotators to conveniently specify the clues that they use for distinguishing between the various senses of a word is quite crucial in the task of word sense disambiguation. It is further important to utilize these clues so as to build a structure or a framework which allows for reducing the uncertainty of the sense of a particular word.

We imagine that constructing a discrimination net in the form of a weighted graph will assist in calculating a score which will say something about this uncertainty. The underlying idea is that there are words with multiple senses as well as ones with unique senses present and by traversing this graph, we will eventually reach these unique senses and then determine the score.

The discrimination net can help us disambiguate the senses of all the words present in a sentence or corpus text fed to it. We hope that this framework will bring about a newer understanding of WSD. We believe that these clues will give us an in depth analysis of how humans disambiguate and using them, if we can bring down the noise in word senses using these clues, We will easily be able to reach the target sense. Bringing down the noise or the entropy as we have thought of it, will depend on positive clues, which will be available in our database.

Clues initially, in this phase are only being manifested by users manually. In the next phase, we plan to automate the clue inclusion but that can only be done is we have enough clues to automate the database searching for related synset words. The related synset words can lead us to more clues, which can help disambiguate related words in the corpus text.

The tool in its first phase is completely working and can be found working online on:

www.cfilt.iitb.ac.in/~diptesh

Chapter 6

Future work

In this report, we have presented the development of a tool which facilitates the collection of clues which help in disambiguating a synset word. These clues can further help in the sense disambiguation of the synset when we build a "Discrimination Net". This tool does not yet retrieve the clues automatically or in any way helps discriminate the sense of a word.

But it does form a solid base for a future rule based framework. It would be self-sufficient or rather and independent framework which would be able to discriminate between polysemous word senses on its own, using the context word set. We have had limited no. of instances for each word as well, which if increased substantially, could lead to a healthy coverage of all word sense meanings.

This tool would track contextual clues during annotation of a word with a particular sense would record the clue words aiding in the disambiguation from the lexicographers themselves hence avoiding future errors.

Our limitation still is the collection of clue sets, which annotators have to feed in manually. Our rule based framework will be built only if we have substantial amount of clues present in the database. We have to keep working on obtaining clues, or else, without clues, we are handicapped and cannot work further on this path.

References

Bibliography

Agirre, & Rigau. (1996). Word sense disambiguation using conceptual density. *16th International Conference on Computational Linguistics.* Copenhagen, Denmark.

Arindam Chatterjee, S. J. (2012). A Study of the Sense Annotation Process: Man v/s Machine. *International Conference on Global Wordnets (GWC 2011)*, (p. 8). Matsue, Japan.

Google. (n.d.). Retrieved from http://www.google.co.in

Lesk, M. (1986). Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. *5th annual international conference on Systems.* Toronto, Ontario, Canada.

Merja Lehtinen, K. S. (2005). *Usability Research Methods.* University of Tampere, Department of Computer Sciences.

Mitesh Khapra, S. S. (2009). Projecting Parameters for Multilingual Word Sense Disambiguation. *Empirical Methods in Natural Language Prfocessing (EMNLP09).* Singapore.

Navigli, R. (2009). Word Sense Disambiguation: A Survey. *ACM Computing Surveys*, 69.

Navigli, R., & Velardi, P. (2005). Structural Semantic Interconnections: A Knowledge-Based Approach to Word Sense Disambiguation. *IEEE Transactions On Pattern Analysis and Machine Intelligence.*

Rada, M. (2005). Large vocabulary unsupervised word sense disambiguation with graph-based algorithms. *Joint Human Language Technology and Empirical Methods in Natural Language Processing Conference (HLT/EMNLP)*, (pp. 411-418). Vancouver, Canada.

Walker, & Amsler. (1986). The Use of Machine Readable Dictionaries in Sublanguage Analysis. *In Analyzing Language in Restricted Domains*, 69-83.

Wikipedia. (n.d.). Retrieved from http://en.wikipedia.org/wiki/Wiki

InTechgrity. (n.d.). Retrieved from http://www.intechgrity.com/