Discrimination-net for Hindi

Diptesh Kanojia Arindam Chatterjee Salil Joshi
Pushpak Bhattacharyya
(1) Gautam Budh Technical University, Delhi, India
(2) Symantec Labs, Pune, India
(3) IBM Research India, Bangalore, India
(4) CSE Department, IIT Bombay, Mumbai, India
dipteshkanojia@gmail.com, arindam.chatterjee23@gmail.com,
saljoshi@in.ibm.com, pb@cse.iitb.ac.in

ABSTRACT
Current state-of-the-art Word Sense Disambiguation (WSD) algorithms are mostly supervised and use the $P(Sense|Word)$ statistic for annotation. This $P(Sense|Word)$ statistic is obtained after training the model on an annotated corpus. The performance of WSD algorithms do not match the efficiency and quality of human annotation. It is therefore important to know the role of the contextual clues in WSD. Human beings in turn, actuate the task of disambiguating the sense of a word, by gathering hints from the context words in the neighbourhood of the word. Contextual clues thus form the basic building block for the human sense disambiguation task. The need was thus felt for a tool, which could help us get a deeper insight into the human mind, while disambiguating polysemous words. As mentioned earlier, in the human mind, sense disambiguation highly depends on finding clues in corpus text, which finally lead to a winner sense. In order to make WSD algorithms more efficient, it is highly desirable to assimilate knowledge regarding contextual clues of words. In order to make WSD algorithms more efficient, it is highly desirable to assimilate knowledge regarding contextual clues of words, which aid in finding correct senses of words in that context. Hence, we developed a tool which could help a lexicographer mark the clues for disambiguating a word in a context. 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.

KEYWORDS: Sense discrimination, tool for generating discrimination-net.

1 Introduction

Human annotators form a hypothesis as soon as they start reading the text. When they reach the target word sufficient information is gathered and they gain enough evidence to disambiguate it. Although in some cases, even reading the whole textmight 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 senses idea (Khapra et al., 2010). 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 lines from different documents of the corpus and altered their order, such that, each sentence of text is taken from a separate sort of contextual scenario.

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 ones 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.

The key features of the system are as follows:

1. **Minimized human interaction:** The system requires the user to provide very less amount of input. All the user has to do, is to select on the contextual clues once a synset is displayed.
2. **User friendliness:** Our system interface provides a nice visual experience. The design of the interface makes the operation of the interface completely clear to the end user.
3. **System independence:** The system is independent of the web browser and the operating system it is used on. Since the business logic is written in Java, the system can be easily ported on another machine.
4. **Ease of configuration:** Our system currently uses the Hindi wordnet as the back-end knowledge source. However, it can be easily ported to support any language wordnet.
5. **Easily interpretable output:** Our interface is designed in such a way that the user can easily understand the ongoing process and the clues entered so far.

This paper is organized as follows. Section 2 lays down the system architecture of our wordnet linking system. In section 3 we describe the operation of the tool. We conclude the paper with section 4.

2 System Architecture

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 an 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. 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.

3 Interface Design

To support various web browsers on different operating systems, we have designed the web interface using standard open technologies. The interface runs using PHP5[1] on the server side, and the back-end database is maintained using MySQL[2]

Figure 1 shows the system interface. 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.

Once the user has a login approved, (s)he needs to follow the steps mentioned below to use the tool:

[1] `http://php.net/downloads.php`
[2] `http://mysql.net/`

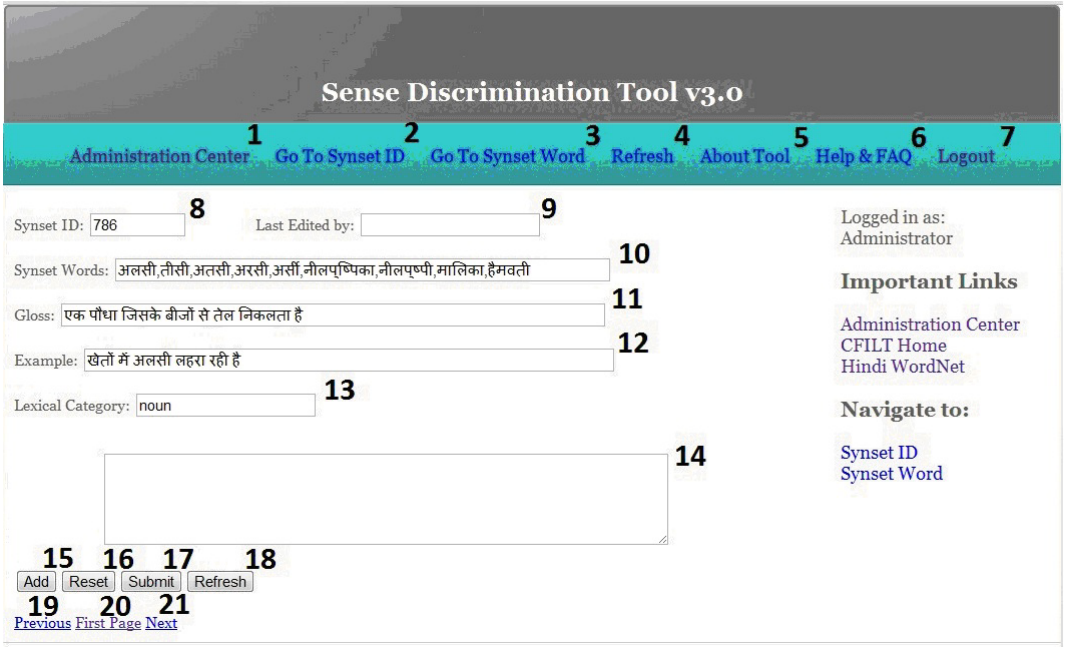

Figure 1: Screen-shot showing the main interface of the system

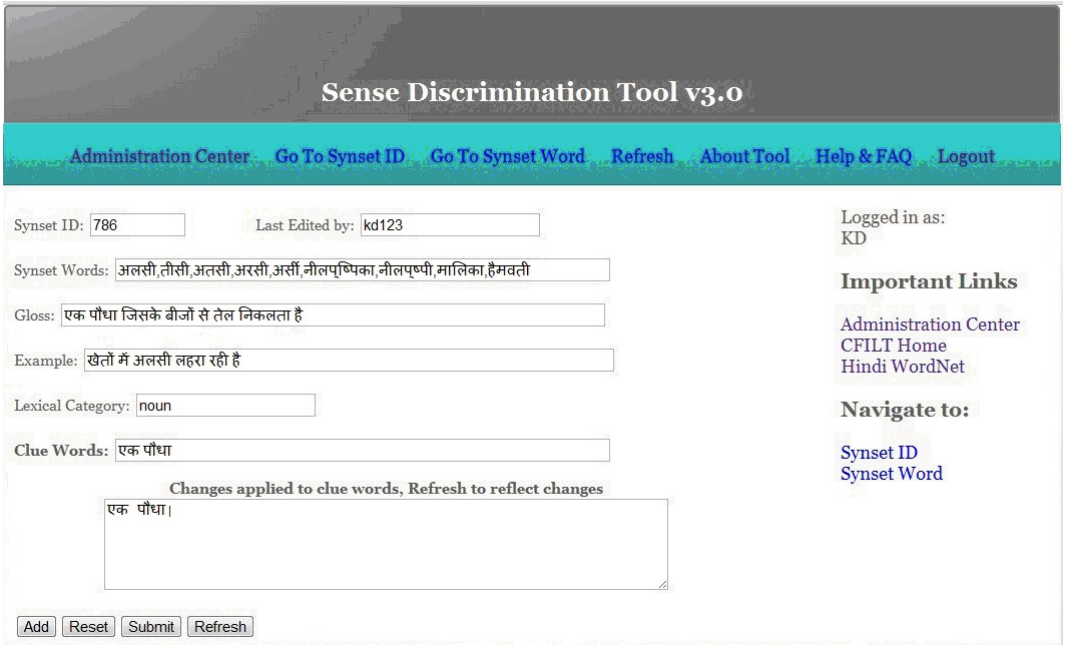

Figure 2: Screen-shot showing the working of the system

1. **Identify the synset word :** user has to identify the synset word in Synset Words and select

the word/phrase which you think helps disambiguate the word meaning and leads to the winner sense. Selection can be made by highlighting that word/phrase using mouse or using SHIFT key on the keyboard. The clues will be available in gloss and example.

2. **Gather the clues:** User needs to click on add to add them to the Clues Text Box14 and edit them for any changes, if needed. This makes the clues set final for addition to database.
3. **Submit the clues:** The user can then simply click on Submit to add the phrases to the database.
4. **Navigation across synsets:** Once the synset is done with, the user can move to the next or previous synset. This working is shown in figure 2.

The tool also provides several other facilities for searching a particular word, carrying out administrative tasks, searching for a particular synset, *etc*. Figure 3 shows the operation of searching a word. After entering the word in the text box, the user needs to click OK or press Enter. The navigation will take the user to a page where all the resulting instances of the input word in the Hindi WN database are present.

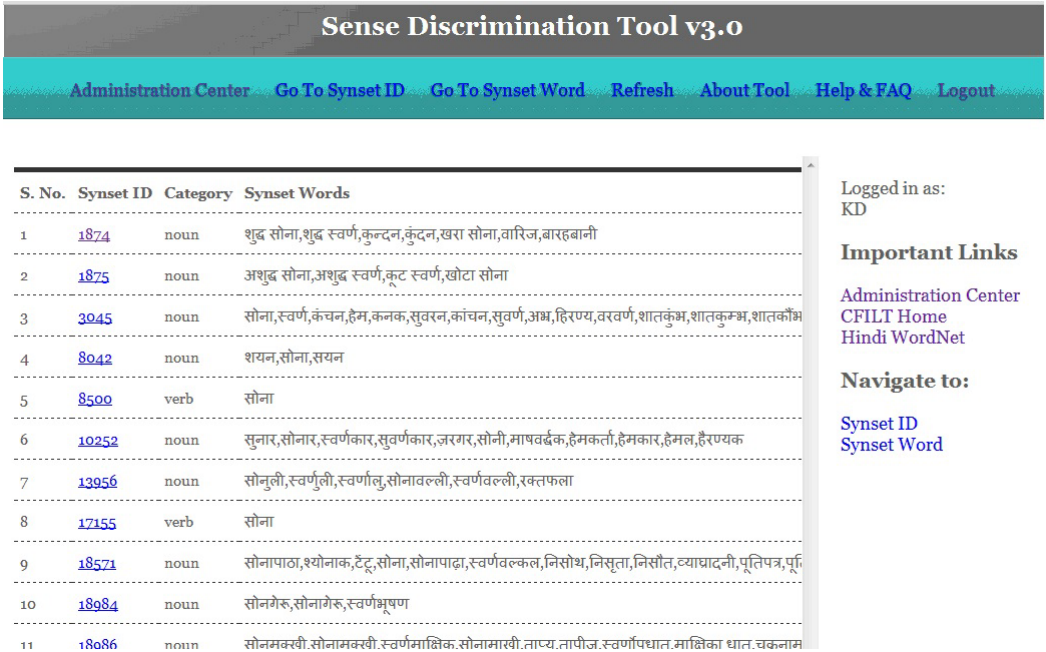

Figure 3: Screen-shot showing the navigational facility of the system

4 Conclusion

The aim of this paper was to illustrate a tool which allows 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, and by traversing this

graph, we will eventually reach these unique senses and then determine the score.

In the future, we would like to help the users in generating the clues using the clues which are accumulated in the system so far.

References

Khapra, M., Shah, S., Kedia, P., and Bhattacharyya, P. (2010). Domain-specific word sense disambiguation combining corpus based and wordnet based parameters. In *5th International Conference on Global Wordnet (GWC2010)*.