**NATURAL LANGUAGE PROCESSING**

**PROJECT**

**TOPIC: MATCHER**

Submitted by,

Praveen Erode Murugesan

pem150030

**PROBLEM DESCRIPTION**

**Problem Statement:**

The project aims at finding the appropriate service provider for a particular consumer where both of them expresses their interests in a common forum.

**Explanation:**

Let us consider a common forum where users expresses their interests and needs. The users are classified as a consumer or a service provider based on their expressed statement. The service needed by a consumer is extracted from the statement and checked against all the service providers’ statements, the ones that gets matched are listed as result to that particular consumer.

Eg:

Consumer Statements Service Provider Statements

I need a cab I sell vegetables

I want to buy carrots I am a taxi driver

\*Note - matching statements are shown in same color

**PROPOSED SOLUTION**

The solution for this problem can be designed by extracting syntactic, semantic and lexical features from the statements expressed by the user. The problem is solved in two steps. They are

1. Classifying the user statement as producer and consumer
2. Finding the matching results in case consumer statements

To complete these two steps, a POS Tagger(Syntactic feature) has to be used to find the verbs and nouns of the sentence.

(i) can be solved by finding the verbs from the tagged sentence and its associated meaning (eg: need – consumer verb, sell – producer verb)

(ii) The service needed or provided by the user has to be extracted from the statement. This is done by finding the nouns after the verbs.

* In case of producer statement, the statement is stored with extracted services
* In case of consumer statements the extracted service is used to find the match with the producer statements that are already stored

The system aims at matching the consumer needs(Lexical feature) not only with the exact producer queries but also with the appropriate related queries(Semantic feature) that could potentially satisfy the consumer.

Eg:

Consumer statement: I need coffee

This statement gets matched with following producer statements

I supply coffee

I sell tea

I provide hot beverages., etc..

The related forms are retrieved from wordnet library.

**ARCHITECTURE & TOOLS**

**Architecture Diagram:**

C:\Users\Praveen\Downloads\NLP_Project.png

**Programming Tools:**

Programming Language: Java

Packages used: Stanford POSTagger , Wordnet, Porter Stemmer

Web server Call: WS4j Demo (http://ws4jdemo.appspot.com/)

**IMPLEMENTATION**

**Data Store:**

There are a few data that is stored to implement the system. A set of verbs which are classified as producer, consumer and neutral are stored in a file. These verbs are taken from the thesaurus. The classified consumer and producer statements are stored in two different files for later use(matching).

**Baseline System:**

Base line system is implemented for exactly matched statements. (i.e) *’I want to buy carrots* is matched only with Service Provider Statements that has *carrots* in it(*I sell carrots*). It will not match with the statement *‘I sell vegetables’*.

The user statement is POS tagged(Syntactic feature - 1) first. The verb of the statement is extracted. The verb is checked against the set of verbs stored in the file(data store) to classify the statement as producer or consumer statement. If the verb is a neutral one, then the system asks for user to choose the appropriate option.

Eg: I go to Florida.

Go – neutral verb. So the statement has two senses from the system perspective. They are

User is going to Florida in car, so he likes to car pool - Producer

User wants to go to Florida, so he is looking for a co-passenger to travel with – Need a co-passenger - Consumer

So the system asks the user to choose the appropriate option.

Once the user statement is classified, the nouns following it are extracted and stored in a corresponding repository. In case of consumer statement, the system looks for the exact match of requirements in the producer statements’ repository. The statements having the exact nouns in it are matched and retrieved as a result.

**Problems with baseline system:**

The base line system has a number of shortcomings as follows

1. The services are not normalized, so the match will not be found if either of the statements has the services in different cases (Uppercase and Lowercase).
2. Base line system will not consider the related terms of the services, so the producer statements with related services will not get listed as a part of the final result to the consumer.
3. The system do not account for interrogative sentences.

Eg: *Anyone want grocery?* - This is actually a producer statement, but the base line system classify it as consumer statement because it has the *want* - consumer verb.

**Improvement Strategy:**

The system is improved to match with the statements that has related terms, in addition to exact match statements by using the wordnet. The hyponyms(Semantic – 1), hypernyms(Semantic – 2), and synsets(Semantic – 3) are employed to increase the efficiency of the system and thereby helping the consumer to get matched with the service provider who could most probably satisfy the need.

**Examples:**

Consumer Statements Service Provider Statements

I need a latte I sell vegetables

I want to buy carrots We sell espresso

In the above examples

(i)Latte – one type of coffee where espresso is another type of coffee. These two gets matched by retrieving all the related forms of latte from wordnet.(espresso found as hypernym of latte in wordnet).

(ii)Vegetables are found as hypernym of Carrots in wordnet and is used to find this particular match.

**Improvement Strategy implementation:**

Initially, using the lexemes(Lexical - 1) as such in the user statements created a lot of mismatches. So the nouns and verbs are stemmed(Lexical - 2) by using a Stemmer. Porter Stemmer’s code is used with a slight modification. The code is modified because it did not handled the case for plurals ending with ‘es’ and system did not required all the steps that are done in the stemmer.(i.e) Only the stemming of the verb required all the steps performed in the stemmer code but the nouns(services) required only stemming of plural form. So the modified Porter Stemmer code is used for stemming. The nouns are also normalized to a particular case, so that it could be matched easily.

The interrogative statements are handled by framing rules that worked efficiently in most of the cases. The rules are

Consumer verb in interrogative sentence – Producer statement

Eg:

*Anyone need car driver?* – need(consumer verb) in interrogative statement – Producer statement

*Who want taxi?* – want(consumer verb) in interrogative statement – Producer statement

Producer verb in interrogative statement – Consumer Statement

Eg:

*Anyone sell medicine?* – sell(producer verb) in interrogative statement – Consumer statement

*Who provides taxi?* – provides(producer verb) in interrogative statement – Consumer statement

The related words of the services are retrieved from wordnet. Wordnet API(JAWS) is used to extract the data form wordnet database. For each noun(service) all the associated synsets, hyponyms, hypernyms are retrieved. This process is repeated so that the synsets, hyponyms, hypernyms of the next levels are retrieved. The nesting is restricted to two levels since the higher levels gave more generalized forms resulting in more number of false positives. In addition to the wordnet API, the similarity value(Semantic - 4) between the noun(service) in the statement and the extracted related term from wordnet is found. Only if the relatedness value is above certain threshold value(0.8), the particular word is considered for the final result matching otherwise those related words are ignored.

**Results**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Consumer statements | Baseline system results | | | | | | | Complete system results | | | | | | |
| TP | FP | TN | FN | Pr | Re | F | TP | FP | TN | FN | Pr | Recall | F |
| Anyone delivering letters? | Got recognized as a producer statement instead of consumer statement because of the verb *‘deliver’* | | | | - | - | - | 1 | 0 | 99 | 0 | 1.0 | 1.0 | 1.0 |
| I desire for coffee | 4 | 0 | 88 | 8 | 1.0 | 0.33 | 0.5 | 12 | 0 | 88 | 0 | 1.0 | 1.0 | 1.0 |
| Need car repair | 3 | 2 | 94 | 1 | 0.6 | 0.75 | 0.672 | 3 | 7 | 89 | 1 | 0.3 | 0.75 | 0.43 |
| Looking for Pizza | 0 | 0 | 95 | 5 | 0 | 0 | 0 | 2 | 0 | 95 | 3 | 1.0 | 0.4 | 0.57 |
| Anyone sell batteries? | 0 | 94 | 5 | 1 | 0 | 0 | 0 | 1 | 1 | 98 | 0 | 0.5 | 1.0 | 0.67 |
| I demand a Dress | 0 | 0 | 94 | 6 | 0 | 0 | 0 | 6 | 4 | 90 | 0 | 0.6 | 1.0 | 0.75 |
| I require tomatoes | 0 | 0 | 95 | 5 | 0 | 0 | 0 | 3 | 0 | 95 | 2 | 1.0 | 0.6 | 0.75 |
| I want to consult a doctor | 2 | 0 | 97 | 1 | 1.0 | 0.66 | 0.80 | 3 | 2 | 95 | 0 | 0.6 | 1.0 | 0.75 |
| I like to buy bouquet | 1 | 0 | 99 | 0 | 1.0 | 1.0 | 1.0 | 1 | 0 | 98 | 1 | 1.0 | 0.5 | 0.67 |
| Seeking for a blood bank | 3 | 0 | 95 | 2 | 1.0 | 0.6 | 0.75 | 4 | 0 | 95 | 1 | 1.0 | 0.8 | 0.89 |

Pr – Precision , Re - Recall, F – Fmeasure.

**Summary of Problems encountered**

1. Sentence having two verbs created problems in classifying the user query.
   1. Eg: *I want to go to Florida*. Want and go are two verbs. In such cases, the sentence had the structure as (VB TO VB). The user query is checked for this structure(Syntactic - 2) and it showed that the first verb is used for classifying the query.
2. Porter Stemmer is used to find the stem of a word. The system requires stemming of both verbs and nouns, but the stemmer code produced satisfying results only to the verbs and hence a few changes are done make it suitable for stemming nouns. The change is

* The last few steps in stemming seemed unnecessary for nouns, so those steps are removed for nouns stemming
* Then plurals ending with *‘ies’* were not handled properly and so required changes are done

1. User queries as a question gave different sense from the verb it had. Considering only the verb is not sufficient to find the actual sense
   1. Eg: *Anyone want grocery?* – Producer statement, even it has *want*(consumer verb). This is handled by framing certain rules.
2. The required amount of related words could not be found using normal thesaurus. So, this initiated the use of wordnet.
3. Single level of synsets, hypernyms, and hyponyms are not sufficient to extract the required amount of related words. This is handled by recursively finding one more level of the synsets, hypernyms and hyponyms
4. The two levels of hyponyms and hypernyms gives innumerous related terms that results with more false positives. The number of false positives are restricted by finding the similarity values between the words extracted from the wordnet. This is done by making a call to the WS4j webserver, since the product has not been released with API[Thus the system needs internet connection always to communicate with the WS4j webserver].
5. The gloss of the related synsets of a word are retrieved to match with the requirements, but this attempt is unsuccessful resulting in more false positives.
   1. Eg: consumer statement: *I need dress*. While searching related words, the system found ‘*dinner dress’* as a related synset using wordnet. The system retrieved the corresponding gloss that had a word ‘*dinner*’in it. So it made the system to include any producer statement that had ‘*dinner*’ in it which is completely out of scope from dress. So the usage of gloss had been dropped.

**Pending issues & future IMPROVEMENTS**

**Pending Issues:**

1. Shorter versions of user queries are not working properly *Need water?*. The query alone is a consumer statement but the symbol ‘?’ has transformed the query to a producer query. So this needs to be fixed. The system handles the queries having question words like ‘Wh’ question words and the queries starting with auxillary verbs.
2. The system fails to handle the compound nouns properly. Though it handles a few cases, it is generating a lot of false positives.

Eg: Consumer statement – *I like to have a car repair*, here *car* is a noun and *repair* is an another noun. These two combine to give a compound service called *car repair.* The system finds match in the producer corpus by matching individual nouns and then compound noun as a whole, therefore Producer statement – *I drive taxi* will also get matched because taxi is a synonym for one of the constituent of compound entity – *car.*

1. The usage of adjectives to give more appropriate results is tried but not completed fully.

**Future Improvements:**

1. The system presumes that the queries will be error(spelling and syntactic) free. But the issue of correcting error before applying the algorithm needs to be implemented.
2. The system does not understand the implicit semantic meaning.
   * 1. Eg: Consumer statement – *I want money* is not getting matched with the Producer statement – *I can donate 70$*
     2. Eg: Consumer statement - *I need blood* is not getting matched with producer statement – *I can provide first aid* because the system is not trained to understand that the blood is needed in the situation of emergency in the above example.

Thus the system must be enhances to understand the implicit meaning conveyed in the user query to give the most optimum results.

1. System should be improved to support the use of passive statements also
2. Eg: Driver needed – consumer statement that will not be recognized by the system now.

**CONCLUSION:**

Thus the problem statement is implemented with few pending issues. The project helped to understand the role of different features (syntactic, semantic and lexical) in Natural Language Processing and importance of applying it to produce the efficient results.