Creating a Multilingual Lexical Resource

Richard Seliga
School of Computer Science
University of Colorado, Colorado Springs
Colorado Springs, Colorado 80920
rseliga@uccs.edu

Abstract - **In this paper we describe how to create a multilingual lexical resource using corpora. This resource will be available to researchers and the public using a web interface. The input data includes millions of words in different languages. These words will of course have to go through some preprocessing before finally being uploaded to a MySQL database. Keeping the data indexed is crucial and will allow easy access and quick searches. The searchable information will include the number of occurrences of a particular word, significant bigrams and trigrams that include this word, significant right or left neighbors, parts-of-speech for words, relationship graph among the words and example sentences where this word appears. This can all be done creating four to five tables for each language, which we will be using. Later on a user based dictionary will be created to add another aspect to this linguistic recourse. We will create a different database for each language as it seems the most convenient. The data will stress languages like Slovak and Assamese, but will also include other languages like Czech, English, Polish, Bengali and Kannada. The reason for creating this website is creating a resource for languages that are not as common to find recourses for on the web.**

I. Introduction

This project is intended to create a large linguistic library of words, their uses and their definitions for those who speak an Indic and Slavic languages, but our initial focus will include languages like Assamese and Slovak. A website will be created to allow the user to research relationships among words in terms of occurrences, bigram or trigram frequencies, and significant right or left neighbors; part of speech for words and usages of the words in different part of speech; and will also allow the user to add definitions to a user based dictionary. Another feature that will be developed will include the ability to search for root words. After all this is finished the website should show about two pages worth of information for every word entered. Part of speech tagging will be a challenge and we will either use an existing POS tagger or work on to develop a new one. All the information for these different options will be extracted from the corpora that are free around the web. This will be a great resource for those who want to:

- have access search engine for lexical resources
- get statistical information about your query word
- have access to lexical information on uncommon languages like Slovak and Assamese.

II. Related Work

A project has been started by the Leipzig University, Computer Science Institute in the turn of the millennium and their website now receives more than 170,000 monthly visits. They have come up in their proposal with a table that shows how many words are in their main corpora (see Table 1). [4][1] Obviously with the languages we will be using, we will not have as many courses, but using Wikipedia dumps and free corpora we will works ourselves up to a nice number. Biemamn wanted to create a flexible website that allows people from around the world to research relationships among words. He also created a great standard of comparison for new websites.

Table 1

	German	*English*	*Italian*	*Korean*
Word Tokens	500 Mill.	260 Mill.	140 Mill.	38 Mill.
Sentences	36 Mill.	13 Mill.	9 Mill.	2.3 Mill.
Word Types	9 Mill.	1.2 Mill.	0.8 Mill.	3.8 Mill.

Another future addition might include graphing the relationships of word. Hatzigeorgiu created a project that displays the data about this. In his paper he describes how world length and occurrences show up on a graph. When he mapped out word length against the number of occurrences he came out with some pretty interesting results. It will be fascinating to see how it is mapped out in Slovak and Assamese, and other Indic and Slavic languages. This is another thing that might be added to our website later on. [3]

III. Data Recourses

A. Data Sources

The main and only source of the data is free corpora which are available on the web or have been developed by universities. These collections of text provide anywhere from 2 million to 36 million sentences in each language. The corpora that we will be using include the American National Corpus and Slovak National Corpus for the beginning and later on expand the languages to mostly Indic and Slavic languages, like Assamese, Bengali, Kannada and Czech, Polish. Another option for more text to work with will include extracting Wikipedia arti-

cles in their respective languages. Here is some of the data we obtained from the three languages so far (see table 2).

Table 2

	English	*Slovak*	*Assamese*
Word Tokens	11.1 Mill.	17.7 Mill.	2.431 Mill.
Sentences	1.8 Mill.	2 Mill.	.
Word Types	0.23 Mill.	0.94 Mill.	0.21 Mill.

The American National Corpus[1] contains over 6000 documents while the Emille[2] corpus, developed by the University of Lancaster one is much smaller. The 17.7 million words in the Slovak language were obtained from a XML Wikipedia dump. We used this Wikipedia dump because the Slovak national corpus is not as accessible as we first thought.

B. Text preprocessing

All the text processing in our project will be done using Perl, since it has great capabilities with word processing. In this section we will describe the steps to construct a text database

1. Create a two dimensional array of strings that includes the sentence and their respectable words.
2. Strip all punctuation from the text document.

Each word will have the index of what sentence it is in and the index of the position in the sentence. This will eliminate getting inaccurate data. This is necessarily because the end of a sentence and a beginning of another is not a bigram. The next step will involve counting the number of accurateness of each word in the documents and sending this data into a database of unigrams which will include the primary key of the unigrams, the spelling of the word and how many times it has occurred in the corpus. The index will provide us with an easy way to connect two or three words. The connecting of words will be based on the significant right or left neighbors of the word. This will create collocation which is the occurrence of two or more words within a sentence or a document. We will keep count of this data and display it on request of the user. By indexing the sentences, we will create a table including the id of the sentence and merge it with the unigram table to display what unigrams appear in what sentences. One of the other tables we will create is a root table, which will display the root word.

A. Creating the Database

The database will be one of the most crucial parts of this project. It will supply our users with quick responses to their queries which will be accessed through a web interface. The main table will be the unigram table. The unigram table will include the id, spelling and the count. To get bi-grams and trigrams all that will be done is connect two and three word ids. The sentences will be in another table and the only relation it will have to the unigram table is that it receives the data from the same document collection.

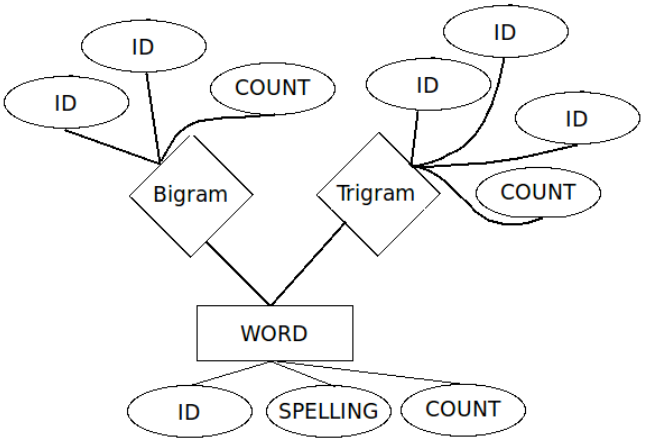
[1]American National Corpus {http://www.americannationalcorpus.org/}
[2]The Emille Corpus{www.ling.lancs.ac.uk/corplang/emille/}

Most available sentence splitters are not very good and sometimes cause errors with one word sentences. This problem will be avoided by querying only sentences between a certain character lengths. Indexing the data will make it extremely quick to access, and will keep the users happy. The entity relationship diagram can be seen for our data can be seen on figure 1 and some of the most popular unigrams in the English language can be seen it table 3.

Table 3

ID	*SPELLING*	*COUNT*	*ID*	*SPELLING*	*COUNT*
1	the	721015	41	is	121803
21	of	385945	33	for	115616
49	and	298341	159	with	85187
11	to	268803	213	as	76022
35	in	241763	71	on	70113
4	a	235332	7	by	67411
40	that	132009	247	was	64532

Figure 1



IV. FUTURE WORK

A. POS Tagging

Using a part of speech tagging algorithm we will create another resource for our users. It will show the word queried by the user and see what POS it is and display it. This will be different for the Slovak language as it has a much more complex grammar. We will start with Slovak, but later extend the work to other Slavic languages and one or more Indic languages. The biggest problem of tagging the Slovak language is that the tools for the Slovak language are underdeveloped since Slovakia claimed its independence from Czechoslovakia in 1993 and most of the people that did research did it in Czech. On the positive side, the Slovak language is very similar to Czech and a POS tagger has been created named ajka for the Czech language.

1) Czech POS Tagging:

For illustration, let's assume word-form zdi (walls). One of the morphological annotations corresponds to the genitive

singular for feminine nouns, other to the dative, vocative and locative singular, or nominative and accusative plural of the same word. The other corresponds to the imperative of singular of the verb and so on. Each morphological category (case, gender, number...) may take a set of possible values (gender - masculine animate, masculine inanimate, neuter, and feminine). The morphological annotations of a word form represent the combinations of morphological categories for the particular part of speech classes. [7] This creates a lot of different variations and will make the tagging extremely tough.

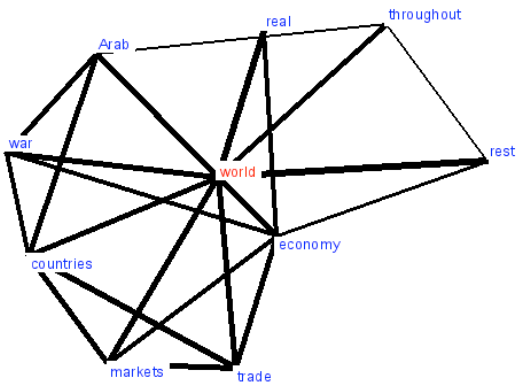
B. *User Input Definitions*

One of the later on additions to this website will include the ability for the user to enter their own definition of the word. Then the definition will be run through a Spam filter and if it successfully runs through it then the definition will be added to the website automatically and be viable by the users. One of the options that we see is using a summarizing posts algorithm to summarize the contents of the user entry and combine this to create some very reliable definitions.

C. *Word Relation Graphs*

Another future addition to this website will include a diagram between different words and their most common neighbors. Even though this seems simple, connecting the words neighbors and their neighbors' neighbors all within one graph will become quite complex and difficult to display properly. Figure 3 shows the graph located on wortzchatz[4]. We will try to make our relationship model a little different this. Also the bolder the line, the more occurrences are related to this subject.

Figure 2



D. *Other Minor Additions*

Some of the minor additions that would be added in the future is that every word displayed on the site would beside it have its count in the corpus and if clicked would display the definition of the word.

V. EXPERIMENTS

Some of the experiments so far have included testing of a website that connects to a database and searches for the most common bigrams and trigrams. The results posted look very promising. The text documents tested included punctuation marks, random lines and double spaces. After testing the same code on the Slovak and Assamese language we had to tweak our code a little to use different encoding. Following this we had 3 databases with all the information needed. Subsequent to testing, we tested some different way to display sentences and we figured out that most sentences that are important are between 50 to 120 characters and this helped us displaying some very good data.

VI. APPROACH TO SOLVING PROBLEMS

Our approach to solving the problems will involve using the scripting language Perl. It is very good with text manipulation and that is what this project needs. Using Perl we first figured out how to count the letter frequencies and word frequencies. Currently we are working on bigrams and trigrams matching by finding out the word that comes before or after the word entered by the user. We will achieve the frequencies by checking the database during insertion. After this is finished we will start working on a basic sentence splitter that will allow us to create a database of sentences. Part of speech tagging will be the final part of this project. After all this has been tested and is working we will implement the different languages and start working on the website. What we did first was made our document available for extraction of information by deleting out some of the punctuation and other unnecessary content. When this was finished, the extraction was quite simple using a *foreach* statement. What we did was put my text document into an array of words and later on used a while loop to look for the appropriate word, while keeping a counter of how many words there are before the current word. When the word was finally found we just looked for the word that had an index of one higher than the counter and one lower than the counter. The solution for the parts of speech tagging will be very interesting to figure out.

VII. SLOVAK LANGUAGE

A. *Slovak Morphology*

Like most other Slavic languages, and contrary to English or German. Slovak is an inflected language. Basically, there are three major types of word-forming processes inflection, derivation, and compounding. Inflection refers to the systematic modification of a stem by means of prefixes and suffixes. Inflected forms express morphological distinctions like case or number, but do not change meaning or POS. In contrast, the process of derivation usually causes change in meaning and often changes of POS. Compounding deals with the process of merging several word bases to form a new word.

B. *Slovak Data Collected*

Some of the data collected from the Wikipedia dump has very good size and quality. In table 4 you can see the most common

words in our Slovak database. Since the sentence structure in Slovak is very similar to the one in English we used our English splitter and the results looked very good. The only problem that we ran into was that our Wikipedia dump included a lot of Wikipedia titles and this counted as sentences. We avoided displaying these by only selecting the sentences that had above a certain character count. This proved to be a solution to that problem and after testing some words the data displayed proved to be good sentences.

Table 4

ID	*SPELLING*	*COUNT*
119	v	575566
136	a	517223
154	na	295515
132	sa	286466
116	je	257444
120	roku	109615
190	aj	99273

VIII. ASSAMESE LANGUAGE

A. *Assamese Data Collected*

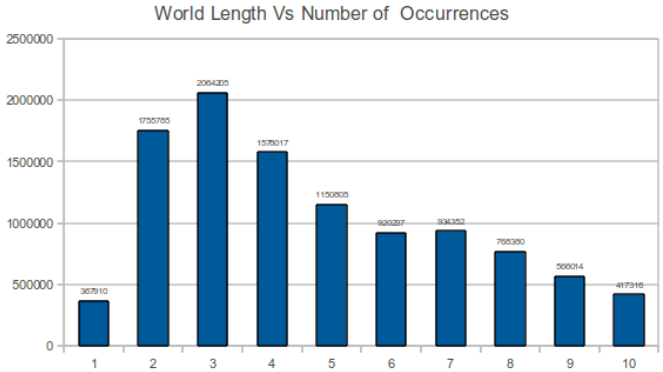
We used an Assamese corpus which was not huge in size but it had a lot of quality text. It was all pre parsed and was a freeze to use. In table 5 you can see the most common words in our Assamese database. Since the Assamese language uses Eastern Nagari Script we had to change our code to a different encoding to handle it. The data displayed looked great.

Table 5

ID	SPELLING	COUNT
13	আৰু	45792
41	এই	22480
30	কৰি	15765
226	হয়	13431
40	পৰা	13415
55	কৰা	12708
398	এটা	12548

IX. WORD LENGTH VS. NUMBER OF OCCURRENCES

A. *English*



X. CONCLUSION AND FUTURE ADDITIONS

Part of speech tagging is going to be one of the more challenging topics addressed and implemented into the website. Horak, the developer of this tagger uses a similar tool that has been used for the Czech language [5]. It's called AJKA and has been developed by Sedlacek. He describes three major parts of word-forming processes: inflection, derivation, and compounding. Inflection refers to the systematic modification of a stem by means of prefixes and suffixes. Inflected forms express morphological distinctions like case or number, but do not change meaning or POS. In contrast, the process of derivation usually causes change in meaning and often change of POS[6]. Compounding all of these together causes problems that are unlike anything in the English language.

In conclusion getting this project will be very challenging and time consuming to work properly. This project would be a good challenge for an intermediate programmer. The final product is something that I am very much looking forward too as it will be extremely satisfying. The final thoughts on this project and some later additions would include an on screen keyboard pop up that would allow users in India, where the amount of characters is so extreme it forces the population to only type in English causing only ten percent of the population to have access to computers. Another addition would be a dictionary that would allow the users to enter the definition in their own language through this on screen keyboard and make the site a very good resource that is user based.

REFERENCES

[1] U. Quasthoff and M. Richter and C. Biemann, Corpus Portal for Search in Monolingual Corpora, Leipzig, Germany: Augustusplatz 11, 04109, 2006.
[2] L. Egghe, The distribution of N-grams, Akadmiai Kiad, Budapest, 2000.
[3] N Hatzigeorgiu, G Mikros, and G Carayannis, Word Length, Word Frequencies and Zipfs Law in the Greek Language, Maroussi, Greece, 2001.
[4] U. Quasthoff and M. Richter and C. Biemann, Language- Independent Methods for Compiling Monolingual Lexical Data, Leipzig, Germany: Augustusplatz 11, 04109, 2004.
[5] A. Hork, L. Gianitsov, M. imkov, M. motlk, and R. Garabk, Slovak National Corpus, Ludovt tr Institute of Linguistics, Slovak Academyof Sciences Bratislava, Slovakia, 2004.
[6] Radek Sedlacek and Pavel Smrz, LA New Czech Morphological Analyser ajka, Faculty of Informatics, Masaryk University Brno Bota nicka 68a, 602 00 Brno, Czech Republic, 2001.
[7] Jan Hajic and Barbora Hladka, Czech Language Processing – PoS Tagging, Institute of Formal and Applied Linguistics Charles University Malostransk nm. 25 118 00 Prague, Czech Republic
[8] Benot Sagot, Automatic Acquisition of a Slovak Lexicon from a Raw Corpus, INRIA-Rocquencourt, Projet Atoll, Domaine de Voluceau, Rocquencourt B.P. 105 78 153 Le Chesnay Cedex, France