Automatic Article Generator from Extracted Databases

Ianick Boudreault
Lund University, Sweden
ianboudreault@hotmail.com

Abstract - The primary goal of an automatic content generator is to bring available new information to the public by the means of search engines. In fact, nowadays there is a rush for accumulating a lot of data which is not always "humanly readable", mostly formatted into databases. The recent success of search engines has revolutionized our way to search for information, allowing them our trust in delivering us the most relevant results to our search requests. The goal of this paper is to present a tool to manipulate into articles such unformatted sources of data so it can be indexed by search engines ready for the end users to consult through their searches.

1. Introduction

Humanly readable information: this seems to be the new fashion in nowadays best search engines to deliver what they believe to be the most relevant results to its users. Wondering why they have attained such standards makes us wonder what the end user really wants. Will a database containing, for example thousands of species of insects be a better alternative then reading a websites presenting articles on each of these species? The answer to this question makes us believe that articles are much more convenient to read. However the main advantage of articles is that search engine will "like" such formatted data and will rank them in their results. This information will then be available to searches through specific keywords like "Hymenoptera bugs" or "Orthoptera insects". The optic of this tool is to get available information to the public that wouldn't be available through the World Wide Web.

2. The ContentBot tool

Contentbot is the name given to the article generator tool built here. Its goal is to build unique articles automatically with a template built by the user with its inbuilt template creator tools. It mainly works in two main operations: a template sentence creator containing "holes" to put the database entries and a synonym creator to make the articles more unique. An XML derived syntax is used to insert the desired information at the right places in the template sentences. An extraction tool is also available to extract web information aimed at creating database from online websites. In this paper, examples will be made with a beer database containing 1600 entries extracted from a list of beers taken on the web.

2.1 The ContentBot Interface

ContentBot has been built as a web application tool to guide the user through the steps of creating new articles from its raw database. The tool has been built to ease the creation of this template that will then be used to generate the articles. It handles adding, modifying and deleting sentences and synonyms while presenting the information in a way to make it easy to the user to write his articles. All the information is stored in a database on the server until the articles are generated and downloaded to the user. In this way, a user can create many projects and finish them later on. When the process is finished and the articles generated, the result can be viewed through the ContentBot result navigator and then be downloaded to the client station in different formats.

2.2 The ContentBot Extraction tool

The beer database presented as an example in this paper has been extracted with the ContentBot extraction tool. This tool is a script that basically finds unique HTML information before and after the desired information. This unique code has to be identical through each pages of the website to extract the same information in each beer page. Doing so for each data desired, we run the script to extract each information for each page of the website. This has been done on a beers website to furnish this database of 1600 beer type, having seven relevant information of each beer: the beers name, the style and sub style, the country, the brewery, the level of alcohol and a score assigned to the beer. For the moment, the database needs to be in the form of a single table, it will later be extended to support relational databases.

2.3 The Sentence Creator Tool

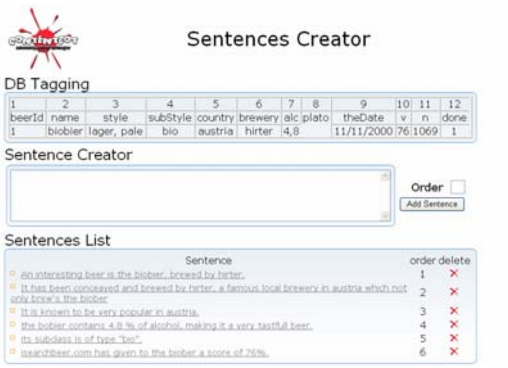
This is the first step to create the template. The tool first presents the first row of the database on top of the page as a resource for writing the sentences. For the beer database, we have the following items:

- Name: biobier
- Style: lager, pale
- Sub style: bio
- Brewery: Hirter
- Country: Austria
- Alcohol level: 4.8 %
- Score: 76 %

We then write an article as if it was written about the current row. Sentences are added one by one as they will be treated separately, for example:

> *"An interesting beer is the biober being classed as a lager, pale."*

The sentence creator tool is presented here with already composed sentences.



Sentences Creator

DB Tagging

1	2	3	4	5	6	7	8	9	10	11	12
beerId	name	style	subStyle	country	brewery	alc	plato	theDate	v	n	done
1	biobier	lager, pale	bio	austria	hirter	4,8		11/11/2000	76	1069	1

Sentence Creator

Order □

Add Sentence

Sentences List

Sentence	order	delete
An interesting beer is the biobier, brewed by hirter.	1	✕
It has been conceaved and brewed by hirter, a famous local brewery in austria which not only brew's the biobier	2	✕
It is known to be very popular in austria.	3	✕
the biobier contains 4.8 % of alcohol, making it a very tastfull beer.	4	✕
its subclass is of type "bio".	5	✕
isearchbeer.com has given to the biobier a score of 76%.	6	✕

The next step is to add the tags related to the database so that the sentences don't depend anymore on the first row but on the assigned column. Each column is identified by a unique id. Adding the id to the tag is of the form:

<-DB item=2 /DB->

During the final generation of the article, this tag will make the system replace the tag for the correct entry in the database. Here it would change *<-DB item=2 /DB->* for biobier. Replacing each tag on the example sentence will look like:

"An interesting beer is the
<-DB item=2 /DB-> being
classed as a <-DB item=3 /DB->"

2.4 The Synonym Creator tool

The article that would be produced from the only use of the Sentence creator tool would result in 1600 articles having the same text for each beer. That would result in a really bad website and our main goal to rank in search engines would fail since all major search engines use what is called “similar content filters” on websites. This means that having 1600 sentences with a high percentage being the same text would be detected and the website containing the articles would probably be dropped. This is where the synonym creator tool gets interesting as it is used to catch sections of a text and allocate synonyms. Having many synonyms for different parts of the sentence will result in a high level of diversity for each sentence during the final generation.

The Synonym generator asks the user to insert in the sentences “synonym tags” enclosing the desired part to work with. Using the same sentence as before for the example, the tag would look like this:

“<-syn item=a1->An interesting beer<-/syn-> is the <-DB item=2 /DB-> being classed as a <-DB item=3 /DB->”

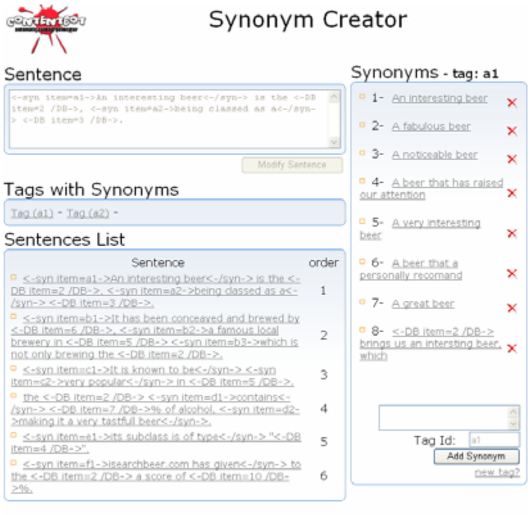
The id of the tag can be anything as long as it is made in one word. The synonym tool now lets the user add as much synonyms for the “a1” tag as wanted. We could for example add:

- *A fabulous beer*
- *A noticeable beer*
- *A beer that has raised our attention*
- *A beer that we would recommend*

We can add an infinite amount of synonym in a sentence. The more synonyms there is the highest are the chances of having all different sentence. The different possibilities then go exponential. For example for a sentence having three synonym tags in a sentence having eight synonyms each, this would make:

$$8^3 = 512$$

512 possibilities for this sentence alone. The interface of the synonym creator tool with the completed tagged sentences is presented below:



```
<-syn item=a1->An interesting beer<-/syn-> is the <-DB item=2 /DB->, <-syn item=a2->being classed as a<-/syn-> <-DB item=3 /DB->.
```

Sentence	order
<-syn item=a1->An interesting beer<-/syn-> is the <-DB item=2 /DB->, <-syn item=a2->being classed as a<-/syn-> <-DB item=3 /DB->.	1
<-syn item=b1->It has been conceaved and brewed by <-DB item=6 /DB->, <-syn item=b2->a famous local brewery in <-DB item=5 /DB-> <-syn item=b3->which is not only brewing the <-DB item=2 /DB->.	2
<-syn item=c1->It is known to be<-/syn-> <-syn item=c2->very popular<-/syn-> in <-DB item=5 /DB->.	3
the <-DB item=2 /DB-> <-syn item=d1->contains<-/syn-> <-DB item=7 /DB->% of alcohol. <-syn item=d2->making it a very tastfull beer<-/syn->.	4
<-syn item=e1->its subclass is of type<-/syn-> "<-DB item=4 /DB->".	5
<-syn item=f1->isearchbeer.com has given<-/syn-> to the <-DB item=2 /DB-> a score of <-DB item=10 /DB->%.	6

1- An interesting beer ✕
2- A fabulous beer ✕
3- A noticeable beer ✕
4- A beer that has raised our attention ✕
5- A very interesting beer ✕
6- A beer that a personally recomand ✕
7- A great beer ✕
8- <-DB item=2 /DB-> brings us an intersting beer, which ✕

The example sentence with the proper synonym tags would like this:

“<-syn item=a1->An interesting beer <-/syn-> is the <-DB item=2 /DB->, <-syn item=a2->being classed as a <-/syn-> <-DB item=3 /DB->.”

Several tools are available on the internet to help finding synonyms and ways to say things differently such as the Prinstons wordnet[1] tool and synonym dictionaries.

2.5 The generation of the articles

The generation tool then uses a random number to select the good synonym for

[1] WordNet : http://wordnet.princeton.edu/

each tag and adds for each of the articles the correct database entry. In our example, we have a total of six sentences, having a total of eleven synonym tags. To find out the amount of different sentence possibilities we multiply the amount of synonym per synonym tags together to find:

$$7x5x6x5x4x3x5x6x4x4x5 = 30\ 240\ 000$$

This makes a total of more than 30 million possibilities. Having 1600 articles to generate the chances a same article appears twice is

$$30\ 240\ 000 / 1600 = 18\ 900$$

This makes a probability of 0.06 % which is extremely low.

3. Conclusion

The results generated by ContentBot are very interesting as each article really seems like they have been written by a real author. These articles can then be added to a website that will be indexed by the search engines, ready for user consulting. A parallel project of mine will actually use the information generated by this beer example. This project called ISearch[2] is using article formatted information to attach to its search engine and making the information searchable through the pages of a new built website. The information is then available in the isearch site, which will be called www.isearchbeers.com. The articles will finally be easley spiderable through the major search engines and will be available to be found using keywords in the articles. Now someone searching for "*merlin's pilsener beer*" will probably find our article on the merlin's pilsener!

Annex A: Example of the beer articles generated

A beer that a personally recommend is the pilsner urquell, knowing to be a lager, pale. We appreciate this beer, thanks to plzn (pilsen), a famous local brewery in czech republic well known for its other conception than the pilsner urquell. This famous brewery is known to be loved in Czech Republic. the pilsner urquell contains an amount of 5,0% of alcohol, which makes it an interesting beer. It is also described as a "pilsen". isearchbeers.com has given to the pilsner urquell a score of 80%.

A beer that has raised our attention is the merlin's pilsener, knowing to be a lager, pale. All the credits are accorded to bextrim, a talented brewery from germany which is not only brewing the merlin's pilsener. This famous brewery is known to be very successful in Germany. The Merlin's pilsener contains an amount of 4,9% of alcohol, making it a very tasteful beer. It is also characterized as a "pilsen". We have accorded to the Merlin's pilsener a score of 70%.

An interesting beer is the dachsenfranz kellerbier, characterized as a lager, pale. It is actually brewed by Herbert Werner, an award winning brewery coming from Germany having much more to know than the dachsenfranz kellerbier. This beer is known to be very successful in Germany. The dachsenfranz kellerbier is powered by an amount of 5,2% of alcohol, enough for a good night of fun.

[2] Examples of Isearch at:
www.isearchquotations.com
www.isearchjokes.com
www.isearchbible.com

isearchbeers.com has granted to the dachsenfranz kellerbier a score of 70%.

A noticeable beer is the warsteiner, knowing to be a lager, pale. All the credits are accorded to warsteiner, a famous local brewery in Germany which is not only brewing the warsteiner. It is known to be loved in Germany. The warsteiner includes about 5,0% of alcohol, enough to cheer you up. isearchbeers.com has granted to the warsteiner a score of 86%.