Handover: Innovation Project ---- Comment Analysis System

By Yuming Yang

**Part I: innovation summary**

This innovation idea is E-shop Comment Analysis System. This is about how we serve for E-commerce companies.

Shopping online is more and more popular today. People buy products and leave comments on big E-commerce platforms such as Amazon. Most E-commerce companies have E-shops on these platforms. Their products received many customer comments there and it was inefficient for them to get feedback about their products by manually reading the comments.

So what they need is an assistant that could provide multi-dimensional comment analysis and extract customers’ opinions.

Our solution is to build an analysis server and integrate it with ERP system like B1. Our customers could finally see the comment analysis result in a dashboard in their ERP systems.

With regard to the proof of concept (POC) of this innovation idea, I wrote a Django project that could process the raw comments text and return the analysis result in a JSON. You can see the details in Part III and Part IV.

In terms of how to present this idea, I just use a mockup dashboard to show just one particular product’s comment analysis result instead of a real-time html front end. The dashboard has been inserted into B1 by an add-on. You can see about the presentation materials in Part II.

**Part II: Presentation materials**

Our presentation is based on a mockup dashboard, sample JSON result, PPT slides, presentation scripts and a demo on B1.

There are two versions of presentation. First is an all-round one with a story line and Second focus on the dashboard and solution details.

The procedure of the first version presentation is: PPT storyline & solution -> PPT describe the dashboard -> PPT solution detail -> show sample JSON -> demo on B1 -> PPT summary

The procedure of the second version presentation is: PPT outline -> demo on B1 -> describe the dashboard -> PPT details

**#PPT slides & scripts**

1st version: CAslide1.pptx CAscript1.docx

2nd version: CAslide2.pptx CAscript2.docx

**#Dashboard**

I build the dashboard on build.me ([www.build.me](http://www.build.me)).

My dashboard link: <https://standard.build.me/api/projects/35dcc810afcfb82d0ce860cd/prototype/snapshot/latest/index.html>

First version dashboard isn’t saved and just have pictures now. The linked page of the URL above is the second version dashboard. I think second version is much better. Pictures of both versions are attached.

**#Sample JSON**

The JSON file attached is generated from our POC Django project. It contains all comment analysis result of one particular product. It shows that we do analysis on comments genuinely.

If you want to show JSON result, you need to put this JSON file into a JSON interpreter to visualize it. I recommend this website <http://www.kjson.com/jsonformat/?f=1>.

**#Demo on B1**

Keguo wrote an add-on so that we could insert our dashboard in B1’s product information page. The add-on code is attached, you need to open it with MS Visual Studio: Open VS and select ‘open project’ then select this file ‘…\WindowsFormsApplication1\WindowsFormsApplication1\ WindowsFormsApplication1.csproject’ to open.

After you configure your B1 environment (If you didn't configure the B1 environment before, you could turn to Keguo or other developers for help), you can show the demo on B1 following directions below:

* login in the system as a company
* run the VS project you just opened

(These 2 procedures above could be done before the presentation)

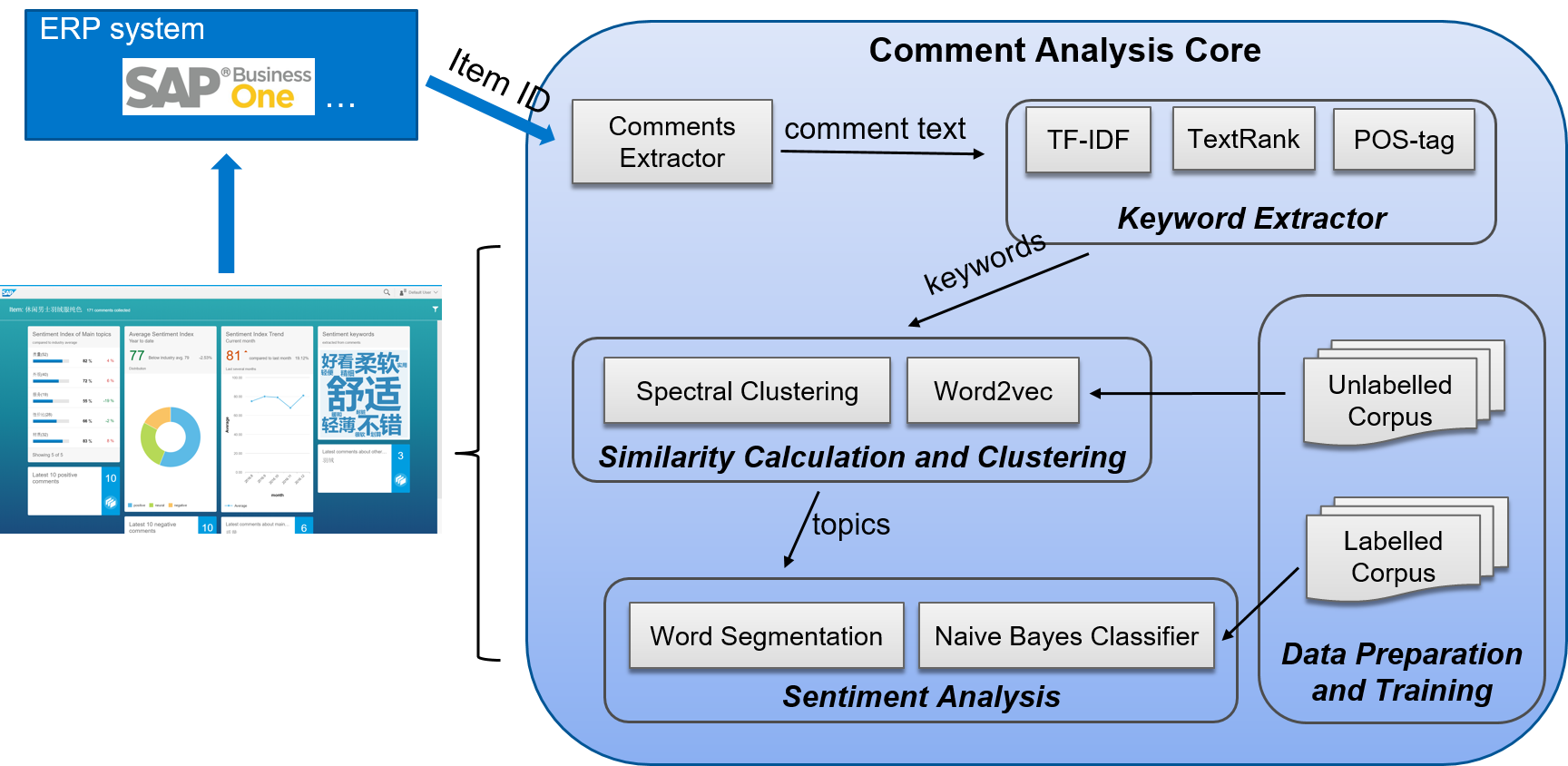
* go to Inventory -> Item Master Data -> click the arrows on top of B1 interface to a specific product information page
* click the ‘Comment Analysis Dashboard’ tab and you can see the dashboard (you need to login in Build at the first time).

**Part III: Solution details**

This part you could refer to slides and scripts in P6 - P8 in version 1 and P3 in version 2.

Here’s the explanation towards this flowchart.

The following flowchart will show how we analysis the comments. First our analysis server get the product’s item ID on E-commerce platforms from ERP systems (B1). Then our comments extractor will extract the product’s comments from the E-commerce platforms. Next, we use technology of TF-IDF, TextRank and POStag to mine keywords in the comments. These keywords will then be put into a similarity calculation unit. In this unit, we use a trained word2vec model to embed these keywords in a vector space. So that we can cluster the keywords with similar meanings as opinions by spectral clustering. Next, we do sentiment analysis toward each opinion. We use technology of word segmentation and Naive Bayes Classifier to quantify customers’ sentiment according to their expressions on each opinion. To ensure the accuracy of this Naive Bayes Classifier as well as the word2vec model mentioned above, we collected corpus from Wiki and E-commerce platforms and labelled some of them. The labelled corpus are used to train the Naive Bayes Classifier and the labelled corpus are used to train Word2vec model. Finally, all the analysis result are returned to the ERP systems and displayed in one dashboard.



Notice that the comments extractor section is not complemented in our POC, we use a web crawler to get the comments of the product we want to analyze before we start our analysis. And when we get the item ID, we just search for the corresponding comments text.

**Part IV: Codes and environment**

The Django project folder attached includes the codes of comment analysis core, the raw comments of products in different categories on Tmall crawled before and the already trained word2vector file. If you want the original code of the web crawler and word2vec training, they are in the folder ‘otherCode’.

To run the Django project, first you need to configure your environment to fulfill the requirements below.

Requirements:

Python 2.7.12

Django 1.10

Python package:

snownlp

numpy

jieba

genism

Recommended but not necessary:

Anaconda 4.1.1 32-bit, which includes the proper Python and Django version

Pycharm IDE, which is convenient to run Django.

After you configure the environment, go to the directory of manage.py and open the command line there. Input the followings:

python manage.py runserver 8000

python manage.py startanalysis

(If you use pycharm IDE, you need to open the folder as Django project and run it. And then find the terminal tab below and type ‘python manage.py startanalysis’)

Once the server is on and the analysis server is listening, you can let it do comment analysis through ‘127.0.0.1:8000/CA/api?q=’ now. For example, if you want to analyze the comments of product whose ID is 536499893821, you can try ‘http://127.0.0.1:8000/CA/api?q=536499893821’ in your browser. Then you should get the JSON result just the same as the sample JSON file attached. Notice that the POC could only analyze the product whose comments we have already crawled, have a look at this folder ‘…\PycharmProjects\CommentAnalysis\CommentAnalysis\management\commands\analysis\tbcomment’ to know the available product ID.